

Revisão do Projeto

Disciplina: Trabalho de Conclusão de Curso I – BCC

Caro, orientando,

segue abaixo a tabela de cálculo da média das notas obtidas no Pré-Projeto e Projeto, as DUAS revisões do seu projeto contendo a avaliação do professor “avaliador” e professor “TCC1”. Lembro que os ajustes indicados nestas revisões não precisam ser feitos no projeto, mas sim quando levarem o conteúdo do projeto para o artigo (se for o caso). Este material contendo todo o histórico das revisões é encaminhado para o professor de TCC2.

Tabela de cálculo da média das notas:

	Nome	Pessoa	PreProjeto										Banca			Projeto										Média
			TCC1					Avaliador								TCC1					Avaliador					
			A	P	N	Nota	A	P	N	Nota	Ori.	Esp.	Nota	A	P	N	Nota	A	P	N	Nota					
	Christian Trisotto Alegri	203693	15	1	0	16	9,79	15	0	0	15	10,00	8,00	8,00	8,00	16	0	0	16	10,00	15	0	0	15	10,00	9,8

Atenciosamente,

CURSO DE CIÊNCIA DA COMPUTAÇÃO – TCC		
() PRÉ-PROJETO	(X) PROJETO	ANO/SEMESTRE: 2023/1

SISTEMAS MULTIAGENTES APLICADO AO FUTEBOL DE ROBÔS

Christian Trisotto Alegri

Prof. Aurélio Faustino Hoppe – Orientador

1 INTRODUÇÃO

A robótica tem evoluído significativamente desde sua criação, há mais de 60 anos. Inicialmente, as máquinas eram projetadas para executar tarefas simples e repetitivas na indústria, mas hoje em dia, robôs são utilizados em uma variedade de áreas, incluindo saúde, segurança, exploração espacial, entre outras. Segundo Gupta (2013), as tecnologias de robótica têm avançado rapidamente nas últimas décadas, com o surgimento de novas técnicas de fabricação, sensores, atuadores, sistemas de controle e algoritmos de inteligência artificial, permitindo que robôs se tornem cada vez mais sofisticados e autônomos. A robótica tem o potencial de transformar radicalmente a maneira como as pessoas vivem e trabalham, e sua evolução continuará a impulsionar novas possibilidades e aplicações no futuro.

Dentro do campo interdisciplinar da robótica, uma área em rápido crescimento é a dos sistemas multiagentes, que são compostos por vários agentes autônomos que cooperam ou competem uns com os outros para atingir um objetivo comum. Um exemplo particularmente interessante de aplicação de sistemas multiagentes é o futebol de robôs, que tem o potencial de ser um desafio altamente complexo e desafiador para a inteligência artificial e a robótica. Segundo Huang *et al.* (2021), o futebol de robôs é uma das aplicações mais ambiciosas de sistemas multiagentes, que exige a integração de diversas habilidades e estratégias de tomada de decisão em um ambiente altamente dinâmico e imprevisível. O uso de sistemas multiagentes no futebol de robôs pode ajudar a resolver alguns dos principais desafios enfrentados pelos robôs individuais, como a limitação de sua percepção do ambiente e a dificuldade de tomada de decisões em tempo real.

Wooldridge (2009) destaca que um dos campos de aplicação mais fascinantes dessa tecnologia é o esporte. É nesse contexto que surge o futebol de robôs, uma competição que coloca em jogo a criatividade e a engenhosidade dos pesquisadores e entusiastas da robótica. O futebol de robôs é uma competição que envolve equipes de robôs jogando futebol em um campo especialmente projetado. Essa competição tem como objetivo incentivar o desenvolvimento de sistemas robóticos capazes de jogar futebol de forma autônoma. Para alcançar esse objetivo, os pesquisadores utilizam técnicas avançadas de inteligência artificial, como a visão computacional e os sistemas multiagentes.

De acordo com Chen *et al.* (2020), os Sistemas Multiagentes (SMA) são particularmente importantes no futebol de robôs, pois permitem que os robôs se comuniquem entre si e com o ambiente, coordenando suas ações para alcançar os objetivos do jogo. Esses sistemas são compostos por vários agentes autônomos que interagem entre si e com o ambiente em busca de alcançar os objetivos específicos do jogo. No entanto, a aplicação de sistemas multiagentes no futebol de robôs ainda enfrenta vários desafios, como a dinamicidade e imprevisibilidade do jogo, além das situações de conflito que podem ocorrer entre os robôs durante a partida. Por isso, é importante avaliar se os sistemas multiagentes podem ser utilizados de forma eficiente no futebol de robôs.

Tisue e Wilensky (2004) afirma que o desenvolvimento de simuladores é uma das principais ferramentas utilizadas para avaliar o desempenho de sistemas robóticos em diversas áreas de aplicação. No futebol de robôs, os simuladores são particularmente importantes para testar e avaliar novas estratégias e abordagens de forma segura e controlada. Dentre os simuladores utilizados para essa finalidade, destaca-se o NetLogo, uma plataforma de modelagem e simulação de sistemas complexos baseada em agentes. O NetLogo permite modelar e simular sistemas multiagentes de forma intuitiva e visual, além de possuir uma ampla variedade de ferramentas de análise.

Diante desse contexto, a pergunta de pesquisa deste trabalho é: sistemas multiagentes podem ser utilizados de forma eficiente no futebol de robôs, considerando as características dinâmicas e imprevisíveis do jogo e as situações de conflito que podem ocorrer entre os robôs?

1.1 OBJETIVOS

O objetivo deste trabalho é utilizar a plataforma NetLogo para simular futebol de robôs a partir de sistemas multiagentes, com o propósito de avaliar diferentes estratégias de jogo e estudar o comportamento dos agentes em um ambiente controlado.

Os objetivos específicos são:

- modelar cada um dos elementos do jogo de futebol como um agente, definindo seu conhecimento,

- funções, comportamento e modos de interação no NetLogo;
- b) analisar o comportamento dos agentes em diferentes situações de jogo, como jogadas de ataque, defesa, cobranças de falta, entre outras;
- c) avaliar o desempenho do modelo em simulações e testes reais, utilizando métricas relevantes para futebol de robôs, como a taxa de sucesso em marcar gols, a precisão na tomada de decisão e a eficiência na coordenação de múltiplos robôs.

2 TRABALHOS CORRELATOS

Nesta seção são apresentados trabalhos com características semelhantes aos principais objetivos do estudo proposto. A subseção 2.1 apresenta um modelo para predição do comportamento dos jogadores (OMIDSHAFEI *et al.*, 2022). Na subseção 2.2 é descrito um método para treinar times de agentes virtuais a jogar futebol em alto nível (LIN *et al.*, 2023). Por fim, a subseção 2.3 apresenta uma abordagem para jogar partidas completas de futebol multiagentes a partir de demonstrações de um único agente (HUANG *et al.*, 2021).

2.1 MULTIAGENT OFF-SCREEN BEHAVIOR PREDICTION IN FOOTBALL

Omidshafei *et al.* (2022) abordam o problema de imputação de séries temporais multiagente no contexto do futebol, onde subconjuntos de jogadores podem entrar e sair do campo de visão das imagens de vídeo transmitidas, enquanto os jogadores não observados continuam a interagir fora da tela. O objetivo dos autores era desenvolver um método de aprendizado de máquina que pudesse usar observações passadas e futuras de subconjuntos de jogadores de futebol para estimar as observações ausentes dos jogadores fora da tela.

A abordagem de Omidshafei *et al.* (2022), chamada de Graph Imputer, é um método de aprendizado de máquina que combina redes de grafo e *autoencoders* variacionais para permitir a aprendizagem de uma distribuição de trajetórias imputadas para jogadores de futebol fora da tela. O método envolve várias etapas. Primeiro, os autores constroem um grafo que representa as interações entre jogadores observados fora da tela. Os nós no grafo representam os jogadores e as arestas representam as interações entre eles. Em seguida, os autores treinam um AutoEncoder Variacional (AEV) nas trajetórias observadas dos jogadores para aprender uma representação de baixa dimensão do comportamento dos jogadores. O AEV é treinado para reconstruir as trajetórias observadas e gerar novas trajetórias que são semelhantes às observadas. Por fim, os autores usam a representação aprendida pelo AEV para codificar as trajetórias observadas dos jogadores e gerar trajetórias imputadas para os jogadores fora da tela. As trajetórias imputadas são geradas demonstrando a distribuição de comportamento dos jogadores aprendida condicionada às trajetórias observadas. A Figura 1 apresenta uma sequência de trajetórias geradas a partir do AEV.

Figura 1 – Visualização das trajetórias



Fonte: Omidshafei *et al.* (2022).

Para testar a eficácia de sua abordagem proposta, Omidshafei *et al.* (2022) realizaram experimentos a partir de jogos de futebol do mundo real. Eles utilizaram o módulo de câmera para simular a estimativa do estado dos jogadores fora da tela. Os autores avaliaram o método usando o *framework* de controle de campo (*pitch control*), ao qual depende de dados totalmente observados. Segundo Omidshafei *et al.* (2022), o controle de campo é um método para avaliar a habilidade de uma equipe em controlar o espaço no campo e restringir os movimentos da equipe adversária. Além disso, os autores também compararam o método com várias abordagens de ponta, incluindo aquelas feitas manualmente para o futebol, e mostraram que o método superou todas as outras abordagens em todas as métricas consideradas, conforme pode ser visto na Figura 2.

Figura 2 – Comparação com outros métodos

	Model	Pitch control MAE
Restricted	Role-invariant VRNN	0.0447 \pm 0.0204
	Bidir. Role-invariant VRNN	0.0209 \pm 0.0096
General	LSTM	0.0592 \pm 0.0274
	Bidir. LSTM	0.0296 \pm 0.0139
	Social LSTM ⁷	0.0474 \pm 0.0216
	Bidir. Social LSTM	0.0219 \pm 0.0098
	GVRNN ¹²	0.0418 \pm 0.0189
	Graph Imputer (Ours)	0.0207 \pm 0.0094

Fonte: Omidshafei *et al.* (2022).

Omidshafei *et al.* (2022) concluem que a abordagem é eficaz na previsão do comportamento de jogadores fora da tela em jogos de futebol. Os autores também demonstraram que o método supera várias abordagens de ponta, incluindo aquelas feitas exclusivamente para o futebol. Porém, ressaltam que o método depende de dados das trajetórias verdadeiras, que nem sempre estão disponíveis na prática. Omidshafei *et al.* (2022) sugerem avaliar a eficácia do método em configurações mais realistas, onde apenas observações parciais do comportamento dos jogadores estão disponíveis. Como possíveis expansões do método, apontam a necessidade de incorporação de fontes adicionais de informação, como atributos de jogadores ou formações de equipe, para melhorar a precisão das previsões. Os autores também sugeriram que o método poderia ser aplicado a outros domínios, como dinâmica de multidões ou robótica, onde sistemas multiagentes são comuns.

2.2 TIZERO: MASTERING MULTI-AGENT FOOTBALL WITH CURRICULUM LEARNING AND SELF-PLAY

Lin *et al.* (2023) desenvolveram um método para treinar times de agentes virtuais a jogar futebol em alto nível. Os autores combinaram duas técnicas de aprendizado de máquina: aprendizado de currículo e autojogo. O aprendizado de currículo envolve aumentar gradualmente a dificuldade das tarefas que os agentes devem executar, enquanto o autojogo envolve fazê-los jogar uns contra os outros para melhorar suas estratégias ao longo do tempo. Ao utilizar essas técnicas, os autores esperavam desenvolver agentes que pudessem cooperar e competir efetivamente como um time, melhorando o desempenho geral do jogo de futebol.

Segundo Lin *et al.* (2023), na abordagem de aprendizado de currículo aumentou-se gradualmente a dificuldade das tarefas aos quais os agentes foram solicitados a executar, começando com tarefas mais simples, como passar e driblar, e progredindo para cenários mais complexos que exigiam coordenação e trabalho em equipe. De acordo com os autores, ao usar tal técnica, os agentes foram capazes de construir sobre suas experiências anteriores e melhorar gradualmente seu desempenho geral.

Lin *et al.* (2023) também utilizaram o autojogo para treinar seus agentes, onde cada agente jogava contra uma cópia de si mesmo com uma política ligeiramente modificada. Essa abordagem permitiu que os agentes aprendessem com suas próprias experiências e ajustassem suas políticas para melhorar seu desempenho ao longo do tempo. Neste caso, os autores utilizaram uma variante do algoritmo de Otimização de Política Próxima de Múltiplos Agentes (OPMA) para treinar seus agentes por meio do autojogo. Esse algoritmo permitiu que os agentes aprendessem com as experiências uns dos outros e ajustassem suas políticas com base em suas observações.

Além disso, Lin *et al.* (2023) introduziram uma abordagem inovadora para o autojogo que envolveu emparelhar os agentes de uma maneira que encorajava a cooperação em vez da competição. Essa abordagem, segundo os autores, resultou no desenvolvimento de agentes que poderiam trabalhar juntos efetivamente como um time, melhorando o desempenho do jogo de futebol. Em termos de resultados, Figura 3 apresenta a comparação do desempenho do TiZero em relação a vários *baselines*, incluindo TiKick, WeKick, JiDi_3rd, Built-in, Hard Rule-Based-1 e Rule-Based-2.

Figura 3 – Comparação TiZero em relação a outros métodos *baselines*

Metric	TiZero (Ours)	TiKick	WeKick	JiDi_3rd	Built-in Hard	Rule-Based-1	Rule-Based-2
Assist	1.30(1.02)	0.61(0.79)	0.20(0.47)	0.35(0.62)	0.20(0.55)	0.28(0.59)	0.22(0.53)
Pass	19.2(3.44)	6.99(2.71)	5.33(2.44)	3.96(2.33)	11.5(4.63)	7.28(2.77)	7.50(3.12)
Pass Rate	0.73(0.07)	0.65(0.17)	0.53(0.18)	0.44(0.19)	0.66(0.12)	0.64(0.17)	0.63(0.19)
Goal	3.42(1.69)	1.79(1.41)	0.88(0.88)	1.43(1.34)	0.52(0.91)	0.73(0.69)	0.64(0.82)
Goal Difference	2.27(1.93)	0.71(2.08)	-0.47(1.68)	-0.02(2.14)	-1.06(1.93)	-0.60(1.03)	-0.71(1.45)
Draw Rate (%)	8.50	22.2	29.0	23.2	24.8	28.7	27.8
Lose Rate (%)	6.50	23.5	44.2	33.8	59.6	48.2	49.5
Win Rate (%)	85.0	54.3	26.8	43.0	15.6	23.1	22.7
TrueSkill	45.2	37.2	30.9	35.0	24.9	28.2	27.1

Fonte: Lin *et al.* (2023).

Segundo Lin *et al.* (2023), o TiZero superou todos os outros métodos em termos de assistência, passe, taxa de passe, gol, diferença de gol, taxa de empate, taxa de derrota, taxa de vitória e TrueSkill. O TiZero apresentou uma média de assistências de 1,30, uma média de passes de 19,2 e uma taxa de passe média de 0,73. Em termos de gols, TiZero marcou uma média de 3,42, com a segunda melhor pontuação de TiKick sendo apenas 1,79. Além disso, também teve a maior diferença de gols, com uma média de 2,27.

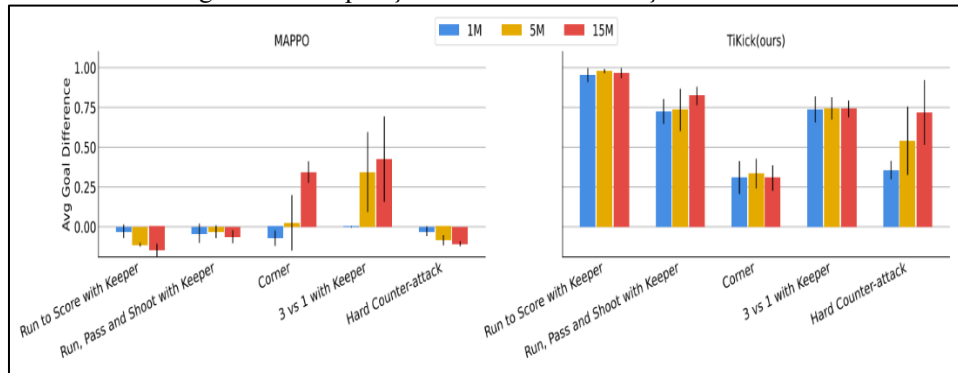
Lin *et al.* (2023) acrescentaram várias inovações em técnicas existentes para treinar agentes fortes para o modo de jogo GFootball 11 vs. 11 a partir do zero. Os autores concluem que o TiZero consegue superar todos os sistemas, por uma grande margem, em termos de taxa de vitória e diferença de gols, utilizando comportamentos de coordenação mais complexos do que seus anteriores. Por fim, os autores sugerem a aplicação do sistema para outros domínios e ambientes com cenários mais desafiadores.

2.3 TIKICK: TOWARDS PLAYING MULTI-AGENT FOOTBALL FULL GAMES FROM SINGLE-AGENT DEMONSTRATIONS

Huang *et al.* (2021) propuseram uma abordagem para jogar partidas completas de futebol multiagentes a partir de demonstrações de um único agente. Eles buscavam resolver o desafio de treinar múltiplos agentes para jogar juntos como uma equipe em um ambiente dinâmico e complexo. Os autores combinaram técnicas de aprendizado por imitação e aprendizado por reforço, utilizando uma rede neural para modelar a política do agente e um ambiente de simulação para testar a eficácia da abordagem.

Huang *et al.* (2021) também elaboraram uma abordagem baseada em hierarquia para lidar com o problema da coordenação entre os agentes. Segundo os autores, tal abordagem treina uma política global que define a estratégia da equipe e políticas individuais para cada agente levando em consideração a interação entre eles. A média da diferença de gols em todos os cenários acadêmicos para o MAPPO (esquerda) e TiKick (direita). Os resultados mostram que o método proposto alcança pontuações melhores em todos os cenários, conforme pode ser visto na Figura 4.

Figura 4 – Comparação do MAPPO em relação ao TiKick



Fonte: Huang *et al.* (2021).

Segundo Huang *et al.* (2021), o método possui seis *baselines*, cada uma aprimorando a etapa anterior:

- CQL: CQL (Kumar *et al.*, 2020) é um algoritmo de aprendizado por reforço off-line que tenta aprender valores Q conservadores adicionando penalidades nas funções Q;
- MABC: o algoritmo de clonagem de comportamento multiagente e usa uma perda supervisionada ingênua;
- α -Balance: adiciona um peso de equilíbrio α à baseline MABC para aliviar o problema do desequilíbrio de classes;
- Min Build-in: adiciona uma perda de minimização de ação embutida à baseline anterior para forçar o jogador designado a escolher a ação não embutida;
- Buffer Ranking: adiciona o truque de classificação de buffer para aproveitar as boas experiências passadas;
- Advantage Weight: adiciona uma perda ponderada por vantagem à baseline anterior e obtém a perda final. Consiste no método utilizado no modelo final do TiKick.

De acordo com Huang *et al.* (2021), os resultados dos experimentos mostraram que a abordagem proposta é capaz de treinar múltiplos agentes para jogar partidas completas de futebol multiagentes a partir de demonstrações de um único agente. A abordagem hierárquica proposta pelos autores também se mostrou eficaz para lidar com o problema da coordenação entre os agentes. Huang *et al.* (2021) também apresentaram uma análise mais detalhada dos resultados, incluindo uma análise de sensibilidade dos hiperparâmetros e uma análise da robustez da abordagem em relação a diferentes cenários de jogo. Segundo os autores, os resultados mostram que a abordagem proposta é capaz de lidar com diferentes configurações do ambiente e é robusta em relação a variações na densidade do agente e na complexidade do ambiente.

Huang *et al.* (2021) ressaltam que a abordagem hierárquica se mostrou eficaz para lidar com o problema da coordenação entre os agentes. No entanto, os autores também apontam algumas limitações em sua abordagem. Em particular, a abordagem requer que o agente aprenda com um único demonstrador, o que pode limitar a variedade de estratégias aprendidas pelos agentes. Além disso, a abordagem é específica para o ambiente de futebol multiagentes e pode não ser diretamente aplicável a outros ambientes multiagentes.

Por fim, Huang *et al.* (2021) sugerem como extensão a incorporação de mais fontes de informação, como informações de vídeo, e a exploração de técnicas de aprendizado por reforço com curadoria, que podem ajudar a melhorar a diversidade de estratégias aprendidas pelos agentes.

3 PROPOSTA DE MODELO

Esta seção apresenta a justificativa para o desenvolvimento deste trabalho, que tem como objetivo simular futebol de robôs no ambiente NetLogo utilizando sistemas multiagentes. A metodologia adotada para o desenvolvimento do modelo será detalhada, bem como os requisitos necessários para a sua implementação.

3.1 JUSTIFICATIVA

No Quadro 1 é possível observar as principais características de cada trabalho correlato apresentado, onde as colunas são cada trabalho e as linhas as características, com elas podemos entender como será solucionado os problemas propostos por este trabalho.

Quadro 1 – Comparativo dos trabalhos correlatos

Características	Trabalhos correlatos	Omidshafei <i>et al.</i> (2022)	Lin <i>et al.</i> (2023)	Huang <i>et al.</i> (2021)
Objetivo		Aprimorar o comportamento dos multiagentes fora do campo de visão	Desenvolver um algoritmo estado-da-arte voltado ao futebol de multiagentes	Aplicar AWR ao futebol de multiagentes
Arquitetura de modelagem dos SMAs		Graph Imputer	JRPO	MAPPO
Linguagem de programação		Não informou	Python	Python
Cenários testados		Comparação com dados reais utilizando Pitch Control	Partidas 11 vs 11, 5 minutos	1, 2 e 3 vs goleiro Escanteios, Contra-ataque
Forma de interação dos agentes		LSTM bidirecional Graph Network, VRNN	Recompensa	Recompensa
Quantidade de repetições por simulação		Não informou	3000	3000

Fonte: elaborado pelo autor.

A partir do Quadro 1 pode-se observar que Huang *et al.* (2021) propuseram uma abordagem mais simples e com menor complexidade computacional em comparação com TiZero. Eles apresentam um sistema de aprendizado por demonstração em que um único agente aprende a jogar futebol multiagente por meio de demonstrações de um jogador experiente. Os autores destacam que os resultados são promissores, embora ainda haja espaço para melhorias em termos de desempenho.

Em relação ao trabalho de Lin *et al.* (2023), é importante destacar que ele apresenta uma abordagem inovadora para treinamento de agentes em ambientes complexos de futebol multiagente, utilizando técnicas de aprendizado por reforço distribuído e autojogo. Os resultados obtidos mostraram que o TiZero superou amplamente os sistemas anteriores em termos de taxa de vitórias e diferença de gols, além de utilizar comportamentos de coordenação mais complexos.

Por fim, Omidshafei *et al.* (2022) apresentam uma abordagem diferente dos dois primeiros trabalhos, focando na predição de comportamentos fora da tela em jogos de futebol multiagente. Os autores propõem um modelo de aprendizado profundo que utiliza informações de rastreamento de jogadores para prever o comportamento de um jogador fora da tela, com resultados positivos. Em resumo, os três trabalhos apresentam abordagens diferentes para lidar com desafios em ambientes de futebol multiagente, cada um com suas vantagens e limitações. TiZero apresenta um alto desempenho, TiKick propõe uma abordagem mais simples e com menor complexidade computacional, enquanto o terceiro artigo foca na predição de comportamentos fora da tela.

Diante deste contexto, este trabalho é relevante por sua contribuição ao avanço da pesquisa em Sistemas Multiagentes (SMA) aplicado à robótica, especialmente em jogos de futebol de robôs. A implementação de uma arquitetura de SMA para coordenação das ações dos robôs em um jogo de futebol pode levar a uma melhoria significativa no desempenho do jogo, possibilitando a participação de equipes mais eficientes e com maior capacidade de adaptação em competições de robótica. O estudo também pode contribuir para o desenvolvimento

de sistemas robóticos mais flexíveis, adaptativos e autônomos, capazes de lidar com incertezas e ambientes dinâmicos. Além disso, a aplicação de SMA em jogos de futebol de robôs pode contribuir quanto a coordenação de múltiplos agentes em outros domínios, como em sistemas de transporte inteligente, logística e redes de sensores distribuídos.

3.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

O trabalho proposto deverá contemplar os seguintes Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF):

- permitir que o ambiente de jogo seja configurável, permitindo a definição de diferentes tipos de campo, tamanho, obstáculos, entre outros fatores que podem influenciar o desempenho dos robôs (RF);
- permitir a criação de equipes de robôs, com diferentes características, tais como velocidade, agilidade, habilidade de passe etc (RF);
- permitir que os robôs sejam capazes de se comunicar entre si, trocando informações sobre a posição da bola, posição dos adversários etc (RF);
- permitir a implementação de diferentes estratégias de jogo, tais como táticas de defesa, ataque, marcação etc (RF);
- ter uma interface com o usuário intuitiva e amigável, permitindo a fácil configuração dos parâmetros do jogo, controle da partida e visualização das estatísticas (RF);
- ser capaz de simular partidas em tempo real, garantindo a fluidez da experiência do usuário (RNF);
- utilizar o paradigma de programação orientada a agentes (RNF);
- ser desenvolvido na linguagem de programação Python e na ferramenta NetLogo (RNF).

3.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- levantamento bibliográfico: escolher trabalhos correlatos e entender os assuntos relacionados à futebol de robôs e SMAs;
- definição do cenário de simulação: identificar os aspectos relevantes (estado global, dinâmicas globais/ entidades locais) que representem o ambiente dos agentes;
- definição de parâmetros iniciais: determinar as informações do ambiente, as quais devem ser fornecidas aos agentes para que estes possam selecionar as ações a serem praticadas;
- modelagem dos agentes: determinar das ações básicas dos agentes, e suas reações às entidades do ambiente;
- definição e escolha da arquitetura de agentes: pesquisar as principais arquiteturas que possibilitem explorar a conexão entre o comportamento ao nível micro dos indivíduos e os padrões ao nível macro, escolhendo o mais adequado para o desenvolvimento do trabalho;
- implementação: desenvolver o artefato computacional de simulação multiagente a partir dos itens (b) até (e), considerando inicialmente a arquitetura BDI, utilizando a linguagem de programação Python e a ferramenta NetLogo;
- validação e calibragem: avaliar o comportamento e capacidade dos agentes em se adaptar a diferentes situações e estratégias de jogo. Para isso, serão utilizadas métricas relevantes para futebol de robôs, como a taxa de sucesso em marcar gols, a precisão na tomada de decisão e a eficiência na coordenação de múltiplos robôs.

As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 2 – Cronograma de atividades a serem realizadas

etapas / quinzenas	2023									
	jul.		ago.		set.		out.		nov.	
	1	2	1	2	1	2	1	2	1	2
levantamento bibliográfico										
definição do cenário de simulação										
definição de parâmetros iniciais										
modelagem do agente										
definição e escolha da arquitetura de agentes										
implementação										
validação e calibragem										

Fonte: elaborado pelo autor.

4 REVISÃO BIBLIOGRÁFICA

Nesta seção serão apresentados os principais conceitos que fundamentam o estudo proposto. A subseção 4.1 discorre sobre futebol de robôs. Por fim, a subseção 4.2 fundamenta sistemas multiagentes.

4.1 FUTEBOL DE ROBÔS

Futebol de robôs, uma fascinante interseção entre esportes e tecnologia, tem sido objeto de extensa pesquisa e desenvolvimento ao longo dos anos. A evolução do futebol de robôs não apenas levou a avanços na robótica, mas também contribuiu para o campo mais amplo da Inteligência Artificial (IA). A primeira geração de jogadores de futebol de robôs, conforme descrito por Kitano *et al.* (1997), foi projetada para participar da RoboCup, uma competição internacional de robótica. O objetivo da competição era promover a pesquisa em robótica, oferecendo um desafio publicamente atraente, mas formidável. Os robôs foram projetados para operar totalmente autonomamente e tomar decisões estratégicas em tempo real, demonstrando o potencial da IA em ambientes dinâmicos (KITANO *et al.*, 1997).

À medida que o campo avançava, os pesquisadores começaram a se concentrar nos aspectos cooperativos do futebol de robôs. Stone *et al.* (2005) desenvolveram um método para sistemas multiagentes, onde cada jogador de robô, equipado com sua própria IA, trabalhava em equipe. Essa pesquisa contribuiu para a compreensão de como entidades individuais de IA podem trabalhar juntas para alcançar um objetivo comum, um conceito aplicável além do futebol de robôs para áreas como veículos autônomos e robótica de enxame (STONE *et al.*, 2005).

Segundo Laue *et al.* (2008), o desenvolvimento do futebol de robôs também levou a avanços no design e nos sistemas de controle de robôs. Por exemplo, os autores citam o desenvolvimento de uma nova abordagem para a locomoção omnidirecional em robôs humanoides, o que permitiu que os robôs se movessem de forma mais eficiente no campo de futebol. Essa pesquisa não apenas melhorou o desempenho dos jogadores de futebol de robôs, mas também contribuiu para o campo mais amplo do design de robôs humanoides.

Mais recentemente, os pesquisadores começaram a explorar o conceito de "digital twins" no contexto do futebol de robôs. Um estudo realizado por Houtman *et al.* (2019) focou no desenvolvimento de um "digital twin" para um robô jogador de futebol, especificamente os robôs Turtle soccer. Os autores descobriram que o reuso de artefatos fornecia uma boa base para o desenvolvimento inicial do Turtle Digital Twin. Eles também destacaram a importância dos casos de uso no direcionamento do desenvolvimento de um "Digital Twin", em vez de buscar a maior semelhança possível com a realidade. Essa percepção foi especialmente importante em seu trabalho com os Turtles, onde o objetivo era visualizar um jogo de futebol em 3D.

De acordo com Houtman *et al.* (2019), existem diversas abordagens para a implementação de um time de futebol de robôs, que variam desde técnicas baseadas em visão computacional até técnicas baseadas em aprendizado de máquina. Dentre as principais, encontram-se:

- a) técnicas de visão computacional: Essas técnicas envolvem o uso de câmeras e algoritmos de processamento de imagem para identificar a bola e os jogadores no campo. Por meio da detecção e rastreamento da bola e dos jogadores, os robôs podem obter informações visuais para tomar decisões e coordenar suas ações;
- b) aprendizado de máquina: Essa abordagem envolve a criação de modelos de aprendizado de máquina que podem aprender a jogar futebol com base em exemplos. Os robôs podem ser treinados usando algoritmos de aprendizado, como redes neurais, para aprender a tomar decisões e executar ações apropriadas durante o jogo;
- c) técnicas de planejamento de trajetória: Essas técnicas visam planejar trajetórias para os robôs no campo, levando em consideração a posição da bola, dos jogadores e dos obstáculos. O objetivo é definir o caminho mais eficiente para que os robôs se movam e executem suas ações, como chutar a bola ou posicionar-se defensivamente;
- d) técnicas de controle de robôs: Essas técnicas se concentram no desenvolvimento de algoritmos de controle que permitem aos robôs executar movimentos precisos e coordenados. Isso envolve o controle dos motores e atuadores dos robôs, garantindo que eles se movam de acordo com as instruções recebidas;
- e) técnicas de tomada de decisão: Essas técnicas envolvem o desenvolvimento de algoritmos que permitem aos robôs tomar decisões inteligentes durante o jogo. Isso inclui decidir quando atacar, defender, passar a bola ou posicionar-se estrategicamente no campo, levando em consideração o estado atual do jogo e as informações disponíveis.

Segundo Houtman *et al.* (2019), essas técnicas representam algumas das áreas de pesquisa essenciais em futebol de robôs, buscando aprimorar a percepção, o planejamento, o controle e a tomada de decisão dos robôs, a fim de melhorar seu desempenho em partidas de futebol de forma simples ou através de sistemas multiagentes.

4.2 SISTEMAS MULTIAGENTES

De acordo com Hübner (1995), os sistemas multiagentes (SMA) são um subcampo da inteligência artificial que se concentra na coordenação e cooperação de múltiplos agentes autônomos. Esses agentes podem ser entidades físicas, como robôs, ou entidades virtuais, como programas de software. Eles interagem entre si para alcançar objetivos individuais ou coletivos, muitas vezes em ambientes complexos e dinâmicos. Este ensaio

explorará as complexidades dos sistemas multiagentes, suas aplicações e os desafios enfrentados em sua implementação.

Segundo com Hübner (1995), as bases teóricas dos sistemas multiagentes estão enraizadas nos conceitos de agentes inteligentes e suas interações. De acordo com Wooldridge e Jennings (1995), um agente inteligente é um sistema que possui autonomia, habilidade social, reatividade e proatividade. Em um sistema multiagente, esses agentes inteligentes interagem entre si em um ambiente compartilhado, resultando em dinâmicas complexas. O estudo dessas dinâmicas, incluindo questões de cooperação, competição e coordenação entre agentes, constitui o cerne da teoria dos sistemas multiagentes.

Stone e Veloso (2000) destacam o uso de sistemas multiagentes em domínios como resolução distribuída de problemas, comércio eletrônico e robótica. Na resolução distribuída de problemas, por exemplo, os sistemas multiagentes podem ajudar a coordenar as ações de vários solucionadores de problemas para alcançar um objetivo comum. No comércio eletrônico, eles podem ser usados para modelar e analisar o comportamento de compradores e vendedores. Na robótica, os sistemas multiagentes podem possibilitar a coordenação de múltiplos robôs, resultando em conclusão de tarefas mais eficiente e efetiva.

Apesar de seu potencial, Zhang *et al.* (2020), a implementação de sistemas multiagentes apresenta vários desafios. Os autores ainda destacam as questões relacionadas à coordenação, comunicação e tomada de decisões entre os agentes. A presença de incerteza no ambiente ou nas ações dos agentes pode complicar o projeto e a operação dos sistemas multiagentes, além da necessidade de técnicas robustas de otimização para lidar com os objetivos conflitantes que muitas vezes surgem em sistemas multiagentes.

Segundo Omidshafiei *et al.* (2022), uma das técnicas usadas para abordar os desafios em sistemas multiagentes é a Multi Agent Collaborative Search (MASC). De acordo com os autores, essa abordagem pode ajudar a melhorar o desempenho do sistema multiagente encontrando um conjunto de soluções que representam um compromisso entre os objetivos conflitantes., além da necessidade de incorporar informações de preferência ao processo de otimização para orientar a busca pelas soluções mais preferidas.

Beard e Ren (2005) apontam que outro aspecto importante dos sistemas multiagentes é o rastreamento de consenso. Os autores apontam que isso envolve garantir que todos os agentes no sistema concordem com certas variáveis ou estados, o que é crucial para sua ação coordenada. Eles destacam que alcançar o consenso em um sistema multiagente pode ser desafiador devido à natureza dinâmica do sistema e às incertezas no ambiente. Beard e Ren (2005) discutem vários protocolos de consenso e suas propriedades de estabilidade, destacando a importância de escolher o protocolo certo para um determinado sistema multiagente.

De acordo com Juchem e Bastos (2001), a estrutura e a arquitetura de modelagem de agentes podem variar de acordo com os objetivos desse agente. A primeira coisa a se fazer para definir qual arquitetura utilizar está baseada em entender qual o tipo de agente do qual se está tratando. Os agentes podem ser cognitivos, que podem ser considerados racionais e tem a capacidade de escolher as ações a executar, dentre as existentes em seu repertório, para chegar em seu objetivo, bem como podem ser reativos, que não necessariamente são agentes racionais, pois eles podem ter seu comportamento definido através de um padrão estímulo-resposta (JUCHEM; BASTOS, 2001). A partir dos dois tipos de agentes descritos, é possível desenvolver uma arquitetura cognitiva (ou deliberativa), que adota o princípio de que agentes são racionais, definindo assim que os agentes cognitivos têm suas deliberações feitas por meio de um processo baseado em raciocínio lógico, escolhendo sua ação por meio da representação simbólica do mundo, um plano e uma função utilizada para a ação. Também pode-se desenvolver uma arquitetura funcional, na qual o agente é composto por módulos que representam cada uma das funcionalidades necessárias para sua operação. Outra arquitetura tradicional muito comum é a arquitetura BDI, que considera três estados mentais do agente: Crença, Desejo e Interação (*Belief, Desire and Intention*). A arquitetura BDI permite que o agente possa decidir, momento a momento, qual ação desempenhar na direção de seus objetivos (JUCHEM; BASTOS, 2001).

REFERÊNCIAS

BEARD, Randall; REN, Wei. Consensus seeking in multiagent systems under dynamically changing interaction topologies. **IEEE Transactions on Automatic Control**, Utah, v. 50, n. 5, p. 655-661, 2005.

CHEN, Liang *et al.* Survey of Multi-Agent Strategy Based on Reinforcement Learning. **Chinese Control And Decision Conference (CCDC)**, Hefei, p. 604-609, 2020.

GUPTA, Sen *et al.* **Recent Advances in Robotics and Automation**. 2013. Disponível em: <https://www.researchgate.net/publication/321613321_Recent_Advances_in_Robotics_and_Automation>. Acesso em: 01 maio 2023.

HÜBNER, Jomi F. Migração de agentes em sistemas multi-agentes abertos. 1995. 126 f. Dissertação (Mestrado em Ciência da Computação) - Programa de Pós-Graduação em Ciência da Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre.

HUANG, Shiyu *et al.* **TiKick: Toward Playing Multi-agent Football Full Games from Single-agent Demonstrations**. 2021. Disponível em: <<https://arxiv.org/abs/2110.04507>>. Acesso em: 16 abr. 2023.

- HOUTMAN, Wouter *et al.* **Tech United Eindhoven Team Description**. Eindhoven, Holanda. 2019 Disponível em: <<https://tdp.robocup.org/wp-content/uploads/tdp/robocup/2019/robocupsoccer-middle-size/tech-united-eindhoven-65/robocup-2019-robocupsoccer-middle-size-tech-united-eindhoven5tvhgXklBF.pdf>>. Acesso em 24 jun. 2023.
- JUCHEM, Murilo; BASTOS, Ricardo Melo. **Arquitetura de Agentes**. Relatório Técnico, n. 013 arquivado na Pró-Reitoria de Pesquisa, Faculdade de Informática PUCRS, Porto Alegre. 2001.
- KITANO, Hiroaki *et al.* RoboCup: The Robot World Cup Initiative. **AGENTS '97: Proceedings of the first international conference on Autonomous agents**, Nagoya, p. 340-347. 1997.
- KUMAR, Aviral *et al.* Conservative q-learning for offline reinforcement learning. **Advances in Neural Information Processing Systems**, v. 33, p. 1179-1191. 2020.
- LAUE, Tim *et al.* SimRobot – A General Physical Robot Simulator and Its Application in RoboCup. **RoboCup 2005: Robot Soccer World Cup IX**, p. 173-183. 2006.
- LIN, Fanqi *et al.* **TiZero: Mastering Multi-Agent Football with Curriculum Learning and Self-Play**. 2023. Disponível em: <<https://arxiv.org/abs/2302.07515>>. Acesso em 16 abr. 2023.
- OMIDSHAFIEI, Shayegan *et al.* Multiagent off-screen behavior prediction in football. **Scientific reports**, v. 12, n. 1, p. 1–13 2022.
- TISUE, Seth; WILENSKY, Uri. NetLogo: A simple environment for modeling complexity. International Conference on Complex Systems. **Agent 2004 Conference on Social Dynamics: Interaction, Reflexivity and Emergence**. 2004.
- STONE, Peter *et al.*, **Reinforcement Learning for RoboCup-Soccer Keepaway**, Adaptive Behavior, Texas, v. 13 n. 3. p. 165–188, 2005.
- STONE, Peter; VELOSO, Manuela. Multiagent Systems: A Survey from a Machine Learning Perspective. **Autonomous Robots**, Texas, v. 8, n. 3. p. 345–383. 2000.
- WOOLDRIDGE, Michael. **An Introduction to MultiAgent Systems**. 2 ed. Estados Unidos: Wiley, 2009.
- WOOLDRIDGE, Michael; JENNINGS, Nicholas. Intelligent Agents: Theory and Practice. **The Knowledge Engineering Review**, v. 10, n. 2, p. 115-152, 1995.
- ZHANG, Haifeng *et al.* Bi-level actor-critic for multi-agent coordination. **Proceedings of the AAAI Conference on Artificial Intelligence**. v. 34, n 5, p. 7325-7332, 2020.

FORMULÁRIO DE AVALIAÇÃO BCC – PROFESSOR AVALIADOR – PROJETO

Avaliador(a): **Miguel Alexandre Wisintainer**

Atenção: quando o avaliador marcar algum item como atende parcialmente ou não atende, deve obrigatoriamente indicar os motivos no texto, para que o aluno saiba o porquê da avaliação.

ASPECTOS AVALIADOS		Atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?	X		
	O problema está claramente formulado?	X		
	2. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?	X		
	Os objetivos específicos são coerentes com o objetivo principal?	X		
	3. TRABALHOS CORRELATOS São apresentados trabalhos correlatos, bem como descritas as principais funcionalidades e os pontos fortes e fracos?	X		
	4. JUSTIFICATIVA Foi apresentado e discutido um quadro relacionando os trabalhos correlatos e suas principais funcionalidades com a proposta apresentada?	X		
	São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?	X		
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?	X		
	5. REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO Os requisitos funcionais e não funcionais foram claramente descritos?	X		
	6. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?	X		
	Os métodos, recursos e o cronograma estão devidamente apresentados e são compatíveis com a metodologia proposta?	X		
	7. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?	X		
	As referências contemplam adequadamente os assuntos abordados (são indicadas obras atualizadas e as mais importantes da área)?	X		
ASPECTOS METODOLÓGICOS	8. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?	X		
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?	X		

O projeto de TCC será reprovado se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos **4 (quatro)** itens dos **ASPECTOS** tiverem resposta ATENDE PARCIALMENTE.

PARECER: (X) APROVADO () REPROVADO

CURSO DE CIÊNCIA DA COMPUTAÇÃO – TCC		
() PRÉ-PROJETO	(X) PROJETO	ANO/SEMESTRE: 2023/1

SISTEMAS MULTIAGENTES APLICADO AO FUTEBOL DE ROBÔS

Christian Trisotto Alegri

Prof. Aurélio Faustino Hoppe – Orientador

5 INTRODUÇÃO

A robótica tem evoluído significativamente desde sua criação, há mais de 60 anos. Inicialmente, as máquinas eram projetadas para executar tarefas simples e repetitivas na indústria, mas hoje em dia, robôs são utilizados em uma variedade de áreas, incluindo saúde, segurança, exploração espacial, entre outras. Segundo Gupta (2013), as tecnologias de robótica têm avançado rapidamente nas últimas décadas, com o surgimento de novas técnicas de fabricação, sensores, atuadores, sistemas de controle e algoritmos de inteligência artificial, permitindo que robôs se tornem cada vez mais sofisticados e autônomos. A robótica tem o potencial de transformar radicalmente a maneira como as pessoas vivem e trabalham, e sua evolução continuará a impulsionar novas possibilidades e aplicações no futuro.

Dentro do campo interdisciplinar da robótica, uma área em rápido crescimento é a dos sistemas multiagentes, que são compostos por vários agentes autônomos que cooperam ou competem uns com os outros para atingir um objetivo comum. Um exemplo particularmente interessante de aplicação de sistemas multiagentes é o futebol de robôs, que tem o potencial de ser um desafio altamente complexo e desafiador para a inteligência artificial e a robótica. Segundo Huang *et al.* (2021), o futebol de robôs é uma das aplicações mais ambiciosas de sistemas multiagentes, que exige a integração de diversas habilidades e estratégias de tomada de decisão em um ambiente altamente dinâmico e imprevisível. O uso de sistemas multiagentes no futebol de robôs pode ajudar a resolver alguns dos principais desafios enfrentados pelos robôs individuais, como a limitação de sua percepção do ambiente e a dificuldade de tomada de decisões em tempo real.

Wooldridge (2009) destaca que um dos campos de aplicação mais fascinantes dessa tecnologia é o esporte. É nesse contexto que surge o futebol de robôs, uma competição que coloca em jogo a criatividade e a engenhosidade dos pesquisadores e entusiastas da robótica. O futebol de robôs é uma competição que envolve equipes de robôs jogando futebol em um campo especialmente projetado. Essa competição tem como objetivo incentivar o desenvolvimento de sistemas robóticos capazes de jogar futebol de forma autônoma. Para alcançar esse objetivo, os pesquisadores utilizam técnicas avançadas de inteligência artificial, como a visão computacional e os sistemas multiagentes.

De acordo com Chen *et al.* (2020), os Sistemas Multiagentes (SMA) são particularmente importantes no futebol de robôs, pois permitem que os robôs se comuniquem entre si e com o ambiente, coordenando suas ações para alcançar os objetivos do jogo. Esses sistemas são compostos por vários agentes autônomos que interagem entre si e com o ambiente em busca de alcançar os objetivos específicos do jogo. No entanto, a aplicação de sistemas multiagentes no futebol de robôs ainda enfrenta vários desafios, como a dinamicidade e imprevisibilidade do jogo, além das situações de conflito que podem ocorrer entre os robôs durante a partida. Por isso, é importante avaliar se os sistemas multiagentes podem ser utilizados de forma eficiente no futebol de robôs.

Tisue e Wilensky (2004) afirma que o desenvolvimento de simuladores é uma das principais ferramentas utilizadas para avaliar o desempenho de sistemas robóticos em diversas áreas de aplicação. No futebol de robôs, os simuladores são particularmente importantes para testar e avaliar novas estratégias e abordagens de forma segura e controlada. Dentre os simuladores utilizados para essa finalidade, destaca-se o NetLogo, uma plataforma de modelagem e simulação de sistemas complexos baseada em agentes. O NetLogo permite modelar e simular sistemas multiagentes de forma intuitiva e visual, além de possuir uma ampla variedade de ferramentas de análise.

Diante desse contexto, a pergunta de pesquisa deste trabalho é: sistemas multiagentes podem ser utilizados de forma eficiente no futebol de robôs, considerando as características dinâmicas e imprevisíveis do jogo e as situações de conflito que podem ocorrer entre os robôs?

5.1 OBJETIVOS

O objetivo deste trabalho é utilizar a plataforma NetLogo para simular futebol de robôs a partir de sistemas multiagentes, com o propósito de avaliar diferentes estratégias de jogo e estudar o comportamento dos agentes em um ambiente controlado.

Os objetivos específicos são:

- f) modelar cada um dos elementos do jogo de futebol como um agente, definindo seu conhecimento,

- funções, comportamento e modos de interação no NetLogo;
- g) analisar o comportamento dos agentes em diferentes situações de jogo, como jogadas de ataque, defesa, cobranças de falta, entre outras;
- h) avaliar o desempenho do modelo em simulações e testes reais, utilizando métricas relevantes para futebol de robôs, como a taxa de sucesso em marcar gols, a precisão na tomada de decisão e a eficiência na coordenação de múltiplos robôs.

6 TRABALHOS CORRELATOS

Nesta seção são apresentados trabalhos com características semelhantes aos principais objetivos do estudo proposto. A subseção 2.1 apresenta um modelo para predição do comportamento dos jogadores (OMIDSHAFEI *et al.*, 2022). Na subseção 2.2 é descrito um método para treinar times de agentes virtuais a jogar futebol em alto nível (LIN *et al.*, 2023). Por fim, a subseção 2.3 apresenta uma abordagem para jogar partidas completas de futebol multiagentes a partir de demonstrações de um único agente (HUANG *et al.*, 2021).

6.1 MULTIAGENT OFF-SCREEN BEHAVIOR PREDICTION IN FOOTBALL

Omidshafei *et al.* (2022) abordam o problema de imputação de séries temporais multiagente no contexto do futebol, onde subconjuntos de jogadores podem entrar e sair do campo de visão das imagens de vídeo transmitidas, enquanto os jogadores não observados continuam a interagir fora da tela. O objetivo dos autores era desenvolver um método de aprendizado de máquina que pudesse usar observações passadas e futuras de subconjuntos de jogadores de futebol para estimar as observações ausentes dos jogadores fora da tela.

A abordagem de Omidshafei *et al.* (2022), chamada de Graph Imputer, é um método de aprendizado de máquina que combina redes de grafo e *autoencoders* variacionais para permitir a aprendizagem de uma distribuição de trajetórias imputadas para jogadores de futebol fora da tela. O método envolve várias etapas. Primeiro, os autores constroem um grafo que representa as interações entre jogadores observados fora da tela. Os nós no grafo representam os jogadores e as arestas representam as interações entre eles. Em seguida, os autores treinam um AutoEncoder Variacional (AEV) nas trajetórias observadas dos jogadores para aprender uma representação de baixa dimensão do comportamento dos jogadores. O AEV é treinado para reconstruir as trajetórias observadas e gerar novas trajetórias que são semelhantes às observadas. Por fim, os autores usam a representação aprendida pelo AEV para codificar as trajetórias observadas dos jogadores e gerar trajetórias imputadas para os jogadores fora da tela. As trajetórias imputadas são geradas demonstrando a distribuição de comportamento dos jogadores aprendida condicionada às trajetórias observadas. A Figura 1 apresenta uma sequência de trajetórias geradas a partir do AEV.

Figura 5 – Visualização das trajetórias



Fonte: Omidshafei *et al.* (2022).

Para testar a eficácia de sua abordagem proposta, Omidshafei *et al.* (2022) realizaram experimentos a partir de jogos de futebol do mundo real. Eles utilizaram o módulo de câmera para simular a estimativa do estado dos jogadores fora da tela. Os autores avaliaram o método usando o *framework* de controle de campo (*pitch control*), ao qual depende de dados totalmente observados. Segundo Omidshafei *et al.* (2022), o controle de campo é um método para avaliar a habilidade de uma equipe em controlar o espaço no campo e restringir os movimentos da equipe adversária. Além disso, os autores também compararam o método com várias abordagens de ponta, incluindo aquelas feitas manualmente para o futebol, e mostraram que o método superou todas as outras abordagens em todas as métricas consideradas, conforme pode ser visto na Figura 2.

Figura 6 – Comparação com outros métodos

	Model	Pitch control MAE
Restricted	Role-invariant VRNN	0.0447 \pm 0.0204
	Bidir. Role-invariant VRNN	0.0209 \pm 0.0096
General	LSTM	0.0592 \pm 0.0274
	Bidir. LSTM	0.0296 \pm 0.0139
	Social LSTM ⁷	0.0474 \pm 0.0216
	Bidir. Social LSTM	0.0219 \pm 0.0098
	GVRNN ¹²	0.0418 \pm 0.0189
	Graph Imputer (Ours)	0.0207 \pm 0.0094

Fonte: Omidshafei *et al.* (2022).

Omidshafei *et al.* (2022) concluem que a abordagem é eficaz na previsão do comportamento de jogadores fora da tela em jogos de futebol. Os autores também demonstraram que o método supera várias abordagens de ponta, incluindo aquelas feitas exclusivamente para o futebol. Porém, ressaltam que o método depende de dados das trajetórias verdadeiras, que nem sempre estão disponíveis na prática. Omidshafei *et al.* (2022) sugerem avaliar a eficácia do método em configurações mais realistas, onde apenas observações parciais do comportamento dos jogadores estão disponíveis. Como possíveis expansões do método, apontam a necessidade de incorporação de fontes adicionais de informação, como atributos de jogadores ou formações de equipe, para melhorar a precisão das previsões. Os autores também sugeriram que o método poderia ser aplicado a outros domínios, como dinâmica de multidões ou robótica, onde sistemas multiagentes são comuns.

6.2 TIZERO: MASTERING MULTI-AGENT FOOTBALL WITH CURRICULUM LEARNING AND SELF-PLAY

Lin *et al.* (2023) desenvolveram um método para treinar times de agentes virtuais a jogar futebol em alto nível. Os autores combinaram duas técnicas de aprendizado de máquina: aprendizado de currículo e autojogo. O aprendizado de currículo envolve aumentar gradualmente a dificuldade das tarefas que os agentes devem executar, enquanto o autojogo envolve fazê-los jogar uns contra os outros para melhorar suas estratégias ao longo do tempo. Ao utilizar essas técnicas, os autores esperavam desenvolver agentes que pudessem cooperar e competir efetivamente como um time, melhorando o desempenho geral do jogo de futebol.

Segundo Lin *et al.* (2023), na abordagem de aprendizado de currículo aumentou-se gradualmente a dificuldade das tarefas aos quais os agentes foram solicitados a executar, começando com tarefas mais simples, como passar e driblar, e progredindo para cenários mais complexos que exigiam coordenação e trabalho em equipe. De acordo com os autores, ao usar tal técnica, os agentes foram capazes de construir sobre suas experiências anteriores e melhorar gradualmente seu desempenho geral.

Lin *et al.* (2023) também utilizaram o autojogo para treinar seus agentes, onde cada agente jogava contra uma cópia de si mesmo com uma política ligeiramente modificada. Essa abordagem permitiu que os agentes aprendessem com suas próprias experiências e ajustassem suas políticas para melhorar seu desempenho ao longo do tempo. Neste caso, os autores utilizaram uma variante do algoritmo de Otimização de Política Próxima de Múltiplos Agentes (OPMA) para treinar seus agentes por meio do autojogo. Esse algoritmo permitiu que os agentes aprendessem com as experiências uns dos outros e ajustassem suas políticas com base em suas observações.

Além disso, Lin *et al.* (2023) introduziram uma abordagem inovadora para o autojogo que envolveu emparelhar os agentes de uma maneira que encorajava a cooperação em vez da competição. Essa abordagem, segundo os autores, resultou no desenvolvimento de agentes que poderiam trabalhar juntos efetivamente como um time, melhorando o desempenho do jogo de futebol. Em termos de resultados, Figura 3 apresenta a comparação do desempenho do TiZero em relação a vários *baselines*, incluindo TiKick, WeKick, JiDi_3rd, Built-in, Hard Rule-Based-1 e Rule-Based-2.

Figura 7 – Comparação TiZero em relação a outros métodos *baselines*

Metric	TiZero (Ours)	TiKick	WeKick	JiDi_3rd	Built-in Hard	Rule-Based-1	Rule-Based-2
Assist	1.30(1.02)	0.61(0.79)	0.20(0.47)	0.35(0.62)	0.20(0.55)	0.28(0.59)	0.22(0.53)
Pass	19.2(3.44)	6.99(2.71)	5.33(2.44)	3.96(2.33)	11.5(4.63)	7.28(2.77)	7.50(3.12)
Pass Rate	0.73(0.07)	0.65(0.17)	0.53(0.18)	0.44(0.19)	0.66(0.12)	0.64(0.17)	0.63(0.19)
Goal	3.42(1.69)	1.79(1.41)	0.88(0.88)	1.43(1.34)	0.52(0.91)	0.73(0.69)	0.64(0.82)
Goal Difference	2.27(1.93)	0.71(2.08)	-0.47(1.68)	-0.02(2.14)	-1.06(1.93)	-0.60(1.03)	-0.71(1.45)
Draw Rate (%)	8.50	22.2	29.0	23.2	24.8	28.7	27.8
Lose Rate (%)	6.50	23.5	44.2	33.8	59.6	48.2	49.5
Win Rate (%)	85.0	54.3	26.8	43.0	15.6	23.1	22.7
TrueSkill	45.2	37.2	30.9	35.0	24.9	28.2	27.1

Fonte: Lin *et al.* (2023).

Segundo Lin *et al.* (2023), o TiZero superou todos os outros métodos em termos de assistência, passe, taxa de passe, gol, diferença de gol, taxa de empate, taxa de derrota, taxa de vitória e TrueSkill. O TiZero apresentou uma média de assistências de 1,30, uma média de passes de 19,2 e uma taxa de passe média de 0,73. Em termos de gols, TiZero marcou uma média de 3,42, com a segunda melhor pontuação de TiKick sendo apenas 1,79. Além disso, também teve a maior diferença de gols, com uma média de 2,27.

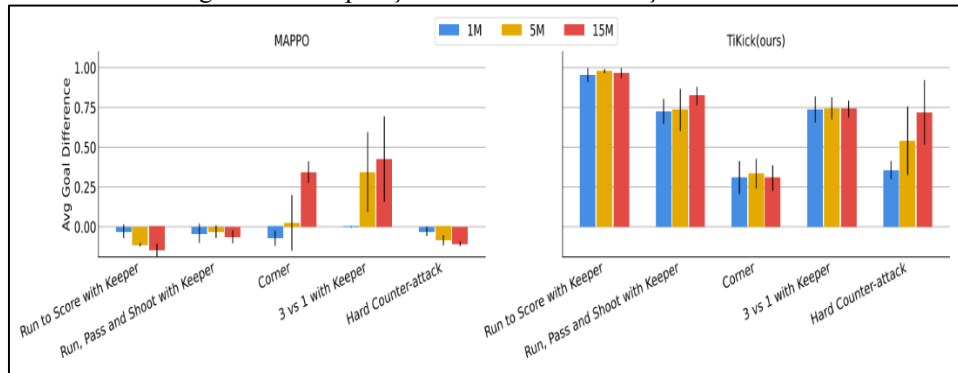
Lin *et al.* (2023) acrescentaram várias inovações em técnicas existentes para treinar agentes fortes para o modo de jogo GFootball 11 vs. 11 a partir do zero. Os autores concluem que o TiZero consegue superar todos os sistemas, por uma grande margem, em termos de taxa de vitória e diferença de gols, utilizando comportamentos de coordenação mais complexos do que seus anteriores. Por fim, os autores sugerem a aplicação do sistema para outros domínios e ambientes com cenários mais desafiadores.

6.3 TIKICK: TOWARDS PLAYING MULTI-AGENT FOOTBALL FULL GAMES FROM SINGLE-AGENT DEMONSTRATIONS

Huang *et al.* (2021) propuseram uma abordagem para jogar partidas completas de futebol multiagentes a partir de demonstrações de um único agente. Eles buscavam resolver o desafio de treinar múltiplos agentes para jogar juntos como uma equipe em um ambiente dinâmico e complexo. Os autores combinaram técnicas de aprendizado por imitação e aprendizado por reforço, utilizando uma rede neural para modelar a política do agente e um ambiente de simulação para testar a eficácia da abordagem.

Huang *et al.* (2021) também elaboraram uma abordagem baseada em hierarquia para lidar com o problema da coordenação entre os agentes. Segundo os autores, tal abordagem treina uma política global que define a estratégia da equipe e políticas individuais para cada agente levando em consideração a interação entre eles. A média da diferença de gols em todos os cenários acadêmicos para o MAPPO (esquerda) e TiKick (direita). Os resultados mostram que o método proposto alcança pontuações melhores em todos os cenários, conforme pode ser visto na Figura 4.

Figura 8 – Comparação do MAPPO em relação ao TiKick



Fonte: Huang *et al.* (2021).

Segundo Huang *et al.* (2021), o método possui seis *baselines*, cada uma aprimorando a etapa anterior:

- i) CQL: CQL (Kumar *et al.*, 2020) é um algoritmo de aprendizado por reforço off-line que tenta aprender valores Q conservadores adicionando penalidades nas funções Q;
- j) MABC: o algoritmo de clonagem de comportamento multiagente e usa uma perda supervisionada ingênua;
- k) α -Balance: adiciona um peso de equilíbrio α à baseline MABC para aliviar o problema do desequilíbrio de classes;
- l) Min Build-in: adiciona uma perda de minimização de ação embutida à baseline anterior para forçar o jogador designado a escolher a ação não embutida;
- m) Buffer Ranking: adiciona o truque de classificação de buffer para aproveitar as boas experiências passadas;
- n) Advantage Weight: adiciona uma perda ponderada por vantagem à baseline anterior e obtém a perda final. Consiste no método utilizado no modelo final do TiKick.

De acordo com Huang *et al.* (2021), os resultados dos experimentos mostraram que a abordagem proposta é capaz de treinar múltiplos agentes para jogar partidas completas de futebol multiagentes a partir de demonstrações de um único agente. A abordagem hierárquica proposta pelos autores também se mostrou eficaz para lidar com o problema da coordenação entre os agentes. Huang *et al.* (2021) também apresentaram uma análise mais detalhada dos resultados, incluindo uma análise de sensibilidade dos hiperparâmetros e uma análise da robustez da abordagem em relação a diferentes cenários de jogo. Segundo os autores, os resultados mostram que a abordagem proposta é capaz de lidar com diferentes configurações do ambiente e é robusta em relação a variações na densidade do agente e na complexidade do ambiente.

Huang *et al.* (2021) ressaltam que a abordagem hierárquica se mostrou eficaz para lidar com o problema da coordenação entre os agentes. No entanto, os autores também apontam algumas limitações em sua abordagem. Em particular, a abordagem requer que o agente aprenda com um único demonstrador, o que pode limitar a variedade de estratégias aprendidas pelos agentes. Além disso, a abordagem é específica para o ambiente de futebol multiagentes e pode não ser diretamente aplicável a outros ambientes multiagentes.

Por fim, Huang *et al.* (2021) sugerem como extensão a incorporação de mais fontes de informação, como informações de vídeo, e a exploração de técnicas de aprendizado por reforço com curadoria, que podem ajudar a melhorar a diversidade de estratégias aprendidas pelos agentes.

7 PROPOSTA DE MODELO

Esta seção apresenta a justificativa para o desenvolvimento deste trabalho, que tem como objetivo simular futebol de robôs no ambiente NetLogo utilizando sistemas multiagentes. A metodologia adotada para o desenvolvimento do modelo será detalhada, bem como os requisitos necessários para a sua implementação.

7.1 JUSTIFICATIVA

No Quadro 1 é possível observar as principais características de cada trabalho correlato apresentado, onde as colunas são cada trabalho e as linhas as características, com elas podemos entender como será solucionado os problemas propostos por este trabalho.

Quadro 3 – Comparativo dos trabalhos correlatos

Características	Trabalhos correlatos	Omidshafei <i>et al.</i> (2022)	Lin <i>et al.</i> (2023)	Huang <i>et al.</i> (2021)
Objetivo		Aprimorar o comportamento dos multiagentes fora do campo de visão	Desenvolver um algoritmo estado-da-arte voltado ao futebol de multiagentes	Aplicar AWR ao futebol de multiagentes
Arquitetura de modelagem dos SMAs		Graph Imputer	JRPO	MAPPO
Linguagem de programação		Não informou	Python	Python
Cenários testados		Comparação com dados reais utilizando Pitch Control	Partidas 11 vs 11, 5 minutos	1, 2 e 3 vs goleiro Escanteios, Contra-ataque
Forma de interação dos agentes		LSTM bidirecional Graph Network, VRNN	Recompensa	Recompensa
Quantidade de repetições por simulação		Não informou	3000	3000

Fonte: elaborado pelo autor.

A partir do Quadro 1 pode-se observar que Huang *et al.* (2021) propuseram uma abordagem mais simples e com menor complexidade computacional em comparação com TiZero. Eles apresentam um sistema de aprendizado por demonstração em que um único agente aprende a jogar futebol multiagente por meio de demonstrações de um jogador experiente. Os autores destacam que os resultados são promissores, embora ainda haja espaço para melhorias em termos de desempenho.

Em relação ao trabalho de Lin *et al.* (2023), é importante destacar que ele apresenta uma abordagem inovadora para treinamento de agentes em ambientes complexos de futebol multiagente, utilizando técnicas de aprendizado por reforço distribuído e autojogo. Os resultados obtidos mostraram que o TiZero superou amplamente os sistemas anteriores em termos de taxa de vitórias e diferença de gols, além de utilizar comportamentos de coordenação mais complexos.

Por fim, Omidshafei *et al.* (2022) apresentam uma abordagem diferente dos dois primeiros trabalhos, focando na predição de comportamentos fora da tela em jogos de futebol multiagente. Os autores propõem um modelo de aprendizado profundo que utiliza informações de rastreamento de jogadores para prever o comportamento de um jogador fora da tela, com resultados positivos. Em resumo, os três trabalhos apresentam abordagens diferentes para lidar com desafios em ambientes de futebol multiagente, cada um com suas vantagens e limitações. TiZero apresenta um alto desempenho, TiKick propõe uma abordagem mais simples e com menor complexidade computacional, enquanto o terceiro artigo foca na predição de comportamentos fora da tela.

Diante deste contexto, este trabalho é relevante por sua contribuição ao avanço da pesquisa em Sistemas Multiagentes (SMA) aplicado à robótica, especialmente em jogos de futebol de robôs. A implementação de uma arquitetura de SMA para coordenação das ações dos robôs em um jogo de futebol pode levar a uma melhoria significativa no desempenho do jogo, possibilitando a participação de equipes mais eficientes e com maior capacidade de adaptação em competições de robótica. O estudo também pode contribuir para o desenvolvimento

de sistemas robóticos mais flexíveis, adaptativos e autônomos, capazes de lidar com incertezas e ambientes dinâmicos. Além disso, a aplicação de SMA em jogos de futebol de robôs pode contribuir quanto a coordenação de múltiplos agentes em outros domínios, como em sistemas de transporte inteligente, logística e redes de sensores distribuídos.

7.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

O trabalho proposto deverá contemplar os seguintes Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF):

- o) permitir que o ambiente de jogo seja configurável, permitindo a definição de diferentes tipos de campo, tamanho, obstáculos, entre outros fatores que podem influenciar o desempenho dos robôs (RF);
- p) permitir a criação de equipes de robôs, com diferentes características, tais como velocidade, agilidade, habilidade de passe etc (RF);
- q) permitir que os robôs sejam capazes de se comunicar entre si, trocando informações sobre a posição da bola, posição dos adversários etc (RF);
- r) permitir a implementação de diferentes estratégias de jogo, tais como táticas de defesa, ataque, marcação etc (RF);
- s) ter uma interface com o usuário intuitiva e amigável, permitindo a fácil configuração dos parâmetros do jogo, controle da partida e visualização das estatísticas (RF);
- t) ser capaz de simular partidas em tempo real, garantindo a fluidez da experiência do usuário (RNF);
- u) utilizar o paradigma de programação orientada a agentes (RNF);
- v) ser desenvolvido na linguagem de programação Python e na ferramenta NetLogo (RNF).

7.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- w) levantamento bibliográfico: escolher trabalhos correlatos e entender os assuntos relacionados à futebol de robôs e SMAs;
- x) definição do cenário de simulação: identificar os aspectos relevantes (estado global, dinâmicas globais/ entidades locais) que representem o ambiente dos agentes;
- y) definição de parâmetros iniciais: determinar as informações do ambiente, as quais devem ser fornecidas aos agentes para que estes possam selecionar as ações a serem praticadas;
- z) modelagem dos agentes: determinar das ações básicas dos agentes, e suas reações às entidades do ambiente;
- aa) definição e escolha da arquitetura de agentes: pesquisar as principais arquiteturas que possibilitem explorar a conexão entre o comportamento ao nível micro dos indivíduos e os padrões ao nível macro, escolhendo o mais adequado para o desenvolvimento do trabalho;
- bb) implementação: desenvolver o artefato computacional de simulação multiagente a partir dos itens (b) até (e), considerando inicialmente a arquitetura BDI, utilizando a linguagem de programação Python e a ferramenta NetLogo;
- cc) validação e calibragem: avaliar o comportamento e capacidade dos agentes em se adaptar a diferentes situações e estratégias de jogo. Para isso, serão utilizadas métricas relevantes para futebol de robôs, como a taxa de sucesso em marcar gols, a precisão na tomada de decisão e a eficiência na coordenação de múltiplos robôs.

As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 4 – Cronograma de atividades a serem realizadas

etapas / quinzenas	2023									
	jul.		ago.		set.		out.		nov.	
	1	2	1	2	1	2	1	2	1	2
levantamento bibliográfico										
definição do cenário de simulação										
definição de parâmetros iniciais										
modelagem do agente										
definição e escolha da arquitetura de agentes										
implementação										
validação e calibragem										

Fonte: elaborado pelo autor.

8 REVISÃO BIBLIOGRÁFICA

Nesta seção serão apresentados os principais conceitos que fundamentam o estudo proposto. A subseção 4.1 discute sobre futebol de robôs. Por fim, a subseção 4.2 fundamenta sistemas multiagentes.

8.1 FUTEBOL DE ROBÔS

Futebol de robôs, uma fascinante interseção entre esportes e tecnologia, tem sido objeto de extensa pesquisa e desenvolvimento ao longo dos anos. A evolução do futebol de robôs não apenas levou a avanços na robótica, mas também contribuiu para o campo mais amplo da Inteligência Artificial (IA). A primeira geração de jogadores de futebol de robôs, conforme descrito por Kitano *et al.* (1997), foi projetada para participar da RoboCup, uma competição internacional de robótica. O objetivo da competição era promover a pesquisa em robótica, oferecendo um desafio publicamente atraente, mas formidável. Os robôs foram projetados para operar totalmente autonomamente e tomar decisões estratégicas em tempo real, demonstrando o potencial da IA em ambientes dinâmicos (KITANO *et al.*, 1997).

À medida que o campo avançava, os pesquisadores começaram a se concentrar nos aspectos cooperativos do futebol de robôs. Stone *et al.* (2005) desenvolveram um método para sistemas multiagentes, onde cada jogador de robô, equipado com sua própria IA, trabalhava em equipe. Essa pesquisa contribuiu para a compreensão de como entidades individuais de IA podem trabalhar juntas para alcançar um objetivo comum, um conceito aplicável além do futebol de robôs para áreas como veículos autônomos e robótica de enxame (STONE *et al.*, 2005).

Segundo Laue *et al.* (2008), o desenvolvimento do futebol de robôs também levou a avanços no design e nos sistemas de controle de robôs. Por exemplo, os autores citam o desenvolvimento de uma nova abordagem para a locomoção omnidirecional em robôs humanoides, o que permitiu que os robôs se movessem de forma mais eficiente no campo de futebol. Essa pesquisa não apenas melhorou o desempenho dos jogadores de futebol de robôs, mas também contribuiu para o campo mais amplo do design de robôs humanoides.

Mais recentemente, os pesquisadores começaram a explorar o conceito de "digital twins" no contexto do futebol de robôs. Um estudo realizado por Houtman *et al.* (2019) focou no desenvolvimento de um "digital twin" para um robô jogador de futebol, especificamente os robôs Turtle soccer. Os autores descobriram que o reuso de artefatos fornecia uma boa base para o desenvolvimento inicial do Turtle Digital Twin. Eles também destacaram a importância dos casos de uso no direcionamento do desenvolvimento de um "Digital Twin", em vez de buscar a maior semelhança possível com a realidade. Essa percepção foi especialmente importante em seu trabalho com os Turtles, onde o objetivo era visualizar um jogo de futebol em 3D.

De acordo com Houtman *et al.* (2019), existem diversas abordagens para a implementação de um time de futebol de robôs, que variam desde técnicas baseadas em visão computacional até técnicas baseadas em aprendizado de máquina. Dentre as principais, encontram-se:

- dd) técnicas de visão computacional: Essas técnicas envolvem o uso de câmeras e algoritmos de processamento de imagem para identificar a bola e os jogadores no campo. Por meio da detecção e rastreamento da bola e dos jogadores, os robôs podem obter informações visuais para tomar decisões e coordenar suas ações;
- ee) aprendizado de máquina: Essa abordagem envolve a criação de modelos de aprendizado de máquina que podem aprender a jogar futebol com base em exemplos. Os robôs podem ser treinados usando algoritmos de aprendizado, como redes neurais, para aprender a tomar decisões e executar ações apropriadas durante o jogo;
- ff) técnicas de planejamento de trajetória: Essas técnicas visam planejar trajetórias para os robôs no campo, levando em consideração a posição da bola, dos jogadores e dos obstáculos. O objetivo é definir o caminho mais eficiente para que os robôs se movam e executem suas ações, como chutar a bola ou posicionar-se defensivamente;
- gg) técnicas de controle de robôs: Essas técnicas se concentram no desenvolvimento de algoritmos de controle que permitem aos robôs executar movimentos precisos e coordenados. Isso envolve o controle dos motores e atuadores dos robôs, garantindo que eles se movam de acordo com as instruções recebidas;
- hh) técnicas de tomada de decisão: Essas técnicas envolvem o desenvolvimento de algoritmos que permitem aos robôs tomar decisões inteligentes durante o jogo. Isso inclui decidir quando atacar, defender, passar a bola ou posicionar-se estrategicamente no campo, levando em consideração o estado atual do jogo e as informações disponíveis.

Segundo Houtman *et al.* (2019), essas técnicas representam algumas das áreas de pesquisa essenciais em futebol de robôs, buscando aprimorar a percepção, o planejamento, o controle e a tomada de decisão dos robôs, a fim de melhorar seu desempenho em partidas de futebol de forma simples ou através de sistemas multiagentes.

8.2 SISTEMAS MULTIAGENTES

De acordo com Hübner (1995), os sistemas multiagentes (SMA) são um subcampo da inteligência artificial que se concentra na coordenação e cooperação de múltiplos agentes autônomos. Esses agentes podem ser entidades físicas, como robôs, ou entidades virtuais, como programas de software. Eles interagem entre si para alcançar objetivos individuais ou coletivos, muitas vezes em ambientes complexos e dinâmicos. Este ensaio

explorará as complexidades dos sistemas multiagentes, suas aplicações e os desafios enfrentados em sua implementação.

Segundo com Hübner (1995), as bases teóricas dos sistemas multiagentes estão enraizadas nos conceitos de agentes inteligentes e suas interações. De acordo com Wooldridge e Jennings (1995), um agente inteligente é um sistema que possui autonomia, habilidade social, reatividade e proatividade. Em um sistema multiagente, esses agentes inteligentes interagem entre si em um ambiente compartilhado, resultando em dinâmicas complexas. O estudo dessas dinâmicas, incluindo questões de cooperação, competição e coordenação entre agentes, constitui o cerne da teoria dos sistemas multiagentes.

Stone e Veloso (2000) destacam o uso de sistemas multiagentes em domínios como resolução distribuída de problemas, comércio eletrônico e robótica. Na resolução distribuída de problemas, por exemplo, os sistemas multiagentes podem ajudar a coordenar as ações de vários solucionadores de problemas para alcançar um objetivo comum. No comércio eletrônico, eles podem ser usados para modelar e analisar o comportamento de compradores e vendedores. Na robótica, os sistemas multiagentes podem possibilitar a coordenação de múltiplos robôs, resultando em conclusão de tarefas mais eficiente e efetiva.

Apesar de seu potencial, Zhang *et al.* (2020), a implementação de sistemas multiagentes apresenta vários desafios. Os autores ainda destacam as questões relacionadas à coordenação, comunicação e tomada de decisões entre os agentes. A presença de incerteza no ambiente ou nas ações dos agentes pode complicar o projeto e a operação dos sistemas multiagentes, além da necessidade de técnicas robustas de otimização para lidar com os objetivos conflitantes que muitas vezes surgem em sistemas multiagentes.

Segundo Omidshafiei *et al.* (2022), uma das técnicas usadas para abordar os desafios em sistemas multiagentes é a Multi Agent Collaborative Search (MASC). De acordo com os autores, essa abordagem pode ajudar a melhorar o desempenho do sistema multiagente encontrando um conjunto de soluções que representam um compromisso entre os objetivos conflitantes., além da necessidade de incorporar informações de preferência ao processo de otimização para orientar a busca pelas soluções mais preferidas.

Beard e Ren (2005) apontam que outro aspecto importante dos sistemas multiagentes é o rastreamento de consenso. Os autores apontam que isso envolve garantir que todos os agentes no sistema concordem com certas variáveis ou estados, o que é crucial para sua ação coordenada. Eles destacam que alcançar o consenso em um sistema multiagente pode ser desafiador devido à natureza dinâmica do sistema e às incertezas no ambiente. Beard e Ren (2005) discutem vários protocolos de consenso e suas propriedades de estabilidade, destacando a importância de escolher o protocolo certo para um determinado sistema multiagente.

De acordo com Juchem e Bastos (2001), a estrutura e a arquitetura de modelagem de agentes podem variar de acordo com os objetivos desse agente. A primeira coisa a se fazer para definir qual arquitetura utilizar está baseada em entender qual o tipo de agente do qual se está tratando. Os agentes podem ser cognitivos, que podem ser considerados racionais e tem a capacidade de escolher as ações a executar, dentre as existentes em seu repertório, para chegar em seu objetivo, bem como podem ser reativos, que não necessariamente são agentes racionais, pois eles podem ter seu comportamento definido através de um padrão estímulo-resposta (JUCHEM; BASTOS, 2001). A partir dos dois tipos de agentes descritos, é possível desenvolver uma arquitetura cognitiva (ou deliberativa), que adota o princípio de que agentes são racionais, definindo assim que os agentes cognitivos têm suas deliberações feitas por meio de um processo baseado em raciocínio lógico, escolhendo sua ação por meio da representação simbólica do mundo, um plano e uma função utilizada para a ação. Também pode-se desenvolver uma arquitetura funcional, na qual o agente é composto por módulos que representam cada uma das funcionalidades necessárias para sua operação. Outra arquitetura tradicional muito comum é a arquitetura BDI, que considera três estados mentais do agente: Crença, Desejo e Interação (*Belief, Desire and Intention*). A arquitetura BDI permite que o agente possa decidir, momento a momento, qual ação desempenhar na direção de seus objetivos (JUCHEM; BASTOS, 2001).

REFERÊNCIAS

BEARD, Randall; REN, Wei. Consensus seeking in multiagent systems under dynamically changing interaction topologies. **IEEE Transactions on Automatic Control**, Utah, v. 50, n. 5, p. 655-661, 2005.

CHEN, Liang *et al.* Survey of Multi-Agent Strategy Based on Reinforcement Learning. **Chinese Control And Decision Conference (CCDC)**, Hefei, p. 604-609, 2020.

GUPTA, Sen *et al.* **Recent Advances in Robotics and Automation**. 2013. Disponível em: <https://www.researchgate.net/publication/321613321_Recent_Advances_in_Robotics_and_Automation>. Acesso em: 01 maio 2023.

HÜBNER, Jomi F. Migração de agentes em sistemas multi-agentes abertos. 1995. 126 f. Dissertação (Mestrado em Ciência da Computação) - Programa de Pós-Graduação em Ciência da Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre.

HUANG, Shiyu *et al.* **TiKick: Toward Playing Multi-agent Football Full Games from Single-agent Demonstrations**. 2021. Disponível em: <<https://arxiv.org/abs/2110.04507>>. Acesso em: 16 abr. 2023.

- HOUTMAN, Wouter *et al.* **Tech United Eindhoven Team Description**. Eindhoven, Holanda. 2019 Disponível em: <<https://tdp.robocup.org/wp-content/uploads/tdp/robocup/2019/robocupsoccer-middle-size/tech-united-eindhoven-65/robocup-2019-robocupsoccer-middle-size-tech-united-eindhoven5tvhgXklBF.pdf>>. Acesso em 24 jun. 2023.
- JUCHEM, Murilo; BASTOS, Ricardo Melo. **Arquitetura de Agentes**. Relatório Técnico, n. 013 arquivado na Pró-Reitoria de Pesquisa, Faculdade de Informática PUCRS, Porto Alegre. 2001.
- KITANO, Hiroaki *et al.* RoboCup: The Robot World Cup Initiative. **AGENTS '97: Proceedings of the first international conference on Autonomous agents**, Nagoya, p. 340-347. 1997.
- KUMAR, Aviral *et al.* Conservative q-learning for offline reinforcement learning. **Advances in Neural Information Processing Systems**, v. 33, p. 1179-1191. 2020.
- LAUE, Tim *et al.* SimRobot – A General Physical Robot Simulator and Its Application in RoboCup. **RoboCup 2005: Robot Soccer World Cup IX**, p. 173-183. 2006.
- LIN, Fanqi *et al.* **TiZero: Mastering Multi-Agent Football with Curriculum Learning and Self-Play**. 2023. Disponível em: <<https://arxiv.org/abs/2302.07515>>. Acesso em 16 abr. 2023.
- OMIDSHAFIEI, Shayegan *et al.* Multiagent off-screen behavior prediction in football. **Scientific reports**, v. 12, n. 1, p. 1–13 2022.
- TISUE, Seth; WILENSKY, Uri. NetLogo: A simple environment for modeling complexity. International Conference on Complex Systems. **Agent 2004 Conference on Social Dynamics: Interaction, Reflexivity and Emergence**. 2004.
- STONE, Peter *et al.*, **Reinforcement Learning for RoboCup-Soccer Keepaway**, Adaptive Behavior, Texas, v. 13 n. 3. p. 165–188, 2005.
- STONE, Peter; VELOSO, Manuela. Multiagent Systems: A Survey from a Machine Learning Perspective. **Autonomous Robots**, Texas, v. 8, n. 3. p. 345–383. 2000.
- WOOLDRIDGE, Michael. **An Introduction to MultiAgent Systems**. 2 ed. Estados Unidos: Wiley, 2009.
- WOOLDRIDGE, Michael; JENNINGS, Nicholas. Intelligent Agents: Theory and Practice. **The Knowledge Engineering Review**, v. 10, n. 2, p. 115-152, 1995.
- ZHANG, Haifeng *et al.* Bi-level actor-critic for multi-agent coordination. **Proceedings of the AAAI Conference on Artificial Intelligence**. v. 34, n 5, p. 7325-7332, 2020.

FORMULÁRIO DE AVALIAÇÃO BCC – PROFESSOR TCC I – PROJETO

Avaliador(a): Dalton Solano dos Reis

ASPECTOS AVALIADOS		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	9. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?	X		
	O problema está claramente formulado?	X		
	10. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?	X		
	Os objetivos específicos são coerentes com o objetivo principal?	X		
	11. JUSTIFICATIVA São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?	X		
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?	X		
	12. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?	X		
	Os métodos, recursos e o cronograma estão devidamente apresentados?	X		
	13. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?	X		
ASPECTOS METODOLÓGICOS	14. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?	X		
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?	X		
	15. ORGANIZAÇÃO E APRESENTAÇÃO GRÁFICA DO TEXTO A organização e apresentação dos capítulos, seções, subseções e parágrafos estão de acordo com o modelo estabelecido?	X		
	16. ILUSTRAÇÕES (figuras, quadros, tabelas) As ilustrações são legíveis e obedecem às normas da ABNT?	X		
	17. REFERÊNCIAS E CITAÇÕES As referências obedecem às normas da ABNT?	X		
	As citações obedecem às normas da ABNT?	X		
	Todos os documentos citados foram referenciados e vice-versa, isto é, as citações e referências são consistentes?	X		

O projeto de TCC será reprovado se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos **4 (quatro)** itens dos **ASPECTOS TÉCNICOS** tiverem resposta ATENDE PARCIALMENTE; ou
- pelo menos **4 (quatro)** itens dos **ASPECTOS METODOLÓGICOS** tiverem resposta ATENDE PARCIALMENTE.

PARECER:

(X) APROVADO

() REPROVADO