

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

HÉBER RENATO FADEL DE MORAIS

**AMBIENTE CIBER-FÍSICO PARA A EXECUÇÃO DE TAREFAS COOPERATIVAS
EM UM SISTEMA MULTIRROBÔ**

CURITIBA

2023

HÉBER RENATO FADEL DE MORAIS

**AMBIENTE CIBER-FÍSICO PARA A EXECUÇÃO DE TAREFAS
COOPERATIVAS EM UM SISTEMA MULTIRROBÔ**

Cyber-physical environment for executing cooperative tasks in a multi-robot system

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de “Mestre em Ciências” – Área de Concentração: Engenharia de Automação e Sistemas.

Orientadora: Profa. Dra. Lúcia Valeria Ramos de Arruda

Coorientador: Prof. Dr. João Paulo Lima Silva de Almeida

CURITIBA

2023



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es).

Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.



HEBER RENATO FADEL DE MORAIS

**AMBIENTE CIBER-FÍSICO PARA A EXECUÇÃO DE TAREFAS COOPERATIVAS EM UM SISTEMA
MULTIRROBÔ**

Trabalho de pesquisa de mestrado apresentado como requisito para obtenção do título de Mestre Em Ciências da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Engenharia De Automação E Sistemas.

Data de aprovação: 19 de Dezembro de 2022

Dra. Lucia Valeria Ramos De Arruda, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Joao Paulo Lima Silva De Almeida, Doutorado - Instituto Federal de Educação, Ciência e Tecnologia do Paraná (Ifpr)

Dr. Marco Antonio Simoes Teixeira, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Ricardo Breganon, Doutorado - Instituto Federal do Paraná

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 19/12/2022.

Dedico este trabalho as minhas filhas, esposa,
pai e mãe.

AGRADECIMENTOS

A Deus pela vida, família, saúde e pelas oportunidades proporcionadas para a realização deste trabalho.

Aos meus pais por sempre me incentivar e apoiar nos estudos.

À minha esposa, Renata, pelo companheirismo, apoio e suporte que sempre me deu em todas as situações.

Às minhas filhas, Maria e Ana, que compartilharam o tempo que estaria com elas com o tempo destinado aos meus estudos.

À professora Valéria por todo conhecimento compartilhado, pelas orientações e disponibilidade em auxiliar no desenvolvimento deste trabalho.

Ao professor João Paulo, por toda disposição, ajuda e companheirismo na execução deste trabalho.

Ao Instituto Federal do Paraná Campus Jacarezinho, pelo incentivo, e ao Laboratório GaJac pelo suporte ao desenvolvimento do ambiente e do sistema multirrobo.

RESUMO

MORAIS, Héber Renato Fadel de. **Ambiente ciber-físico para a execução de tarefas cooperativas em um sistema multirrobo**. 2023. 72 f. Dissertação (Mestrado em Engenharia Elétrica e Informática Industrial) – Universidade Tecnológica Federal do Paraná. Curitiba, 2023.

A quarta revolução industrial é marcada pela utilização de sistemas ciber-físicos, que propõem uma produção industrial mais integrada e inteligente, uma vez que realizam a junção de elementos reais e virtuais, utilizando estruturas de computação e comunicação. Este trabalho tem por objetivo o desenvolvimento de um sistema ciber-físico (*Cyber-Physical Systems* (CPS)) composto por agentes de *software* para suportar a utilização de um sistema multirrobo (SMR) de baixo custo na execução de tarefas cooperativas. Uma camada virtual foi criada para a complementação sensorial dos robôs desenvolvidos, já que os robôs utilizados possuem *hardware* limitado. Para a troca de informações entre os agentes do CPS e o SMR, foi utilizado o *framework Robot Operating System* (ROS) como intermediador da comunicação. Nesta dissertação, um ambiente experimental inspirado em cenários da indústria 4.0 foi criado a fim de estabelecer as tarefas cooperativas que o SMR deve realizar bem como caracterizar o ambiente real que os robôs estão inseridos. Por outro lado, o CPS desenvolvido insere características virtuais nos cenários, permitindo criar experimentos variados e dinâmicos. Desta forma, foram realizados vários experimentos distintos para demonstrar a capacidade do SMR suportar a execução de tarefas cooperativas em um SMR homogêneo e/ou heterogêneo. Os resultados mostram que o CPS e o SMR desenvolvidos atingem os objetivos propostos e são flexíveis para a criação de novas tarefas, novos experimentos e integração em novos cenários reais e/ou virtuais.

Palavras-chave: Robótica. Sistemas Multirrobo. Sistemas Ciber-físicos. Cooperação.

ABSTRACT

MORAIS, Héber Renato Fadel de. **Cyber-physical environment for executing cooperative tasks in a multi-robot system**. 2023. 72 p. Dissertation (Master's Degree in Electrical Engineering and Industrial Informatic) – Universidade Tecnológica Federal do Paraná. Curitiba, 2023.

The use of cyber-physical systems (CPS) characterizes the fourth industrial revolution. Such systems propose a more integrated and intelligent industrial production since they combine real and virtual elements using computing and communication structures. This work aims to develop a cyber-physical system composed of software agents to support the use of a low-cost multi-robot system (SMR) in the execution of cooperative tasks. A virtual layer was added for sensory complementation of the developed robots since the robots used have limited *hardware*. For exchanging information between the CPS agents and the SMR, the *framework Robot Operating System* (ROS) is used as a communication intermediary. This dissertation builds an experimental environment inspired by Industry 4.0 scenarios to establish the cooperative tasks that the SMR must perform and to characterize the real environment in which the robots are inserted. On the other hand, the developed CPS inserts virtual characteristics in the scenarios, allowing the realization of varied and dynamic experiments. In this way, several different experiments are carried out to demonstrate the ability of the SMR to support the execution of cooperative tasks in a homogeneous and/or heterogeneous SMR. The results show that the developed CPS and SMR reach the proposed objectives and are flexible for implementing new tasks, experiments, and their integration in new real and/or virtual scenarios.

Keywords: Robotic. Multi-robotic Systems. Cyber-Physical Systems. Cooperation.

LISTA DE ILUSTRAÇÕES

Figura 1 – Evolução da indústria 1.0 à indústria 4.0.	18
Figura 2 – Estrutura proposta por (LEE; BAGHERI; KAO, 2015).	20
Figura 3 – Visão geral do ambiente criado.	27
Figura 4 – Arquitetura do ambiente com características ciber-físicas desenvolvido.	28
Figura 5 – Dimensões dos robôs desenvolvidos.	29
Figura 6 – Robôs móveis desenvolvidos.	30
Figura 7 – Tópicos ROS utilizados para comunicação de robôs com o AGT.	31
Figura 8 – Funcionamento do sensor virtual dos robôs.	32
Figura 9 – Variável de Entrada: Distância.	34
Figura 10 – Variável de Entrada: Ângulo.	35
Figura 11 – Variável de Saída: Velocidade Angular.	35
Figura 12 – Superfície de resposta do controlador <i>Fuzzy</i>	37
Figura 13 – Situação onde a estratégia de comunicação foi utilizada.	38
Figura 14 – Fluxograma do sistema de navegação.	39
Figura 15 – Tópicos criados pelo AAL após a identificação de robôs e alvos.	41
Figura 16 – Diagrama de classes utilizadas pelo AGT.	44
Figura 17 – Fluxograma referente à execução de uma tarefa do AGT.	45
Figura 18 – Tela do Agente Supervisório e imagem capturada pelo AAL.	46
Figura 19 – Aplicação da arquitetura 5C no CPS criado.	48
Figura 20 – Disposição do experimento para validar a função de transição.	49
Figura 21 – Execução sem a função de transição.	50
Figura 22 – Execução com a função de transição.	51
Figura 23 – Quadros sobrepostos capturados do AAL na execução com a função de transição.	51
Figura 24 – Quadros que representam a execução da estratégia de comunicação.	52
Figura 25 – Representação do momento do processamento da estratégia de comunicação.	52
Figura 26 – Inspiração para as configurações experimentais.	53
Figura 27 – Configuração do experimento 01.	54
Figura 28 – Sequência de quadros capturados do AAL e do AS durante o experimento 01.	55
Figura 29 – Status dos robôs durante a execução no experimento 01.	56
Figura 30 – Caminho percorrido pelos robôs no experimento 01.	57
Figura 31 – Configuração do experimento 02.	58
Figura 32 – Sequência de quadros capturados do AAL e do AS durante o experimento 02.	59
Figura 33 – Caminho percorrido pelos robôs no experimento 02.	60
Figura 34 – Configuração do experimento 03.	60
Figura 35 – Sequência de quadros capturados do AAL e do AS durante o experimento 03.	62
Figura 36 – Caminho percorrido pelos robôs no experimento 3 - 1º ciclo.	63
Figura 37 – Caminho percorrido pelos robôs no experimento 3 - 2º ciclo.	63
Quadro 1 – Estrutura da Mensagem ROS <i>Info</i>	42
Quadro 2 – Estrutura da Mensagem <i>Lista_info</i>	43
Quadro 3 – Estrutura da Mensagem <i>Point_r</i>	43
Quadro 4 – Estrutura da Mensagem <i>Points_r</i>	43

LISTA DE TABELAS

Tabela 1 – Características dessa dissertação.	24
Tabela 2 – Comparação com trabalhos relacionados (Parte 01).	24
Tabela 3 – Comparação com trabalhos relacionados (Parte 02).	24
Tabela 4 – Componentes de cada robô.	29
Tabela 5 – Base de regras do controlador <i>Fuzzy</i>	36
Tabela 6 – Tópicos ROS produzidos pelo AGT.	45
Tabela 7 – Recursos ciber-físicos do sistema multirrobô.	47

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

SIGLAS

AAL	Agente Auxiliar de Localização
AGT	Agente Gerenciador de Tarefas
ARTag	<i>Augmented Reality Tag</i>
AS	Agente Supervisório
CP	Computador Principal
CPS	<i>Cyber-Physical Systems</i>
EFL	<i>Embedded Fuzzy Logic Library</i>
IA	Inteligência Artificial
IoT	<i>Internet of Things</i>
PD	Proporcional-Derivativo
PID	Proporcional, Integral e Derivativo
ROS	<i>Robot Operating System</i>
SMR	Sistema Multirrobo

SUMÁRIO

1	INTRODUÇÃO	11
1.1	MOTIVAÇÃO	11
1.2	OBJETIVO GERAL	13
1.3	OBJETIVOS ESPECÍFICOS	13
1.4	ARTIGOS RELACIONADOS	14
1.5	ORGANIZAÇÃO DO TRABALHO	14
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	SISTEMAS MULTIROBÔ	15
2.2	NAVEGAÇÃO AUTÔNOMA	17
2.3	SISTEMAS CIBER-FÍSICOS	18
2.4	TRABALHOS RELACIONADOS	21
2.5	CONSIDERAÇÕES	23
3	AMBIENTE CIBER-FÍSICO E ROBÔS DESENVOLVIDOS	26
3.1	VISÃO GERAL DA ARQUITETURA DESENVOLVIDA	26
3.2	ROBÔS MÓVEIS DESENVOLVIDOS	28
3.2.1	Habilidades de <i>software</i> dos robôs	30
3.2.2	Sensoriamento Virtual	31
3.2.3	Controladores utilizados na navegação	33
3.2.4	Estratégia de comunicação entre robôs	37
3.3	AGENTE AUXILIAR DE LOCALIZAÇÃO (AAL)	39
3.4	AGENTE GERENCIADOR DE TAREFAS (AGT)	41
3.4.1	Mensagens do AGT	42
3.4.2	Gerenciamento de Tarefas	44
3.5	AGENTE SUPERVISÓRIO (AS)	46
3.6	CONSIDERAÇÕES	47
4	RESULTADOS	49
4.1	VALIDAÇÕES DAS ESTRATÉGIAS DE CONTROLE DOS ROBÔS	49
4.1.1	Função de Transição	49
4.1.2	Estratégia de Comunicação para Desvios de Obstáculos Dinâmicos	51
4.2	VALIDAÇÕES DO AMBIENTE CIBER-FÍSICO	53
4.2.1	Experimento 01	53
4.2.2	Experimento 02	57
4.2.3	Experimento 03	58
4.3	CONSIDERAÇÕES	64
5	CONSIDERAÇÕES FINAIS	65
5.1	CONCLUSÕES	65
5.2	TRABALHOS FUTUROS	67
	REFERÊNCIAS	68

1 INTRODUÇÃO

Neste capítulo são apresentadas as principais motivações para o desenvolvimento deste trabalho, seus objetivos gerais e específicos, as produções geradas pela pesquisa bem como a organização do trabalho.

1.1 MOTIVAÇÃO

A indústria 4.0 é marcada fortemente pela integração de conceitos relacionados aos Sistemas Ciber-Físicos (CPS), robótica, Internet das Coisas (*Internet of Things* (IoT)), computação em nuvem e Inteligência Artificial (IA). No entanto, devido a complexidade na integração dessas tecnologias, sua implementação ainda representa um grande desafio (YOU; FENG, 2020; RIKALOVIC *et al.*, 2022).

No contexto da indústria 4.0, é comum sistemas ciber-físicos trabalharem em conjunto com múltiplos robôs para a execução de trabalhos em diversas situações, tais como: linha de produção, armazéns inteligentes, dentre outras aplicações industriais. Como uma das características principais de um CPS é a junção de elementos reais e virtuais, este atributo traz benefícios para aplicações no contexto da robótica, uma vez que robôs podem ser entidades autônomas com a capacidade de interagir com um ambiente real, desencadeando assim ações dentro de um meio físico. Tais ações podem ser coordenadas de forma centralizada, por um "gestor", o qual possui um conhecimento de todo o ambiente de trabalho, incluindo suas características virtuais. Assim o CPS, através de tal "gestor" é responsável pelo planejamento e distribuição das tarefas aos robôs do Sistema Multirrobo (SMR) (AFRIN *et al.*, 2021; YE *et al.*, 2021).

De uma forma geral, um CPS se caracteriza como um sistema de entidades computacionais colaborativas, podendo ser agentes de *software* que estão constantemente conectados ao mundo físico e seus processos em andamento, cuja interação entre elementos físicos e cibernéticos é de fundamental importância (MONOSTORI *et al.*, 2016).

Um sistema multi-agente se caracteriza por resolver problemas complexos subdividindo-os em tarefas de menor complexidade, as quais são alocadas a entidades autônomas (agentes), que convergem para um objetivo em comum. Um agente inserido em um ambiente processa parâmetros para a tomada de decisão de acordo com o objetivo. Um agente pode ser um *software* (agente virtual), um *hardware* (agente físico) ou a combinação dos dois (um robô). Dessa forma

um sistema composto por múltiplos robôs podem ser considerado um sistema multi-agente (DORRI; KANHERE; JURDAK, 2018).

A utilização de agentes autônomos, tais que robôs móveis, veículos aéreos não tripulados, carros, dentre outros, em sistemas ciber-físicos para realização de tarefas complexas é um tópico atual de pesquisa em SMR. Em um CPS, o ambiente de navegação além de apresentar características físicas virtuais e reais, sendo por isso denominado ambiente de realidade aumentada ou de realidade mista (EGGER; MASOOD, 2020; PIARDI *et al.*, 2019), possui também agentes reais e virtuais com os quais os robôs devem interagir e compartilhar informações. Por exemplo, um SMR inserido em um ambiente de realidade mista se beneficia de recursos de sensoriamento virtual que os robôs não possuem e também de estratégias de comunicação envolvendo informações compartilhadas de uma camada virtual do CPS, ao qual o SMR tem acesso.

Uma vez que um SMR, especificamente formado por robôs móveis, necessita trabalhar em um ambiente compartilhado, devendo tomar decisões sobre desvio de obstáculos, seguir um determinado caminho e atingir metas enquanto realizam tarefas de forma cooperativa, a navegação autônoma deve levar em consideração o trabalho em equipe, principalmente no compartilhamento de informações a respeito do ambiente de trabalho (MADRIDANO *et al.*, 2021; RIZK; AWAD; TUNSTEL, 2020).

Dessa forma, se faz necessário a aplicação de técnicas que permitem a cooperação e a colaboração de robôs móveis dentro de um ambiente para a execução de tarefas. A capacidade de se trabalhar em conjunto requer de robôs colaborativos a realização de operações que exigem comunicação dinâmica, otimização, aprendizado, dentre outras. A inserção do SMR dentro de um CPS, ou de outro modo a construção de um SMR como um CPS, possibilita que os robôs utilizem as facilidades de tais sistemas ciber-físicos na execução colaborativa de suas tarefas (WEISS; WORTMEIER; KUBICEK, 2021).

Neste contexto, a presente dissertação apresenta o desenvolvimento de um ambiente com características ciber-físicas a ser utilizado como suporte para a implementação de um SMR. O CPS desenvolvido é composto por agentes computacionais responsáveis por: auxiliar na localização dos robôs e alvos; gerenciar atribuição de tarefas e por criar uma visualização dos elementos virtuais e reais inseridos no ambiente de trabalho. Por outro lado, o SMR desenvolvido é composto por robôs móveis reais com capacidades básicas de locomoção e de comunicação, o *hardware* construído é simples e de baixo custo. Para este trabalho, foram desenvolvidos dois

tipos de robôs com habilidades virtuais distintas, assim o SMR formado é do tipo heterogêneo em que robôs distintos realizam sub-tarefas diferentes em cooperação. Foi utilizado o *framework Robot Operating System (ROS)* (QUIGLEY *et al.*, 2009) como intermediador de comunicação entre agentes e robôs móveis, permitindo a criação de mensagens personalizadas para a troca de informações entre agentes e robôs.

Sendo assim, a principal motivação desta dissertação é a criação de uma ferramenta que possa ser utilizada para o ensino de conceitos relacionados aos sistemas ciber-físicos e robótica, uma vez que é fundamental que se integrem técnicas relativas à indústria 4.0 no desenvolvimento de ferramentas didáticas que reproduzam ambientes e permitam a interação com equipamentos reais, tornando fácil a aplicação destes conceitos em ambiente acadêmico (FUERTES *et al.*, 2021).

1.2 OBJETIVO GERAL

O objetivo geral desta dissertação consiste na criação de um ambiente em escala laboratorial que apresente características ciber-físicas e que permita a execução de tarefas de forma cooperativas, intermediadas pelo *framework ROS*, de múltiplos robôs móveis reais de baixo custo.

1.3 OBJETIVOS ESPECÍFICOS

- Criar um ambiente com características ciber-físicas que permita a inclusão e a comunicação de agentes de *software* individuais a fim de torná-lo flexível e escalável;
- Construir múltiplos robôs móveis de baixo custo, que possam processar de maneira individualizada e descentralizada decisões a cerca de sua navegação;
- Desenvolver um agente que possa auxiliar o processo de localização dos robôs dentro da área experimental e conferir habilidades de sensoriamento virtual ao SMR;
- Desenvolver um agente que possa gerenciar as tarefas a serem executadas pelo SMR;
- Desenvolver uma estrutura de classes e mensagens personalizadas ROS para dar suporte na elaboração de tarefas personalizadas;

- Desenvolver um agente com a habilidade de monitorar a execução de tarefas realizadas pelo SMR e com a capacidade de visualizar os elementos reais e virtuais do ambiente;
- Validar as estratégias adotadas na codificação embarcada do SMR e a capacidade do ambiente criado na execução de tarefas cooperativas.

1.4 ARTIGOS RELACIONADOS

Durante o desenvolvimento desta dissertação alguns artigos foram submetidos a congressos correlatos à área de estudo, abaixo segue a relação dos trabalhos aprovados e apresentados.

- MORAIS, H. R. F.; ALMEIDA, J. P. L. S.; ARRUDA, L. V. R. Ambiente de realidade mista para experimentos com múltiplos robôs de baixo custo. **CBA 2022 - Congresso Brasileiro de Automática**. Fortaleza-CE, Brasil. 2022.
- MORAIS, H. R. F.; ALMEIDA, J. P. L. S.; ARRUDA, L. V. R. *Running Cooperative Tasks in a Multi-Robot System With Cyber-Physical Features*. **LARS 2022 - IEEE Latin American Robotics Symposium**. São Paulo-SP, Brasil. 2022.

1.5 ORGANIZAÇÃO DO TRABALHO

O Capítulo 2 apresenta a fundamentação teórica para embasar os principais conceitos utilizados nessa dissertação, além de comparar a atual implementação com alguns trabalhos relacionados. O Capítulo 3 tem por intuito apresentar as principais características do CPS criado bem como detalha o SMR desenvolvido, em especial os robôs construídos. O Capítulo 4 apresenta os principais resultados extraídos de experimentos que puderam validar toda a arquitetura desenvolvida. Já o Capítulo 5 apresenta as considerações finais obtidas durante o desenvolvimento dessa dissertação.

2 FUNDAMENTAÇÃO TEÓRICA

Serão descritos neste capítulo os principais tópicos da literatura que nortearam o desenvolvimento desta dissertação. A seção 2.1 detalha as características básicas de um SMR. Na seção 2.2 são apresentadas as propriedades elementares de navegação autônoma em robôs móveis. Na seção 2.3 é apresentada a contextualização do conceito de sistemas ciber-físico, na seção 2.4 são apresentados os trabalhos relacionados a esta dissertação, por fim, na seção 2.5 é apresentada uma breve consideração sobre o capítulo.

2.1 SISTEMAS MULTIRROBÔ

Um SMR consiste em um sistema em que múltiplos robôs compartilham um ambiente de trabalho a fim de alcançar um objetivo global, em que um único robô não seria capaz ou teria limitações em fazê-lo. Em diversas abordagens, o objetivo global atribuído ao SMR é decomposto em sub-objetivos de tarefas de menor complexidade, mas que convergem para o objetivo em comum (YAN; JOUANDEAU; CHERIF, 2013; ALMADHOUN *et al.*, 2019).

Dividir uma tarefa complexa em sub-tarefas mais simples e atribuí-las aos robôs inseridos em um SMR e permitir a interação entre eles para encontrar soluções para essas sub-tarefas é um dos conceitos que está por traz da robótica multi-agente. Neste contexto, os SMRs são mais econômicos comparando-se à construção de um único robô com muitos recursos e capaz de resolver sozinho a tarefa original (GAUTAM; MOHAN, 2012).

Um grupo de robôs trabalhando em conjunto pode ser homogêneo ou heterogêneo (YAN; JOUANDEAU; CHERIF, 2013). Os SMRs homogêneos são formados por robôs cuja capacidade individual é igual, mesmo que suas estruturas físicas sejam diferentes. Já os SMRs heterogêneos possuem robôs com capacidades distintas, onde cada um tem a capacidade especializada na execução tarefas específicas.

Os SMRs heterogêneos foram propostos, principalmente, para que cooperem na execução de tarefas complexas, uma vez que, devido a sua versatilidade na realização de tarefas especializadas, exigem colaboração entre diferentes tipos de robôs (MAYYA *et al.*, 2021).

A execução cooperativa de um SMR engloba tarefas com vários graus de complexidade, onde a dificuldade de execução afeta o número e o tipo de robôs necessários para completá-la. Rizk, Awad e Tunstel (2020) agrupam estas tarefas a serem realizadas por um SMR em fraca-

mente coordenadas e fortemente coordenadas. O primeiro grupo inclui as tarefas de exploração e mapeamento em grande escala, limpeza de materiais perigosos, rastreamento e vigilância. Para a realização dessas tarefas, acontece uma sub divisão do ambiente e dos robôs que trabalham dentro de áreas específicas. Já as tarefas fortemente coordenadas não podem ser executadas se forem decompostas em sub-tarefas, uma vez que requerem execução coordenada dos robôs do SMR que as desempenham, demandando grande interação entre eles. São exemplo de tais tarefas o transporte de objetos e construção em larga escala.

A comunicação dentro de um SMR é imprescindível para a execução de tarefas cooperativas, uma vez que é necessário a coordenação e o compartilhamento de informações entre os robôs (ANGUELOV; TRIFONOV; NAKOV, 2021). Em sistemas multi-agentes, uma rede é frequentemente formada para comunicação sem fio e roteamento de mensagens entre os robôs (QUERALTA *et al.*, 2020).

Para permitir a comunicação entre robôs, duas abordagens podem ser utilizadas, a direta ou a indireta. A comunicação direta ocorre quando robôs tem a capacidade de se comunicar diretamente entre si, sem a intervenção de um agente intermediador, ela pode ser estabelecida por meio de rádio frequência, infravermelho, entre outras tecnologias. Já a comunicação indireta, permite que robôs se comuniquem por intermédio de um outro agente ou pelo próprio ambiente, podendo ser um agente central, outro robô, feromônios e/ou marcações no ambiente, dentre outros dispositivos capazes de intermediar a comunicação (DORIYA; MISHRA; GUPTA, 2015).

Em relação ao processamento e a tomada de decisão dentro de um SMR, uma arquitetura centralizada possui um agente que se encarrega de organizar o trabalho dos demais agentes, onde um líder está envolvido no processo de decisão de toda a equipe, tendo uma visão global de todo o sistema. Tal implementação facilita o processo de tomada de decisão, uma vez que um agente centralizador está por traz da gerência de tarefas, no entanto a dependência de um agente central pode tornar o SMR menos robusto e sujeito a falhas, uma vez que se houver comunicação comprometida, devida a situações inesperadas do ambiente, as tarefas podem ser prejudicadas. Já um sistema descentralizado não depende de uma unidade central, podendo tomar decisões e realizar processamentos de forma individual (GAO; CAI; YU, 2009; MADRIDANO *et al.*, 2021).

Os conceitos acima mencionados compreendem o que se julga como elementos básicos para a implementação de sistemas compostos por múltiplos robôs móveis, onde mesmo possuindo características distintas, para trabalharem de forma cooperativa devem se comunicar diretamente

ou indiretamente para a realização de tarefas.

2.2 NAVEGAÇÃO AUTÔNOMA

Para a realização de tarefas em um ambiente, um robô deve navegar com segurança dentro de sua área de trabalho. Para isto, seu planejamento de rota deve levar em consideração variáveis específicas que o conduza de um estado inicial para um estado objetivo determinado pela tarefa que ele deve realizar. Um estado é caracterizado pelos valores que um conjunto de variáveis (posição, velocidade, status de operação) do robô pode assumir, ou ainda pelo conjunto de ações de controle aplicadas aos seus atuadores (ANTONYSHYN *et al.*, 2022).

A implementação de técnicas de movimentação e de desvio de obstáculos tanto em ambientes conhecidos ou desconhecidos é essencial quando se pretende implementar navegação autônoma de robôs móveis, onde o mesmo deverá ter a capacidade de percorrer um determinado ambiente, prevenir colisões e atingir seus objetivos.

Considerando um SMR, a navegação multi-agente é um problema fundamental que deve ser abordado, já que o gerenciamento da locomoção de vários robôs consiste em tarefas que tem o objetivo de direcioná-los a seus respectivos alvos (posições de objetivo ou pontos de interesse em geral) evitando colisões durante seu percurso. A navegação multi-agente é crítica em tarefas práticas, como logística de armazéns inteligentes, navegação autônoma, dentre outras aplicações onde os ambientes estão em constante mudança (QIU *et al.*, 2021).

Durante a navegação em um SMR, a comunicação entre robôs é importante para evitar colisões entre eles, e se torna essencial no caso em que os robôs têm uma visão limitada sobre o ambiente onde está inserido. Em tarefas de navegação de vários robôs em um ambiente desconhecido, o compartilhamento de informações específicas de cada robô (processamento interno, tomadas de decisão, controladores utilizados, dentre outras informações produzidas por cada robô do SMR) torna-se necessária à coordenação de seus movimentos para a realização de um trabalho cooperativo com sucesso. Em SMR cooperativos, é natural que os robôs se comuniquem entre si para gerar ações considerando as intenções de ação de outros robôs, bem como suas próprias observações do ambiente para o cumprimento de objetivos e na realização de tarefas (ZHAI *et al.*, 2021).

Considerando a aplicação de um SMR cooperativo e utilizando uma arquitetura centralizada e robôs com restrições sensoriais, a construção de sensores virtuais utilizando recursos cibernéticos pode ser uma alternativa principalmente em aplicações em ambientes experimentais,

em que os dispositivos periféricos devem ser testados de forma virtual antes de sua aquisição real (compra) (MILLARD *et al.*, 2018). A aplicação de conceitos relacionados a construção de sensores virtuais para que robôs móveis evitem colisão com objetos (reais ou virtuais sejam eles estáticos ou dinâmicos), se demonstrou eficaz para conduzir com segurança robôs móveis de uma coordenada de origem até uma de destino (ALMEIDA *et al.*, 2020).

Em resumo, a navegação autônoma de um SMR requer a locomoção segura de todos os robôs pelo ambiente de trabalho, além de uma comunicação efetiva entre robôs para superar entraves relacionados à navegação coletiva.

2.3 SISTEMAS CIBER-FÍSICOS

A história já testemunhou três grandes revoluções industriais, em que a primeira foi marcada pelo avanço tecnológico provido pela utilização da força da água e do vapor no processo de produção. A segunda revolução industrial foi marcada pela utilização da energia elétrica e a produção em massa. A terceira se destacou pelo avanço da eletrônica, utilização de computação e internet para a automação de processos (XU; XU; LI, 2018). A quarta revolução industrial é marcada fortemente pela junção de tecnologias como sistemas ciber-físicos, IoT, computação em nuvem, 5ª geração da rede móvel (5G), sensores de rede sem fio, robótica, dentre outros (LIU *et al.*, 2018). A Figura 1 apresenta as principais características que marcaram desde a 1ª revolução industrial até a 4ª, destacando de acordo com (XU; XU; LI, 2018), que a 4ª revolução industrial é marcada pela era do CPS.

Figura 1 – Evolução da indústria 1.0 à indústria 4.0.

Era do vapor	Era da Eletricidade	Era da Informação	Era do CPS
<ul style="list-style-type: none"> • 1ª Revolução Industrial • Introdução aos sistemas de manufatura mecânica utilizando água e a força do vapor. 	<ul style="list-style-type: none"> • 2ª Revolução Industrial • Introdução à produção em massa utilizando a energia elétrica. 	<ul style="list-style-type: none"> • 3ª Revolução Industrial • Introdução à eletrônica e sistemas de informação e comunicação para a automação. 	<ul style="list-style-type: none"> • 4ª Revolução Industrial • Em 2013, o conceito de indústria 4.0 foi oficialmente apresentado. Junção de CPS, IoT, robótica, 5G, etc.

Fonte: Adaptado de (XU; XU; LI, 2018).

O termo CPS refere-se a uma nova geração de sistemas com capacidades computacionais, que por meio de uma estrutura de rede, elementos reais (robôs, sensores, máquinas, etc)

possam interagir de forma a expandir as limitações do mundo físico, com elementos virtuais, permitindo comandos remotos, análises de processos, e demais atividades de gerenciamento da indústria, com o intuito de se criar um ambiente industrial moderno, robusto e inteligente (RAJKUMAR *et al.*, 2010). Assim, um CPS se caracteriza como um sistema físico cujas operações são monitoradas, controladas, coordenadas e integradas por um núcleo de computação e comunicação (MONOSTORI *et al.*, 2016).

Em resumo, os CPS se caracterizam pela integração da computação, comunicação e controle. As técnicas e metodologias dessas áreas são usadas em conjunto para a realização de tarefas de detecção em tempo real, controle dinâmico e coleta de informações em grandes sistemas de engenharia. Os CPS também podem ser entendido como um sistema distribuído heterogêneo, que mescla a utilização de sistemas físicos com subsistemas virtuais distribuídos na execução de tarefas, para isso, a comunicação com ou sem fio necessária para que estes sistemas coexistam (LIU *et al.*, 2017; SHI *et al.*, 2011).

Como um CPS integra elementos físicos (reais) a elementos cibernéticos (virtuais), os reais são compostos por sensores de rede, robôs móveis, braços robóticos, computadores, câmeras, entre outros elementos físicos, já os elementos cibernéticos geralmente são compostas por sistemas computacionais com a capacidade de supervisionar, analisar e avaliar o desempenho de elementos físicos, podendo criar simulações em tempo real sobre o funcionamento de periféricos e permitir a criação de ambientes de realidade virtual que podem ser executados em paralelo ao ambiente real (TAO; ZHANG, 2017).

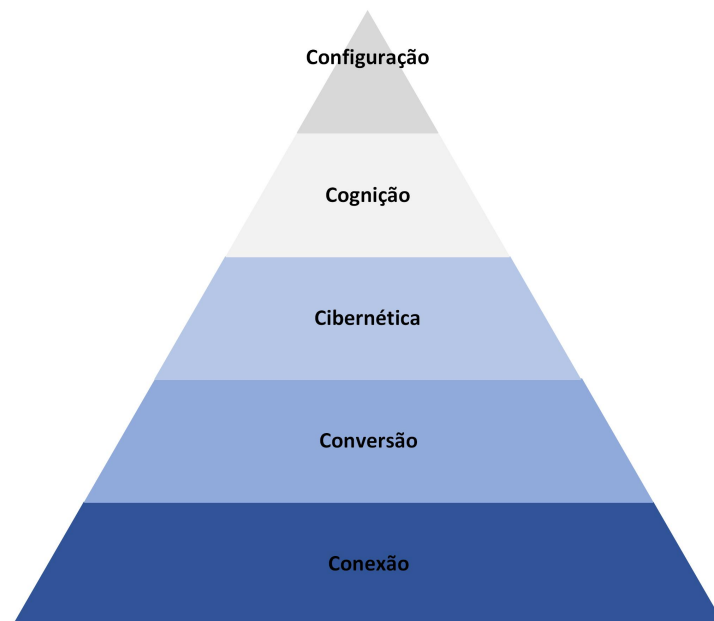
Outra característica importante do CPS é a sua interação com os humanos, que visa expandir as capacidades do mundo físico. Uma das aplicações deste conceito são em desenvolvimento de aviões, veículos espaciais, veículos autônomos e também em próteses que permitem que sinais cerebrais controlem objetos físicos (BAHETI; GILL, 2011; PEREIRA; THOMAS, 2020).

Lee e seus colaboradores definiram uma arquitetura geral de 5 níveis denominada 5C, que serve como um guia para a implementação de um CPS (LEE; BAGHERI; KAO, 2015). As camadas apresentadas são de conexão, conversão, cibernética, cognição e configuração, conforme ordem apresentada na Figura 2. Suas funcionalidades são brevemente descritas como segue:

- **Conexão:** atua no espaço físico, onde os elementos reais são responsáveis pela aquisição dos dados (provenientes de sensores, robôs, máquinas e seus componentes) que são transmitidos a um servidor central;

- Conversão: atua no espaço cibernético (elementos virtuais), traz autoconsciência para as máquinas, converte os dados vindos da primeira camada em informações;
- Cibernética: atua no espaço cibernético, centralizando toda a informação produzida pela arquitetura, tem o papel de compilar estas informações com o intuito de fornecer melhorias sobre a situação dos elementos inseridos no CPS, fornecendo-lhes uma capacidade de auto comparação;
- Cognição: atua no espaço cibernético, analisa de forma aprofundada as informações produzidas com o intuito de fornecer aos usuários conhecimento aprofundado do sistema a ser monitorado;
- Configuração: é um retorno do ambiente virtual para o físico com o intuito de tornar as máquinas auto configuráveis e auto adaptativas;

Figura 2 – Estrutura proposta por (LEE; BAGHERI; KAO, 2015).



Fonte: Adaptado de (LEE; BAGHERI; KAO, 2015).

Dentro da classificação de 5 níveis para um CPS, a presente dissertação propõe a criação de um CPS que atue nas três primeiras camadas (conexão, conversão e cibernética) dentro de um contexto de aplicação com um SMR contendo robôs móveis reais de baixo custo. Para o CPS, agentes de *software* são criados visando: complementar a capacidade sensorial dos robôs, intermediar a comunicação entre robôs e agentes, coordenar trabalhos cooperativos e

proporcionar uma visualização dos elementos reais e virtuais do ambiente de trabalho a um usuário humano.

2.4 TRABALHOS RELACIONADOS

São mostrados nesta seção alguns trabalhos que utilizam múltiplos robôs móveis inseridos em sistemas ciber-físicos. Ao final da seção a Tabela 1 apresenta as características do CPS desenvolvido neste trabalho, já nas Tabelas 2 e 3 é apresentada uma comparação destes trabalhos com a implementação proposta nesta dissertação.

Em Semwal *et al.* (2015) foi desenvolvida uma plataforma cujo objetivo é integrar robôs móveis com demais dispositivos estáticos e heterogêneos para formar um CPS. A plataforma permite a criação de agentes que se ajustam em tempo de execução sem interferir na execução do CPS. Para a validação do ambiente, são apresentados experimentos em que um robô móvel está navegando por um ambiente real e após a identificação de obstáculos, este robô aciona um agente central que tem por objetivo enviar um outro robô para auxiliar na remoção do obstáculo. Neste processo uma iteração com um administrador humano acontece, o qual também tem acesso ao CPS desenvolvido. O trabalho utiliza robôs móveis desenvolvidos com o kit LEGO *Mindstorms NXT* e um sistema de monitoramento estático com uma câmera acoplada a um *Raspberry Pi*¹.

Foi desenvolvido em Xu *et al.* (2019), um CPS com o intuito de controlar a formação de robôs móveis reais permitindo que realizem manobras de desvio de obstáculo. Para isso, um dos robôs móveis desenvolvidos atua como o líder do pelotão e os demais como seus seguidores. Nos experimentos apresentados, três robôs são utilizados, e o robô líder direciona os seguidores a constituírem uma formação triangular e a seguirem uma trajetória circular em uma área experimental, desviando de um obstáculo estático. Todos os robôs possuem sensores ultrassônicos que compartilham informações a respeito de obstáculos e utilizam controladores *Fuzzy* para a realização da tarefa (formação, navegação e desvio). O trabalho também utiliza o conceito 5C para a definição do seu CPS (LEE; BAGHERI; KAO, 2015).

Em Bemani e Bjorsell (2020), foi desenvolvido um CPS que tem por objetivo centralizar informações a respeito de uma navegação cooperativa de robôs móveis. Utiliza dispositivos *bluetooth* de baixo consumo de energia para auxiliar os robôs móveis no processo de auto localização dentro de ambientes internos. Os robôs móveis são equipados com um *Raspberry PI*, sensores de distância e de detecção de linha. Nos experimentos apresentados, robôs móveis

¹ <https://www.raspberrypi.org/>

percorrem uma linha previamente estabelecida, e desconhecida para eles, enquanto compartilham com um agente central informações a respeito de sua localização e identificação dos sensores. Durante a execução de suas tarefas, o CPS realiza um mapeamento do ambiente em tempo real com informações produzidas pelos robôs móveis. Tal mapeamento é considerado pelos autores como um gêmeo virtual, que reflete a junção das informações reais e virtuais de seu ambiente.

Foi desenvolvido por Escobar *et al.* (2020) uma plataforma didática para aplicação dos princípios de sistemas ciber-físicos juntamente com múltiplos robôs móveis. Os robôs utilizados possuem movimentação omnidirecional e foram construídos combinando um *Raspberry Pi*, motores, sensores de localização e velocidade. O *framework* ROS foi utilizado para unificar as informações produzidas pelos robôs, mas nenhum sensor ou estratégia de desvio de obstáculos foi implementada no trabalho. Nos experimentos, os autores analisaram dados relacionados a movimentos circulares dentro de formas geométricas previamente estabelecidas, tanto para avaliação de desempenho individual dos robôs quando para realizarem movimentos que envolvem uma coordenação entre os robôs.

Em Rohrich *et al.* (2021) foi proposto o desenvolvimento de sensores móveis (robôs) com o objetivo de monitorar vazamento de gases. Os robôs móveis desenvolvidos não possuem sensores reais e sua movimentação é por meio de motores de vibração. Eles são construídos com componentes eletrônicos convencionais e de baixo custo. Para validar a proposta, o trabalho aplica o conceito de realidade mista para criar tanto simulações a respeito da dissipação de gases quanto para fornecer a capacidade de detecção aos robôs (sensoriamento virtual). O *framework* ROS é utilizado para unir informações do sistema real e de realidade mista. Foi criado um algoritmo inspirado no comportamento de bactérias para a identificação do vazamento de gases. A partir de diversas configurações experimentais, foi analisado o comportamento individual e coletivo do SMR, além de comparar resultados onde o número de robôs inseridos no ambiente era aumentado a cada execução. A configuração experimental possibilitou, com o auxílio de um CPS, simular situações sem expor os pesquisadores a ambientes insalubres reais.

Já em Shin *et al.* (2021) é proposto um CPS distribuído, onde somente a comunicação direta e localizada entre robôs móveis é utilizada. Para isso, desenvolveram robôs móveis utilizando kits LEGO *Mindstorms* Ev3 dotados de sensores ultrassônicos para o desvio de obstáculos e sensores de cor para o seguimento de linhas. Nos experimentos, uma pista com um traçado no formato de um quadrado (linha por onde os robôs devem se guiar) foi estabelecida, possibilitando que robôs naveguem tanto na parte interior da marcação quando na parte exterior.

Inicialmente foram posicionados na parte externa deste trajeto três robôs, sendo um líder e os demais seguidores, devendo seguir esta trajetória. A comunicação é feita em cascata, onde o primeiro robô envia informações para o segundo e o segundo para o terceiro. Assim que um obstáculo é identificado pelo robô líder, é enviada uma informação para que o robô que o segue mude sua trajetória para o traçado interno do circuito, o inverso ocorre quando o robô líder já está na parte interna. Os autores concluíram que mesmo um experimento simples validou uma estratégia de comunicação direta entre robôs.

A partir do que foi apresentado, o SMR e o CPS desenvolvidos nesta pesquisa possuem as características apresentadas na Tabela 1. O SMR proposto, do ponto de vista de *hardware*, todos os robôs são iguais, mas do ponto de vista da execução de tarefas, os robôs podem ter atribuições e habilidades distintas e, neste caso, o SMR é considerado heterogêneo. Os robôs processam de forma individualizada ações a respeito de sua navegação, porém dependem de informações de um agente central para conhecer sua localização, a localização de alvos e obstáculos, além de informações de outros robôs, dessa forma a arquitetura do SMR é centralizada por depender de um agente central para a atribuição de tarefas e intermediar a comunicação entre os robôs do SMR. Para permitir uma fácil replicação dos robôs, seu *hardware* e *software* foi feito utilizando tecnologias *open source*. Os robôs utilizam *Wi-Fi* como tecnologia de comunicação sem fio e sua locomoção se dá por tração diferencial. Em relação ao desenvolvimento do CPS, utilizou-se o *framework* ROS como intermediador de comunicação e foram utilizadas as linguagens de programação *Python* e *C++* no desenvolvimento dos algoritmos dos agentes de *software* e dos robôs. Para o gerenciamento das tarefas foi criado um agente gerenciador e para permitir a visualização de elementos virtuais e reais das execuções experimentais foi criado um agente supervisor. Os experimentos foram conduzidos em um ambiente real construído em escala laboratorial.

De acordo com as características do trabalho atual detalhadas anteriormente e sintetizadas na Tabela 1, são apresentadas nas Tabelas 2 e 3 uma comparação dos trabalhos relacionados acima descritos com as características principais do trabalho desenvolvido.

2.5 CONSIDERAÇÕES

O presente capítulo apresentou os principais conceitos relacionados aos sistemas multirrobô, navegação autônoma e de sistemas ciber-físicos, expondo uma breve contextualização sobre suas definições.

Tabela 1 – Características dessa dissertação.

Característica	Esse trabalho
1 - Tipo do SMR	Heterogêneo
2 - Arquitetura do SMR	Centralizado
3 - Arquitetura de construção e programação do SMR	<i>Open Source</i>
4 - Protocolo de comunicação com o SMR	Wi-Fi
5 - Locomoção do SMR	Diferencial
6 - Tecnologias de implementação do CPS	ROS, <i>Python</i> , C++
7 - Atuação de um Agente Gerenciador no CPS	Sim
8 - Atuação de um Agente Supervisório no CPS	Sim
9 - Ambiente Real em escala laboratorial	Sim

Fonte: Autoria própria.

Tabela 2 – Comparação com trabalhos relacionados (Parte 01).

Característica	(SEMWAL <i>et al.</i> , 2015)	(XU <i>et al.</i> , 2019)	(BEMANI; BJORSELL, 2020)
1	Homogêneo	Homogêneo	Homogêneo
2	Descentralizado	Descentralizado	Descentralizado
3	Proprietário e <i>Open Source</i>	<i>Open Source</i>	<i>Open Source</i>
4	<i>Bluetooth</i>	<i>Wi-Fi</i>	<i>Bluetooth</i>
5	Diferencial	Omnidirecional	<i>Ackerman</i>
6	Prolog, C++	Não especificado	Matlab, <i>Python</i>
7	Sim	Sim	Sim
8	Não	Não	Sim
9	Sim	Sim	Sim

Fonte: Autoria própria.

Tabela 3 – Comparação com trabalhos relacionados (Parte 02).

Característica	(ESCOBAR <i>et al.</i> , 2020)	(ROHRICH <i>et al.</i> , 2021)	(SHIN <i>et al.</i> , 2021)
1	Homogêneo	Homogêneo	Homogêneo
2	Descentralizado	Centralizado	Descentralizado
3	<i>Open Source</i>	<i>Open Source</i>	Proprietário
4	<i>Wi-Fi</i>	<i>Wi-Fi</i>	<i>Bluetooth</i>
5	Omnidirecional	Vibração Diferencial	Diferencial
6	ROS, <i>Python</i> , C, C++	ROS,	<i>Python</i>
7	Sim	Sim	Não
8	Não	Sim	Não
9	Sim	Sim	Sim

Fonte: Autoria própria.

Por fim, foi apresentada uma comparação entre abordagens que utilizam múltiplos robôs inseridos em sistemas ciber-físicos, onde o trabalho atual se diferencia pela forma com que foi desenvolvido, utilizando tecnologias *open source* e possibilitando a atuação de agentes que tornam a proposta desenvolvida expansível com a fácil inclusão de elementos virtuais durante a execução dos trabalhos. O próximo capítulo apresenta com detalhes toda a metodologia utilizada tanto na criação dos agentes que fazem parte do CPS quanto dos robôs desenvolvidos.

3 AMBIENTE CIBER-FÍSICO E ROBÔS DESENVOLVIDOS

Neste capítulo é apresentado o ambiente com características ciber-físicas desenvolvido bem como as especificidades dos múltiplos robôs construídos. Com o intuito de se executar experimentos em escala laboratorial e de se replicar características básicas de um ambiente ciber-físico, foi proposta a criação de um ambiente que realiza o controle da navegação de múltiplos robôs móveis em um ambiente real onde elementos virtuais e físicos são utilizados para interagir entre si.

3.1 VISÃO GERAL DA ARQUITETURA DESENVOLVIDA

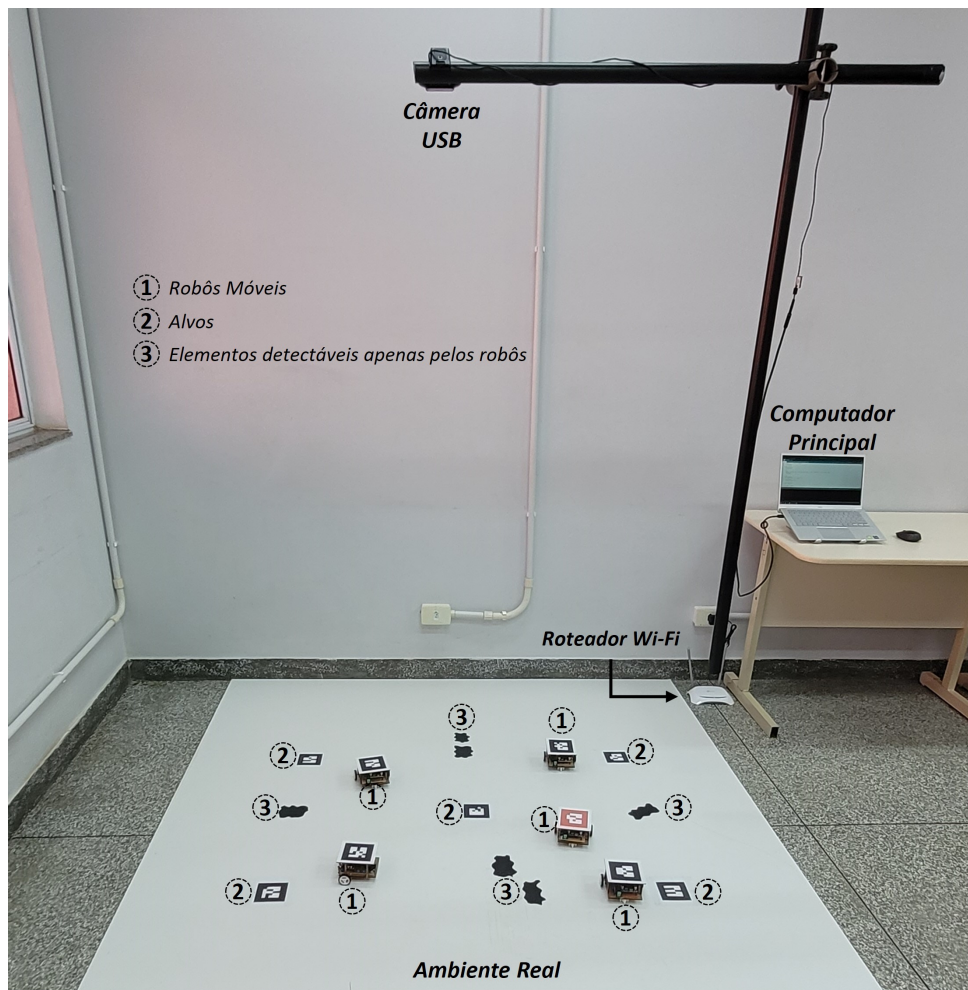
Visando o desenvolvimento CPS proposto e principalmente a execução de experimentos em escala laboratorial, um ambiente físico foi criado. Seus detalhes são vistos na Figura 3, que apresenta os cenários reais de trabalho. A área experimental que servirá de ambiente para os múltiplos robôs é composta de uma chapa de MDF de aproximadamente 1,8 m x 1,8 m correspondendo a região onde ocorre a movimentação de todos os elementos envolvidos na execução do SMR. Um computador é utilizado para o gerenciamento das tarefas dos robôs móveis, via *framework* ROS. Uma câmera USB é conectada a ele com o intuito de realizar a captura de imagens da área experimental a fim de auxiliar no processo de localização dos robôs, alvos e demais objetos dotados de marcadores *Augmented Reality Tag* (ARTag) (FIALA, 2005). A Figura 3 também apresenta manchas em preto espalhadas pela área experimental, que representam elementos que devem ser detectados pelos robôs. A comunicação ocorre apenas entre os robôs e o computador principal via rede *Wi-Fi* por intermédio do roteador presente na imagem.

O ambiente criado é preparado para trabalhar com a inserção dinâmica de elementos reais, que são: robôs móveis, alvos e obstáculos, desde que estejam dotados de marcadores ARTag. Também está preparado para aceitar elementos virtuais, dos tipos alvos e obstáculos, sejam eles estáticos ou dinâmicos.

Tanto alvos quanto obstáculos são elementos reais ou virtuais. Os alvos representam as estações de trabalho, estações de recarga, áreas a serem protegidas, objetos a serem capturados, dentre outros objetos, de acordo com a aplicação do SMR. Os elementos que constituem os obstáculos são formados pelos próprios robôs, áreas a serem evitadas, paredes virtuais, dentre

outros. Vale destacar que tanto os alvos quanto os obstáculos são elementos que se movem ou são fixos. Existem ainda elementos que são reais ou virtuais detectados apenas pelo robôs que tem o potencial de prejudicar a plena navegação dos robôs pelo ambiente como: buracos, manchas de óleo, degraus, objetos a serem carregados, limpos além de outros itens que possam interagir com os robôs a depender da configuração experimental adotada. Estes elementos correspondem às manchas pretas mostradas na Figura 3.

Figura 3 – Visão geral do ambiente criado.

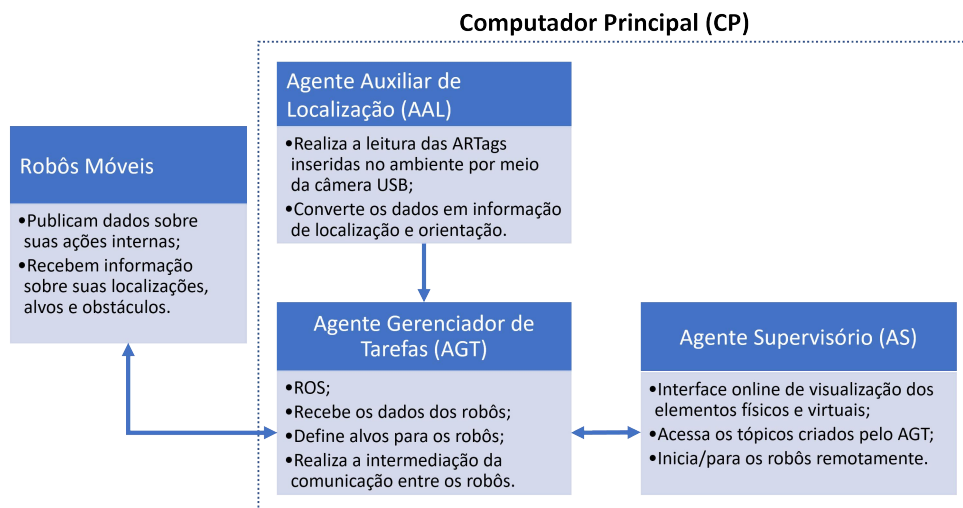


Fonte: Autoria própria.

De acordo com a arquitetura de cinco camadas (5C) para um CPS definida por (LEE; BAGHERI; KAO, 2015) e discutida na seção 2.3, o ambiente criado apresenta apenas as 3 camadas: conexão, conversão e cibernética. Dessa forma, foi desenvolvido um SMR com robôs móveis reais de baixo custo e agentes de *software* instalados em um mesmo computador, denominado Computador Principal (CP). Esses agentes são integrados de forma coordenada e compõem o CPS desenvolvido. São eles: o Agente Auxiliar de Localização (AAL) que realiza a leitura de marcadores inseridos nos robôs, alvos e demais objetos que possam interagir com o

ambiente, ele é responsável por gerar informações para os robôs a respeito de sua localização e orientação no ambiente; o Agente Gerenciador de Tarefas (AGT) que utiliza o *framework* ROS para intermediar a comunicação entre o SMR e demais agentes e é o responsável por gerenciar e conduzir os trabalhos colaborativos durante as execuções de tarefas, nele são criados mensagens e tópicos ROS para serem utilizados por todo o sistema; e o Agente Supervisório (AS) que fornece aos usuários humanos uma visualização amigável dos elementos reais e virtuais inseridos dentro de uma determinada execução de trabalho. A Figura 4 apresenta as características e atribuições principais do SMR e dos agentes de *software* do CPS.

Figura 4 – Arquitetura do ambiente com características ciber-físicas desenvolvido.



Fonte: Autoria própria.

Os agentes foram programados em algoritmos separados, e apesar de neste trabalho residirem no computador principal, podem ser executados em equipamentos distintos, possibilitando uma melhor escalabilidade e flexibilidade. É apresentado nas seções seguintes o funcionamento e configuração de cada um dos elementos que integram a arquitetura desenvolvida.

3.2 ROBÔS MÓVEIS DESENVOLVIDOS

Os robôs reais que compõem o SMR desenvolvido e interagem com o ambiente com características ciber-físicas, são robôs móveis de pequeno porte, baixo custo e de simples construção. Eles possuem apenas habilidades básicas de comunicação e de locomoção e foram desenvolvidos para aplicação em experimentos em escala laboratorial.

Os robôs desenvolvidos são do tipo diferencial, possuem dois atuadores, um em cada roda e uma esfera deslizante como apoio de sua estrutura, seu corpo foi feito utilizando chapas de

madeira de MDF cortadas a laser e os suportes dos motores foram desenvolvidos em impressão 3D. Como cada robô necessita de conexão sem fio, foi utilizado o microcontrolador ESP32 para embarcar sua codificação. Os demais componentes utilizados na construção dos robôs são mostrados na Tabela 4.

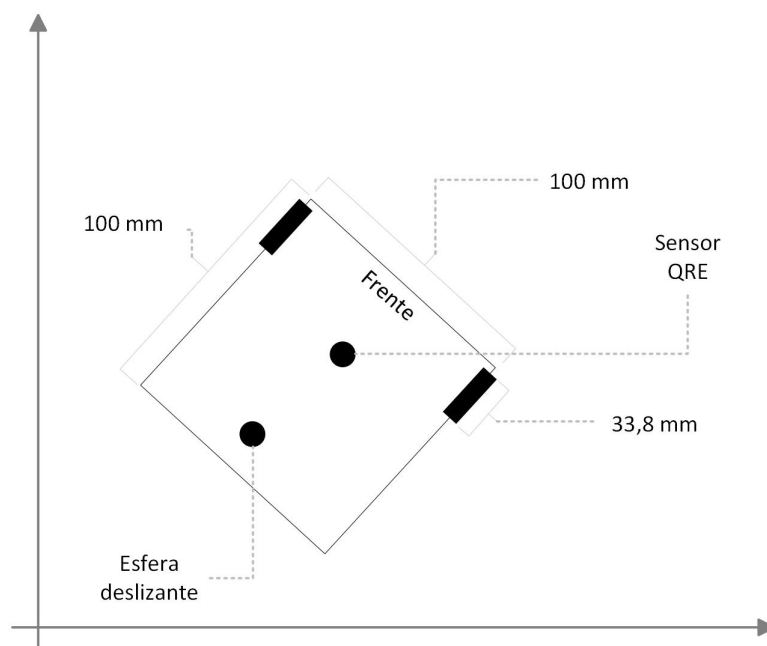
Tabela 4 – Componentes de cada robô.

Qtde	Componente
1	ESP32
2	Micro-motor DC 6V 30 RPM (N20)
2	Rodas pequenas compatíveis com o motor
1	Módulo Regulador de Tensão LM2596
1	Mini Ponte H L298N
1	Bateria 9v
1	Sensor QRE

Fonte: Autoria própria.

Todos os robôs móveis utilizados neste trabalho são de pequeno porte e compartilham as mesmas características físicas, suas dimensões são apresentadas com detalhes na Figura 5.

Figura 5 – Dimensões dos robôs desenvolvidos.

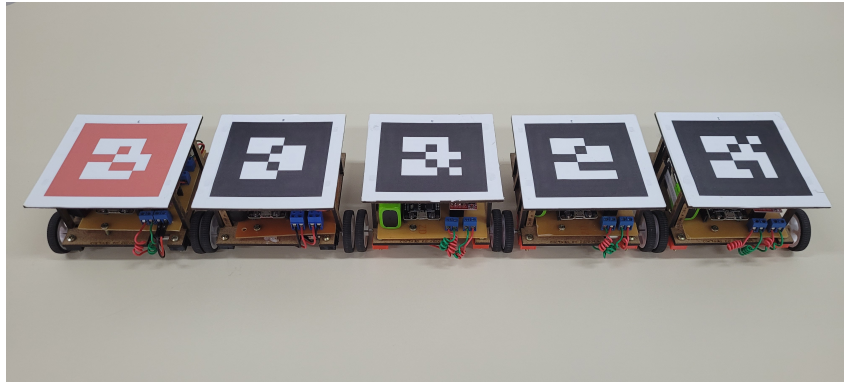


Fonte: Autoria própria.

Marcadores ARTag individuais foram posicionados na parte superior dos robôs para permitir o reconhecimento individual de cada robô pelo CPS. A Figura 6 apresenta os robôs móveis desenvolvidos utilizados neste trabalho. Mesmo possuindo características físicas semelhantes (mesmo *hardware*), o SMR desenvolvido é do tipo heterogêneo, pois diferentes habilidades são

conferidas de forma virtual a cada um dos robôs de acordo com as tarefa a serem realizadas por cada um deles.

Figura 6 – Robôs móveis desenvolvidos.



Fonte: Autoria própria.

Como a comunicação entre os agentes do CPS desenvolvido é intermediada pelo *framework* ROS, conforme já mostrado na Figura 4, os robôs devem ser compatíveis com ele. O *framework* ROS deve estar instalado em um dispositivo, geralmente um laptop ou computador de mesa, e opera como um servidor geral da aplicação, permitindo que outros dispositivos com ROS instalado também possam se conectar a este mestre enviando e recebendo informações (WEST *et al.*, 2018).

O microcontrolador utilizado nos robôs móveis desenvolvidos neste trabalho não é capaz de suportar a instalação do *framework* ROS, uma vez que os robôs foram projetados para serem pequenos, leves e demandarem pouca energia para o seu funcionamento. Para que robôs móveis possam se conectar e acessar os tópicos ROS, sem necessitar de instalação nativa, foi utilizado o pacote *rosserial_server*¹, que permite que microcontroladores possam se comunicar com o ROS executando em um dispositivo central. O CPS através de um agente gerenciador de tarefas que será descrito a seguir, executa o pacote que cria um soquete de rede visando permitir as conexões, via *WI-FI*, dos microcontroladores situados nos robôs móveis.

3.2.1 Habilidades de *software* dos robôs

Como os robôs desenvolvidos não possuem nenhum sensor físico que lhes forneça a capacidade de se auto localizarem dentro da área experimental, cada um deles recebe do AGT as coordenadas de sua localização, dos obstáculos e de seu alvo atual, além de informações sobre

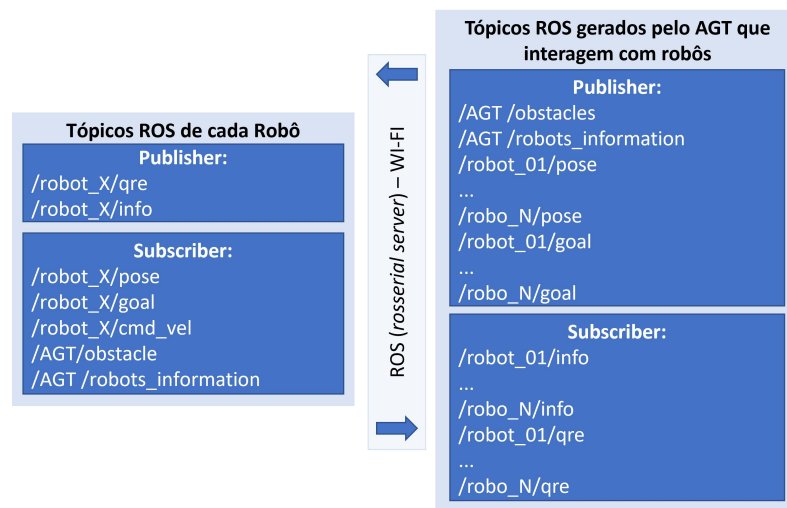
¹ http://wiki.ros.org/rosserial_server

os outros robôs inseridos no ambiente.

Os robôs publicam, via tópico individual */robot_X/qre*, os valores de leitura de um sensor que mede o nível de refletância de luz na parte inferior de sua estrutura. Tal sensor foi desenvolvido com o intuito dos robôs realizarem identificações a respeito do ambiente, visando o compartilhamento desta informação com o restante do CPS durante sua navegação. Os robôs também publicam informações sobre suas ações internas, como dados de navegação, tomadas de decisão e ações dos controladores utilizados.

Uma vez recebendo sua posição e orientação atualizada em relação ao ambiente de trabalho, via tópico */robot_X/pose*, os robôs desenvolvidos tem o conhecimento de sua coordenada atualizada. A partir do momento que o robô identifica uma mensagem publicada no tópico */robot_X/goal*, via AGT, realiza o processamento diretamente em seu *hardware* visando se locomover até a coordenada recebida. Cada robô se move de forma independente utilizando as informações de velocidade linear e angular que são publicadas no tópico */robot_X/cmd_vel* a fim de permitir sua locomoção. A Figura 7 apresenta a estrutura dos tópicos ROS configuradas dentro de cada robô e como ela se relaciona com os tópicos do AGT.

Figura 7 – Tópicos ROS utilizados para comunicação de robôs com o AGT.



Fonte: Autoria própria.

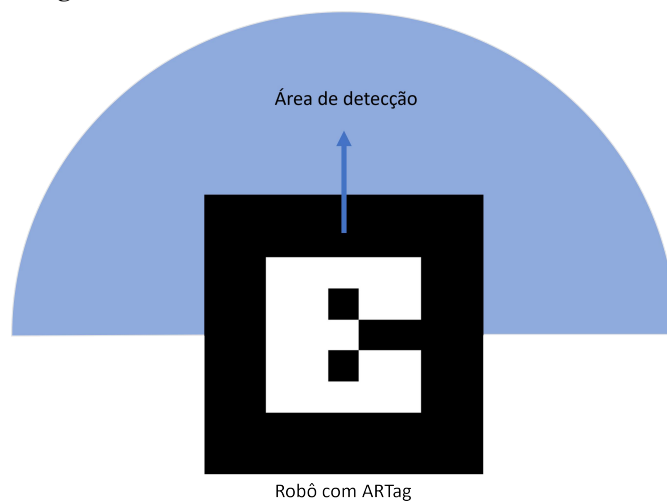
3.2.2 Sensoriamento Virtual

Considerando que locomoção e desvio de obstáculos são habilidades básicas de um robô do SMR, e que os robôs móveis desenvolvidos não possuem sensores reais para sua auto localização e nem sensores que detectam obstáculos a sua frente, foi desenvolvido um sensor

virtual para proporcionar aos robôs a capacidade de identificação de obstáculos próximos a sua posição, sejam eles reais ou virtuais.

Uma vez que cada robô recebe as coordenadas de sua posição e as coordenadas de obstáculos inseridos no ambiente, foi desenvolvido um sensor virtual embarcado dentro de cada microcontrolador que cria uma zona de segurança para os robôs. De posse de suas coordenadas, um semicírculo virtual é criado na parte frontal dos robôs a partir do ponto central dos marcadores ARTag, conforme apresentado na Figura 8. A região em azul representa área de detecção do sensor virtual, criando assim uma zona de segurança para cada robô.

Figura 8 – Funcionamento do sensor virtual dos robôs.



Fonte: Autoria própria.

Cada robô, após ter acesso a lista de obstáculos via tópico */AGT/obstacle*, compartilhados pelo AGT, calcula a distância entre ele e os obstáculos, feito isso, é averiguado se algum dos obstáculos está no interior da zona de detecção do sensor virtual. Em caso afirmativo, a distância entre robô e obstáculo é normalizada entre 0 (obstáculo próximo ao centro robô) e 1 (obstáculo próximo ao limite da área de detecção). Quando é identificado mais de um obstáculo dentro da zona de detecção apenas o mais próximo ao centro do robô é considerado.

Através do sensoriamento virtual, os robôs desenvolvidos neste trabalho tem a capacidade de identificar obstáculos que por ventura possam estar em seu caminho, além de atribuir um risco de colisão para as detecções. A partir deste ponto, os robôs podem executar as estratégias de controle desenvolvidos para desvio de obstáculo e navegação segura.

3.2.3 Controladores utilizados na navegação

Com o objetivo de proporcionar a movimentação de maneira segura dos robôs pela área experimental, sem colidir com obstáculos, sejam eles estáticos, dinâmicos, reais ou virtuais, foi desenvolvida uma estratégia de controle de navegação autônoma buscando guiar os robôs com segurança a partir de sua coordenada atual até uma coordenada de destino, atribuída pelo AGT via tópico individual */robot_X/goal*. Além disso, para algumas situações, os robôs utilizam informações provenientes dos outros robôs envolvidos no SMR para a tomada de decisão.

Cada robô depende do AGT para conhecer sua localização, o AGT por sua vez recebe esta informação do AAL, que realiza a leitura de todas os marcadores ARTag que estão em seu campo de visão. Com eventuais atrasos de leitura e falhas na captura das imagens podem ocorrer, são atribuídas velocidades lineares fixas, deixando a cargo dos controladores dos robôs atuar apenas na movimentação angular.

Visando movimentar o robô de um ponto de origem até um de destino de forma a desviar de possíveis obstáculos, foram desenvolvidos dois controladores que são embarcados em seu microcontrolador: um controlador baseado em Proporcional, Integral e Derivativo (PID) convencional, na configuração Proporcional-Derivativo (PD), que tem por objetivo atuar no acionamento dos motores para manobrar o robô em direção a sua coordenada de destino e um controlador *Fuzzy Mamdani*, que atua nas tomadas de decisões relacionadas ao desvio de obstáculos.

Em relação ao controlador PD, é importante enfatizar que seus ganhos proporcional ($K_p = 1,5$) e derivativo ($K_d = 0,5$) foram ajustados com base em observações de experimentos preliminares realizados, com o objetivo de diminuir manobras abruptas e também garantir a leitura e identificação do marcador ARTag pela câmera USB durante os experimentos.

Uma vez que a caixa de redução dos motores utilizados aplica uma rotação de 30 voltas por minuto, ao mover os robôs para frente em potência máxima, verificou-se que sua velocidade linear máxima é de 5,4 centímetros por segundo, o que é compatível com as dimensões dos robôs e da área experimental, resultando em uma movimentação lenta pelo ambiente. Dessa forma, essa velocidade foi utilizada quando o controlador PD entra em ação, isto é, em navegação livre de obstáculos. Nas situações onde ocorrem as manobras de desvio de obstáculo, atribui-se uma velocidade linear de 65% do valor máximo, ou seja de 3,51 centímetros por segundo, para garantir um desvio de forma segura, uma vez que o obstáculo também pode ser outro robô

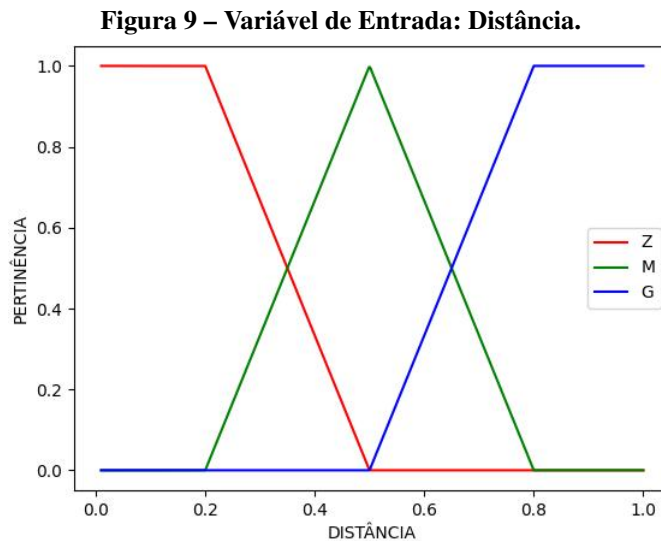
(obstáculo dinâmico).

Como o controlador PD atua apenas na movimentação angular, o erro considerado corresponde à diferença entre a orientação angular atual do robô em relação ao ambiente e a direção do seu destino, dada pela Equação 1. A ação de controle resultante é a velocidade angular desejada, que visa alinhar o robô ao seu objetivo.

$$erroAngular = atan2(y_{destino} - y_{robo}, x_{destino} - x_{robo}) \quad (1)$$

O controlador de desvio de obstáculo é um controlador *Fuzzy Mamdani*, com defuzzificação por meio do centro de gravidade. Sua modelagem e implementação foi feita utilizando a biblioteca *Embedded Fuzzy Logic Library* (EFL), desenvolvida por Kridi *et al.* (2011), e o código resultante foi embarcado no microcontrolador de cada robô.

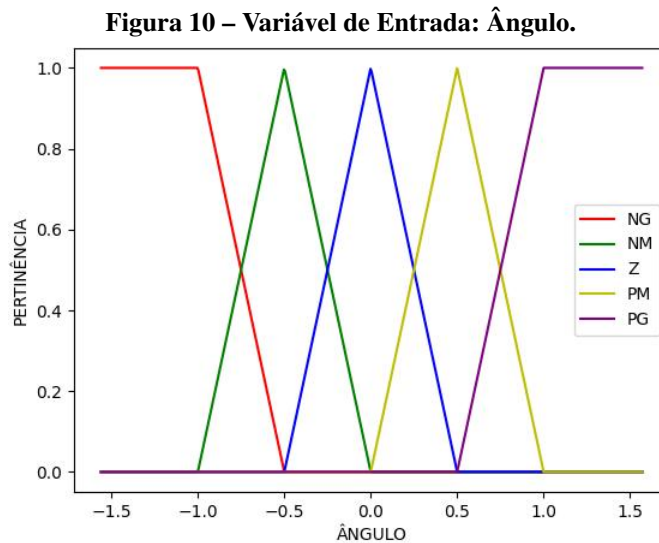
Para o controlador *Fuzzy*, foram consideradas duas variáveis de entrada e uma de saída. A primeira entrada é a distância normalizada entre robô e obstáculo, conforme foi descrito na seção 3.2.2, suas funções de pertinência são apresentadas na Figura 9, cujas variáveis linguísticas são: Z = Zero; M = Médio e G = Grande.



Fonte: Autoria própria.

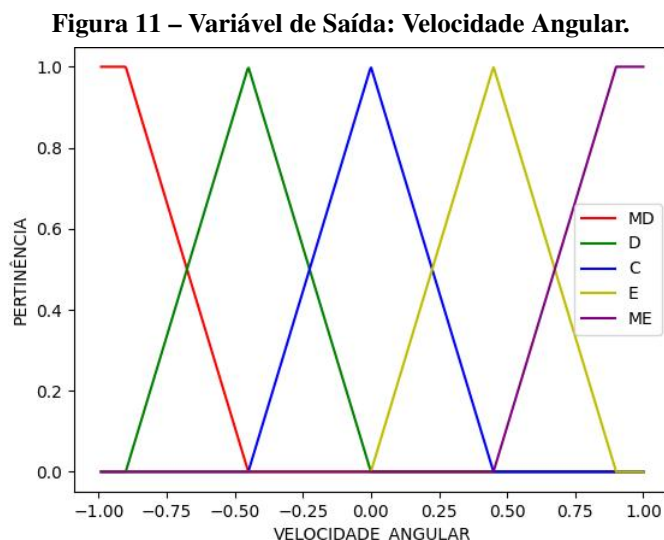
A segunda variável de entrada representa a diferença angular entre a orientação do robô e a direção do obstáculo detectado (α). Como o sensor virtual abrange um semi-círculo na parte da frente dos robôs, essa variável considera valores de $-\frac{\pi}{2}$ à $\frac{\pi}{2}$, onde um obstáculo à direita do robô é representado por um valor negativo e um obstáculo a sua esquerda é um valor positivo. Valores próximos a 0 caracteriza que o obstáculo está à frente do robô. A Figura 10 apresenta

as funções de pertinência para esta variável, cujas variáveis linguísticas são: NG = Negativo Grande; NM = Negativo Médio; Z = Zero; PM = Positivo Médio e PG = Positivo Grande.



Fonte: Autoria própria.

Após o processo de inferência é obtido como saída o valor da velocidade angular, em rad/s, que foi normalizada entre -1 e 1, onde valores negativos representam manobras à direita e valores positivos manobras à esquerda. A Figura 11 apresenta a variável com suas funções de pertinência, cujas variáveis linguísticas são: MD = Muito à Direita; D = Direita; C = Centro; E = Esquerda e ME = Muito à Esquerda.



Fonte: Autoria própria.

A modelagem das variáveis de entrada e de saída foi desenvolvida para garantir uma navegação segura dentro da área experimental, possibilitando aos robôs desviarem de obstáculos estáticos e dinâmicos (outros robôs). É detalhada na Tabela 5, a base de regras que possibilitou

ao controlador *Fuzzy* implementar as ações de controle necessárias ao desvio de obstáculos. A modelagem da base de regras visou criar uma resposta equilibrada com uma simetria na velocidade angular resultante, referente às manobras para a direita ou esquerda, a única exceção acontece para a regra cujas variáveis distância e ângulo são ‘Z’, onde a velocidade angular projetada foi ‘MD’, forçando uma manobra para a direita quando o robô está muito próximo do obstáculo e bem a frente dele. A superfície de resposta do controlador *Fuzzy* é apresentada na Figura 12.

Tabela 5 – Base de regras do controlador *Fuzzy*.

Entradas		Saída
Distância	Ângulo	Velocidade Angular
Z	NG	ME
Z	NM	ME
Z	Z	MD
Z	PM	MD
Z	PG	MD
M	NG	E
M	NM	ME
M	Z	C
M	PM	MD
M	PG	D
G	NG	E
G	NM	E
G	Z	C
G	PM	D
G	PG	D

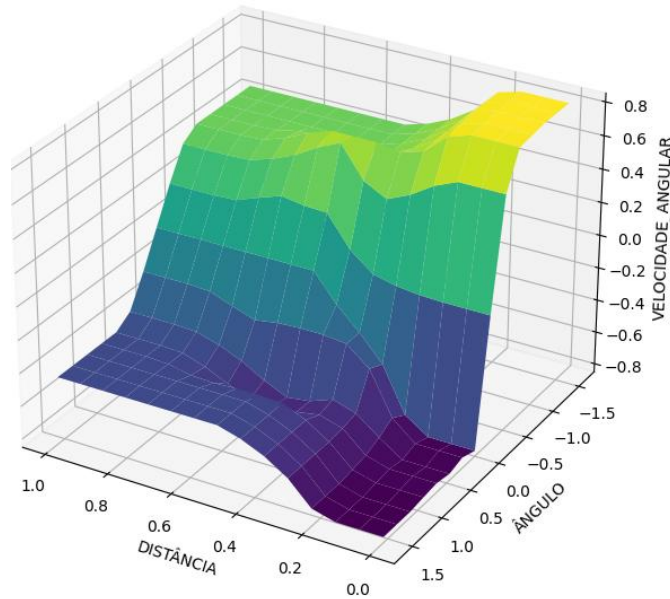
Fonte: Autoria própria.

Após a implementação dos controladores verificou-se que o processo de transição (passagem do comando do robô entre os controladores) entre as ações “navegação livre” e “desvio de obstáculos” não ocorria de forma suave, resultando em manobras abruptas durante os desvios de obstáculos. Por essa razão, se fez necessária a implementação de uma função de transição, Equação 2, entre os controladores, inspirada em (ALMEIDA *et al.*, 2019).

$$u = \beta \cdot u_{PD} + (1 - \beta) \cdot u_{fuzzy} \quad (2)$$

Onde u representa a ação de controle referente à velocidade angular; u_{PD} e u_{fuzzy} são as ações de controle computadas pelos controladores PD e *Fuzzy*, respectivamente; e $\beta \in [0,1]$ é a distância entre o centro do robô e o obstáculo detectado, mesmo valor utilizado para a entrada *distancia* do controlador *Fuzzy*. Desta forma, quando mais próximo do obstáculo ($\beta \approx 0$) maior será a atuação do controlador *Fuzzy* e quanto mais longe ($\beta \approx 1$) maior a atuação do controlador

Figura 12 – Superfície de resposta do controlador *Fuzzy*.



Fonte: Autoria própria.

PD. Uma vez que nenhum obstáculo estiver na zona de detecção do sensor virtual apenas o controlador PD atua.

Por fim, após a implementação dos controladores, cada robô é capaz de se locomover no ambiente por meio dos algoritmos de controle embarcados em seu *hardware*, que os permitem alcançar um alvo (coordenada x,y) e desviar de obstáculos reais ou virtuais, sejam eles estáticos ou dinâmicos. Estas ações são possíveis uma vez que é garantido o acesso a informação de suas respectivas posições no ambiente. Cada robô recebe do AGT sua posição e orientação por meio do tópico `/robot_X/pose` e uma lista de obstáculos presentes no ambiente pelo tópico `/environment/obstacle`, como já visto na Figura 7.

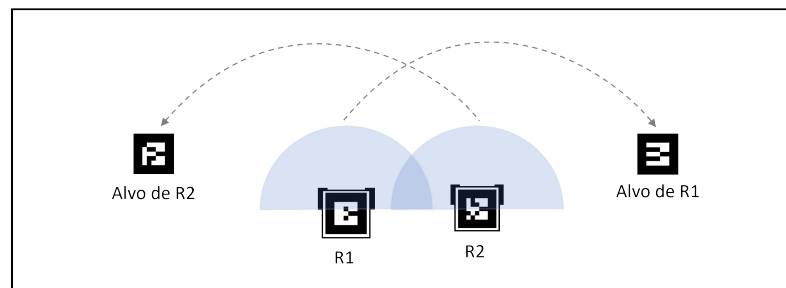
3.2.4 Estratégia de comunicação entre robôs

Os robôs do SMR realizam tarefas de forma sincronizada e cooperativa, Para isto, implementou-se um mecanismo de comunicação em que, com o auxílio do AGT, as decisões individuais dos robôs possam ser tomadas de forma autônoma dentro do seu fluxo de trabalho.

Uma vez que todos os membros do SMR tem acesso ao tópico ROS `/AGT/robots_information`, um robô tem a capacidade de conhecer informações provenientes de outros robôs, assim qualquer informação compartilhada auxilia no processo de tomada de decisão dos robôs. É mostrada na Figura 13, uma situação que geralmente ocorre em um ambiente compartilhado

entre múltiplos robôs móveis, em que os robôs R1 e R2 devem alcançar seus respectivos alvos. Entretanto, conforme se dirigem aos seus objetivos, ocorre uma aproximação entre eles e, conseqüentemente, há um risco de colisão (detectado com base no sensoriamento virtual, representado pela área do semi-círculo em destaque na figura 13). Um agravante nesta situação é que, coincidentemente, eles se encontram com orientação angular (alinhados para a parte superior da figura 13) semelhantes, fato que leva o controlador *Fuzzy* a atuar para o distanciamento e, quando não mais detectados pelos sensores, ocorre uma nova aproximação, fazendo com que este impasse não seja superado pelos robôs.

Figura 13 – Situação onde a estratégia de comunicação foi utilizada.



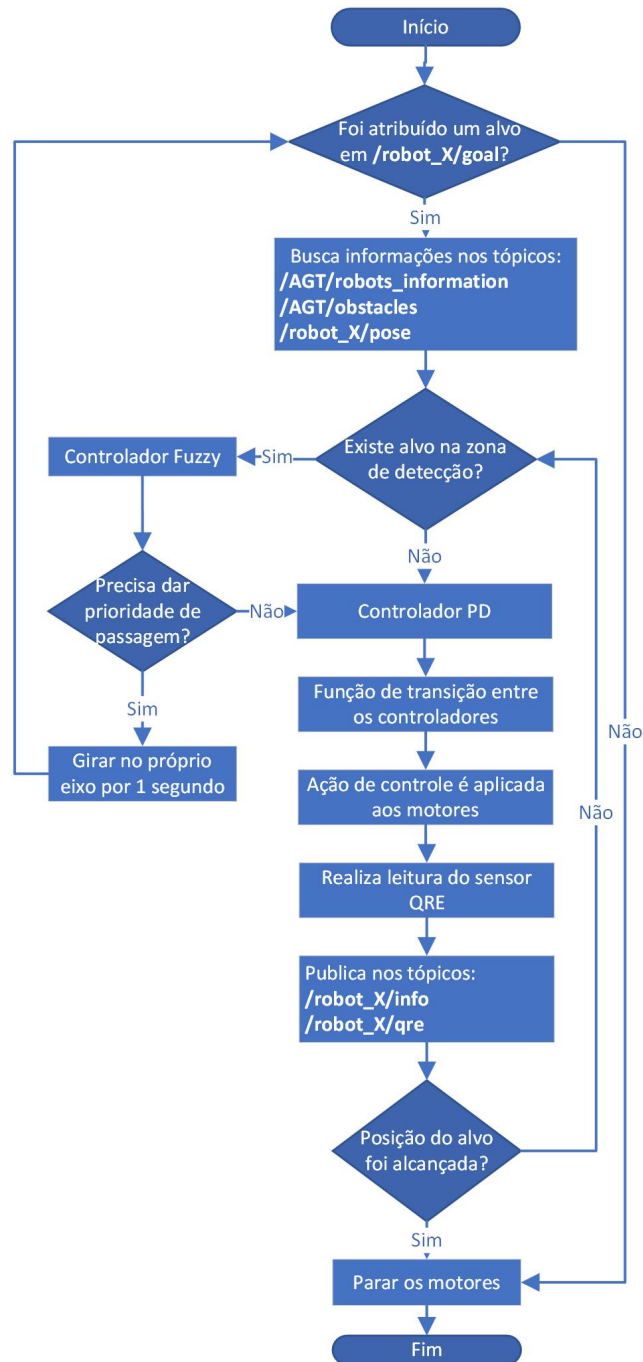
Fonte: Autoria própria.

Para contornar este problema de navegação, adota-se neste trabalho uma política de decisão que considera as informações compartilhadas entre robôs, por meio dos tópicos ROS explicados anteriormente. Para a tomada de decisão, os robôs envolvidos comparam seus respectivos indicadores de *contador* (conforme descrito no Quadro 1), e o que apresentar o maior valor tem prioridade de passagem, enquanto o outro robô executa uma manobra de giro em torno de seu próprio eixo por 1 segundo (tempo suficiente para que o robô prioritário faça sua passagem de forma segura). Caso o valor da variável *contador* seja o mesmo para ambos os robôs, o critério de desempate adotado é o identificador interno (*id*) de cada robô, onde o robô que possuir o maior valor possui prioridade de passagem.

No fluxograma exposto na Figura 14, é possível verificar como cada um dos robôs inseridos no SMR toma suas decisões bem como em que momento ocorre a função de transição entre os controladores e também o momento em que a verificação da necessidade de prioridade de passagem é tomada.

Ao fim dessas implementações observou-se que o SMR desenvolvido desempenha com sucesso tarefas em que devem sair de sua posição de origem e ir até uma coordenada de destino, desviando de obstáculos estáticos e dinâmicos.

Figura 14 – Fluxograma do sistema de navegação.



Fonte: Autoria própria.

3.3 AGENTE AUXILIAR DE LOCALIZAÇÃO (AAL)

Considerando que o processo de auto localização de robôs móveis no ambiente é fundamental para a realização de tarefas, o AAL foi desenvolvido para dar suporte aos robôs na definição de sua localização dentro do ambiente, uma vez que os robôs móveis desenvolvidos neste trabalho não possuem sensores capaz de realizar esta tarefa.

O AAL usa a imagem fornecida pela câmera USB para realizar a identificação de ARTag, inseridos no ambiente real, convertendo estes dados em informações de posição e orientação. Ele usa o *framework* ROS para processar as informações por meio do pacote *ar_track_alvar*². Sua função não se resume apenas a auxiliar os robôs no processo de localização pelo ambiente, a depender do experimento a ser conduzido, também realiza a leitura de marcadores que representam alvos, obstáculos estáticos ou demais objetos dotados de um marcador ARTag. De fato, o AAL faz um mapeamento do ambiente identificando para o CPS cada ponto de interesse do ambiente real.

Após a identificação e tratamento dos marcadores via pacote *ar_track_alvar*, é gerada uma lista de marcadores no tópico *ar_pose_marker*, o papel do AAL é processar essas informações e criar tópicos específicos para cada robô e alvo, contendo informações de posição e orientação em relação ao ambiente real, os quais foram configurados para serem atualizados a uma taxa de 10 Hz. Tal taxa foi implementada com o intuito de não sobrecarregar o CP durante o processamento dos múltiplos marcadores inseridos no ambiente.

O AAL foi codificado para identificar dinamicamente marcadores ARTag inseridos na área experimental. Para diferenciar marcadores de robôs e de alvos, foi definido que são robôs os marcadores que vão de 0 a 10, já para alvos, foi definido o intervalo de 11 a 20. O intervalo definido para robôs e alvos é facilmente configurado, a depender da quantidade de elementos inseridos no ambiente e da necessidade do experimento a ser conduzido.

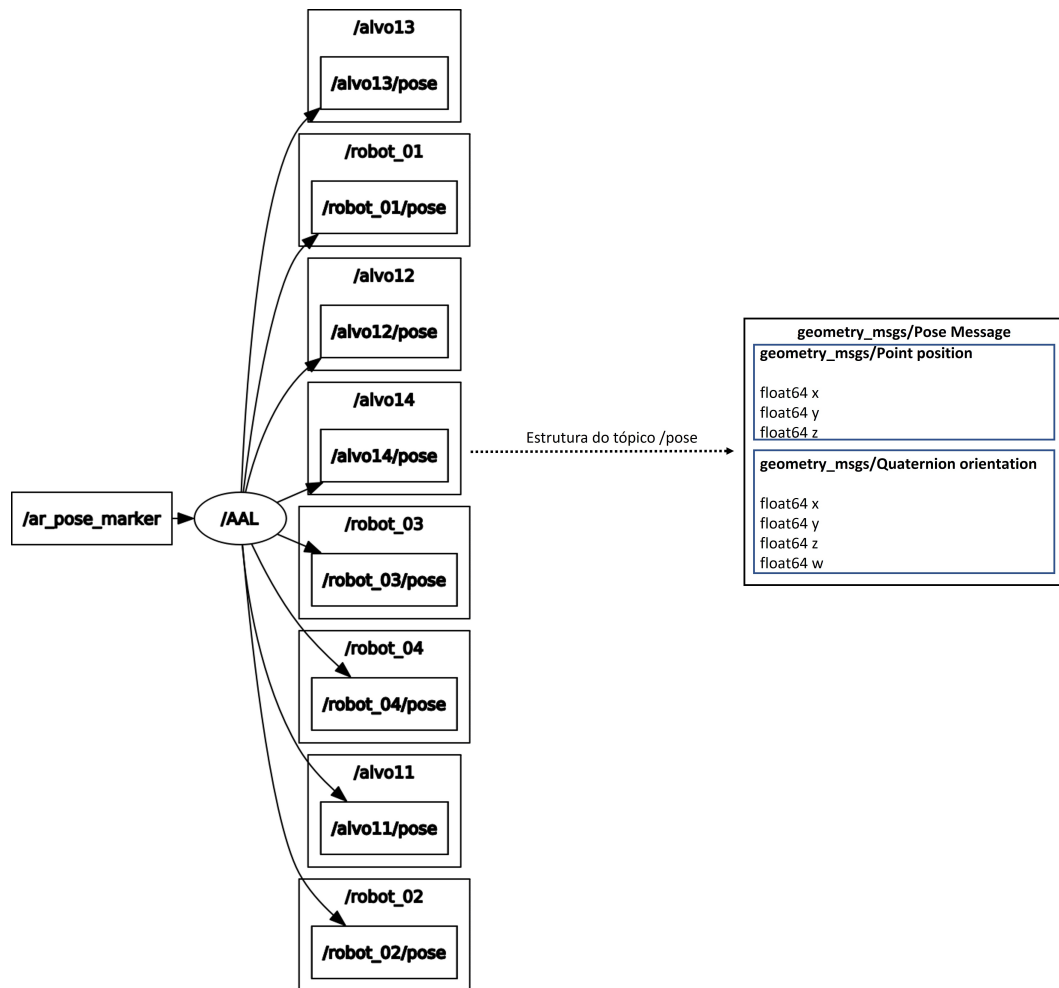
A fim de ilustrar o funcionamento do AAL após a identificação de 4 robôs e 4 alvos, a Figura 15 apresenta um gráfico gerado pelo comando *rqt_graph* referente à estrutura de tópicos ROS gerados por este agente. Também é mostrada a estrutura dos tópicos individuais, que utilizam a mensagem *Pose* padrão do ROS para a definição de posição e orientação.

O AAL é conectado com a AGT de forma unidirecional, pois apenas envia informações a ele. Portanto, para que os robôs recebam as informações referentes a sua localização, é necessária a intermediação do AGT. Na implementação atual, o AAL executa no mesmo computador que o AGT, mas esse agente pode ser usado em qualquer outro dispositivo, por exemplo em um veículo aéreo não tripulado, outro computador ou qualquer dispositivo capaz de realizar capturas de imagens e ser compatível com o *framework* ROS.

É possível implementar o AAL em situações onde a precisão dos sensores de localização embarcados em robôs não possam ser utilizados ou possam sofrer alguma interferência, tais

² http://wiki.ros.org/ar_track_alvar

Figura 15 – Tópicos criados pelo AAL após a identificação de robôs e alvos.



Fonte: Autoria própria.

como áreas internas, subterrâneas, entre outras.

3.4 AGENTE GERENCIADOR DE TAREFAS (AGT)

O AGT centraliza as informações gerais do sistema, processa as informações provenientes do AAL, recebe informações dos robôs, realiza a intermediação da comunicação e gerencia a atribuição de tarefas para o SMR, sendo o principal agente do CPS proposto. Neste trabalho, o AGT utiliza o *framework* ROS para atuar como meio de comunicação para com os demais agentes do CPS.

3.4.1 Mensagens do AGT

Visando a padronização da comunicação entre os agentes e possibilitar a flexibilidade de implementações futuras de expansão do CPS, foi criado um pacote ROS com a capacidade de armazenar os algoritmos responsáveis por delegar tarefas ao SMR e padronizar mensagens utilizadas na comunicação entre agentes e robôs, bem como a integração de elementos reais e/ou virtuais do CPS. Inicialmente foram criadas mensagens ROS (estruturas de dados compostas de variáveis ou objetos que são publicadas por meios de tópicos ROS) que representem as características necessárias para permitir a unificação das informações utilizadas no processo de comunicação.

Como cada robô processa de forma individualizada sua estratégia de controle, conforme já apresentado na Figura 14, para permitir que cada robô envie informações atualizadas sobre seu processamento interno ao AGT, permitindo ao agente utilizar essas informações no gerenciamento de tarefas, no armazenamento dos dados de processamento dos robôs e na análise futura destes dados, foi criada uma mensagem que agrupasse esses dados em um único lugar. O Quadro 1 apresenta a estrutura desta mensagem.

Quadro 1 – Estrutura da Mensagem ROS *Info*.

## Definição da mensagem AGT_msgs/Info.msg	
uint8 id	# identificação interna de cada robô
uint8 status	# status da movimentação do robô
uint8 acao	# ação que o robô está realizando
uint16 setor	# id do alvo que está em processamento
uint8 contador	# indicador de contagem de serviço
int16 controlador	# indicação do controlador utilizado
float32 linearX	# velocidade linear aplicada aos atuadores
float32 angularZ	# velocidade angular aplicada aos atuadores
float32 erroAngular	# entrada do controlador pid
float32 distancia	# entrada 01 do controlador fuzzy
float32 angulo	# entrada 02 do controlador fuzzy
uint16 acao_desvio	# utilizada para indicar ação de desvio

Fonte: Autoria própria.

Considerando que, neste trabalho, a quantidade de robôs do SMR é variável, e que o código embarcado em cada robô permite a troca de informações entre robôs, foi criada uma mensagem agrupando as informações originárias de todos os robôs em um único lugar, sua estrutura é apresentada no Quadro 2. Ao acessar informações vindas desta mensagem, intermediadas via AGT, cada robô tem acesso as informações individuais de todos os outros robôs do ambiente, independentemente da quantidade de robôs inseridos em cada configuração experimental.

Quadro 2 – Estrutura da Mensagem *Lista_info*.

```
## Definição da mensagem AGT_msgs/Lista_info.msg
Info[] informacoes          #informações compiladas dos robôs
```

Fonte: Autoria própria.

Uma vez que as coordenadas identificadas pelo AAL são tanto de robôs quanto de obstáculos e que o próprio AGT tem a capacidade de criar elementos virtuais a serem considerados como obstáculos, foram criadas mensagens que agrupam informações de coordenadas que venham a representar obstáculos dentro do ambiente. Durante a ação de desvio de obstáculos, todos os robôs recebem a mesma lista de obstáculos presentes na área experimental vinda do AGT por meio do tópico */AGT/obstacles*, com isso, cada robô não deve levar em consideração as coordenadas referentes a sua própria posição. Para que ele possa realizar essa diferenciação, é necessário que o AGT, além de enviar uma coordenada para desvio, também envie o seu identificador interno. Assim, foi criada a mensagem *Point_r*, detalhada no Quadro 3, que agrupa informações dos pontos de uma coordenada e o seu identificador. Já a mensagem *Points_r*, detalhada no Quadro 4, possui uma lista de *Point_r*, assim o AGT envia as informações de coordenadas que representam obstáculos aos robôs.

Quadro 3 – Estrutura da Mensagem *Point_r*.

```
## Definição da mensagem AGT_msgs/Point_r.msg
uint16          #identificador de um ponto
geometry_msgs/Point point #coordenadas do ponto
```

Fonte: Autoria própria.**Quadro 4 – Estrutura da Mensagem *Points_r*.**

```
## Definição da mensagem AGT_msgs/Points_r.msg
Point_r[] points # lista de pontos identificados
```

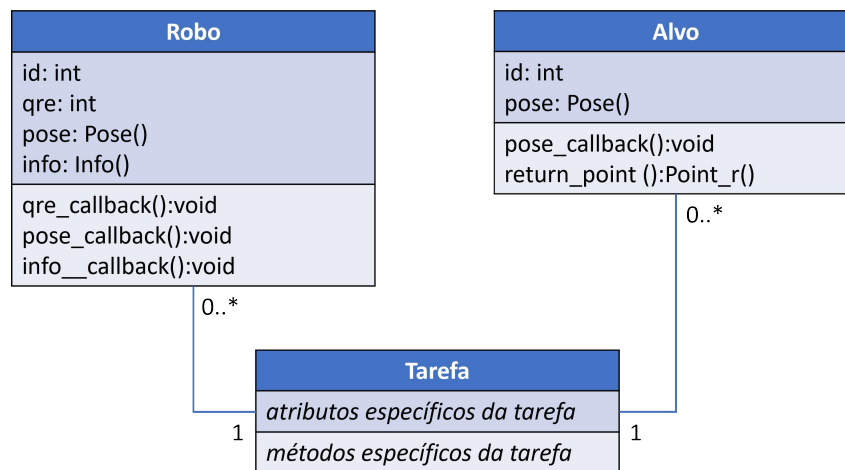
Fonte: Autoria própria.

As mensagens *Point_r* e *Points_r* também podem ser utilizadas em situações em que o AGT tem a função de gerenciar itens identificados por robôs e itens que sofreram alguma interação, além de outras necessidades onde é necessário o armazenamento de uma lista de coordenadas com um respectivo identificador interno.

3.4.2 Gerenciamento de Tarefas

Toda a codificação do AGT foi feita utilizando a linguagem de programação *Python*, juntamente com bibliotecas necessárias para o desenvolvimento integrado ao *framework* ROS. No AGT, cada tarefa tem codificação própria, na qual, em qualquer uma delas é necessário definir o número de robôs e alvos envolvidos além da definição das atribuições do trabalho a ser realizado nesta configuração. Para isso, foram desenvolvidas classes que auxiliam no processo de inclusão de robôs e alvos nos algoritmos que realizam as tarefas. O diagrama da Figura 16 apresenta os atributos e métodos das classes *Robo* e *Alvo*. Assim, ao ser criada uma *Tarefa*, são instanciadas a quantidade necessária de robôs ou alvos presentes na mesma.

Figura 16 – Diagrama de classes utilizadas pelo AGT.

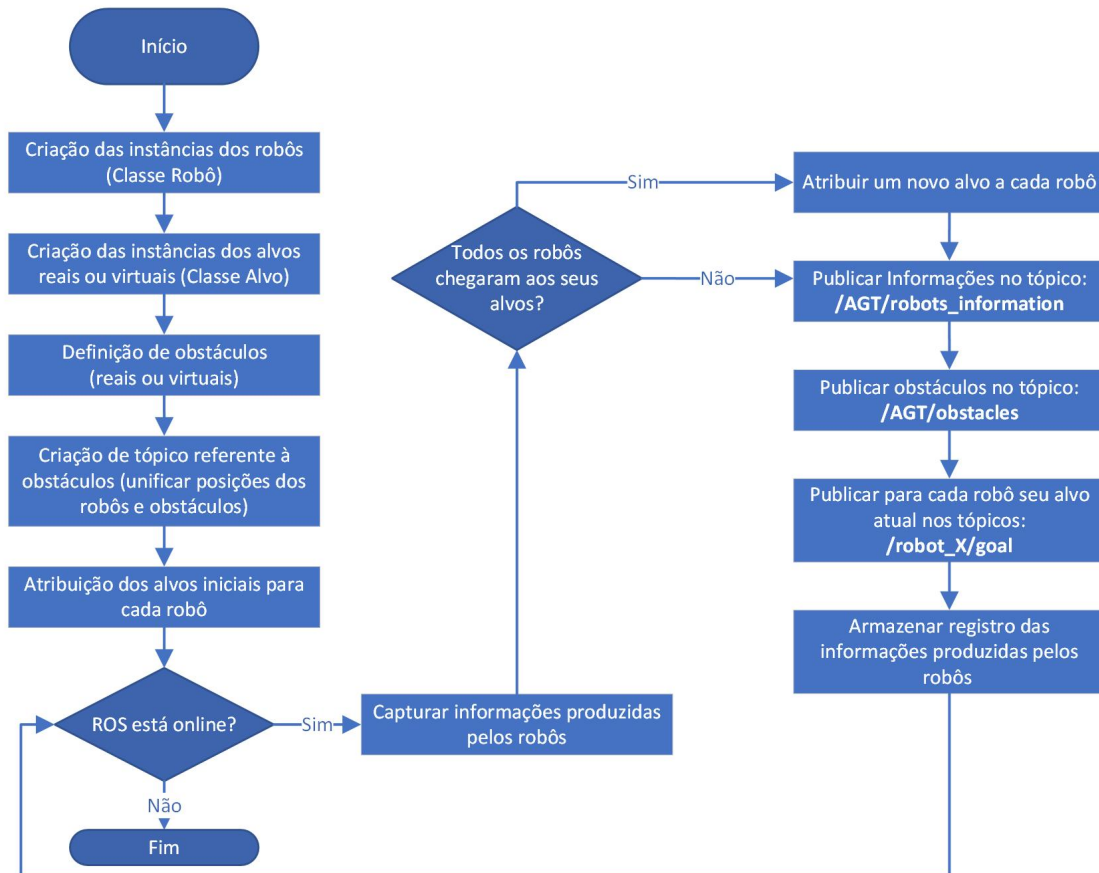


Fonte: Autoria própria.

A classe *Robo* utiliza informações vindas do AAL (*pose*) e produzida pelos próprios Robôs (*id*, *qre* e *info*), as unifica em um único lugar para simplificar a criação e acesso aos tópicos ROS específicos de cada robô. Já a classe *Alvo* sintetiza os atributos referente aos alvos vindos do AAL.

O algoritmo que representa a atribuição de trabalho ao SMR, classe *Tarefa* da Figura 16, deve ser configurado para definir os processos que permitam a comunicação entre os robôs. O fluxograma apresentado na Figura 17 retrata um exemplo de uma situação desenvolvida nesta classe, onde robôs devem visitar ordenadamente alvos e devem trocar suas posições assim que todos cheguem aos seus destinos. O AGT aceita diversas configurações experimentais, sendo necessária a criação e codificação de uma nova classe que realiza o gerenciamento de tarefas para cada uma delas.

Figura 17 – Fluxograma referente à execução de uma tarefa do AGT.



Fonte: Autoria própria.

Os agentes desenvolvidos nesta dissertação se comunicam com o AGT e são previamente configurados para acessarem tópicos ROS produzidos por ele, os quais são mostrados em detalhe na Tabela 6.

Tabela 6 – Tópicos ROS produzidos pelo AGT.

Tópico	Mensagem	Descrição	Agente que utiliza
<i>/AGT/robots_information</i>	Lista_info	Unifica informações vindas dos robôs	Robôs, AGT
<i>/AGT/obstacles</i>	Points_r	Unifica obstáculos geridos pelo AGT	Robôs, AGT e AS
<i>/AGT/goals</i>	Points_r	Unifica os alvos dos robôs	Robôs, AGT e AS
<i>/AGT/identified_obstacles</i>	Points_r	Unifica itens identificados por robôs	Robôs, AGT e AS
<i>/AGT/interacted_items</i>	Points_r	Unifica itens interagidos por robôs	Robôs, AGT e AS

Fonte: Autoria própria.

O AGT se caracteriza como um dos principais agentes do CPS produzido, uma vez que gerencia o controle das tarefas, realiza a intermediação da comunicação entre robôs e mais agentes. Devido a forma que foi estruturado, pode ser interligado a um outro CPS capaz de se comunicar por meio dos tópicos ROS criados e utilizados pelo AGT, permitindo assim a construção de aplicações mais complexas, em que por exemplo vários SMR atuem no mesmo

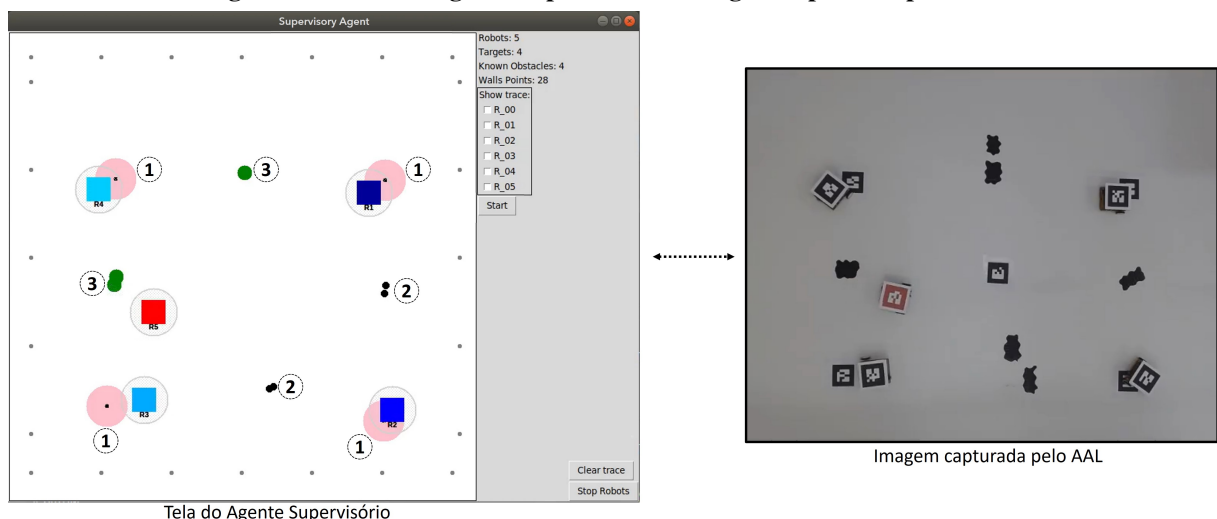
ambiente mas em equipes diferentes.

3.5 AGENTE SUPERVISÓRIO (AS)

O AS consiste em uma interface gráfica que tem o objetivo de permitir uma visualização online dos elementos reais e virtuais envolvidos na realização de uma determinada tarefa, possibilitando a um usuário humano, uma visualização amigável dos elementos envolvidos na execução de tarefas. Para isso, ele acessa os tópicos ROS gerados pelo AGT para proporcionar a visualização dos elementos contidos no ambiente.

A Figura 18 mostra uma captura de tela do AS e da imagem capturada pela câmera do AAL em execução de uma determinada configuração experimental. Os robôs de trabalho são representados por quadrados em tons de azul e identificados pelos rótulos *R1*, *R2*, *R3* e *R4*; um robô de apoio é representado por um quadrado vermelho, rótulo *R5*; os alvos (1) são os círculos rosa; itens detectados pelos robôs (2) são os pontos pretos; itens que foram identificados por robôs e sofreram uma interação (3) são os pontos verdes, e as paredes virtuais são os pontos cinzas ao redor do ambiente. Esta tela ainda permite ao usuário iniciar/parar uma tarefa atribuída pelo AGT aos robôs, visualizar o caminho percorrido por um determinado robô e o quantitativo de alvos, obstáculos conhecidos e os pontos que representam paredes virtuais da área experimental.

Figura 18 – Tela do Agente Supervisório e imagem capturada pelo AAL.



Fonte: Autoria própria.

Para plotar os elementos no AS são acessados os tópicos ROS provenientes do AGT: */AGT/robots_information* para os robôs; */AGT/goals* para os alvos; */AGT/identified_obstacles* para itens identificados por robôs; */AGT/interacted_items* para itens interagidos por robôs e

/AGT/obstacles para os obstáculos.

Assim como o AAL e AGT, o AS está igualmente residente no mesmo computador principal, no entanto ele pode ser embarcado em qualquer dispositivo que tenha a capacidade de se conectar ao *framework* ROS.

3.6 CONSIDERAÇÕES

Ao fim das implementações dos agentes que compõe o CPS e do SMR, o ambiente pode ser configurado com diversas disposições experimentais, e com o desenvolvimento de algoritmos apropriados no AGT, para a realização de tarefas específicas.

Ressalta-se que toda a arquitetura desenvolvida não representa um CPS completo de acordo com os níveis estabelecidos por (LEE; BAGHERI; KAO, 2015). Este CPS atua apenas nas camadas de conexão, conversão e cibernética. De acordo com o que foi apresentado neste capítulo, os agentes desenvolvidos são enquadrados nas camadas de conexão, conversão e cibernética conforme divisão apresentada na Tabela 7 e tal organização também pode ser vista na Figura 19.

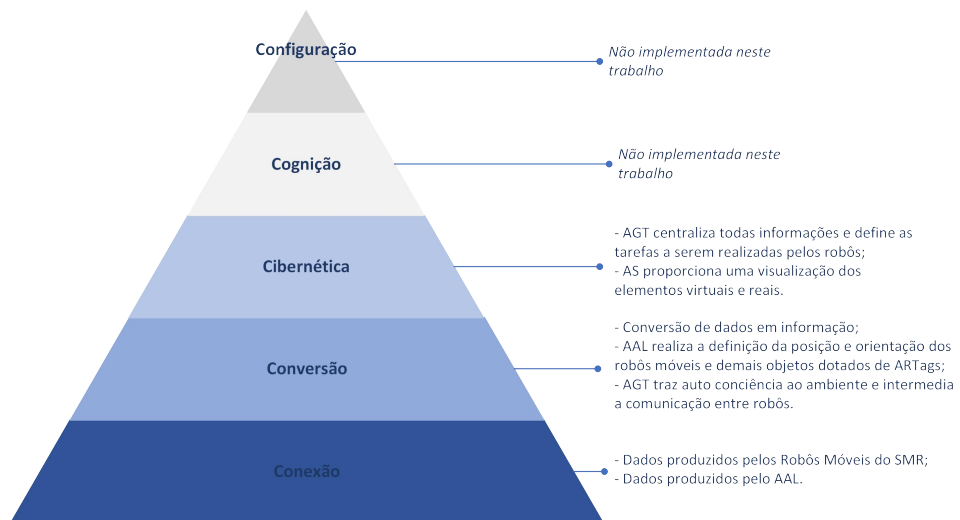
Tabela 7 – Recursos ciber-físicos do sistema multirrobô.

Elemento	Camadas do CPS		
	<i>Conexão</i>	<i>Conversão</i>	<i>Cibernética</i>
Robôs Móveis	X		
Agente Gerenciador de Tarefas		X	X
Agente Auxiliar de Localização	X	X	
Agente Supervisório		X	X

Fonte: Autoria própria.

O CPS desenvolvido atua especificamente para SMRs compatíveis com a estrutura dos agentes desenvolvido neste trabalho. Para etapas futuras, as camadas de cognição e configuração poderão ser implementadas no CPS, podendo-se criar um gêmeo virtual e coletar informações do ambiente para a utilização em treinamento de algoritmos de IA, possibilitando ao CPS tomar decisões automáticas com base no conhecimento adquirido dentre outras implementações.

Figura 19 – Aplicação da arquitetura 5C no CPS criado.



Fonte: Adaptado de (LEE; BAGHERI; KAO, 2015).

4 RESULTADOS

O presente capítulo apresenta os principais resultados provenientes de experimentos realizados para a validação do SMR e CPS propostos, bem como a avaliação das estratégias de controle dos robôs, da comunicação e implementação dos agentes que compõem o CPS criado.

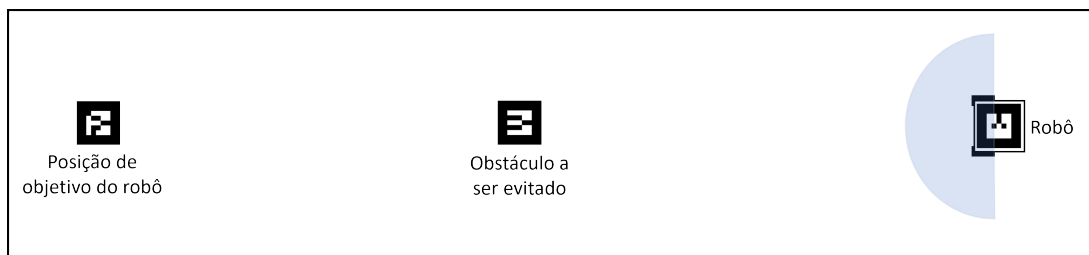
4.1 VALIDAÇÕES DAS ESTRATÉGIAS DE CONTROLE DOS ROBÔS

Visando validar os algoritmos de controle embarcados em cada robô, foram realizados experimentos para se analisar a função de transição entre os controladores implementados e a estratégia de comunicação para desvio de obstáculos dinâmicos.

4.1.1 Função de Transição

Com o intuito de analisar a eficiência da função de transição entre os controladores PD e *Fuzzy* implementados em cada um dos robôs, foi criada uma configuração experimental no AGT, onde um robô deve sair de sua posição inicial e ir até uma posição de objetivo, necessitando desviar de um obstáculo posicionado no meio de sua trajetória, conforme detalhado na Figura 20.

Figura 20 – Disposição do experimento para validar a função de transição.

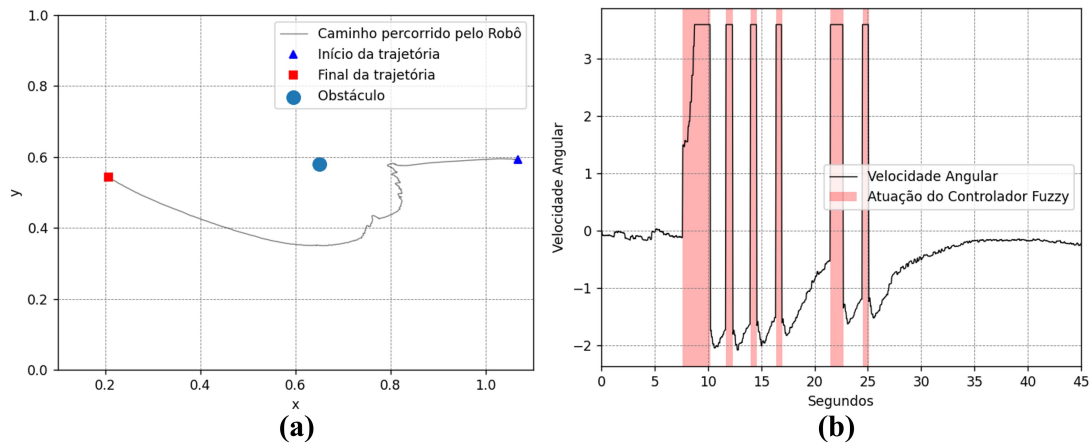


Fonte: Autoria própria.

Primeiramente foi realizada uma execução sem a implementação da função de transição entre os controladores do robô. A Figura 21 (a) apresenta o caminho percorrido pelo robô durante sua trajetória, de sua posição inicial até a posição final de destino. Observa-se que, após a identificação do obstáculo, o robô realiza movimentos de forma sinuosa, isso se dá devido ao chaveamento ocorrido entre os controladores, uma vez que o controlador *Fuzzy* força o robô para se movimentar do lado oposto ao obstáculo até que ele não seja mais identificado pelo sensor virtual, enquanto o controlador PD direciona o robô em relação à sua posição de destino.

A Figura 21 (b) apresenta a velocidade angular aplicada aos motores e também os momentos onde o controlador *Fuzzy* entrou em ação, é possível verificar que a comutação dos controladores proporcionou uma movimentação em zig zag, com movimentos para a direita (velocidade angular negativa) e para a esquerda (velocidade angular positiva).

Figura 21 – Execução sem a função de transição.

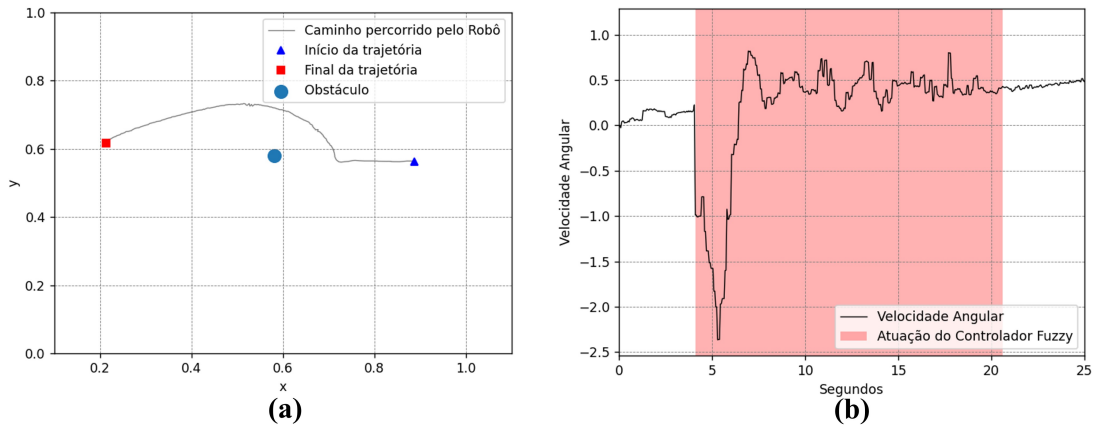


Fonte: Autoria própria.

Posteriormente, foi realizado outro experimento, agora com a implementação da função de transição, descrita pela Equação 2. Na Figura 22 (a), é possível observar a trajetória do robô partindo de sua posição inicial até a final, observa-se uma suavidade na movimentação se comparado com a trajetória apresentada na Figura 21 (a). Já a Figura 22 (b), traz a resposta dos controladores, na parte destacada é possível verificar o momento em que se ponderou a utilização da resposta do controlador *Fuzzy* em conjunto ao controlador PD, proporcionando suavidade na movimentação, uma vez que não houveram trocas consecutivas de controladores, se comparado com a execução sem a implementação da função de transição. Observa-se que durante as execuções os robôs realizaram a manobra de desvio por caminhos distintos (direita ou esquerda), isso se deu devido ao ponto inicial de cada robô, em cada experimento, não ser o mesmo, o que muda a tomada de decisão de desvio.

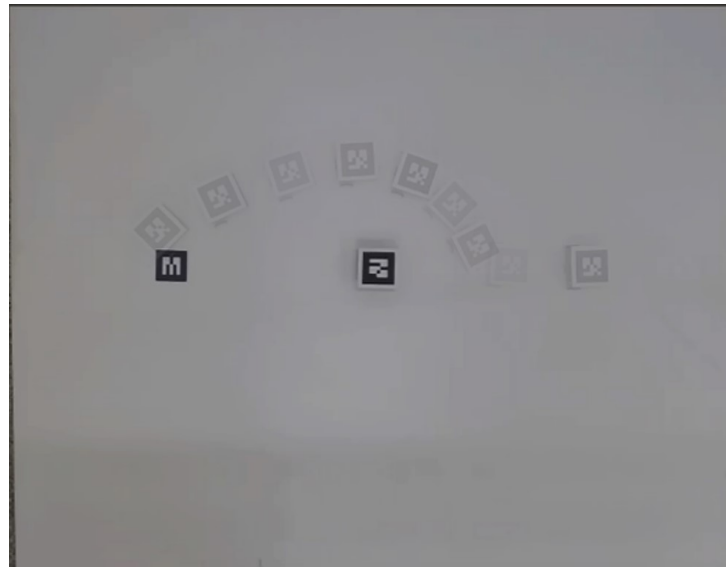
É apresentado na Figura 23 quadros sobrepostos da imagem capturada pelo AAL durante a execução com a implementação da função de transição, onde observa-se o caminho percorrido pelo robô durante esta validação.

Figura 22 – Execução com a função de transição.



Fonte: Autoria própria.

Figura 23 – Quadros sobrepostos capturados do AAL na execução com a função de transição.



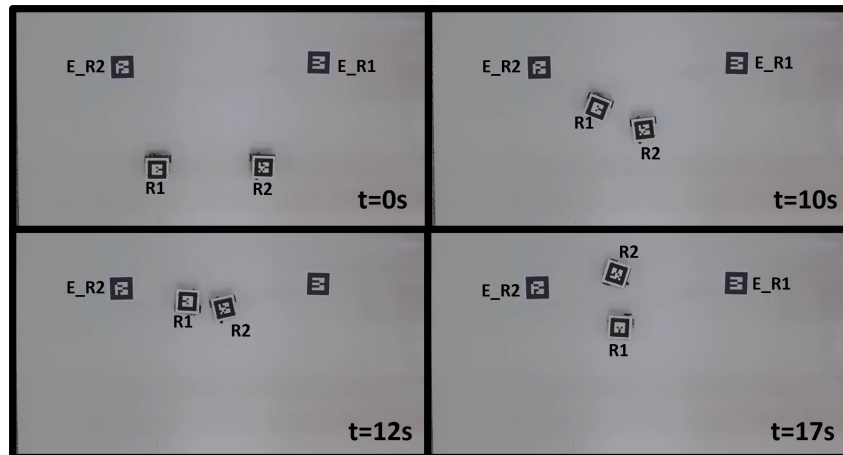
Fonte: Autoria própria.

4.1.2 Estratégia de Comunicação para Desvios de Obstáculos Dinâmicos

A fim de se verificar a estratégia de comunicação entre robôs implementada, foi criado um experimento onde os robôs foram posicionados de forma semelhante à representada pela Figura 13. É apresentado na Figura 24, quatro quadros da execução experimental onde a comunicação entre robôs foi utilizada. No instante $t=0s$ é apresentada a configuração inicial dos robôs ($R1$ e $R2$) e estações de trabalho/alvos (E_{R1} e E_{R2}). Em $t=10s$, $R1$ verifica que necessita dar passagem a $R2$, para isso, dá início a uma manobra abrupta à direita (girar em seu próprio eixo), o fim desta manobra é mostrado no instante $t=12s$. Já em $t=17s$, é apresentado o momento em que não existe mais risco de colisão nem da obrigatoriedade de passagem por nenhum dos robôs

e os mesmos se encaminham aos seus respectivos alvos.

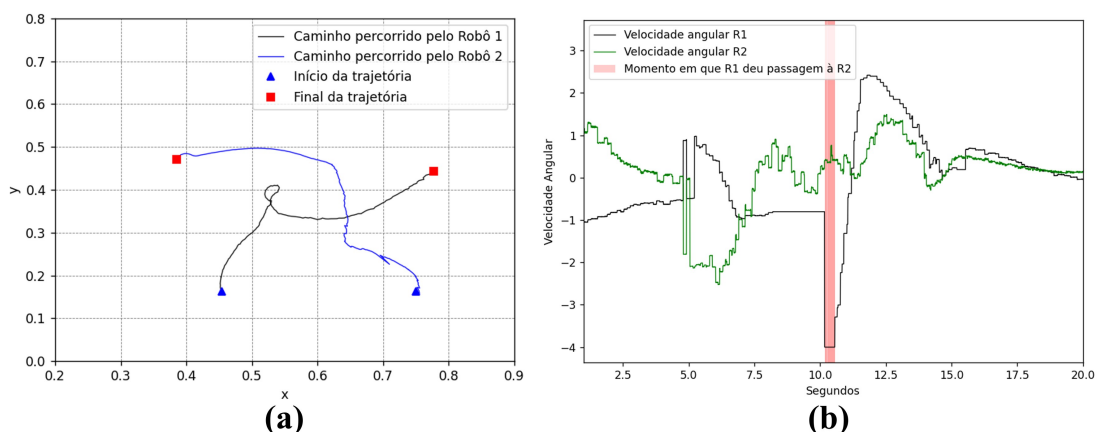
Figura 24 – Quadros que representam a execução da estratégia de comunicação.



Fonte: Autoria própria.

Na execução, ambos robôs possuem o valor da variável *contador* igual ao valor do próprio identificador interno (*id*), o qual foi utilizado por eles para a decisão sobre qual robô irá dar a passagem ao outro. A Figura 25 (a), mostra o início e fim dos caminhos percorridos pelos robôs e o momento em que o *R1* gira sobre seu próprio eixo por 1s para dar passagem a *R2*, o qual possui prioridade neste experimento. Já a Figura 25 (b) apresenta a velocidade angular dos robôs, a área em destaque representa o momento em que o robô *R1* deu passagem ao robô *R2*, situação em que foi aplicada uma velocidade angular de -4, para que o robô gire em seu eixo pela direita.

Figura 25 – Representação do momento do processamento da estratégia de comunicação.



Fonte: Autoria própria.

No experimento realizado, não foi atribuído nenhum valor a variável *contador*, porém a depender do experimento a ser conduzido, é possível registrar nela, a carga de trabalho do robô

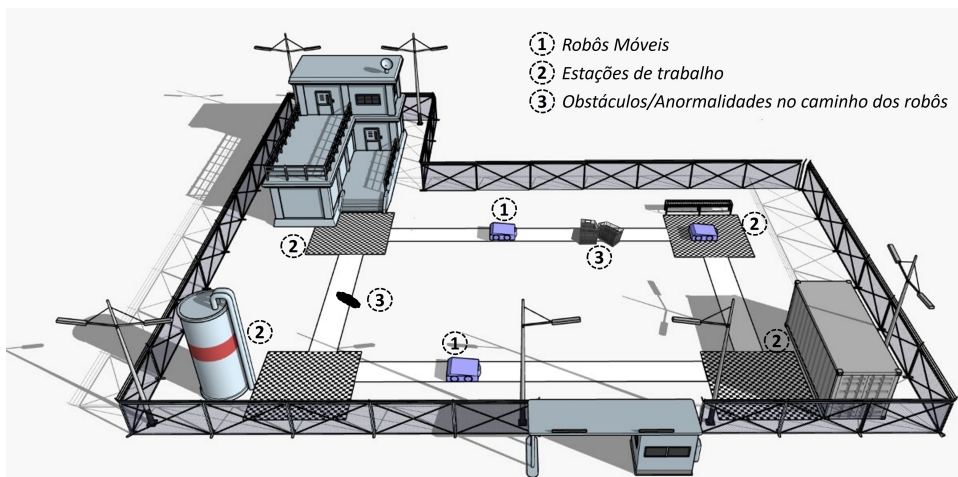
(distância navegada, quantidade de alvos visitados, entre outros), o nível de bateria, e outros indicadores pertinentes ao experimento, e esta informação sendo compartilhada com o restante do SMR, possibilita a adoção de novas estratégias de tomada de decisão.

4.2 VALIDAÇÕES DO AMBIENTE CIBER-FÍSICO

Visando representar algumas situações reais de ambientes que utilizam múltiplos robôs para a realização de tarefas cooperativas, algumas configurações experimentais foram criadas, em escala laboratorial, com o intuito de validar os agentes que compõe o CPS-SMR criado e as técnicas de comunicação desenvolvidas.

A inspiração para estas configurações vieram de situações em que robôs móveis autônomos tem a tarefa de visitar estações de trabalho, seja para auxiliar trabalhadores humanos, transportar ferramentas ou realizar tarefas similares. Neste cenário, os robôs devem desviar de obstáculos (estáticos ou dinâmicos) e compartilhar informações a cerca de suas ações e sobre situações inesperadas que possam prejudicar sua navegação. Um ambiente real que inspirou os experimentos simulados é apresentado na Figura 26.

Figura 26 – Inspiração para as configurações experimentais.



Fonte: Autoria própria.

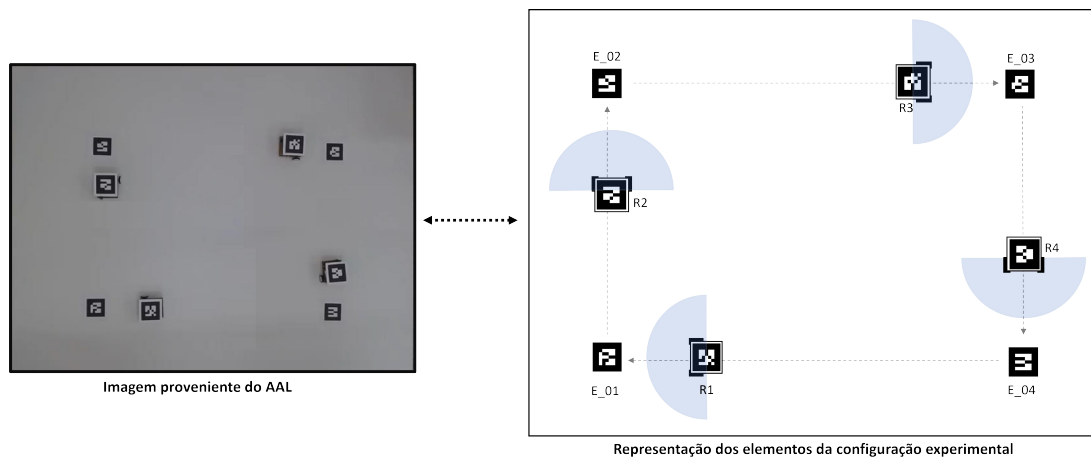
4.2.1 Experimento 01

Este experimento tem por intuito mostrar a capacidade do CPS em gerenciar tarefas cooperativas, em que os robôs do SMR devem visitar ordenadamente as estações de trabalho. Para isso, 4 robôs devem visitar também 4 estações de trabalho de forma ordenada, no sentido

horário.

A Figura 27 apresenta a disposição dos elementos (alvos, robôs e a direção que os robôs devem navegar) para a execução do experimento, onde as 4 estações de trabalho (E_{01} , E_{02} , E_{03} , E_{04}) são representadas por marcadores ARTags e são posicionadas próximas aos cantos da área experimental e os robôs ($R1$, $R2$, $R3$ e $R4$) também possuem suas ARTags. As setas pontilhadas dentro da representação experimental mostra a sequência em que as estações de trabalho serão visitadas. Também é possível observar a imagem capturada pelo AAL que representa a real disposição dos elementos no ambiente real.

Figura 27 – Configuração do experimento 01.



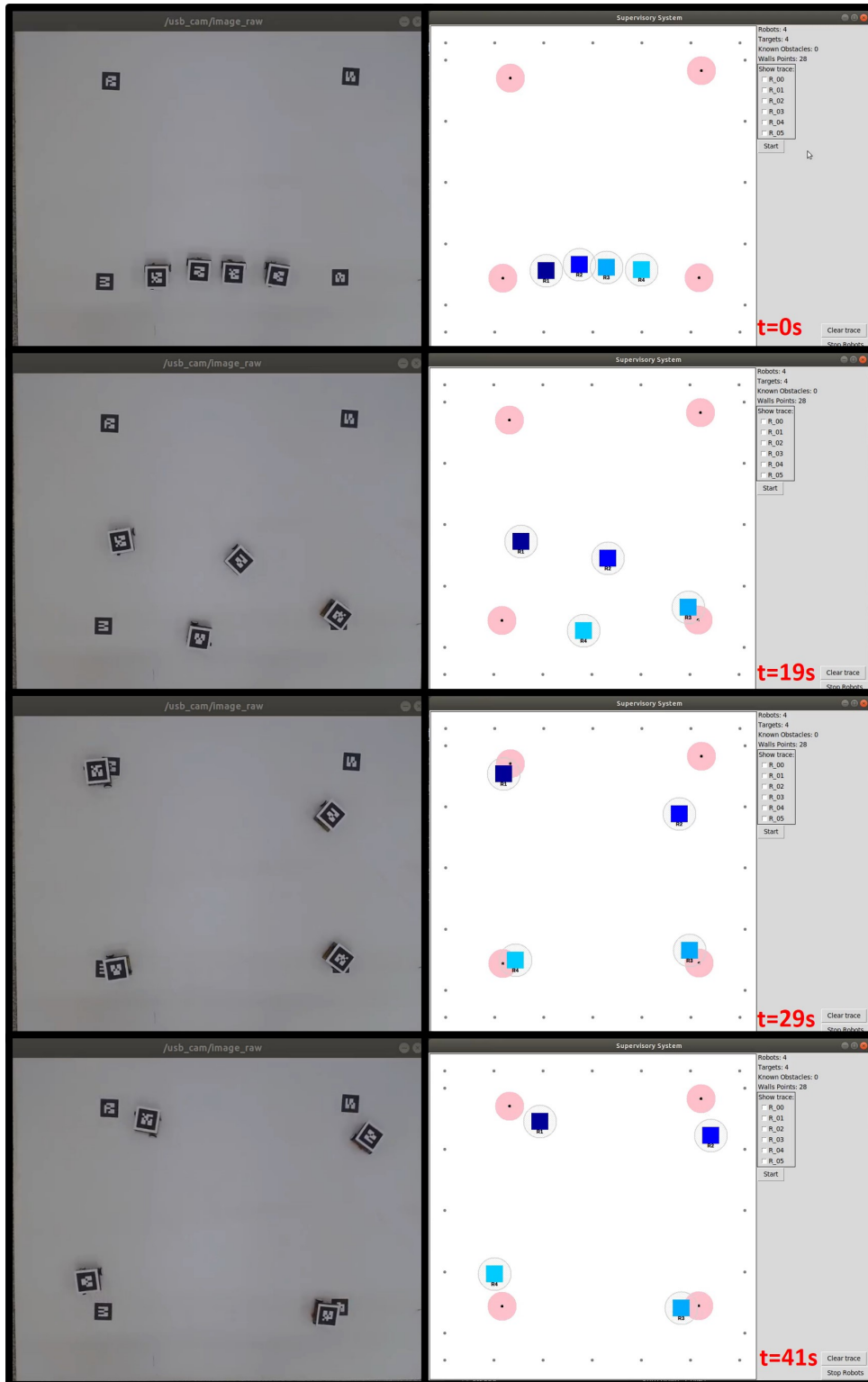
Fonte: Autoria própria.

Neste experimento, o AGT gerencia a delegação de tarefas aos robôs, ajudando-os a realizarem tarefas sincronizadas, onde somente após todos os robôs chegarem às suas estações de trabalho atuais, uma nova estação de trabalho é atribuída a todos, este processo acontece continuamente até que seja concluído o experimento por meio de comando remoto via interface do AS. De fato, esta configuração reflete o fluxograma apresentado na Figura 17.

É apresentado na Figura 28, 4 quadros que representam o início da execução do experimento, vistos tanto pela câmera acoplada ao AAL (ambiente real) quanto pela interface gráfica do AS, o instante $t=0s$ é o momento em que a execução experimental é iniciada, com os robôs posicionados e alinhados na parte inferior do ambiente, também foi definido previamente no AGT as estações de trabalho iniciais a serem visitadas por cada um dos robôs. No instante $t=19s$, é apresentado o momento em que os robôs se encaminham para a suas respectivas estações de trabalhos iniciais, sendo que $R3$ já chegou a sua estação de trabalho e aguarda os demais robôs chegarem. No instante $t=29s$ é mostrada situação em que os robôs $R1$, $R3$ e $R4$ chegam às suas estações e estão aguardando $R2$ chegar para se encaminhar à próxima estação de trabalho. Já

no instante $t=41s$, todos os robôs saem de forma sincronizada de suas respectivas estações em direção à próxima.

Figura 28 – Sequência de quadros capturados do AAL e do AS durante o experimento 01.

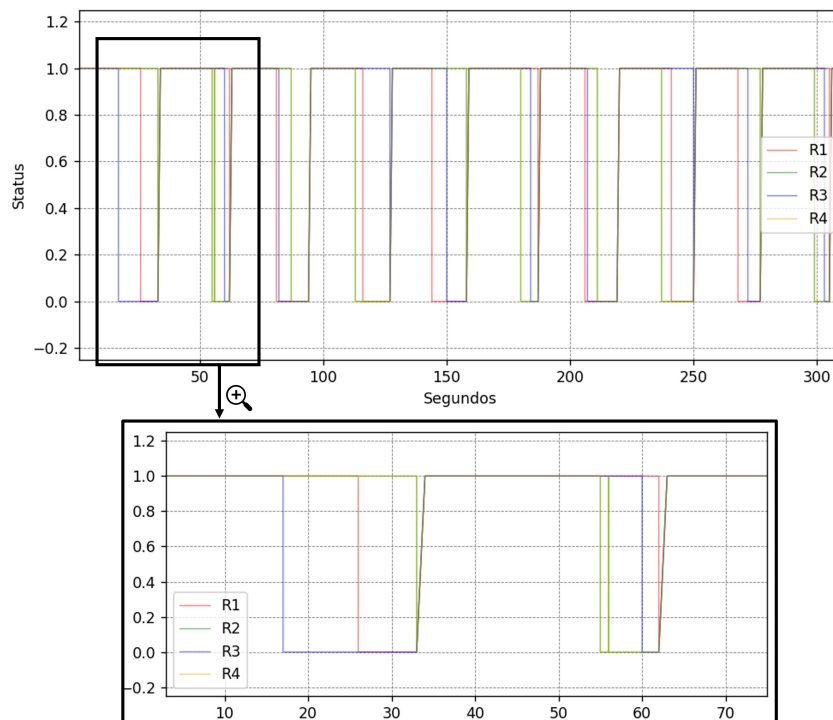


Fonte: Autoria própria.

Durante a realização deste experimento, os robôs visitaram diversas vezes as estações

de trabalho de forma coordenada, o gráfico na Figura 29 mostra o status de operação de todos os robôs durante a execução do experimento, onde 0 representa a situação onde o robô está parado/aguardando e 1 representa que o robô está em movimento. Na área destacada é possível verificar o momento em que R3 foi o primeiro a chegar à sua estação de trabalho, seguido por R1 e R4, até o momento em que R2 chega à estação e todos saem de forma sincronizada em direção ao próximo alvo. O processo acontece de forma ininterrupta até que a execução seja parada pela intervenção de um usuário na interface do AS.

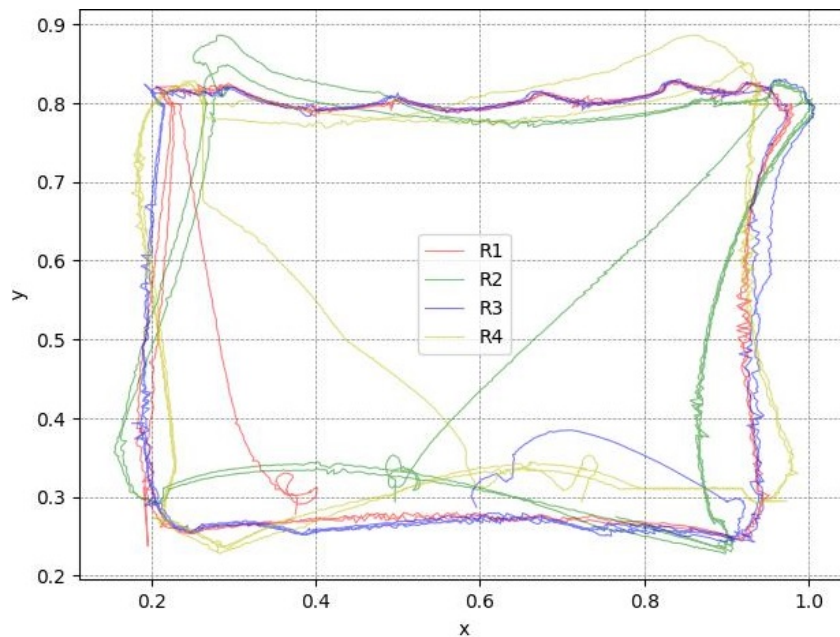
Figura 29 – Status dos robôs durante a execução no experimento 01.



Fonte: Autoria própria.

Ao final do experimento, observou-se que as definições estabelecidas nos agentes (AAL, AGT e AS) bem como os controladores embarcados nos robôs, desempenharam com sucesso as tarefas de execução sincronizada. A Figura 30 apresenta o caminho percorrido pelos robôs durante o experimento. Observa-se que o caminho percorrido pelos robôs não é o mesmo, embora todos possuam o mesmo *hardware*, observou-se que existem diferenças nos atuadores, porém, os controladores implementados nos robôs conseguem compensar esta diferença permitindo que eles cheguem às suas posições de destino.

Figura 30 – Caminho percorrido pelos robôs no experimento 01.



Fonte: Autoria própria.

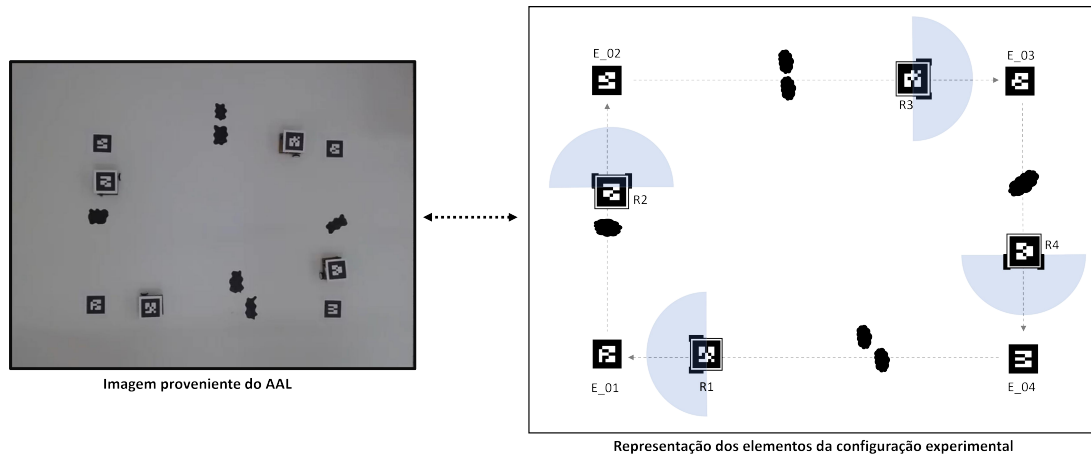
4.2.2 Experimento 02

Neste experimento, os robôs devem visitar estações de trabalho de forma ordenada assim como foi realizada no experimento 01, porém manchas em preto foram inseridas na área experimental com o intuito de representar objetos ou anormalidades que possam ser identificadas apenas por robôs móveis e que representam um risco a sua navegação. Assim, o experimento tem por intuito verificar a capacidade dos robôs em identificar essas manchas e de enviar essa informação ao AGT, que por sua vez deve compartilhar essa informação com todos os robôs para que eles possam desviar destas zonas.

A Figura 31 apresenta a configuração experimental adotada em que 4 robôs móveis foram inseridos na área experimental (R1, R2, R3 e R4) e devem visitar ordenadamente as estações de trabalho (E_01, E_02, E_03, E_04). As manchas foram posicionadas no caminho dos robôs (entre uma estação de trabalho e outra) a fim de forçar a sua identificação. Inicialmente os robôs e os agentes desconhecem a localização das manchas, as quais passam a ser visíveis ao CPS-SMR a medida que são identificadas pelos robôs.

É apresentado na Figura 32, uma sequência de 4 quadros que representa a execução deste experimento, visto tanto pela câmera acoplada ao AAL quanto pelo AS. No instante $t=10s$, os robôs saem de forma sincronizada de sua primeira estação de trabalho em direção a próxima. Em $t=15s$ é possível observar o momento em que R1 e R3 identificam as primeiras manchas, os

Figura 31 – Configuração do experimento 02.



Fonte: Autoria própria.

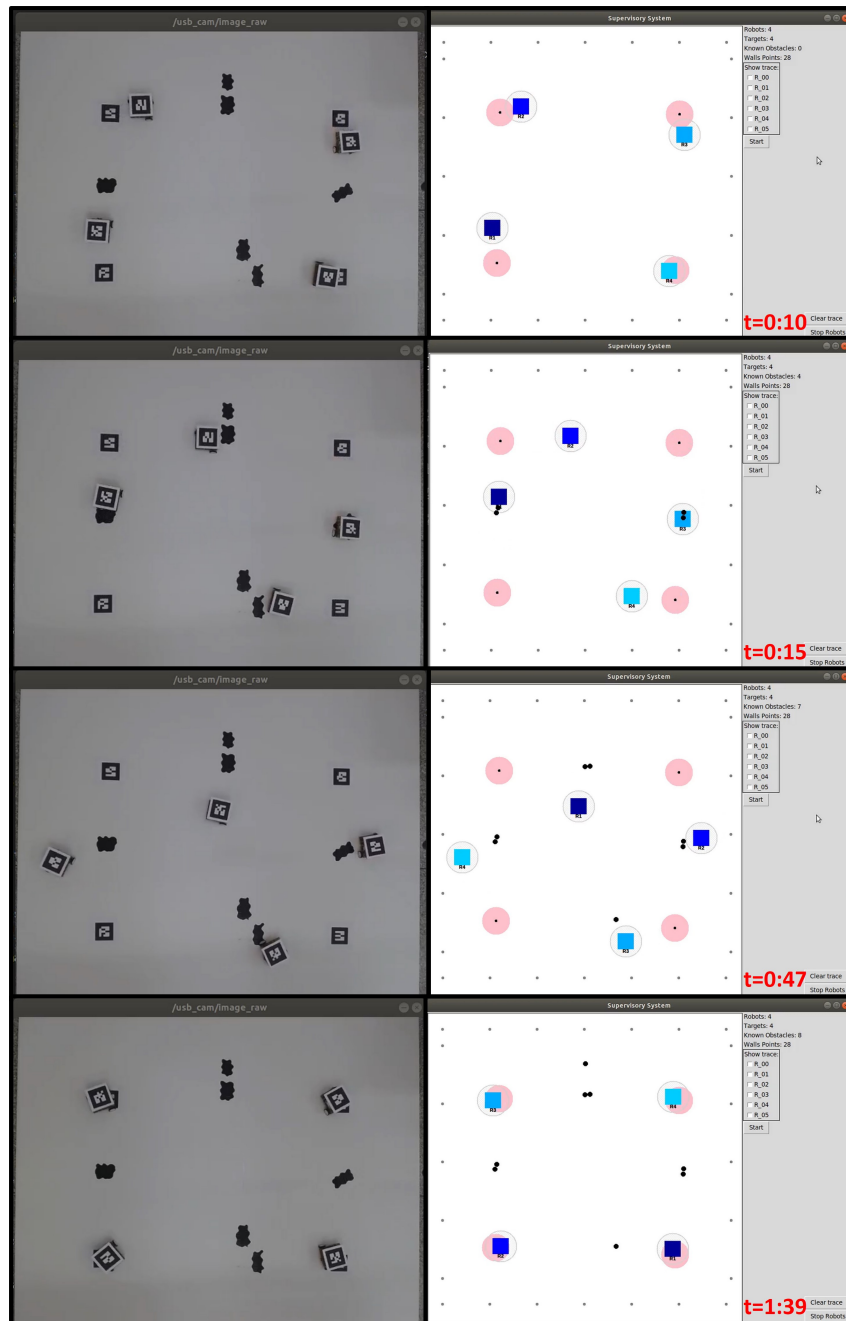
pontos em preto da interface gráfica do AS representa estas identificações, a partir deste momento estas identificações passam a ser obstáculos para todos robôs. No instante $t=47s$, novas manchas foram identificadas e observa-se o momento em que todos os robôs se desviam destas zonas. Já em $t=1:39s$, todas as manchas foram identificadas e os robôs trafegam de maneira segura entre as estações de trabalho, desviando das áreas que representam um risco a sua navegação.

É apresentado na Figura 33, o caminho percorrido pelos robôs durante a execução do experimento 02, observa-se os momentos em que os robôs passaram pelos obstáculos e posteriormente iniciaram as trajetórias de desvio. Nos momentos em que a ação de desvio do controlador *Fuzzy* entra em ação, observa-se manobras de desvio à direita do obstáculo e também em manobras pelo lado esquerdo. Isso se deve à posição/orientação do obstáculo capturada pelo sensor virtual no momento de identificação do obstáculo dentro da zona de detecção, já que o controlador atua para realizar o desvio pelo caminho menos custoso, conforme modelagem descrita na base de regras *Fuzzy* da Tabela 5.

4.2.3 Experimento 03

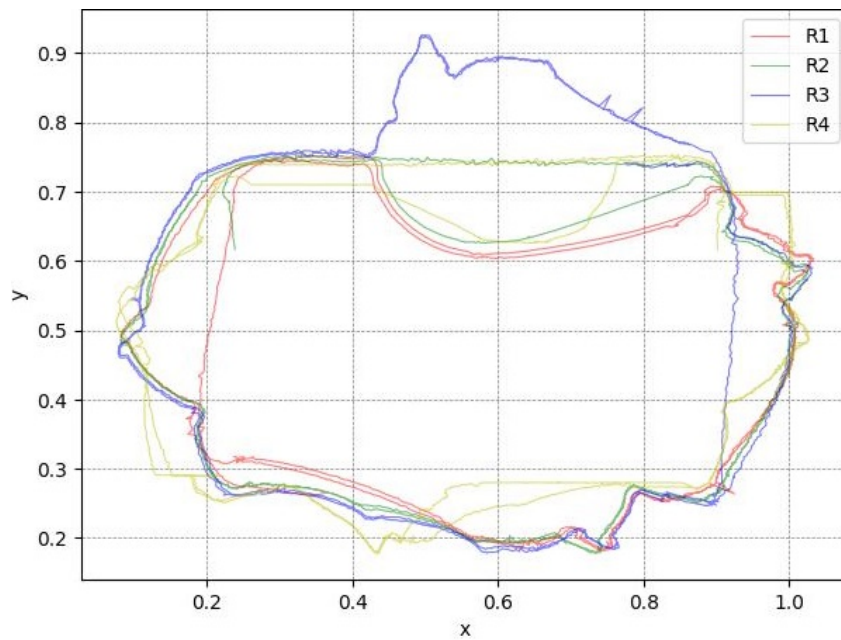
Neste experimento, os robôs devem visitar as estações de trabalho de forma ordenada, identificar anormalidades, compartilhar informações e realizar manobras de desvio destas zonas, assim como foi realizada no experimento 02, porém foi inserido no ambiente um robô com a capacidade de realizar a limpeza da área identificada, a fim de proporcionar uma navegação segura durante a execução do experimento. Ressalta-se que neste experimento, diferente dos demais em que todos os robôs tinham as mesmas habilidades, o SMR é heterogêneo e o objetivo

Figura 32 – Sequência de quadros capturados do AAL e do AS durante o experimento 02.



Fonte: Autoria própria.

Figura 33 – Caminho percorrido pelos robôs no experimento 02.

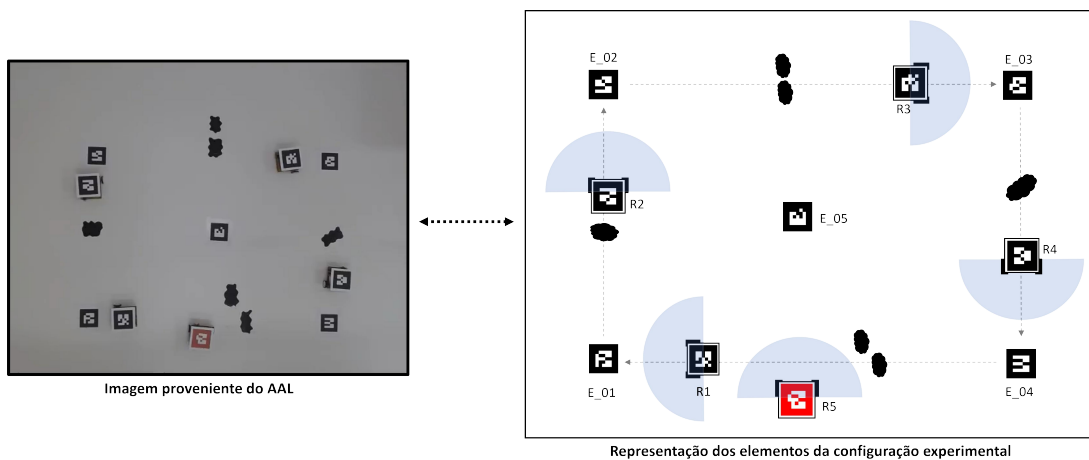


Fonte: Autoria própria.

é verificar a capacidade do CPS em coordenar tarefas para robôs com habilidades diferentes.

A Figura 34 apresenta a disposição inicial dos elementos para este experimento, em que 4 robôs móveis foram inseridos na área experimental (*R1*, *R2*, *R3* e *R4*) e devem visitar ordenadamente as estações de trabalho (*E_01*, *E_02*, *E_03*, *E_04*), esses 4 robôs representam robôs de trabalho inseridos em um ambiente real. Nesta configuração, também foi inserido um robô de suporte (*R5*), destacado pela ARTag impressa na cor vermelha, que possui sua estação de trabalho (*E_05*) posicionada na parte central da área experimental.

Figura 34 – Configuração do experimento 03.



Fonte: Autoria própria.

Neste experimento, foi definido que no momento em que uma mancha é identificada

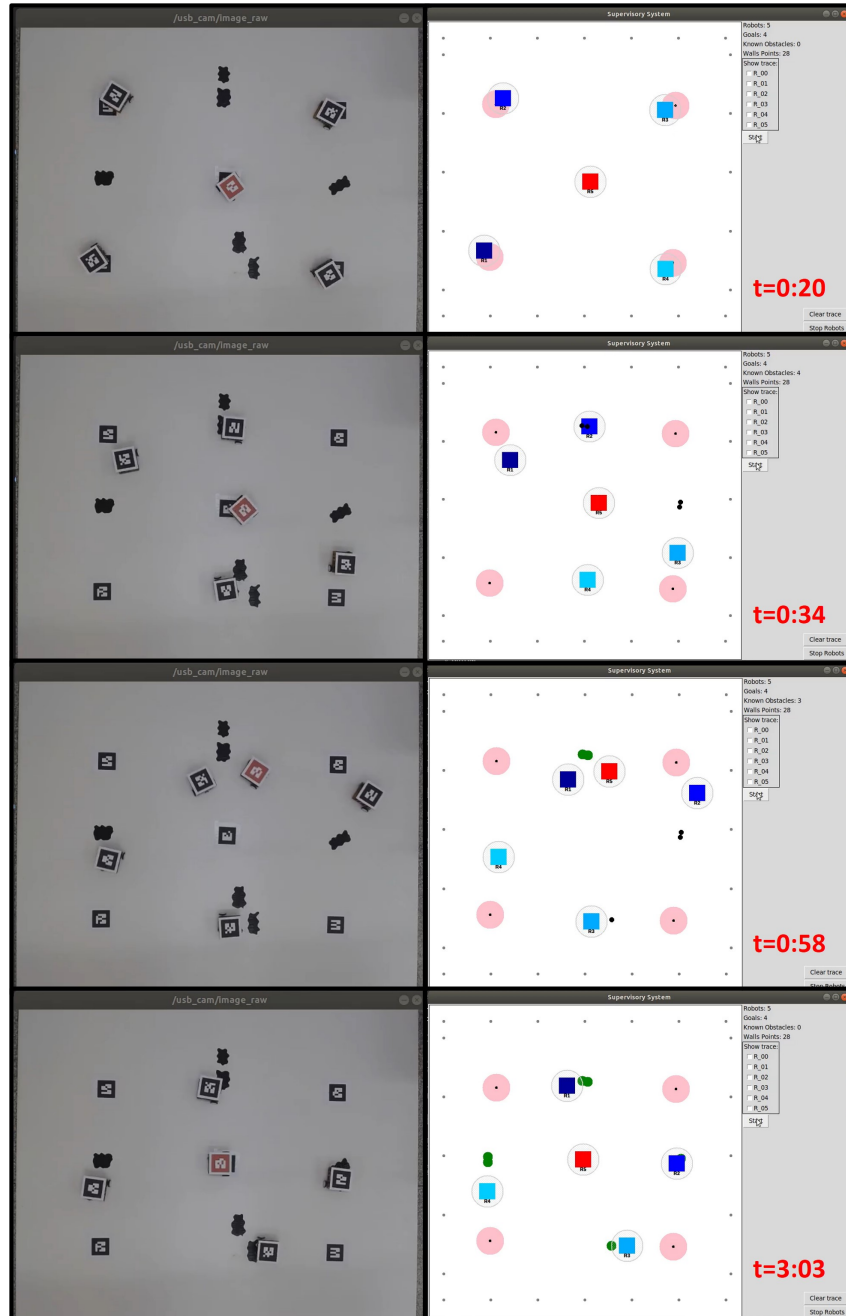
pelos robôs de trabalho, o robô de suporte deve ir em direção à mancha mais próxima (em relação a sua posição atual) e realizar a limpeza da área (operação virtual realizada no AGT para fins experimentais). Não havendo mais áreas a serem limpas, o robô de suporte deve retornar à sua estação de trabalho. Para testar a capacidade do CPS-SMR em tratar situações dinâmicas, foi definido no AGT que uma área fica limpa por apenas 6 minutos. Quando uma área está limpa, a identificação da mancha é ignorada pelos robôs de trabalho, que passam pela área sem realizar a manobra de desvio e sem identificar a mancha.

A Figura 35 apresenta uma sequência de 4 quadros que representa a execução do experimento 03, visto tanto pela câmera acoplada ao AAL quanto pelo AS. O instante $t=20s$, mostra momento em que todos os robôs chegaram às suas estações de trabalho e estão prestes a saírem de maneira sincronizada em direção à próxima estação. No instante $t=34s$, os robôs *R2* e *R3* identificam as primeiras manchas e *R5* sai de sua estação de trabalho para realizar a limpeza da mancha mais próxima. Em $t=58s$, é mostrado o momento em que *R5* já realizou a limpeza de algumas áreas (pontos verdes destacados na parte superior da imagem do AS). Já no instante $t=3:03s$, é apresentado o momento em que todas as manchas identificadas pelos robôs de trabalho foram limpas e o robô de suporte (*R5*) retornou a sua base. Neste instante também é possível observar que os robôs de trabalho passam livremente pelas áreas limpas (operação virtual gerenciada pelo AGT) sem realizar o desvio das manchas.

A Figura 36 apresenta o caminho percorrido pelos robôs neste experimento durante o primeiro ciclo de trabalho, onde saem de sua posição inicial, visitam todas as estações de trabalho e retornam à 1ª estação visitada. Além das movimentações dos robôs *R1*, *R2*, *R3* e *R4* também é possível visualizar o caminho percorrido pelo robô de suporte (*R5*) durante a execução da tarefa de limpeza das zonas identificadas pelos robôs de trabalho.

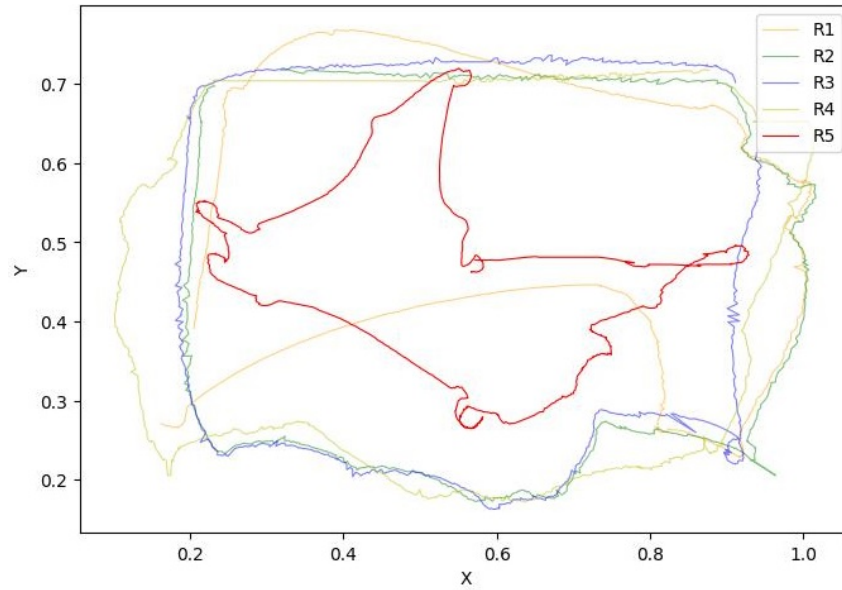
É apresentado na Figura 37, outro ciclo de movimentos dos robôs de trabalho após o robô de suporte realizar a operação virtual de limpeza do ambiente. Nesta análise, os robôs apenas visitam as estações de trabalho, ignorando a identificação das manchas (uma vez que estão virtualmente limpas). Também é possível observar que ao final do experimento o robô de suporte (*R5*) se encontrou parado em sua estação de trabalho (centro da área experimental). Este experimento comprova a capacidade do CPS desenvolvido suportar o gerenciamento de tarefas diferentes de acordo com as habilidades dos robôs em um SMR heterogêneo.

Figura 35 – Sequência de quadros capturados do AAL e do AS durante o experimento 03.



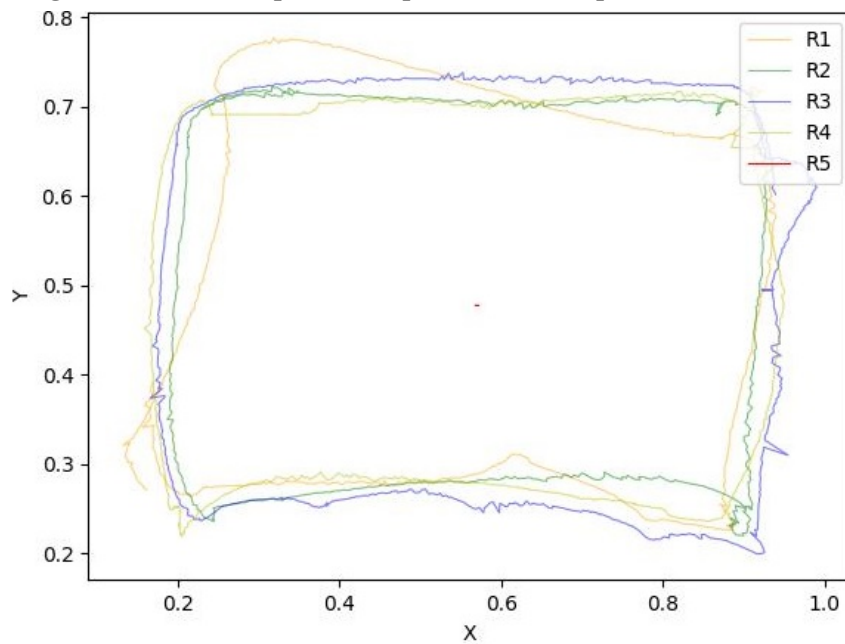
Fonte: Autoria própria.

Figura 36 – Caminho percorrido pelos robôs no experimento 3 - 1º ciclo.



Fonte: Autoria própria.

Figura 37 – Caminho percorrido pelos robôs no experimento 3 - 2º ciclo.



Fonte: Autoria própria.

4.3 CONSIDERAÇÕES

O presente capítulo teve por objetivo validar o CPS-SMR proposto e testar as estratégias de controles embarcadas nos robôs bem como analisar as funcionalidades implementadas no AGT. Os experimentos mostraram a capacidade do CPS em trabalhar com tarefas cooperativas e em compartilhar informações entre agentes e robôs móveis, além da possibilidade de coordenar robôs com habilidades distintas.

Em relação ao conjunto de habilidades do CPS-SMR, testou-se inicialmente a capacidade do ambiente no gerenciamento de trabalho sincronizado entre 4 robôs, posteriormente verificou-se a capacidade do CPS em compartilhar informações produzidas pelo SMR, por fim, foi criado um experimento mostrando a flexibilidade do CPS em trabalhar com robôs heterogêneos. Tais implementações mostraram o quão flexível pode ser a utilização do ambiente ciber-físico desenvolvido em diversas outras configurações.

5 CONSIDERAÇÕES FINAIS

A presente dissertação relatou o desenvolvimento de um ambiente com características ciber-físicas para utilização de múltiplos robôs móveis reais construídos com materiais de baixo custo. Apresentou no capítulo 2 alguns trabalhos que fundamentaram este estudo. No capítulo 3 foi apresentado os principais materiais e métodos utilizado para a construção dos robôs que compõem o Sistema Multirrobô (SMR) e os agentes que fazem parte do ambiente com características ciber-físicas (CPS) desenvolvido. Os resultados de validação do CPS-SMR puderam ser vistos no capítulo 4.

5.1 CONCLUSÕES

O uso de equipamentos comumente encontrados (câmera USB, computador e roteador *Wi-Fi*) permitiu a construção de um ambiente com características ciber-físicas CPS, da mesma forma que a utilização de materiais de baixo custo e *hardwares* simples foram essenciais para a construção de múltiplos robôs móveis (SMR) para a realização de experimentos em escala laboratorial.

A utilização do *framework* ROS foi essencial para realizar a comunicação entre os agentes de *software* que compõem o CPS, codificados em *Python* e os robôs móveis desenvolvidos nesta dissertação, possibilitando intermediar as tecnologias distintas utilizadas tanto na implementação do *software* quando no *hardware* dos robôs.

O desenvolvimento do Agente Auxiliar de Localização (AAL) possibilitou o uso de robôs móveis de baixo custo, uma vez que desprovidos de sensores de localização próprio, esses robôs puderam utilizar as informações do AAL para realizar sua movimentação além de auxiliar no processo de criação do sensorialmente virtual.

As classes *Python* que gerenciam a rápida criação de instâncias de robôs e alvos e a estrutura de mensagens ROS personalizadas desenvolvidas, auxiliaram na implementação de algoritmos para o Agente Gerenciador de Tarefas (AGT), facilitando a definição e gerenciamento de tarefas personalizadas. O AGT fornece suporte a realização de tarefas cooperativas e também no compartilhamento de informações, processando informações proveniente de cada robô e a compartilhando com todo o SMR.

Com o desenvolvimento do Agente Supervisório (AS) foi possível visualizar, por meio

de sua interface de fácil uso, os elementos virtuais contidos no AGT e acessados pelos robôs móveis (paredes virtuais, obstáculos, itens identificados apenas por robôs, itens que sofreram interação por robôs e caminho percorrido pelos robôs) e os elementos reais (robôs e estações de trabalho), além de possibilitar ao operador iniciar e parar a execução de tarefas pelos robôs.

Assim, toda a estrutura de agentes, mensagens e classes desenvolvidas foi capaz de dar suporte aos algoritmos que gerenciam tarefas no AGT, proporcionando implementações de diversas configurações experimentais de forma simplificada e rápida.

Por mais que o AAL, AGT e AS estejam implementados em um mesmo computador, todo o desenvolvimento foi feito visando a implementação distribuída destes agentes em outros equipamentos compatíveis com o *framework* ROS.

Os robôs desenvolvidos executaram com êxito as tarefas atribuídas pelo AGT, seus controladores desempenharam com sucesso as tarefas que envolveram movimentação e desvio de obstáculos. O fato dos controladores PD e *Fuzzy* serem embarcados nos robôs liberou o AGT da execução da tarefa de movimentação e desvio de obstáculos de cada robô, uma vez que para gerenciar a locomoção dos robôs, sua função foi apenas atribuir as coordenadas de destino para cada um deles.

Devido a flexibilidade com que todo o ambiente foi desenvolvido, é simples a inclusão de robôs móveis com características diferentes. O único requisito é a capacidade de se conectar e se comunicar com os agentes do CPS, via *framework* ROS.

Os experimentos apresentados validaram inicialmente os sistemas embarcados nos robôs, como a função de transição entre os controladores PD e *Fuzzy* assim como a estratégia adotada para desvios de obstáculos dinâmicos capturada na base de conhecimento do controlador *Fuzzy*. Os experimentos avaliaram a capacidade do CPS em coordenar tarefas, com o compartilhamento de informações entre os robôs, a atribuição de tarefas para robôs com habilidades diferentes e também testar a comunicação entre os agentes (AAL, AGT e AS) com os robôs móveis construídos.

Por fim, toda a estrutura de agentes e robôs desenvolvidos configuram uma plataforma didática para o ensino de conceitos de SMR e CPS, podendo ser aplicadas em cursos de graduação e de nível técnico dentro da área de estudo.

5.2 TRABALHOS FUTUROS

Como trabalhos futuros, pretende-se implementar o CPS de forma distribuída, alocando os agentes (AAL, AGT e AS) em dispositivos diferentes com o intuito de se validar sua escalabilidade, tais como veículos aéreos não tripulados, mini-computadores, dentre outros.

Em relação ao ambiente real, pode-se utilizar técnicas de realidade mista para projetar elementos virtuais trazendo mais realismo na simulação. Além disso, pode-se implementar novos sensores reais e virtuais nos robôs móveis permitindo a realização de novas tarefas, tais como monitoramento da carga da bateria, sensores reais de detecção de obstáculos, dentre outros. Outra possibilidade é de se aumentar a quantidade de robôs móveis reais desenvolvidos, além da criação e inserção de robôs virtuais. De fato, um experimento que poderia ser realizado de imediato é a implementação de um SMR em que coexistam robôs virtuais e reais com habilidades iguais e/ou diferentes.

Em relação ao AS, pode-se melhorar suas funcionalidades a fim de subsidiar a criação de um gêmeo virtual, que armazene as informações geradas pelos robôs durante a execução de tarefas em uma base de dados, visando o treinamento de algoritmos de inteligência artificial para posteriormente implementar esses algoritmos inteligentes no CPS. Por fim, pode-se desenvolver novos experimentos e guias didáticos para permitir ao CPS funcionar como plataforma didática de apoio as disciplinas de robótica, IoT e sistemas ciber-físicos.

REFERÊNCIAS

AFRIN, Mahbuba; JIN, Jiong; RAHMAN, Akhlaqur; RAHMAN, Ashfaqur; WAN, Jiafu; HOSSAIN, Ekram. Resource allocation and service provisioning in multi-agent cloud robotics: A comprehensive survey. **IEEE Communications Surveys & Tutorials**, v. 23, n. 2, p. 842–870, 2021.

ALMADHOUN, Randa; TAHA, Tarek; SENEVIRATNE, Lakmal; ZWEIRI, Yahya. A survey on multi-robot coverage path planning for model reconstruction and mapping. **SN Applied Sciences**, Springer Science and Business Media LLC, v. 1, n. 8, jul. 2019. Disponível em: <https://doi.org/10.1007/s42452-019-0872-y>.

ALMEIDA, João Paulo Lima Silva de; NAKASHIMA, Renan Taizo; NEVES-JR, Flávio; ARRUDA, Lúcia Valéria Ramos de. Bio-inspired on-line path planner for cooperative exploration of unknown environment by a multi-robot system. **Robotics and Autonomous Systems**, Elsevier BV, v. 112, p. 32–48, feb 2019.

ALMEIDA, João Paulo Lima Silva de; NAKASHIMA, Renan Taizo; NEVES-JR, Flávio; OLIVEIRA, André Schneider de; ARRUDA, Lúcia Valéria Ramos de. Autonomous navigation of multiple robots with sensing and communication constraints based on mixed reality. **Journal of Control, Automation and Electrical Systems**, Springer Science and Business Media LLC, v. 31, n. 5, p. 1165–1176, jul 2020.

ANGUELOV, Antouan; TRIFONOV, Roumen; NAKOV, Ognian. Analysis of swarm application layer protocols (SALP) used in event-driven communication. *In: 2021 International Conference Automatics and Informatics (ICAI)*. IEEE, 2021. Disponível em: <https://doi.org/10.1109/icai52893.2021.9639663>.

ANTONYSHYN, Luka; SILVEIRA, Jefferson; GIVIGI, Sidney; MARSHALL, Joshua. Multiple mobile robot task and motion planning: A survey. **ACM Computing Surveys**, Association for Computing Machinery (ACM), set. 2022. Disponível em: <https://doi.org/10.1145/3564696>.

BAHETI, Radhakisan; GILL, Helen. Cyber-physical systems. **The impact of control technology**, v. 12, n. 1, p. 161–166, 2011.

BEMANI, Ali; BJORSELL, Niclas. Cyber-physical control of indoor multi-vehicle testbed for cooperative driving. *In: 2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS)*. IEEE, 2020. Disponível em: <https://doi.org/10.1109/icps48405.2020.9274688>.

DORIYA, Rajesh; MISHRA, Siddharth; GUPTA, Swati. A brief survey and analysis of multi-robot communication and coordination. *In: International Conference on Computing, Communication & Automation*. [S.l.: s.n.], 2015. p. 1014–1021.

DORRI, Ali; KANHERE, Salil S.; JURDAK, Raja. Multi-agent systems: A survey. **IEEE Access**, v. 6, p. 28573–28593, 2018.

EGGER, Johannes; MASOOD, Tariq. Augmented reality in support of intelligent manufacturing – a systematic literature review. **Computers & Industrial Engineering**, v. 140, p. 106195, 2020. ISSN 0360-8352. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0360835219306643>.

ESCOBAR, Luis; MOYANO, Cristian; AGUIRRE, Gabriel; GUERRA, Gabriel; ALLAUCA, Luis; LOZA, David. Multi-robot platform with features of cyber-physical systems for education applications. *In: 2020 IEEE ANDESCON*. IEEE, 2020. Disponível em: <https://doi.org/10.1109/andescon50619.2020.9272030>.

FIALA, M. Artag, a fiducial marker system using digital techniques. *In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. [S.l.: s.n.], 2005. v. 2, p. 590–596. ISSN 1063-6919.

FUERTES, Juan Jose; PRADA, Miguel Angel; RODRIGUEZ-OSSORIO, Jose Ramon; GONZALEZ-HERBON, Raul; PEREZ, Daniel; DOMINGUEZ, Manuel. Environment for education on industry 4.0. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 9, p. 144395–144405, 2021. Disponível em: <https://doi.org/10.1109/access.2021.3120517>.

GAO, Ping-an; CAI, Zi-xing; YU, Ling-li. Evolutionary computation approach to decentralized multi-robot task allocation. *In: 2009 Fifth International Conference on Natural Computation*. [S.l.: s.n.], 2009. v. 5, p. 415–419.

GAUTAM, Avinash; MOHAN, Sudeept. A review of research in multi-robot systems. *In: 2012 IEEE 7th International Conference on Industrial and Information Systems (ICIIS)*. IEEE, 2012. Disponível em: <https://doi.org/10.1109/iciinfs.2012.6304778>.

KRIDI, Douglas Santiago; ALVES, Antônio José de Oliveira; LEMOS, Marcus Vinícius de Sousa; RABELO, Ricardo de Andrade Lira. Desenvolvimento de uma biblioteca fuzzy para o controle autônomo de um robô móvel em ambiente desconhecido. *In: Anais da I Mostra Nacional de Robótica*. [S.l.: s.n.], 2011. p. 155–161.

LEE, Jay; BAGHERI, Behrad; KAO, Hung-An. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. **Manufacturing Letters**, Elsevier BV, v. 3, p. 18–23, jan. 2015. Disponível em: <https://doi.org/10.1016/j.mfglet.2014.12.001>.

LIU, Xiulong; CAO, Jiannong; YANG, Yanni; JIANG, Shan. CPS-based smart warehouse for industry 4.0: A survey of the underlying technologies. **Computers**, MDPI AG, v. 7, n. 1, p. 13, fev. 2018. Disponível em: <https://doi.org/10.3390/computers7010013>.

LIU, Yang; PENG, Yu; WANG, Bailing; YAO, Sirui; LIU, Zihe. Review on cyber-physical systems. **IEEE/CAA Journal of Automatica Sinica**, v. 4, n. 1, p. 27–40, 2017.

MADRIDANO Ángel; AL-KAFF, Abdulla; MARTÍN, David; ESCALERA, Arturo de la. Trajectory planning for multi-robot systems: Methods and applications. **Expert Systems with Applications**, Elsevier BV, v. 173, p. 114660, jul. 2021. Disponível em: <https://doi.org/10.1016/j.eswa.2021.114660>.

MAYYA, Siddharth; D'ANTONIO, Diego S.; SALDANA, David; KUMAR, Vijay. Resilient task allocation in heterogeneous multi-robot systems. **IEEE Robotics and Automation Letters**, Institute of Electrical and Electronics Engineers (IEEE), v. 6, n. 2, p. 1327–1334, abr. 2021. Disponível em: <https://doi.org/10.1109/lra.2021.3057559>.

MILLARD, Alan G.; REDPATH, Richard; JEWERS, Alistair M.; ARNDT, Charlotte; JOYCE, Russell; HILDER, James A.; MCDAID, Liam J.; HALLIDAY, David M. ARDebug: An augmented reality tool for analysing and debugging swarm robotic systems. **Frontiers in Robotics and AI**, Frontiers Media SA, v. 5, jul. 2018. Disponível em: <https://doi.org/10.3389/frobt.2018.00087>.

MONOSTORI, L.; KÁDÁR, B.; BAUERNHANSL, T.; KONDOH, S.; KUMARA, S.; REINHART, G.; SAUER, O.; SCHUH, G.; SIHN, W.; UEDA, K. Cyber-physical systems in manufacturing. **CIRP Annals**, Elsevier BV, v. 65, n. 2, p. 621–641, 2016. Disponível em: <https://doi.org/10.1016/j.cirp.2016.06.005>.

PEREIRA, Ana; THOMAS, Carsten. Challenges of machine learning applied to safety-critical cyber-physical systems. **Machine Learning and Knowledge Extraction**, MDPI AG, v. 2, n. 4, p. 579–602, nov. 2020. Disponível em: <https://doi.org/10.3390/make2040031>.

PIARDI, Luis; KALEMPA, Vivian Cremer; LIMEIRA, Marcelo; OLIVEIRA, André Schneider de; LEITÃO, Paulo. Arena—augmented reality to enhanced experimentation in smart warehouses. **Sensors**, v. 19, n. 19, 2019. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/19/19/4308>.

QIU, Jiantao; YU, Chao; LIU, Weiling; YANG, Tianxiang; YU, Jincheng; WANG, Yu; YANG, Huazhong. Low-cost multi-agent navigation via reinforcement learning with multi-fidelity simulator. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 9, p. 84773–84782, 2021. Disponível em: <https://doi.org/10.1109/access.2021.3085328>.

QUERALTA, Jorge Pena; TAIPALMAA, Jussi; PULLINEN, Bilge Can; SARKER, Victor Kathan; GIA, Tuan Nguyen; TENHUNEN, Hannu; GABBOUJ, Moncef; RAITOHARJU, Jenni; WESTERLUND, Tomi. Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 8, p. 191617–191643, 2020. Disponível em: <https://doi.org/10.1109/access.2020.3030190>.

QUIGLEY, Morgan; CONLEY, Ken; GERKEY, Brian P.; FAUST, Josh; FOOTE, Tully; LEIBS, Jeremy; WHEELER, Rob; NG, Andrew Y. Ros: an open-source robot operating system. *In: ICRA Workshop on Open Source Software. [S.l.: s.n.], 2009.*

RAJKUMAR, Rangunathan (Raj); LEE, Insup; SHA, Lui; STANKOVIC, John. Cyber-physical systems. *In: Proceedings of the 47th Design Automation Conference on - DAC '10.* ACM Press, 2010. Disponível em: <https://doi.org/10.1145/1837274.1837461>.

RIKALOVIC, Aleksandar; SUZIC, Nikola; BAJIC, Bojana; PIURI, Vincenzo. Industry 4.0 implementation challenges and opportunities: A technological perspective. **IEEE Systems Journal**, Institute of Electrical and Electronics Engineers (IEEE), v. 16, n. 2, p. 2797–2810, jun. 2022. Disponível em: <https://doi.org/10.1109/jsyst.2021.3101673>.

RIZK, Yara; AWAD, Mariette; TUNSTEL, Edward W. Cooperative heterogeneous multi-robot systems. **ACM Computing Surveys**, Association for Computing Machinery (ACM), v. 52, n. 2, p. 1–31, mar. 2020. Disponível em: <https://doi.org/10.1145/3303848>.

ROHRICH, Ronnier Frates; TEIXEIRA, Marco Antonio Simoes; LIMA, Jose; OLIVEIRA, Andre Schneider de. Collective gas sensing in a cyber-physical system. **IEEE Sensors Journal**, Institute of Electrical and Electronics Engineers (IEEE), v. 21, n. 12, p. 13761–13771, jun. 2021. Disponível em: <https://doi.org/10.1109/jsen.2021.3069687>.

SEMWAL, Tushar; BODE, Manoj; SINGH, Vivek; JHA, Shashi Shekhar; NAIR, Shivashankar B. Tartarus: A multi-agent platform for integrating cyber-physical systems and robots. *In: Proceedings of the 2015 Conference on Advances In Robotics - AIR '15.* ACM Press, 2015. Disponível em: <https://doi.org/10.1145/2783449.2783469>.

SHI, Jianhua; WAN, Jiafu; YAN, Hehua; SUO, Hui. A survey of cyber-physical systems. *In: 2011 International Conference on Wireless Communications and Signal Processing (WCSP).* [S.l.: s.n.], 2011. p. 1–6.

SHIN, Yong-Jun; LIU, Lingjun; HYUN, Sangwon; BAE, Doo-Hwan. Platooning LEGOs: An open physical exemplar for engineering self-adaptive cyber-physical systems-of-systems. *In: 2021 International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS).* IEEE, 2021. Disponível em: <https://doi.org/10.1109/seams51251.2021.00038>.

TAO, Fei; ZHANG, Meng. Digital twin shop-floor: A new shop-floor paradigm towards smart manufacturing. **IEEE Access**, v. 5, p. 20418–20427, 2017.

WEISS, Astrid; WORTMEIER, Ann-Kathrin; KUBICEK, Bettina. Cobots in industry 4.0: A roadmap for future practice studies on human–robot collaboration. **IEEE Transactions on Human-Machine Systems**, Institute of Electrical and Electronics Engineers (IEEE), v. 51, n. 4, p. 335–345, ago. 2021. Disponível em: <https://doi.org/10.1109/thms.2021.3092684>.

WEST, Andrew; ARVIN, Farshad; MARTIN, Horatio; WATSON, Simon; LENNOX, Barry. ROS integration for miniature mobile robots. *In: Towards Autonomous Robotic Systems*. Cham: Springer International Publishing, 2018, (Lecture notes in computer science). p. 345–356.

XU, Li Da; XU, Eric L.; LI, Ling. Industry 4.0: state of the art and future trends. **International Journal of Production Research**, Informa UK Limited, v. 56, n. 8, p. 2941–2962, mar. 2018. Disponível em: <https://doi.org/10.1080/00207543.2018.1444806>.

XU, Sendren Sheng-Dong; HUANG, Hsu-Chih; KUNG, Yu-Chieh; LIN, Shao-Kang. Collision-free fuzzy formation control of swarm robotic cyber-physical systems using a robust orthogonal firefly algorithm. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 7, p. 9205–9214, 2019. Disponível em: <https://doi.org/10.1109/access.2018.2888881>.

YAN, Zhi; JOUANDEAU, Nicolas; CHERIF, Arab Ali. A survey and analysis of multi-robot coordination. **International Journal of Advanced Robotic Systems**, SAGE Publications, v. 10, n. 12, p. 399, jan. 2013. Disponível em: <https://doi.org/10.5772/57313>.

YE, Xun; HONG, Seung Ho; SONG, Won Seok; KIM, Yu Chul; ZHANG, Xiongfeng. An industry 4.0 asset administration shell-enabled digital solution for robot-based manufacturing systems. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 9, p. 154448–154459, 2021. Disponível em: <https://doi.org/10.1109/access.2021.3128580>.

YOU, Zhijia; FENG, Lingjun. Integration of industry 4.0 related technologies in construction industry: A framework of cyber-physical system. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 8, p. 122908–122922, 2020. Disponível em: <https://doi.org/10.1109/access.2020.3007206>.

ZHAI, Yuanzhao; DING, Bo; LIU, Xuan; JIA, Hongda; ZHAO, Yong; LUO, Jie. Decentralized multi-robot collision avoidance in complex scenarios with selective communication. **IEEE Robotics and Automation Letters**, Institute of Electrical and Electronics Engineers (IEEE), v. 6, n. 4, p. 8379–8386, out. 2021. Disponível em: <https://doi.org/10.1109/lra.2021.3102636>.