

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E
INFORMÁTICA INDUSTRIAL

JOÃO PAULO LIMA SILVA DE ALMEIDA

**PLANEJAMENTO INTELIGENTE DE TRAJETÓRIAS PARA
NAVEGAÇÃO AUTÔNOMA E COOPERATIVA DE MÚLTIPLOS
ROBÔS**

TESE

CURITIBA

2019

JOÃO PAULO LIMA SILVA DE ALMEIDA

**PLANEJAMENTO INTELIGENTE DE TRAJETÓRIAS PARA
NAVEGAÇÃO AUTÔNOMA E COOPERATIVA DE MÚLTIPLOS
ROBÔS**

Tese 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 “Doutor em Ciências” – Área de Concentração: Engenharia de Automação e Sistemas.

Orientadora: Prof. Dra. Lúcia Valéria Ramos de Arruda

CURITIBA

2019

Dados Internacionais de Catalogação na Publicação

Almeida, João Paulo Lima Silva de

Planejamento inteligente de trajetórias para navegação autônoma e cooperativa de múltiplos robôs [recurso eletrônico] / João Paulo Lima Silva de Almeida.-- 2019.

1 arquivo texto (134 f.) : PDF ; 13,8 MB

Modo de acesso: World Wide Web

Título extraído da tela de título (visualizado em 29 maio 2019)

Texto em português com resumo em inglês

Tese (Doutorado) - Universidade Tecnológica Federal do Paraná. Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial, Curitiba, 2019

Bibliografia: f. 129-134

1. Engenharia elétrica – Teses. 2. Robôs - Sistemas de controle. 3. Robôs autônomos. 4. Robôs móveis. 5. Robótica. 6. Sistemas difusos. 7. Robôs - Programação. I. Arruda, Lúcia Valéria Ramos de. II. Universidade Tecnológica Federal do Paraná. Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial. III. Título.

CDD: Ed. 23 – 621.3

Biblioteca Central da UTFPR, Câmpus Curitiba

Bibliotecário: Adriano Lopes CRB-9/1429

TERMO DE APROVAÇÃO DE TESE Nº 192

A Tese de Doutorado intitulada “**Planejamento Inteligente de Trajetórias para Navegação Autônoma e Cooperativa de Múltiplos Robôs**”, defendida em sessão pública pelo(a) candidato(a) **João Paulo Lima Silva de Almeida**, no dia **26 de abril de 2019**, foi julgada para a obtenção do título de Doutor em Ciências, **área de concentração Engenharia de Automação e Sistemas**, e aprovada em sua forma final, pelo Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial.

BANCA EXAMINADORA:

Prof(a). Dr(a). Lucia Valéria Ramos de Arruda - Presidente – (UTFPR)
Prof(a). Dr(a). André Schneider de Oliveira – (UTFPR)
Prof(a). Dr(a). João Alberto Fabro – (UTFPR)
Prof(a). Dr(a). Sílvia Silva da Costa Botelho – (FURG)
Prof(a). Dr(a). Luciano Cunha de Araújo Pimenta - (UFMG)

A via original deste documento encontra-se arquivada na Secretaria do Programa, contendo a assinatura da Coordenação após a entrega da versão corrigida do trabalho.

Curitiba, 26 de abril de 2019.

Dedico esta tese aos meus pais, que sempre incentivaram minha educação formal.

AGRADECIMENTOS

Agradeço a Deus pelo dom da vida e pela força para não desistir.

À Profa. Dra. Lúcia Valéria Ramos de Arruda, pela orientação e apoio incondicional nesta pesquisa.

Agradeço também aos colegas do Laboratório de Automação e Sistemas de Controle Avançado (LASCA), pela amizade e companheirismo prestados durante este período.

Ao Instituto Federal do Paraná - Jacarezinho (IFPR), por proporcionar as condições necessárias para a realização desta pesquisa.

Agradeço à minha família pelo incentivo durante a minha formação acadêmica.

RESUMO

ALMEIDA, J. P. L. S.. PLANEJAMENTO INTELIGENTE DE TRAJETÓRIAS PARA NAVEGAÇÃO AUTÔNOMA E COOPERATIVA DE MÚLTIPLOS ROBÔS. 135 f. Tese – Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

As aplicações da robótica são destacadas em diversas áreas do conhecimento, especialmente aquelas que consideram sistemas multirrobo, em que múltiplos robôs móveis são capazes de desenvolver uma tarefa de forma autônoma e cooperativa. Neste contexto, os sistemas inteligentes são ferramentas válidas para promover a navegação e coordenação de um sistema multirrobo. Este trabalho tem como principal objetivo apresentar soluções para problemas que requerem o planejamento de rotas para múltiplos robôs, definidos a partir de um ambiente que contém pontos de passagem obrigatórios (alvos) e obstáculos estáticos e dinâmicos. Neste ambiente, o sistema multirrobo deve visitar todos os alvos de forma cooperativa e finalizar a navegação em um local pré-definido. Inicialmente, o conhecimento do ambiente é limitado apenas à posição dos alvos (os obstáculos são desconhecidos) e, portanto, um planejador híbrido de rotas é proposto para calcular os caminhos a serem navegados pelos robôs a partir deste conhecimento prévio (etapa *off-line*) e tratar problemas inesperados durante a navegação (etapa *on-line*, desenvolvida individualmente por cada robô), tais como o desvio de obstáculo e perda de localização. O planejador híbrido é modelado através de uma adaptação do problema do caixeiro viajante múltiplo, e solucionado com algoritmos genéticos combinados a heurísticas cooperativas. Em uma segunda etapa, o conhecimento prévio sobre o ambiente consiste apenas nas posições inicial e final dos robôs. Nesse caso, os robôs devem explorar o ambiente para localizar e visitar os alvos, enquanto desviam dos obstáculos, até atingir as suas posições finais. Para isto, uma estratégia *on-line* e bioinspirada em comportamento de estigmergia por feromônios artificiais é proposta para promover uma navegação cooperativa e distribuída. Os robôs considerados neste trabalho são homogêneos, independentes, possuem habilidades de comunicação limitadas e são dotados de controladores *fuzzy* para sua movimentação. Resultados de simulação são apresentados para a validação dos planejadores e uma implementação com robôs reais em um ambiente experimental confirma a factibilidade das estratégias propostas.

Palavras-chave: Sistema multirrobo; Planejamento de rota; Navegação autônoma; Controle *fuzzy*.

ABSTRACT

ALMEIDA, J. P. L. S.. INTELLIGENT PATH PLANNING FOR AUTONOMOUS AND COOPERATIVE NAVIGATION OF MULTIPLE ROBOTS. 135 f. Tese – Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

Robotic applications are highlighted in several areas, especially those that consider multi-robot systems, in which multiple mobile robots are able to develop a task in an autonomous and cooperative way. In this context, intelligent systems are valid tools to provide a navigation and coordination of a multi-robot system. This work has as main objective to present solutions to problems that require the path planning for multiple robots, defined from an environment that contains points of obligatory passage (targets) and static and dynamic obstacles. In this environment, the multi-robot system must visit all targets in a cooperative way and finish the navigation in a pre-defined location. Initially, the knowledge of the environment is limited only on the targets' position (obstacles are unknown), and then a hybrid path planner is proposed to compute the paths to be navigated by the robots from that previous knowledge (off-line step) and handle unexpected problems during navigation (on-line step, individually performed by each robot), such as obstacle avoidance and loss of localization. The hybrid planner is modeled through a multiple traveling salesperson problem adaptation, and solved with genetic algorithms and other cooperative heuristics. In a second step, the previous knowledge about the environment consists only on the initial and final positions of the robots. In this case, robots should explore the environment to locate and visit the targets, while avoiding obstacles, until they reach their final positions. To do this, an on-line and bio-inspired strategy on stigmergic behavior from artificial pheromones is proposed to provide a cooperative and distributed navigation. The robots considered in this work are homogeneous, independent, have limited communication skills and are equipped with fuzzy controllers for their locomotion. Results from simulation are presented for the planners validation and a implementation with real robots in an experimental environment confirms the feasibility of the proposed strategies.

Keywords: Multi-robot system; Path planning; Autonomous navigation; Fuzzy control.

LISTA DE FIGURAS

FIGURA 1	– Robôs <i>Kilobot</i>	23
FIGURA 2	– Uma visão geral dos sistemas multirrobô.	25
FIGURA 3	– Representação simulada de um ambiente com robôs móveis, alvos e obstáculos estáticos.	32
FIGURA 4	– Especificações da estrutura dos robôs móveis: (a) vistas superiores e (b) vistas laterais.	35
FIGURA 5	– Sensores de proximidade dos robôs móveis.	37
FIGURA 6	– Comunicação direta e compartilhamento de informações.	38
FIGURA 7	– Sistemas <i>fuzzy</i> para o controle de locomoção dos robôs móveis.	40
FIGURA 8	– Ilustração das variáveis de entrada do sistema <i>fuzzy</i> para o seguimento de rota.	41
FIGURA 9	– Funções de pertinência das entradas do interpolador <i>fuzzy</i>	42
FIGURA 10	– Funções de pertinência da saída do interpolador <i>fuzzy</i>	43
FIGURA 11	– Superfície de resposta referente ao sistema <i>fuzzy</i> para a tomada de decisão.	44
FIGURA 12	– Ilustração da propagação do erro em sistemas de odometria.	45
FIGURA 13	– Configuração dos experimentos de simulação e com robôs reais.	48
FIGURA 14	– Exemplo de solução para a configuração em que a posição dos alvos é conhecida e dos obstáculos é desconhecida.	50
FIGURA 15	– Exemplo numérico de codificação de um cromossomo.	52
FIGURA 16	– Procedimentos para o operador de cruzamento TCX: (a) cromossomos progenitores (P1 e P2) selecionados; (b) seleção aleatória dos genes de P1; (c) seleção dos genes de P2 e (d) combinação e obtenção de um cromossomo descendente (D1).	53
FIGURA 17	– Ilustração das variáveis definidas para a função de aptidão (<i>fitness</i>) do algoritmo genético. <i>A</i> é a situação inicial; <i>B</i> é a situação em que o primeiro alvo foi visitado; <i>C</i> é o momento em que o robô visita o seu segundo alvo; e em <i>D</i> o robô alcança sua posição final.	54
FIGURA 18	– Manobra para correção de rota devido à presença de obstáculos.	56
FIGURA 19	– Regra de prioridade para o algoritmo de triangulação: (a) O <i>landmark</i> A3 possui prioridade sobre o robô R2. (b) O robô com a menor imprecisão (R2) é priorizado ao invés de R3. (c) Robôs sob a condição de navegação cega não são usados neste procedimento.	58
FIGURA 20	– Procedimentos executados pelo planejamento <i>on-line</i> de rotas.	59
FIGURA 21	– Cenário 4 simulado no V-REP: 20 robôs e 80 alvos.	60
FIGURA 22	– Distâncias planejadas: distribuição das rotas nos cenários 1 e 2.	62
FIGURA 23	– Distâncias planejadas: distribuição das rotas nos cenários 3 e 4.	62
FIGURA 24	– Distribuição de alvos entre os robôs nos cenários 1 e 2.	63
FIGURA 25	– Distribuição de alvos entre os robôs nos cenários 3 e 4.	64
FIGURA 26	– Rotas planejadas para um cenário com $n_A = 40$ e $n_R = 10$	66
FIGURA 27	– Rotas navegadas com $R_{Det} = 1,5 m$	67
FIGURA 28	– Área da elipse de R8.	68

FIGURA 29	– Evolução da variável τ para tomadas de decisão.	69
FIGURA 30	– Ações de controle aplicadas ao robô R8.	69
FIGURA 31	– Rotas navegadas com $R_{Det} = 3 m$	70
FIGURA 32	– Exemplo de um cenário em que a posição dos alvos e dos obstáculos no ambiente é desconhecida.	74
FIGURA 33	– Ilustração da dinâmica de compartilhamento de informações.	76
FIGURA 34	– Trilha de feromônios.	77
FIGURA 35	– Comunicação indireta: o robô R1 detecta o feromônio lançado por R2. ...	78
FIGURA 36	– Tomada de decisão: tarefas e planejamento de rota.	79
FIGURA 37	– Fluxograma do algoritmo heurístico de atribuição de alvos.	81
FIGURA 38	– Ilustração da decisão de explorar o ambiente.	83
FIGURA 39	– Correção da rota devido a presença de feromônio.	84
FIGURA 40	– Ambiente no simulador V-REP.	86
FIGURA 41	– Rotas executadas por 5 robôs no cenário apresentado na Figura 40, sem a tarefa de exploração do ambiente e sem a deposição de feromônios.	87
FIGURA 42	– Rotas dos 5 robôs: (a) $\gamma^+ = 0,015$ e (b) $\gamma^+ = 0,03$	89
FIGURA 43	– Área total explorada: (a) $\gamma^+ = 0,015$ e (b) $\gamma^+ = 0,03$	91
FIGURA 44	– Decisões de R1 com $\gamma^+ = 0,015$: (a) decisão da tarefa a ser realizada e (b) correção da rota calculada.	92
FIGURA 45	– Decisões de R3 com $\gamma^+ = 0,015$: (a) decisão da tarefa a ser realizada e (b) correção da rota calculada.	92
FIGURA 46	– Decisões de R1 e R3 para evitar colisões (τ).	93
FIGURA 47	– Resultado das métricas.	96
FIGURA 48	– Balanceamento da distribuição de tarefas.	97
FIGURA 49	– Ilustração dos casos especiais: (a) Cenário 5 e (b) Cenário 6.	100
FIGURA 50	– AUC: (a) $R_{Det} = 3$ e (b) $R_{Det} = 0,9$. A = estratégia proposta neste capítulo, B = RDPSO, C = AFS, D = GSO, E = PPSO e F = EPSO.	105
FIGURA 51	– Robô Sphero.	107
FIGURA 52	– Ambiente real e seus dispositivos.	108
FIGURA 53	– Diagrama do fluxo de operação da arquitetura experimental.	109
FIGURA 54	– Sistema de localização dos robôs: (a) marcador ARTag e (b) suporte do marcador.	110
FIGURA 55	– Ambiente da Figura 52 em realidade aumentada.	111
FIGURA 56	– Processamento dos <i>scripts</i> dos robôs.	112
FIGURA 57	– Exemplo da comunicação direta na arquitetura proposta: compartilhamento de informações entre robôs.	114
FIGURA 58	– <i>Frames</i> de execução do primeiro cenário da parte 1.	116
FIGURA 59	– Desempenho local de um robô: (a) ações de controle e (b) evolução dinâmica de τ	116
FIGURA 60	– Rotas planejadas para os robôs R1, R2, R3 e R4.	117
FIGURA 61	– <i>Frames</i> de execução do segundo cenário da parte 1.	118
FIGURA 62	– <i>Frames</i> de execução do primeiro cenário da parte 2.	120
FIGURA 63	– Evolução da variável de tomada de decisão τ : experimentos da parte 2. ..	121
FIGURA 64	– Ações de controle aplicada ao robô R: experimentos da parte 2.	122
FIGURA 65	– <i>Frames</i> de execução do segundo cenário da parte 2.	123

LISTA DE TABELAS

TABELA 1	– Comparação com trabalhos relacionados.	31
TABELA 2	– Base de regras do sistema <i>fuzzy</i> para tomadas de decisão.	42
TABELA 3	– Operadores do algoritmo genético para o planejador de rotas <i>off-line</i>	52
TABELA 4	– Características dos cenários simulados para a validação do planejador <i>off-line</i> de rotas.	60
TABELA 5	– Parâmetros usados no algoritmo genético.	61
TABELA 6	– Resultados estatísticos para as distâncias planejadas da melhor solução. ..	63
TABELA 7	– Distâncias planejadas e distribuição de alvos obtidas na melhor solução. ..	65
TABELA 8	– Resultado das distâncias planejadas e navegadas para um ambiente com $n_A = 40$ e $n_R = 10$	67
TABELA 9	– Estudo comparativo referente ao planejador <i>off-line</i> de rotas.	71
TABELA 10	– Parâmetros da simulação.	87
TABELA 11	– Parâmetros das simulações referentes a validação global do planejador. ...	93
TABELA 12	– Número de iterações (k).	96
TABELA 13	– Número de alvos visitados nos casos especiais.	101
TABELA 14	– Áreas exploradas nos casos especiais (m^2).	101
TABELA 15	– Percentual de células não exploradas.	103

LISTA DE SIGLAS

AE	Área Explorada
ACO	<i>Ant Colony Optimization</i>
AFS	<i>Aggregations of Foraging Swarm</i>
AGV	<i>Automated Guided Vehicles</i>
AUC	<i>Area Under the Curve</i>
DE	<i>Differential Evolution</i>
DN	Distância Navegada
DP	Distância Planejada
EPSO	<i>Extended Particle Swarm Optimization</i>
GSO	<i>Glowworm Swarm Optimization</i>
IA	<i>Instantaneous assignment</i>
LQR	<i>Linear Quadratic Regulator</i>
MTSP	<i>Multiple Traveling Salesman Problem</i>
PPSO	<i>Physically-embedded Particle Swarm Optimization</i>
PSO	<i>Particle Swarm Optimization</i>
RDPSO	<i>Robotic Darwinian Particle Swarm Optimization</i>
RNA	Redes Neurais Artificiais
ROS	<i>Robot Operating System</i>
SMR	Sistema Multirrobo
SR	<i>Single-robot tasks</i>
ST	<i>Single-task robots</i>
TA	<i>Time-extended assignment</i>
TCX	<i>Two-Chromosome Crossover</i>
TSP	<i>Traveling Salesman Problem</i>
UAV	<i>Unmanned Aerial Vehicle</i>
V-REP	<i>Virtual Robot Experimental Platform</i>

LISTA DE SÍMBOLOS

R_E	Roda esquerda do robô móvel.
R_D	Roda direita do robô móvel.
RS_1	Roda de suporte frontal do robô móvel.
RS_2	Roda de suporte traseira do robô móvel.
$Sensor_E$	Sensor de proximidade à esquerda do robô móvel.
$Sensor_F$	Sensor de proximidade frontal do robô móvel.
$Sensor_D$	Sensor de proximidade à direita do robô móvel.
d_{max}	Distância máxima de medição dos sensores de proximidade.
R_{Det}	Tamanho da região de detecção/comunicação de um robô móvel.
$\sigma_{x,y}(k)$	Função que descreve o depósito e a evaporação de feromônios em um determinado ponto (x,y) , no instante k .
ρ	Taxa de evaporação dos feromônios artificiais.
$\hat{\sigma}_{x,y}$	Sinal de feromônio lançado por um robô no instante k .
$LVel$	Velocidade linear.
$AVel$	Velocidade angular.
u_{D1}	Ação de controle referente à roda direita do robô, obtida do módulo “Seguimento de rota”.
u_{E1}	Ação de controle referente à roda esquerda do robô, obtida do módulo “Seguimento de rota”.
$Erro_D$	Diferença entre as posições do robô e do seu objetivo.
$\Delta Erro_D$	Varição de $Erro_D$ em uma amostra.
$Erro_\alpha$	Diferença angular entre as posições do robô e do seu objetivo.
$\Delta Erro_\alpha$	Varição de $Erro_\alpha$ em uma amostra.
u_{D2}	Ação de controle referente à roda direita do robô, obtida do módulo “Desvio de obstáculos”.
u_{E2}	Ação de controle referente à roda esquerda do robô, obtida do módulo “Desvio de obstáculos”.
u_D	Ação de controle aplicada à roda direita do robô.
u_E	Ação de controle aplicada à roda esquerda do robô.
τ	Variável <i>fuzzy</i> para tomadas de decisão sobre a navegação.
$Dist_{RA}$	Distância entre as posições atual do robô e do seu objetivo.
$Sensor$	Intensidade do sensor que apontar a maior iminência de colisão.
U	Universo de discurso.
p	Vetor que representa a posição do robô no ambiente, composto pelas coordenadas x,y e orientação.

θ	Orientação de um robô no ambiente bidimensional.
Δs_d	Rotação da roda direita do robô.
Δs_e	Rotação da roda esquerda do robô.
b	Distância entre as rodas direita e esquerda do robô.
k_d	Constante de erro associada à roda direita do robô.
k_e	Constante de erro associada à roda esquerda do robô.
n_R	Número de robôs no sistema multirrobo.
R_i	Notação para o i -ésimo robô.
n_A	Número de alvos inseridos no cenário.
S	Conjunto de alvos do cenário.
A_j	Notação para o j -ésimo alvo do conjunto S .
S_{R_i}	Conjunto de alvos atribuídos ao robô R_i .
$SP_{\alpha k}$	Ângulo entre o vetor de referência no eixo x e o vetor que interliga as posições atual e a final desejada do robô, no instante k .
NA_{R_i}	Número de alvos contidos no subconjunto S_{R_i} .
$Dist_{R_{ij}}$	Distância a ser navegada pelo robô R_i da sua posição atual ao seu j -ésimo objetivo.
$Dist_{R_i}$	Distância total a ser navegada pelo robô R_i .
$Dist_{Total}$	Distância total a ser navegada pelo sistema multirrobo.
$R_{Dist}^{R_i}$	Relação entre $Dist_{R_i}$ e $Dist_{Total}$.
$Erro_{\alpha}^{R_{ij}}$	Desvio angular entre a frente do robô e sua posição final.
$R_{Erro_{\alpha}}^{R_i}$	Soma de $Erro_{\alpha}^{R_{ij}}$, normalizado por 180° .
$R_{NA}^{R_i}$	Relação entre NA_{R_i} e n_A .
$\pm \alpha_{desvio}$	Rotação aplicada ao vetor que interliga a posição atual do robô e seu objetivo.
K_{NC}	Valor de limiar para o tamanho da área da elipse.
SL	É o tamanho da rota em linha reta que interliga as posições inicial e final de um robô.
D_1	Decisão de visitar um alvo.
D_2	Decisão de explorar o ambiente.
D_3	Decisão de ir ao ponto final.
D_4	Decisão de desviar de feromônios.
D_5	Decisão de desviar de obstáculos.
h_i	Notação para a i -ésima variável auxiliar do sistema de tomada de decisão.
γ	Fator de ponderação variável aplicado à decisão D_2 .
η	Relação entre o número de alvos visitados pelo robô e seu limite de carga.
γ^+	Valor de incremento para γ .
α_{Exp}	Rotação aplicada quando a decisão D_2 é tomada pelo robô.
$\pm \alpha_F$	Rotação aplicada quando a decisão D_4 é tomada pelo robô.
ζ_{Exp}	Limiar de decisão entre D_2 e D_3 .

ζ_F	Limiar de decisão para D_4 .
C_{Area}	Custo da exploração de área.
C_{Alvos}	Custo para visitar alvos.
$A_{Visitados}$	Número total de alvos visitados.
$Eficacia_{Area}$	Eficiência na exploração de área.
$Area_{Amb}$	Área total do ambiente.
$Eficacia_{Alvos}$	Eficácia para visitar alvos.
$Eficiencia_{Area}$	Eficiência na exploração de área.
AE_{sob}	Áreas exploradas por mais de um robô (sobrepostas).
N_{Iter}	Número de iterações.

SUMÁRIO

1 INTRODUÇÃO	13
1.1 MOTIVAÇÃO	13
1.2 OBJETIVOS	16
1.2.1 Objetivo Geral	16
1.2.2 Objetivos Específicos	16
1.3 CONTRIBUIÇÕES DA TESE	17
1.4 ORGANIZAÇÃO DO TRABALHO	18
1.5 ARTIGOS RELACIONADOS	18
2 FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS	20
2.1 FUNDAMENTOS DE SISTEMAS MULTIRROBÔ	20
2.2 ENXAME DE ROBÔS	22
2.3 NAVEGAÇÃO AUTÔNOMA E PLANEJAMENTO DE ROTA	24
2.4 APLICAÇÕES GERAIS	25
2.5 ASPECTOS DE IMPLEMENTAÇÃO DE SISTEMAS MULTIRROBÔ	27
2.6 TRABALHOS RELACIONADOS	29
3 DESCRIÇÃO DO AMBIENTE E DO SISTEMA MULTIRROBÔ	32
3.1 DESCRIÇÃO DOS ROBÔS E DO SISTEMA BÁSICO DE NAVEGAÇÃO	35
3.1.1 Habilidade de detecção	36
3.1.2 Habilidade de comunicação direta	37
3.1.3 Habilidade de comunicação indireta	38
3.1.4 Sistema <i>fuzzy</i> de controle de navegação	39
3.1.5 Sistema de autolocalização dos robôs	44
3.2 CONFIGURAÇÃO DOS EXPERIMENTOS	47
4 PLANEJAMENTO DE ROTAS PARA TAREFAS DE VISITAR ALVOS	
CONHECIDOS EM UM AMBIENTE SEMI-DESCONHECIDO	49
4.1 PLANEJAMENTO <i>OFF-LINE</i> DE ROTAS BASEADO EM ALGORITMO GENÉTICO	50
4.2 PLANEJAMENTO <i>ON-LINE</i> DE ROTAS	55
4.3 RESULTADOS SIMULADOS	58
4.3.1 Validação do planejamento <i>off-line</i> de rotas	60
4.3.2 Desempenho do sistema de navegação	64
4.3.3 Estudos comparativos	70
5 PLANEJAMENTO DE ROTAS PARA TAREFAS DE EXPLORAÇÃO DE AMBIENTES	73
5.1 REQUISITOS DE COMUNICAÇÃO	74
5.1.1 Comunicação direta e compartilhamento de informações	75
5.1.2 Comunicação indireta e estigmergia	75
5.2 PLANEJADOR DE ROTAS <i>ON-LINE</i> E BIOINSPIRADO	77
5.2.1 Camada de planejamento inteligente: decisões deliberativas	79
5.2.2 Camada reativa: decisões de navegação	85

5.3	RESULTADOS SIMULADOS	85
5.3.1	Validação do planejamento de rotas bioinspirado	85
5.3.2	Desempenho global do planejamento de rotas	93
5.3.3	Casos especiais	100
5.3.4	Estudos comparativos	102
5.3.4.1	Estudo comparativo 1	102
5.3.4.2	Estudo comparativo 2	104
6	UMA ARQUITETURA BASEADA EM REALIDADE AUMENTADA PARA A	
	VALIDAÇÃO DO SISTEMA MULTIRROBÔ PROPOSTO	106
6.1	DESCRIÇÃO DOS ROBÔS SPHERO	107
6.2	DESCRIÇÃO DA ARQUITETURA EXPERIMENTAL BASEADA EM	
	REALIDADE AUMENTADA	108
6.2.1	Representação dos objetos no ambiente real	109
6.2.2	Metodologia de implementação das estratégias	111
6.3	RESULTADOS EXPERIMENTAIS	115
6.3.1	Parte 1: implementação da estratégia híbrida de planejamento de rotas	115
6.3.2	Parte 2: implementação da estratégia bioinspirada em feromônios	119
7	CONCLUSÕES	125
7.1	TRABALHOS FUTUROS	128
	REFERÊNCIAS	129

1 INTRODUÇÃO

Neste capítulo são apresentadas as principais considerações acerca da motivação do tema de tese, dos objetivos geral e específicos, das contribuições da tese, da organização do texto e dos artigos produzidos a partir dos resultados obtidos.

1.1 MOTIVAÇÃO

Os sistemas multirrobo (SMR) e sua sub-área enxame de robôs (*Swarm Robotics*) vem se destacando em diversas pesquisas recentes em robótica móvel, que consistem na utilização de múltiplos robôs móveis para alcançar um objetivo comum de forma cooperativa e por meio de ações locais em um ambiente compartilhado (BAYINDIR, 2016).

Em sistemas com múltiplos robôs, as tarefas complexas realizadas por um único robô podem ser simplificadas e decompostas em ações de menor complexidade, para que robôs com habilidades limitadas possam cumprí-las de forma distribuída e independente. Alguns exemplos dessa distribuição de tarefas entre robôs compreendem as operações de busca e resgate (YANGUAS-ROJAS; MOJICA-NAVA, 2017; NIETO-GRANDA et al., 2014; PUIG et al., 2011), mapeamento de campo minado (KHAMIS; ELGINDY, 2012), plantio e pulverização em agricultura (BECHAR; VIGNEAULT, 2017), exploração de ambientes nocivos (PALMIERI et al., 2018; RANGO et al., 2018), transporte de cargas (PETITTI et al., 2016), entre outras.

As tarefas mencionadas compartilham características que definem um requisito para os robôs envolvidos a fim de que o objetivo global seja cumprido. Por exemplo, o menor tempo para encontrar sobreviventes de desastres naturais (missões de busca e resgate) é um requisito crucial, assim como a navegação em rotas que favoreçam a economia de recursos, tais como combustível ou bateria até determinados pontos de interesse. Assim, estratégias de planejamento de rota e heurísticas cooperativas são frequentemente utilizadas no desenvolvimento de SMRs para essas aplicações.

Por exemplo, os autores Han et al. (2017) e Li et al. (2017) ressaltam a importância

dos conceitos que envolvem o planejamento de rotas para múltiplos robôs quando aplicados em um ambiente industrial. Nestes ambientes, múltiplos *Automated Guided Vehicles* (AGV) devem seguir rotas previamente calculadas para visitar estações de trabalho ou outros pontos de interesse de forma eficiente, visando a minimização das distâncias a serem percorridas ou outros critérios. A otimização destas rotas é alcançada a partir do conhecimento prévio da posição de todos os pontos a serem visitados e dos obstáculos, dos quais os agentes móveis devem evitar colisão.

Entretanto, nem sempre é possível garantir que a base de conhecimentos mencionada tenha informações suficientes para que as rotas ótimas sejam calculadas previamente. Como exemplo, as rotas planejadas de múltiplos AGVs podem ser interrompidas por obstáculos estáticos ou dinâmicos, que possuem uma posição desconhecida ao grupo de robôs (LI et al., 2017). Uma segunda situação é o caso de ambientes que possuem minas terrestres com localização desconhecida, que devem ser encontradas e desarmadas por múltiplos robôs (PALMIERI et al., 2018).

Uma estratégia de planejamento para as ocasiões exemplificadas deve contar com procedimentos *on-line* e/ou híbridos para o ajuste das rotas conforme os eventos inesperados ocorrem. Tais procedimentos são frequentemente dependentes das habilidades que os robôs devem apresentar localmente para obter as informações necessárias, como a detecção de obstáculos e/ou alvos, compartilhamento de informações, entre outras. Neste contexto, as estratégias de coordenação do SMR e os mecanismos de sensoriamento e comunicação são primordiais para o êxito da etapa de planejamento de rota (ZHOU et al., 2017).

Portanto, observa-se que o planejamento de rota é um passo importante no desenvolvimento de um sistema de navegação autônomo para um SMR. Abordagens por meta-heurísticas de otimização demonstram resultados promissores nesta área de aplicação (LARSEN et al., 2018; JOSE; PRATIHAR, 2016). Duas estratégias de planejamento de rota são frequentemente consideradas na literatura da área: *off-line* e *on-line*. No primeiro caso, são adotados os conhecimentos *a priori* do ambiente de navegação e/ou os dados passados, advindos de sensoriamento, a fim de se calcular as rotas dos diversos robôs em um procedimento *off-line*. Já, no planejamento de rota *on-line*, o procedimento utiliza dados locais do ambiente, adquiridos por meio de interações ao longo da navegação, e assim cada agente tem sua rota planejada de forma individual e *on-line*, isto é, em tempo de navegação.

Com base nas informações apresentadas, problemas relacionados ao planejamento de rota e às estratégias de coordenação de sistemas multirrobô são investigados nesta tese. Esses problemas são decorrentes essencialmente do uso de um ambiente estruturado em que

diversos pontos de passagem obrigatória (alvos) e obstáculos estáticos são inseridos em posições conhecidas ou não. Dependendo da aplicação, os alvos podem representar vítimas de desastres naturais que devem ser encontradas e resgatadas, pontos de recursos (combustível, alimento, etc), objetos a serem manipulados, entre outras interpretações. Neste contexto, os múltiplos robôs móveis devem, a partir de um ponto inicial, visitar todos os alvos de forma cooperativa, evitar colisão com os obstáculos estáticos e com outros robôs (obstáculos dinâmicos), e alcançar uma posição final pré-determinada. Estas características configuram um problema de planejamento de rota e coordenação de grupos de robôs com tomadas de decisão (PARKER, 2009).

Neste trabalho, estratégias de planejamento de rota são propostas a partir de duas situações: (i) o SMR possui a informação prévia da posição de todos os alvos inseridos no ambiente, no entanto os obstáculos são desconhecidos e há incertezas na posição de cada robô durante a navegação; e (ii) todos os elementos do ambiente (alvos e obstáculos) estão em posições desconhecidas pelo SMR. Em ambas as situações, objetiva-se minimizar a distância a ser navegada pelo SMR, distribuir uniformemente (balanceamento) a carga de trabalho entre os robô e maximizar a eficiência na exploração do ambiente. Para isto, é necessário desenvolver um sistema de navegação autônomo para o SMR, com capacidade de planejamento de rota *off-line* e *on-line*, de acordo com as situações elencadas. Todos os robôs móveis são homogêneos e possuem um rol de habilidades que os permitem desempenhar a comunicação direta/indireta e a detecção de obstáculos, ambas com alcance limitado, como usualmente é encontrado em sistemas reais.

As estratégias propostas são validadas por meio de simulações e experimentos práticos com robôs reais. A configuração de simulação utiliza a plataforma *Virtual Robot Experimental Platform* (V-REP) (ROBOTICS, 2017) para implementar o SMR (características dos robôs móveis e do ambiente), o *software* Matlab para executar os algoritmos de planejamento e controle, e o *framework Robot Operating System* (ROS) (QUIGLEY et al., 2009) para intermediar a interação entre V-REP e Matlab. Na validação prática, robôs móveis reais com recursos limitados são usados em uma plataforma experimental desenvolvida neste trabalho, a qual fornece, virtualmente, as habilidades de navegação necessárias para os robôs cumprirem a tarefa definida. O *framework* ROS também é usado como principal meio para a interação entre os robôs reais e os algoritmos de planejamento e controle.

1.2 OBJETIVOS

Os objetivos deste trabalho de doutorado são sintetizados em geral e específicos, conforme explicados a seguir.

1.2.1 OBJETIVO GERAL

O objetivo geral consiste em desenvolver um sistema autônomo de navegação para múltiplos robôs com limitações de sensoriamento e comunicação, inseridos em ambientes semi-desconhecidos e com capacidade de planejamento de rotas *off-line* (com adaptação de caminho) e *on-line*.

1.2.2 OBJETIVOS ESPECÍFICOS

A partir dos conceitos apresentados anteriormente, traçaram-se os seguintes objetivos específicos para a realização deste trabalho:

- investigar as principais e atuais estratégias distribuída para o planejamento de rota e controle inteligente de sistemas multirrobô inseridos em ambientes semi ou totalmente desconhecidos;
- propor um sistema *fuzzy* distribuído para o controle de navegação autônoma dos robôs móveis;
- desenvolver uma estratégia híbrida de planejamento de rotas para um SMR, envolvendo conceitos de heurísticas cooperativas, de algoritmo genético e com aplicações em ambientes semi-desconhecidos;
- propor uma estratégia de planejamento de rota *on-line*, bioinspirada em feromônios artificiais e outras heurísticas cooperativas, para promover a exploração de um ambiente desconhecido por um grupo de robôs autônomos;
- avaliar a capacidade para tomadas de decisão cooperativa e autônoma dos robôs móveis frente a situações inesperadas;
- implementar uma plataforma experimental, composta pelo *framework* ROS e baseada em realidade aumentada, capaz de compensar as limitações de comunicação e sensoriamento dos robôs;

- simular as estratégias propostas na plataforma desenvolvida;
- confirmar a viabilidade da proposta por meio de experimentos com robôs reais.

1.3 CONTRIBUIÇÕES DA TESE

As principais contribuições previstas desta tese são sintetizadas nos seguintes tópicos:

- desenvolvimento de um sistema *fuzzy* de navegação, capaz de conferir aos robôs a capacidade de seguir uma rota calculada e tomar decisões locais para o desvio de obstáculos estáticos e dinâmicos, a partir de sensoriamento local;
- uma estratégia híbrida de planejamento de rota, com módulos *off-line* e *on-line*, é proposta para definir o caminho dos múltiplos robôs em um ambiente semi-desconhecido (a posição dos alvos é conhecida e dos obstáculos é desconhecida). O módulo *off-line* é executado antes do início da navegação e realiza uma otimização combinatória por algoritmo genético, modelada como uma variação do problema do caixeiro viajante múltiplo (do inglês: *Multiple Traveling Salesman Problem* (MTSP)) (YUAN et al., 2013; CARTER; RAGSDALE, 2006). Este módulo calcula uma lista de tarefas para cada robô, que contém apenas os alvos atribuídos a ele e suas posições inicial e final. Durante a navegação, os robôs contam com o módulo *on-line* para corrigir os erros em sua rota, ocasionados por desvios de obstáculos e pela imprecisão de sua posição, modelada a partir dos erros cumulativos da odometria (SIEGWART et al., 2011). A correção da localização é executada de forma cooperativa, por meio de comunicação direta entre os robôs, e inspirada em algoritmos de triangulação (LEE et al., 2016).
- é proposta uma estratégia *on-line* e distribuída para a exploração de ambientes desconhecidos (a posição dos alvos e dos obstáculos é desconhecida). Os robôs são capazes de planejar localmente suas rotas, de acordo com suas interações com outros robôs e com o ambiente. O principal mecanismo de coordenação desta estratégia é bioinspirado no comportamento de estigmergia por feromônios artificiais, semelhante ao utilizado nas técnicas de otimização por colônia de formigas (*Ant Colony Optimization*) (DORIGO et al., 2006) e outras heurísticas cooperativas. Assume-se que cada robô é capaz de lançar feromônios no ambiente conforme se movimentam.

1.4 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado da seguinte forma: no Capítulo 2 são apresentados os principais fundamentos de sistemas multirrobo, juntamente com um resumo dos trabalhos relacionados ao tema desta tese. Em seguida, no Capítulo 3, são apresentadas as características do ambiente e dos robôs considerados neste trabalho. No Capítulo 4 é proposto o planejador *off-line/on-line* de rotas para sistemas multirrobo inseridos em ambientes semi-desconhecidos. Em seguida, um planejador *on-line* de rotas bioinspirado em feromônios artificiais é proposto no Capítulo 5. Uma implementação prática das estratégias propostas em um ambiente experimental, baseado em realidade aumentada, é descrita no Capítulo 6. Os resultados obtidos dos experimentos envolvendo as estratégias propostas são discutidos em seus próprios capítulos, enquanto as conclusões desta tese e seus possíveis trabalhos futuros são apresentadas no Capítulo 7.

1.5 ARTIGOS RELACIONADOS

A partir dos resultados obtidos no decorrer do desenvolvimento desta tese, os seguintes artigos foram publicados (em ordem cronológica):

- MENDONÇA, M.; ARRUDA, L. V. R.; CHRUN, I. R.; ALMEIDA, J. P. L. S. Sistemas Multi-Agentes Cooperativos Inspirados em Robótica de Enxame. **SBAI 2015 - Simpósio Brasileiro de Automação Inteligente**. Natal - RN, Brasil, 2015.
- ALMEIDA, J. P. L. S.; ARRUDA, L. V. R.; NEVES-JR, F. Planejamento de Rota Por Meio de Algoritmo Genético Para Um Enxame de Robôs. **SBAI 2017 - Simpósio Brasileiro de Automação Inteligente**. Porto Alegre - RS, Brasil, 2017.
- NAKASHIMA, R. T.; CAMPERA, B.; ALMEIDA, J. P. L. S.; ARRUDA, L. V. R.; NEVES-JR, F. Uma Arquitetura Centralizada Para a Navegação Autônoma de Múltiplos Robôs Sphero. **CBA 2018 - Congresso Brasileiro de Automática**. João Pessoa - PB, Brasil, 2018.
- ALMEIDA, J. P. L. S.; NAKASHIMA, R. T.; NEVES-JR, F.; ARRUDA, L. V. R. Bio-inspired on-line path planner for cooperative exploration of unknown environment by a multi-robot system. **Robotics and Autonomous Systems**, 112, p. 32-48, 2019.

Os seguintes trabalhos foram submetidos:

- ALMEIDA, J. P. L. S.; NAKASHIMA, R. T.; NEVES-JR, F.; ARRUDA, L. V. R. A global/local path planner for swarm robotics with uncertain robot localization. **Journal of Intelligent & Robotic Systems**, 2019.
- ALMEIDA, J. P. L. S.; NAKASHIMA, R. T.; OLIVEIRA, A. S.; NEVES-JR, F.; ARRUDA, L. V. R. An augmented reality platform to essay multiple robots with limited sensing and communication skills. **IEEE Access**, 2019.
- ALMEIDA, J. P. L. S.; NAKASHIMA, R. T.; NEVES-JR, F.; ARRUDA, L. V. R. A cooperative path planning based on virtual repulsive pheromone for exploration tasks by swarm robotics. **IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**, 2019.

2 FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

Neste capítulo são apresentados os principais fundamentos de sistemas multirrobo que envolvem o tema desta tese, incluindo os conceitos gerais e as potenciais áreas de aplicação. Em especial, são apresentados e discutidos os principais trabalhos encontrados na literatura que abordam os problemas de planejamento de rota e navegação autônoma, uma vez que são conceitos essenciais para o desenvolvimento deste trabalho de doutorado.

2.1 FUNDAMENTOS DE SISTEMAS MULTIRROBÔ

O termo “sistema multirrobo” (SMR) corresponde a um conjunto de robôs dotados de habilidades individuais a fim de desempenhar um comportamento cooperativo e convergente na realização de uma tarefa (PARKER et al., 2016). A partir desta necessidade de cooperação, as seguintes vantagens emergem da aplicação de múltiplos robôs: *(i)* tarefas de alta complexidade, que sobrecarregam um único robô, podem ser decompostas em tarefas de menor complexidade e são solucionadas por vários robôs; *(ii)* os problemas podem ser resolvidos por meio de ações paralelas e individuais, favorecendo requisitos de otimização do tempo; e *(iii)* múltiplos robôs podem ter estruturas e recursos sensoriais mais limitados que os de um único robô para realizar uma tarefa. Além disso, existem tarefas que exigem naturalmente a utilização de múltiplos agentes (PETITTI et al., 2016).

O comportamento de um SMR como um todo pode ser sintetizado em três principais requisitos, que são: robustez, escalabilidade e flexibilidade (KHAMIS et al., 2015). A robustez é medida pela tolerância dos robôs a eventuais falhas, uma vez que um robô pode cooperar com o grupo e compensar a atuação de outro. A escalabilidade está relacionada com o número de agentes considerados no SMR. Por fim, a flexibilidade diz respeito à capacidade do sistema em atingir um objetivo global por meio de objetivos locais, conforme já mencionado.

Em uma perspectiva geral, as aplicações de múltiplos robôs para a realização de tarefas cooperativas requer que cada robô possua as seguintes habilidades de navegação:

- capacidade sensorial: detectar características do ambiente, medir proximidades, identificar objetos, entre outras ações;
- capacidade de processamento: interpretar e tratar os dados advindos de dispositivos periféricos de comunicação e/ou sensoriamento;
- capacidade de interação: com outros robôs, com o controlador (agente) central e com o ambiente;
- capacidade de realizar ações: se locomover, segurar objetos, perseguir alvos, entre outros.

A partir destas habilidades, o comportamento desejado de um SMR pode ser coordenado por meio de duas abordagens: centralizada ou descentralizada (distribuída) (ZHOU et al., 2018). Na abordagem centralizada, as decisões de cada membro do SMR são deliberadas por um agente central, que pode ser representado por um robô líder ou um dispositivo capaz de se comunicar com todos os robôs do sistema (controlador central ou um sistema de supervisão, por exemplo). Geralmente, esta estratégia é considerada quando uma comunicação irrestrita entre o agente central e todos os outros robôs é garantida durante a navegação. Entretanto, as seguintes implicações devem ser consideradas na medida em que aumenta-se o número de robôs: diminuição da robustez e da flexibilidade; aumento da complexidade computacional do sistema; e atrasos no tempo de resposta do agente central (KISDI; TATNALL, 2011; GAZI; PASSINO, 2011).

Por outro lado, a estratégia distribuída se caracteriza como uma alternativa para estas limitações, principalmente quando há restrições ou fatores inesperados que impedem o êxito desta comunicação centralizada, tais como ruídos no ambiente e limitações na capacidade de comunicação dos robôs. Nesse caso, cada robô é independente e autônomo para deliberar sobre suas próprias ações, baseado em interações locais com o ambiente e/ou com outros robôs por meio de um mecanismo de comunicação e de acordo com as limitações dos seus dispositivos e protocolos de comunicação. Além disso, admite-se que não há a presença de um agente central ou de um sistema geral de supervisão (SABATTINI et al., 2017; YAN et al., 2013).

Em SMR, é essencial que os robôs desempenhem suas habilidades de comunicação para interagir e compartilhar informações entre eles. A literatura da área classifica os mecanismos de comunicação nas formas direta e indireta. Na primeira, a comunicação depende da capacidade de cada robô em compartilhar informações diretamente entre eles e/ou com o agente central por meio de um protocolo específico (SABATTINI et al., 2017). Por outro lado, uma vez que há algum impedimento à comunicação direta, os robôs podem se comunicar indiretamente por meio de alterações no próprio ambiente em que estão inseridos,

tais como produtos químicos lançados pelos robôs conforme se movimentam (CALVO et al., 2012; PURNAMADJAJA; RUSSELL, 2005; ZARZHITSKY et al., 2005). As estratégias de comunicação indireta são inspiradas no comportamento de estigmergia desempenhado em diversas espécies de seres vivos, principalmente os feromônios utilizados pelas formigas (KUYUCU et al., 2015), que servem como base para diversas soluções de coordenação de um enxame de robôs (PALMIERI et al., 2018; RANGO et al., 2018; LIMA; OLIVEIRA, 2017).

Os conceitos apresentados até o momento se referem a um panorama geral sobre o comportamento esperado de um SMR, suas arquiteturas de coordenação (centralizada e distribuída) e seus mecanismos de comunicação (direta e indireta). Entretanto estas definições podem ser ampliadas ou restringidas de acordo com as configurações e objetivos do SMR.

2.2 ENXAME DE ROBÔS

A área de enxame de robôs é um caso particular de sistemas multirrobô. Geralmente, um enxame de robôs é caracterizado pela composição de uma ampla quantidade de robôs homogêneos (a partir de dúzias e geralmente centenas e milhares de robôs) para atingir um objetivo global de forma distribuída, tornando evidente as questões que envolvem, principalmente, a escalabilidade de um SMR (GAZI; PASSINO, 2011).

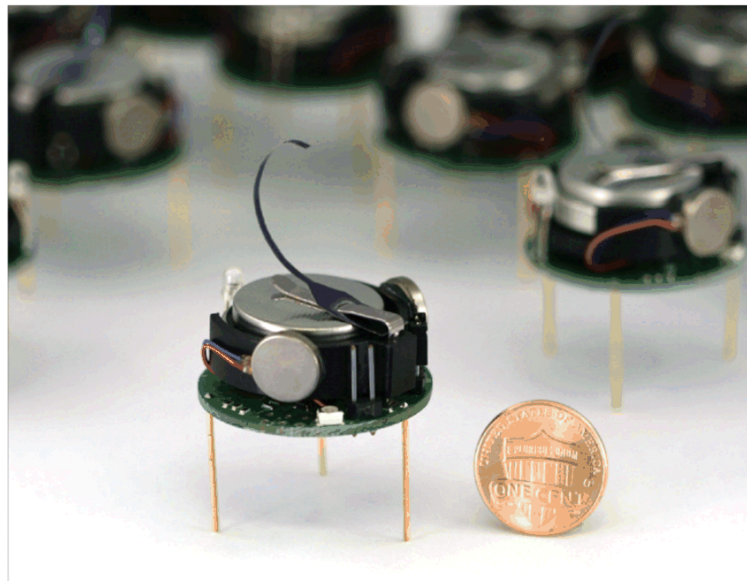
Devido ao número de robôs homogêneos que geralmente compõe um enxame, esses sistemas apresentam um elevado grau de redundância, pois os robôs possuem arquiteturas idênticas e são designados ao cumprimento das mesmas tarefas, contribuindo para o objetivo geral da aplicação.

Em geral, cada robô de um enxame apresenta um comportamento individual, possui arquitetura e dispositivos periféricos simples para o sensoramento/comunicação e desconhece as ações dos outros robôs, exceto as informações sobre proximidade entre eles. Portanto, uma aplicação de enxame de robôs requer naturalmente uma abordagem distribuída para a coordenação dos múltiplos robôs com limitações na comunicação direta e, assim, dependente de um mecanismo de comunicação indireta. Tais características demandam um comportamento similar ao observado em alguns seres vivos que vivem em grupo. O comportamento mencionado é denominado estigmergia, onde uma ação global dos indivíduos emerge de diversas interações locais com o ambiente. Alguns exemplos de estigmergia são as ações das formigas em busca de alimentos e o voo em formação de alguns pássaros a procura de uma região durante a migração (HECKER; MOSES, 2015).

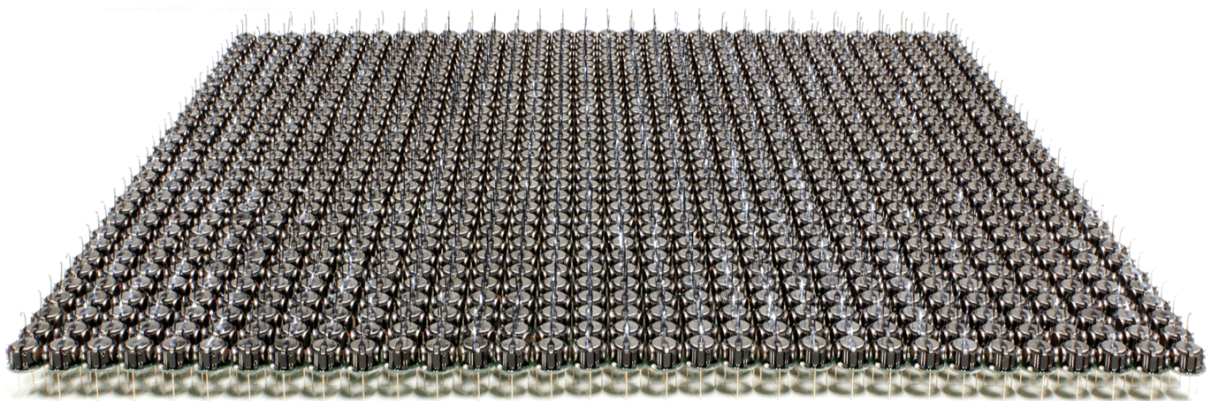
Por exemplo, as aplicações de enxame de robôs nas tarefas de organização espacial são

evidenciadas em diversos trabalhos da área, como a agregação por meio de robôs modulares, onde uma estrutura ou forma complexa pode ser atingida com a conexão ou aproximação dos robôs (RUBENSTEIN et al., 2014).

Na Figura 1 (a) é apresentado o exemplo do robô *Kilobot*, utilizado em diversos experimentos de enxame de robôs. Sua estrutura é composta por uma base para suportar o microcontrolador, uma bateria, um sensor infravermelho e um conector para carregamento. Sua locomoção ocorre por meio de vibrações em suas pernas de apoio, comandadas por um controlador central. Na Figura 1 (b) é apresentado um time composto por mil *Kilobots*.



(a)



(b)

Figura 1: Robôs *Kilobot*.

Fonte: (RUBENSTEIN et al., 2014).

2.3 NAVEGAÇÃO AUTÔNOMA E PLANEJAMENTO DE ROTA

Assim como para um único robô, pode-se dizer que o planejamento de rota de um SMR é um tema importante no desenvolvimento de um sistema de navegação autônoma (SIEGWART et al., 2011). Este planejamento consiste em calcular uma sequência de movimentos a ser realizada pelos robôs móveis, a fim de alcançar uma posição desejada a partir de um ponto inicial, de acordo com as restrições de navegação impostas, tais como o desvio de obstáculos, a passagem por alvos, as interações com outros robôs, entre outros fatores (TANG et al., 2016).

Assim, o planejamento de rota de múltiplos robôs móveis deve contemplar restrições que torna complexa sua execução, tais como o desvio de obstáculos desconhecidos, escolher caminhos com os recursos necessários para completar uma tarefa determinada, tratar situações de ambientes dinâmicos e problemas afins (LADOSZ et al., 2017; ZHOU et al., 2017). Portanto, técnicas de inteligência computacional, em especial as heurísticas evolucionárias, são amplamente utilizadas, conforme compilado por Mac et al. (2016).

Duas abordagens de planejamento de rota podem ser encontradas na literatura, que são: planejamento *off-line* e *on-line*. Na abordagem *off-line*, as expectativas do planejamento podem não ser atingidas se o ambiente de navegação apresentar características que se alteram dinamicamente, pois o conhecimento *a priori* do ambiente e/ou histórico de dados passados, advindos de sensoriamento, são as principais informações a serem consideradas no cálculo das rotas dos diversos robôs que compõem o SMR. Assim, uma vez que o ambiente de navegação apresenta comportamento dinâmico, métodos *on-line* são indicados, pois sua execução depende de dados locais do ambiente, que são adquiridos por meio das interações realizadas em tempo de navegação (MAC et al., 2016), e assim cada robô planeja sua rota de forma independente ou após uma interpretação dos dados realizada por um agente central.

Especialmente sobre os métodos *on-line* de planejamento de rota, em (SCHLOTFELDT et al., 2018) é discutida a influência da capacidade de comunicação e sensoriamento dos robôs na etapa de planejamento de rotas, uma vez que esta tarefa é diretamente dependente das informações locais do ambiente que os robôs são capazes de obter. Dessa forma, admite-se que há uma relação de compromisso entre a eficiência de um planejamento *on-line* e a capacidade de obtenção de informações do ambiente, por meio de uma tarefa de exploração, por exemplo.

A partir de um planejamento de rota, os robôs devem possuir um sistema de navegação capaz de converter as rotas calculadas em ações de controle que, efetivamente, são aplicadas aos atuadores responsáveis por sua locomoção. Estratégias de controle inteligente, em especial

os sistemas *fuzzy*, são opções válidas para promover a locomoção de robôs, pois são capazes de lidar com incertezas inerentes a sua dinâmica de movimentação, como é abordado em (HONG et al., 2016), (ABADI; KHOOBAN, 2015) e (MEHRJERDI et al., 2011).

2.4 APLICAÇÕES GERAIS

As principais aplicações dos SMR e dos enxames de robôs frequentemente abordam a solução dos seguintes problemas: organização espacial (controle de formação e agregação), navegação autônoma (exploração, movimento coordenado, manipulação e transporte de objetos) e tomadas de decisão distribuídas (BAYINDIR, 2016; BRAMBILLA et al., 2013). Essa visão geral dos SMR é sintetizada na Figura 2.

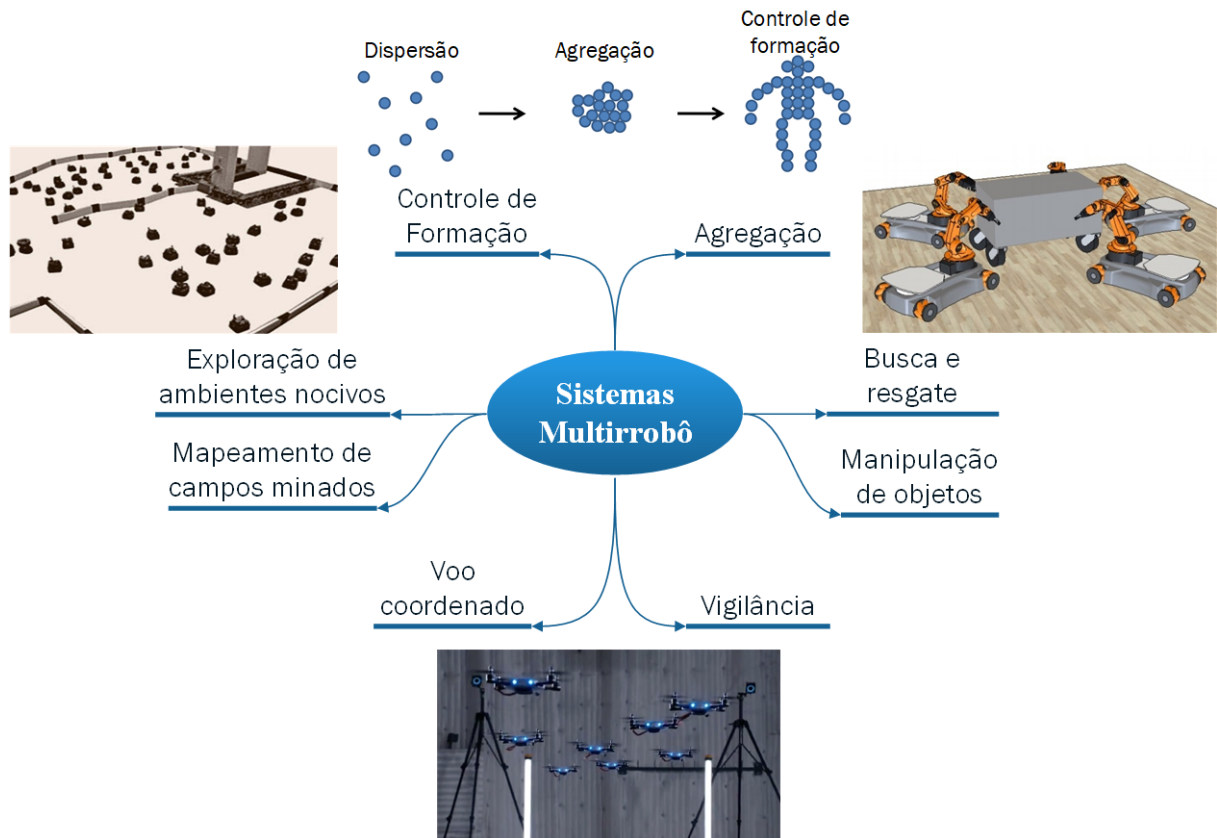


Figura 2: Uma visão geral dos sistemas multirrobo.

Fonte: Autoria própria.

A tarefa de agregação consiste no agrupamento dos robôs móveis em um ambiente comum, com o objetivo de reproduzir o comportamento básico entre diversos seres da mesma espécie (ARVIN et al., 2016). Para a realização desta tarefa, os algoritmos são sintetizados em agregação livre e mediada pelo ambiente. No primeiro caso, os agentes não possuem uma

região específica para se agruparem, enquanto que no segundo, uma região é definida para a realização da tarefa.

Nas tarefas de controle de formação, ou formação de padrões, objetiva-se coordenar os robôs para que fiquem a uma determinada distância uns dos outros, com o intuito de configurar e manter uma forma geométrica desejada. Esta tarefa pode ser estruturada nas seguintes ações/estágios: aquisição da forma desejada (ou estabilização), manutenção da forma, e reconfiguração da forma. Este comportamento pode ser alcançado por meio de diversas estratégias encontradas em trabalhos recentes, como por exemplo a utilização de controle ótimo, abordado em (FUJIMORI et al., 2018), onde uma estratégia baseada em *Linear Quadratic Regulator* (LQR) é utilizada para promover o controle de formação entre *drones* e outros robôs terrestres. Uma outra abordagem é apresentada em (WANG et al., 2017), onde um controle adaptativo é proposto para que múltiplos robôs alcancem formações que se alteram durante os experimentos. Em (YU et al., 2019), uma estratégia baseada em Redes Neurais Artificiais (RNA) é utilizada para levar múltiplos robôs a uma determinada formação considerando o desvio de obstáculos, se necessário. Além disso, diversos algoritmos para o controle de formação são inspirados em comportamentos registrados em grupos de formigas, cardumes e aves e, portanto, tais algoritmos são ditos bioinspirados (BRAMBILLA et al., 2013).

Na exploração cooperativa por um SMR, adotam-se métodos para que os robôs consigam rastrear pontos de interesse em um ambiente e, ao mesmo tempo, evitar regiões que não sejam de interesse do objetivo a ser cumprido. Em (ZARZHITSKY et al., 2005), por exemplo, múltiplos robôs devem explorar o ambiente à procura de uma fonte emissora de produto químico e, como estratégia de exploração, os autores utilizam o método *casting*, que consiste basicamente em movimentos espirais ou *zig-zag* que otimizam a cobertura do ambiente. Em (COLARES; CHAIMOWICZ, 2016, 2015), um SMR combina estratégias de exploração com técnicas de mapeamento para ambientes desconhecidos, em que os robôs devem evitar áreas já exploradas por outros robôs enquanto obtém informações do ambiente para representá-lo em um mapa. Um outro exemplo também é encontrado em (SPURNY et al.,), onde múltiplos *Unmanned Aerial Vehicles* (UAVs) devem cumprir o objetivo de uma tarefa de “caça ao tesouro”, onde objetos espalhados por um ambiente devem ser encontrados, coletados e depositados em um local pré-definido. Os autores destacam que esta tarefa de exploração envolve diversas outras sub-tarefas que devem ser realizadas de forma cooperativa, tais como um planejamento de trajetória para a exploração do ambiente, detecção de objetos (alvos a serem coletados e obstáculos que devem ser desviados), autolocalização, coordenação distribuída, seguimento de uma trajetória calculada e manipulação de objetos (coleta e depósito). Há também as estratégias de exploração bioinspiradas em feromônios artificiais, como é tratado

em (PALMIERI et al., 2018; RANGO et al., 2018), onde múltiplos robôs devem explorar um ambiente para encontrar e desarmar minas terrestres de localização desconhecida.

As soluções para as tarefas que exigem movimentos coordenados, que vão além do controle de formação, possuem inspirações no comportamento social de seres que se movimentam de forma conjunta durante uma exploração, como em algumas espécies de pássaros. Geralmente, o movimento coordenado é regido pela regra do vizinho mais próximo (JADBABAIE et al., 2003), que possui como base três princípios, que são: atração, necessária para que um agente permaneça agrupado aos demais membros do SMR; repulsão, que é responsável por evitar colisões entre os robôs; e alinhamento, para que os robôs naveguem na mesma direção e velocidade (EVERSHAM; RUIZ, 2011).

Nas tarefas de transporte cooperativo, objetiva-se alcançar um comportamento no SMR que permita o transporte e/ou a manipulação de um objeto que geralmente possui uma dimensão maior que a dos robôs envolvidos e/ou requer a utilização de mais de um agente para a realização da tarefa (RUIZ et al., 2016).

Independente do problema abordado, os robôs precisam tomar decisões durante o cumprimento da tarefa. Para isso, heurísticas cooperativas devem ser consideradas para que uma decisão unânime seja tomada por um SMR (SABATTINI et al., 2017; OLFATI-SABER et al., 2007). A unanimidade pode ser atingida por meio de diferentes alternativas avaliadas por cada robô, que geralmente resulta em uma determinada recompensa. É importante salientar que é indispensável o estabelecimento de meios de comunicação entre os robôs e/ou agente central para o êxito deste comportamento cooperativo.

Os problemas de navegação autônoma e planejamento de rota de um SMR são os temas investigados nesta tese e são tratados com destaque ao longo do trabalho. São considerados os conceitos de coordenação centralizada e autônoma, comunicação direta e indireta, e planejamentos *off-line* e *on-line*.

2.5 ASPECTOS DE IMPLEMENTAÇÃO DE SISTEMAS MULTIROBÔ

As implementações de estratégias aplicadas a SMRs, simuladas ou não, podem ser realizadas a partir de diversas considerações, principalmente as relacionadas à representação do ambiente e dos robôs móveis.

Em simulações, o ambiente em que os múltiplos robôs são inseridos pode ser representado de forma contínua ou discreta. O simulador V-REP (ROBOTICS, 2017), por exemplo, permite que robôs sejam inseridos e simulados em ambientes que evolui

continuamente de acordo com as leis dinâmicas consideradas pelo usuário. Esta representação, por meio de um simulador alternativo, também está presente nos trabalhos (PURNAMADJAJA; RUSSELL, 2005) e (ZARZHITSKY et al., 2005). Por outro lado, uma representação discreta significa que os estados do ambiente são alterados instantaneamente por partes de um estado contínuo. Alguns exemplos de representações discretas podem ser encontrados em (PALMIERI et al., 2018; RANGO et al., 2018), nos quais o ambiente de simulação é decomposto em células.

Independente da representação considerada para o ambiente, um robô pode ser modelado a partir de diversas formas. O modelo de acionamento diferencial (SIEGWART et al., 2011) é um das topologias mecânicas frequentemente considerada, principalmente quando os problemas de navegação (derrapagem das rodas, assimetrias em suas dimensões, odometria, entre outros) também são analisados. Por outro lado, o modelo de um ponto omnidirecional, capaz de se mover instantaneamente à qualquer ponto ao seu redor (de acordo com um limite estabelecido), é um modelo alternativo e simplificado de um robô móvel para promover uma análise restrita aos problemas de coordenação (camada de planejamento e controle) dos múltiplos robôs, como são considerados em (LIMA; OLIVEIRA, 2017) e (PALMIERI et al., 2018), por exemplo.

De acordo com a implementação pretendida, um dos fatores que limita os experimentos com robôs reais é o elevado custo ou a inacessibilidade do pesquisador aos dispositivos periféricos de sensoriamento e comunicação requeridos pela aplicação, além da aquisição dos próprios robôs. Nesses casos, os sistemas baseados em realidade aumentada são amplamente utilizados para inserir uma camada virtual ao ambiente experimental real, em que os dispositivos para os propósitos mencionados podem ser virtualmente implementados e, portanto, pode-se testar seu funcionamento básico e confirmar sua validade no contexto do experimento a ser realizado, antes do processo de prototipagem ou aquisição dos mesmos (MILLARD et al., 2018). Tal arquitetura experimental adota os conceitos de realidade mista para robótica, onde características virtuais complementam configurações de ambientes reais (HÖNIG et al., 2015). Em (REINA et al., 2017), por exemplo, uma plataforma experimental baseada em realidade aumentada é utilizada para conferir habilidades virtuais a um enxame composto por *Kilobots*, enquanto em (REINA et al., 2015) focam-se nas habilidades virtuais de sensoriamento para múltiplos robôs *e-puck*.

Além disso, as implementações simuladas e reais de sistemas robóticos também contam frequentemente com a presença do *framework* ROS (QUIGLEY et al., 2009), que consiste em um sistema de código aberto e flexível para a integração de diferentes dispositivos robóticos em uma arquitetura padronizada. Para isso, o ROS possui uma coleção de ferramentas

e bibliotecas que visam simplificar este processo de desenvolvimento. De uma forma geral, o ROS é intermediário entre o robô e os algoritmos de planejamento e controle, cuja comunicação ocorre por meio de tópicos e nós, baseados no padrão *publisher-subscriber*.

Com base nas informações apresentadas, os principais trabalhos relacionados com o tema em análise desta tese são apresentados a seguir, juntamente com suas principais características de modelos e representação dos sistemas de múltiplos robôs considerados.

2.6 TRABALHOS RELACIONADOS

Os diversos trabalhos elencados como referências nesta tese evidenciam a navegação autônoma de múltiplos robôs, juntamente com problemas de planejamento de rota e tomadas de decisão cooperativa. Dessas citações, as principais são descritas nesta seção, com os devidos destaques que as relacionam com a proposta deste trabalho.

Uma estratégia híbrida, composta por um sistema de otimização por enxame de partículas (*Particle Swarm Optimization* (PSO)) com evolução diferencial (*Differential Evolution* (DE)), é usado por Tang et al. (2016) para um planejamento *off-line* de rotas. Uma outra aplicação do algoritmo PSO para estes propósitos é apresentada por Das et al. (2016), cujo objetivo é minimizar o caminho máximo e o tempo gasto entre pontos de partida e destino de múltiplos robôs. O trabalho apresentado por Sundar e Rathinam (2016) aborda uma situação em que um grupo de veículos aéreos e terrestres heterogêneos deve visitar um conjunto de alvos e retornar para suas posições iniciais, tratadas como depósitos. Assim, o objetivo desta aplicação é encontrar um caminho ótimo para cada veículo por meio do algoritmo “*branch-and-cut*”, com o objetivo de minimizar a soma das distâncias a serem percorridas. De forma semelhante, um algoritmo genético com operadores alternativos de cruzamento é proposto por Han et al. (2017) para calcular rotas de múltiplos AGVs, de tal maneira que o grupo deve iniciar em uma posição determinada, passar por todos os pontos de trabalho contidos em um ambiente bidimensional, e retornar ao ponto inicial, sem a presença de obstáculos durante a navegação. O objetivo da otimização é minimizar a soma das distâncias a serem percorridas e também a distância máxima a ser desempenhada por um AGV.

Algumas estratégias híbridas para ajustes de rotas já calculadas também podem ser encontradas na literatura. Como exemplo, há registros de aplicações envolvendo sistemas *fuzzy* com PSO, ACO e outros métodos similares (HONG et al., 2016; CAI; YANG, 2016; ABADI; KHOOBAN, 2015; CASTILLO et al., 2015), assim como a combinação de algoritmos PSO e DE (TANG et al., 2016).

Em especial, a estratégia híbrida de planejamento de rotas, como proposta nesta tese, herda as características dos problemas apresentados por Sundar e Rathinam (2016) e Han et al. (2017). No entanto, tais autores não abordam as situações de possíveis colisões com obstáculos estáticos e dinâmicos conforme os robôs executam a navegação planejada, ou seja, situações que limitam a factibilidade das rotas calculadas em ambientes dinâmicos. Neste contexto, a proposta desta tese é atribuir melhorias nas rotas calculadas pelo agente central e também conferir aos robôs as habilidades locais necessárias para lidar com situações dinâmicas e inesperadas, de forma a garantir a retomada da rota calculada anteriormente.

Para abordagens de planejamento *on-line*, destacam-se trabalhos baseados em feromônios artificiais para promover o comportamento de estigmergia em um SMR, com o objetivo de alcançar um ou mais pontos de interesse e desviar de obstáculos de posição desconhecida. Nos trabalhos de Zarzhitsky et al. (2005) e Purnamadjaja e Russell (2005), os robôs móveis devem procurar e alcançar uma fonte que emite produto químico no ambiente, cuja posição é desconhecida. Especificamente em (PURNAMADJAJA; RUSSELL, 2005), a fonte emissora representa um robô danificado e os outros membros do SMR devem desempenhar uma missão de resgate. Esta ação cooperativa é inspirada no comportamento de abelhas, denominado comportamento necrofórico. Rango et al. (2018) e Palmieri et al. (2018) abordam uma aplicação em que o SMR deve procurar e desarmar minas terrestres distribuídas em um ambiente, cujas soluções são dadas essencialmente pelo conceito de compartilhamento de informações entre os robôs. Rango et al. (2018) descrevem e validam um protocolo *wireless* distribuído para a coordenação dos agentes durante a tarefa cooperativa para desarmar as minas terrestres, enquanto Palmieri et al. (2018) apresentam um mecanismo de tomada de decisão para garantir um balanceamento uniforme na distribuição das tarefas locais entre os robôs.

Tais problemas de exploração serviram de inspiração para as tomadas de decisão cooperativas que compõem planejamento de rotas *on-line* desenvolvido nesta tese. Neste caso, a estratégia de comunicação indireta por feromônio artificial, juntamente com as heurísticas cooperativas propostas, visam aprimorar o resultado das métricas relacionadas a eficácia da exploração apresentadas pelos autores. Além disso, a utilização do ambiente de simulação V-REP, o *framework* ROS e a arquitetura experimental são fatores diferenciais que atribuem aspectos mais realistas ao sistema. Uma comparação com os trabalhos citados e os principais aspectos apresentados nesta tese são elencados na Tabela 1.

Tabela 1: Comparação com trabalhos relacionados.

Principais características	Zarzhitsky et al. (2005)	Purnamadajaja e Russell (2005)	Rango et al. (2018) e Palmieri et al. (2018)	Nesta tese
Abordagem distribuída	Sim.	Sim.	Sim.	Sim.
Representação do ambiente	Contínuo.	Contínuo.	Decomposto em células).	V-REP.
Controle dos robôs	<i>Artificial physics (AP)</i> para o controle de formação.	Não informado.	Movimentos discretos para uma célula vizinha.	Controle <i>fuzzy</i> .
Comunicação direta	Sim.	Não.	Sim. Informação compartilhada.	Sim. Informação compartilhada.
Comunicação indireta	Feromônios atrativos.	Feromônios atrativos.	Feromônios repelentes.	Feromônios repelentes.
Mecanismo de exploração	Estratégia <i>casting</i> (<i>zig-zag</i> ou espiral).	Movimentos aleatórios no sentido anti-horário.	Inspirado em <i>Ant Systems</i> (AS).	Inspirado em vizinhança e <i>Ant Systems</i> (AS).
Balanceamento da carga de trabalho	Não requerido para a tarefa.	Não considerado.	Sim.	Sim. Tomadas de decisão coletiva.
Implementação	Simulação (ambiente Java customizado).	Real e simulação (Java customizado).	Simulação (ambiente Java customizado).	Real e simulação: V-REP, Matlab e ROS.

3 DESCRIÇÃO DO AMBIENTE E DO SISTEMA MULTIRROBÔ

Neste capítulo são estabelecidas as principais características dos problemas de planejamento de rota abordados nesta tese. Para isso, um ambiente bidimensional (2D) é idealizado e utilizado como um plano de referência, tanto para os desenvolvimentos teóricos quanto para a validação por simulação e com robôs reais. Este ambiente é capaz de suportar a inserção de robôs móveis, pontos de interesse (alvos), e obstáculos estáticos. Uma representação simulada de tal ambiente, por meio do *software* V-REP, é apresentada na Figura 3, onde há 5 robôs móveis, 20 alvos (círculos vermelhos) e 12 obstáculos estáticos (cilindros cinzas).

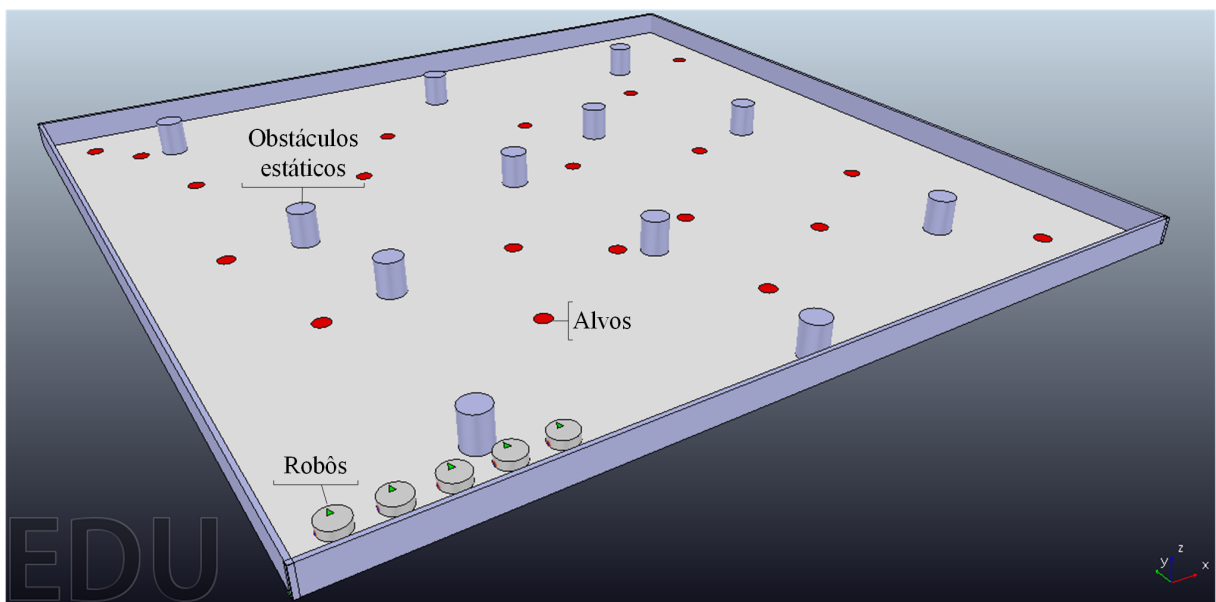


Figura 3: Representação simulada de um ambiente com robôs móveis, alvos e obstáculos estáticos.

Fonte: Autoria própria.

Os alvos presentes no ambiente são pontos obrigatórios de passagem, que podem representar estações de trabalho, nós de comunicação, marcações de áreas importantes, pontos de reabastecimento, ou objetos a serem coletados em aplicações reais, semelhante ao discutido por Han et al. (2017) e Sundar e Rathinam (2016). Neste trabalho, o principal objetivo a ser

cumprido pelos múltiplos robôs no ambiente exemplificado é, a partir de uma posição inicial, visitar todos os alvos de uma forma otimizada, desviar dos obstáculos e dirigir até um ponto final previamente conhecido para concluir a navegação. A realização deste objetivo é baseada em princípios de trabalho cooperativo em um SMR e visa minimizar as rotas percorridas por cada robô e também distribuir de forma igualitária a carga de trabalho entre eles, ou seja, o número de alvos visitados e a distância a ser percorrida por cada robô deve ser equilibrada. Tais requisitos constituem um problema de otimização combinatória do tipo caminho mínimo, sendo portanto NP-completo (PARKER, 2009).

Idealmente, o cálculo de rotas ótimas para o caso de exemplo (Figura 3) depende de um conhecimento prévio e completo acerca do ambiente, tais como a dimensão do espaço disponível para a navegação e a posição de todos os elementos do cenário (robôs, alvos e obstáculos), e do conhecimento dos robôs usados para cumprir esta tarefa, tais como suas dimensões e capacidade de locomoção. A partir disso, as rotas otimizadas devem assegurar que todos os alvos sejam visitados por um robô e também considerar o desvio de obstáculos, para que a integridade física dos robôs seja preservada durante a navegação.

Entretanto, nem sempre é possível garantir que as informações disponíveis sejam suficientes para que as rotas ótimas sejam calculadas previamente por um planejador *off-line*, ou seja, antes de iniciar a navegação dos robôs. Além disso, existem elementos dinâmicos no ambiente, tais como a navegação do próprio SMR, que alteram o ambiente em tempo de navegação. Dessa forma, um sistema de planejamento requer a inclusão de estratégias *on-line* e/ou híbridas de planejamento de rota, que são desempenhadas em tempo real e de forma autônoma pelos próprios robôs, tais como a detecção de obstáculos, comunicação direta e indireta, entre outras.

Com base nestas informações, diferentes ambientes com características semelhantes ao exemplificado na Figura 3 (presença de múltiplos robôs, alvos e obstáculos estáticos) são considerados neste trabalho para validar as duas estratégias de planejamento de rota propostas para a navegação autônoma de SMR. Para ambas as propostas, o conhecimento acerca da posição e existência dos elementos inseridos no ambiente (robôs, alvos e/ou obstáculos) é limitado.

Para o desenvolvimento da primeira estratégia, as informações sobre o ambiente corresponde apenas às posições dos alvos (os obstáculos são desconhecidos). Nesse caso, uma estratégia híbrida de planejamento de rotas é proposta para planejar os caminhos a serem percorridos pelo SMR de forma *off-line*, por meio de um algoritmo genético, enquanto cada robô possui um módulo de planejamento *on-line*, responsável por recalcular sua rota devido

aos desvios ocasionados por obstáculos e/ou incertezas acerca de sua localização no ambiente, devido a erros de odometria.

Na segunda estratégia, todos os elementos estão em posições desconhecidas (alvos e obstáculos) e os robôs devem desempenhar uma tarefa de exploração para encontrar e visitar todos os alvos, enquanto desviam de obstáculos. Para isso, uma estratégia *on-line* e bioinspirada em feromônios artificiais é proposta para suportar a exploração do SMR e encontrar todos os alvos de forma otimizada.

De acordo com a taxonomia acerca dos problemas de alocação de tarefas em SMR, descrita em (GERKEY; MATARIĆ, 2004), o problema abordado no primeiro cenário se enquadra nos conceitos de *Single-task robots* (ST), *Single-robot tasks* (SR) e *Time-extended assignment* (TA) e, portanto representando o problema ST-SR-TA, onde cada robô é capaz de executar no máximo uma tarefa por vez (ST), cada tarefa requer exatamente um único robô para ser realizada (SR) e, a partir do conhecimento de todas as tarefas a serem realizadas (visita aos alvos previamente conhecidos), é possível planejar a alocação de todas elas aos robôs (TA). Por outro lado, o problema do segundo cenário se enquadra em uma configuração semelhante (ST-SR), exceto pelo fato de que a informação das tarefas a serem realizadas (alvos identificados localmente pelos robôs) permite apenas alocações instantâneas, ou seja, um planejamento considerando apenas as informações locais dos robôs (*Instantaneous assignment* – IA) e, portanto, é classificado como ST-SR-IA.

Pra uma melhor compreensão dos conceitos abordados ao longo da descrição das estratégias, apresentadas em capítulos posteriores, as seguintes afirmações devem ser consideradas:

- o cenário de navegação dos robôs possui um ponto de referência fixo (0,0) e todos os elementos inseridos no ambiente (robôs, alvos e obstáculos) tem sua posição estabelecida a partir desta coordenada;
- todos os robôs conhecem, pelo menos, suas posições inicial e final no ambiente no início da navegação, que são definidas a partir do ponto de referência fixo (0,0);
- assume-se que a comunicação entre robôs e as interações com outros elementos (identificação de obstáculos, visita aos alvos, etc) são instantâneas, ou seja, não são considerados os tempos de execução de protocolos de comunicação, taxas de transmissão, entre outras características;
- uma rota é definida como uma linha reta que interliga as posições atual do robô e do objetivo a ser alcançado, ou seja, uma linha reta entre dois pontos (x,y) .

Além disso, cada robô do SMR deve possuir um sistema básico de navegação, composto por um conjunto de habilidades que viabilizam a implementação das estratégias de planejamento de rota mencionadas. Para uma melhor compreensão, as principais características dos robôs e das habilidades do sistema básico de navegação são apresentadas na seção 3.1 e a configuração experimental, utilizada para a implementação destes conceitos, é descrita na seção 3.2.

3.1 DESCRIÇÃO DOS ROBÔS E DO SISTEMA BÁSICO DE NAVEGAÇÃO

O SMR considerado nesta tese é composto por robôs móveis homogêneos de estrutura simples, capazes de incorporar sensores de proximidade e outros dispositivos periféricos para detecção e comunicação. Os principais detalhes de sua estrutura são apresentados na Figura 4, onde os eixos vermelho (X), verde (Y) e azul (Z) são as referências para o plano 3D. A estrutura possui uma forma cilíndrica com uma altura de $0,1\text{ m}$, com duas rodas independentes, R_E e R_D , para a locomoção (acionamento diferencial) e outras duas outras rodas apenas para suporte, denominadas RS_1 e RS_2 .

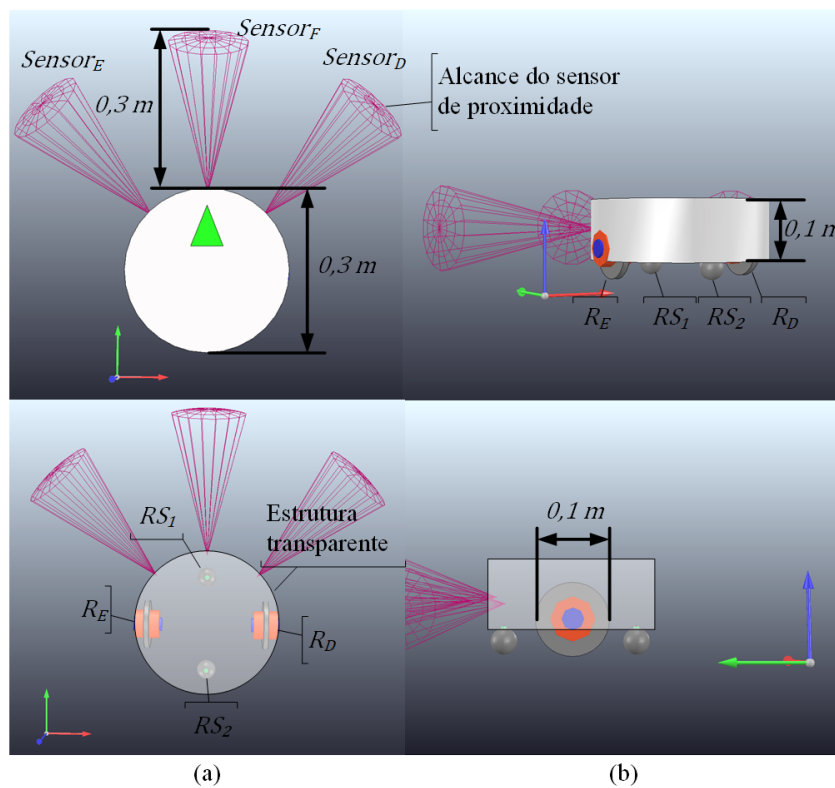


Figura 4: Especificações da estrutura dos robôs móveis: (a) vistas superiores e (b) vistas laterais.

Fonte: Autoria própria.

Os robôs apresentados são capazes de se movimentar em ambientes como o apresentado na Figura 3 e desempenhar uma série de habilidades de detecção, comunicação e locomoção, que são componentes do sistema de navegação definido nesta tese. Cada ação é executada individualmente por cada membro do SMR, de acordo com as suas próprias interações locais e independente da ação dos outros robôs (abordagem distribuída). Estas habilidades são necessárias para estabelecer um comportamento autônomo dos robôs frente aos problemas de planejamento de rota apresentados anteriormente.

Inicialmente, os robôs devem possuir requisitos que os permitam obter informações para superar as dificuldades advindas das características dinâmicas do ambiente e para promover o compartilhamento de informações entre robôs. Estes requisitos são:

- (i) detecção: habilidade de detectar obstáculos e calcular a distância entre o sensor e os elementos detectados;
- (ii) comunicação: promove interações locais com alvos, outros membros do SMR e com o ambiente, por meio de um protocolo de reconhecimento;
- (iii) locomoção: confere ao robô um sistema de controle para seguir uma rota pré-determinada, desviar de obstáculos e ponderar a atuação destas duas manobras. Tal habilidade é desempenhada por um sistema *fuzzy* de navegação;
- (iv) autolocalização: o robô é capaz de estimar a sua posição no ambiente por meio de dados obtidos de sua odometria.

Uma descrição mais detalhada de cada um dos requisitos é apresentada nas seções a seguir.

3.1.1 HABILIDADE DE DETECÇÃO

Cada robô móvel possui um sistema de detecção composto por três sensores de proximidade, direcionados para detecções de obstáculos à esquerda (*Sensor_E*), frente (*Sensor_F*) e à direita (*Sensor_D*) de sua estrutura, com uma distância máxima de medição d_{max} , conforme é ilustrado na Figura 5. As iminências de colisão são caracterizadas de acordo com uma intensidade inversamente proporcional à proximidade medida, no intervalo de 0 a 1, ou seja, a intensidade 0 significa que não há elementos detectados na região de detecção do sensor, enquanto o valor 1 caracteriza um objeto muito próximo da estrutura física do robô. O princípio de funcionamento de detecção destes dispositivos é similar ao dos sensores exteroceptivos ativos

para medir proximidades, tais como os sensores ultrassônicos ou a laser (SIEGWART et al., 2011).

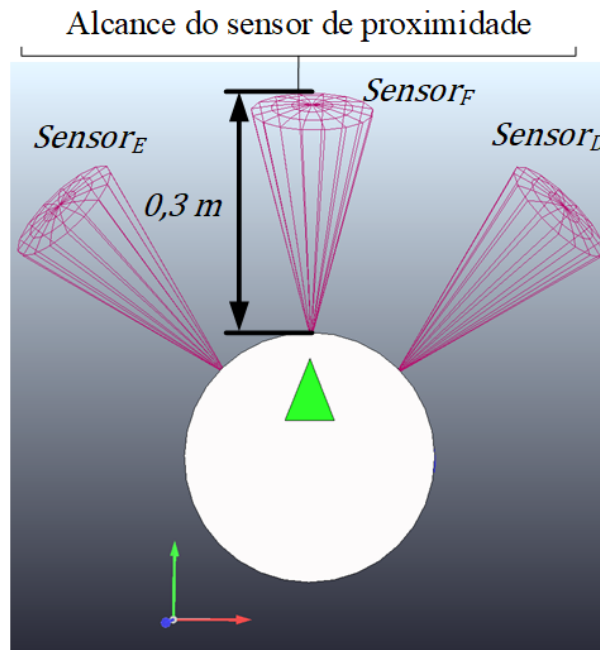


Figura 5: Sensores de proximidade dos robôs móveis.

Fonte: Autoria própria.

3.1.2 HABILIDADE DE COMUNICAÇÃO DIRETA

Em complemento aos sensores de proximidade, admite-se que os robôs são dotados de dispositivos periféricos adicionais de comunicação, que os permitem identificar, reconhecer e se comunicar com outros robôs e alvos (caracterizados como uma estação de trabalho, ou outro tipo de ponto de interação), desde que estejam no interior de uma região delimitada. Mais especificamente, as interações desempenhadas por estes dispositivos ocorrem de acordo com a distância entre os elementos envolvidos. Para isso, uma distância máxima é considerada como o limite desta região, medida a partir do centro de cada robô e formando uma área circular ao seu redor, de raio R_{Det} .

Um exemplo destas interações locais é ilustrado na Figura 6, onde há 3 robôs móveis (R1, R2 e R3) e 4 alvos (A1, A2, A3 e A4). Nesse caso, A1 é detectado por R1, A2 não é detectado, A3 é detectado por R2 e R3, e A4 é detectado apenas por R3. As setas de duas vias representam a troca de informações entre robôs, enquanto a de apenas uma via representa a emissão de dados pelos alvos.

O compartilhamento de informações entre o robô e outros elementos ocorre por meio

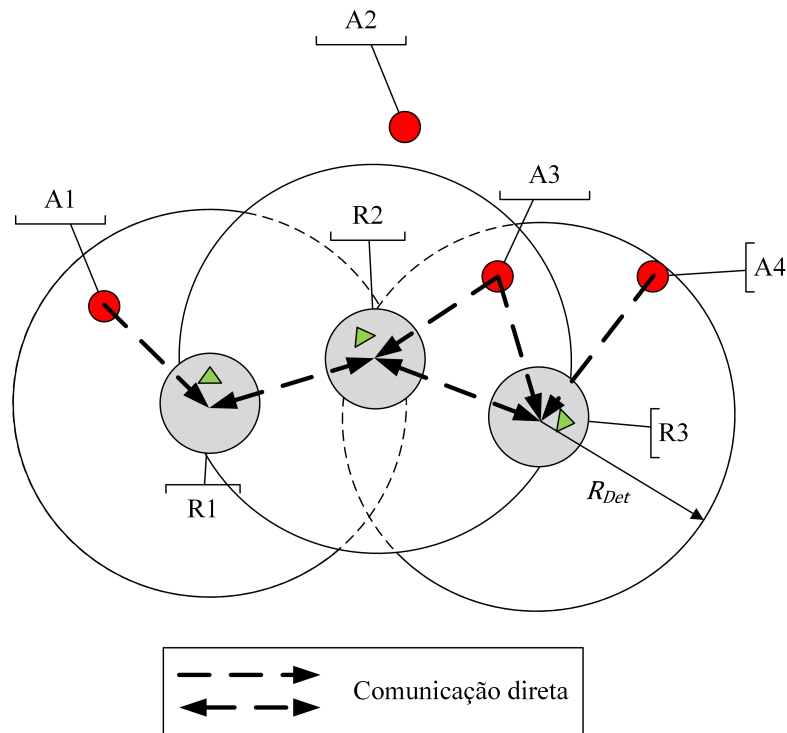


Figura 6: Comunicação direta e compartilhamento de informações.

Fonte: Autoria própria.

de um protocolo de comunicação direta. Os robôs envolvidos são capazes de compartilhar informações simples acerca de sua navegação, tais como a sua identificação, a posição de seu próximo objetivo, e sua posição estimada. Por outro lado, os alvos são inspirados em estações de trabalho, capazes de emitir sua posição exata no ambiente e a sua identificação.

3.1.3 HABILIDADE DE COMUNICAÇÃO INDIRETA

Além da comunicação direta, cada robô também é capaz de se comunicar indiretamente com outros membros do SMR. Esta forma de comunicação ocorre por meio de alterações no ambiente de navegação, que são detectadas e interpretadas pelos robôs. Geralmente, a comunicação indireta é utilizada para promover um comportamento de estigmergia no SMR, bioinspirado em diversas espécies de seres vivos que vivem em grupo (CALVO et al., 2012).

No caso desta tese, os feromônios artificiais são baseados nos conceitos empregados em algoritmos de otimização *Ant Colony Optimization* (ACO), que são inspirados no comportamento de forrageamento desempenhado por diversas espécies de formigas (DORIGO et al., 2006). Em um algoritmo ACO, um grupo de formigas artificiais encontra a solução de um determinado problema de otimização por meio da troca de informações que contém uma medida

de qualidade das diversas soluções possíveis. As formigas envolvidas neste sistema lançam feromônios no ambiente, a fim de que outros membros da colônia detectem e sigam estes sinais, sem que ocorra, de fato, um contato direto entre estes agentes para se comunicarem. Como resultado, tem-se a realização de um comportamento baseado em estigmergia, que consiste em uma forma indireta de comunicação mediada pelo ambiente.

Em analogia aos conceitos utilizados nos algoritmos ACO, os robôs considerados nesta tese representam as formigas artificiais destes sistemas e possuem mecanismos para liberar feromônios artificiais no ambiente, de forma a marcar uma trilha temporária de sua rota e indicar indiretamente sua presença à outro robô. A detecção destes sinais de feromônio é realizada de forma semelhante à detecção de outros robôs e alvos do ambiente (Figura 6), ou seja, quando estão no interior da região de detecção do robô.

Assim como na natureza, os feromônios artificiais são substâncias químicas e possuem uma característica evanescente, de dispersão e evaporação no ambiente (DORIGO et al., 2006). Exemplos deste feromônio em SMR são dados em (PURNAMADAJA; RUSSELL, 2005) e (ZARZHITSKY et al., 2005), onde os robôs lançam produtos químicos no ambiente que exalam espuma e fumaça, respectivamente.

A função que descreve o depósito e a evaporação de um sinal de feromônio lançado por um robô considerado neste trabalho, em um determinado ponto (x, y) do ambiente e no instante k , é dada por $\sigma_{x,y}(k)$. Esta variável é localmente atualizada por todos os robôs que compartilham o mesmo ambiente, de acordo com a Equação (1), onde $\rho \in [0, 1]$ é a taxa de evaporação e $\hat{\sigma}_{x,y}$ é o sinal de feromônio lançado pelo robô no instante k .

$$\sigma_{x,y}(k) = (1 - \rho) \cdot \sigma_{x,y}(k - 1) + \hat{\sigma}_{x,y}(k) \quad (1)$$

Os feromônios artificiais são utilizados nesta tese para promover um comportamento de exploração de ambientes no SMR considerado e a descrição detalhada da implementação e utilização deste mecanismo de comunicação indireta é apresentada conforme as estratégias de planejamento de rota são propostas nos capítulos seguintes.

3.1.4 SISTEMA FUZZY DE CONTROLE DE NAVEGAÇÃO

Dada uma rota a ser executada por um robô, definida por um vetor que interliga as posições (x, y) atual do robô e do seu próximo alvo (objetivo), um sistema *fuzzy* é desenvolvido para promover o seu movimento em direção ao seu objetivo, enquanto evita colisão com obstáculos estáticos e dinâmicos. Este sistema é composto por três módulos, dois controladores

e um interpolador *fuzzy* para tomadas de decisão, que atua em função da saída dos dois controladores. Uma visão geral deste sistema é apresentada na Figura 7.

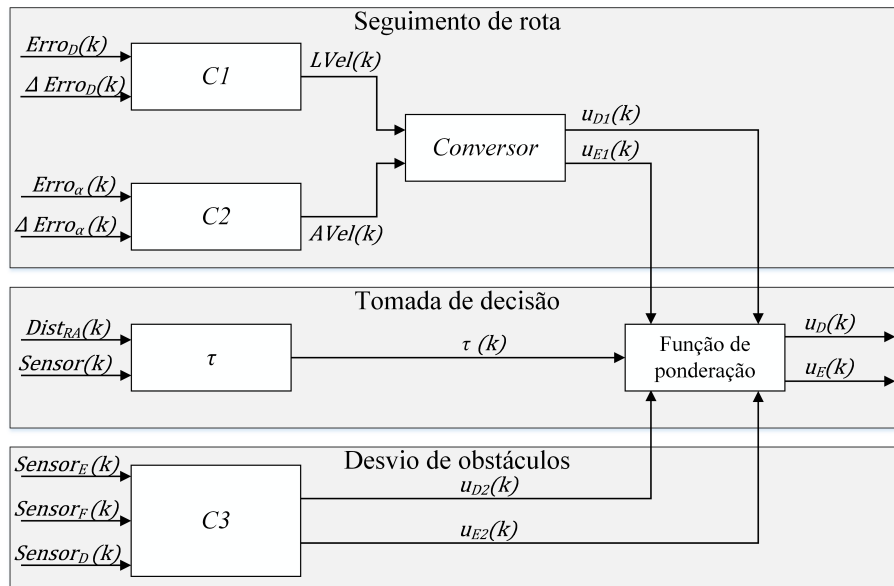


Figura 7: Sistemas *fuzzy* para o controle de locomoção dos robôs móveis.

Fonte: Autoria própria.

O módulo “Seguimento de rota” é responsável por gerar ações de controle para que o robô se alinhe à posição de seu objetivo e execute uma rota linear até este ponto. Para isso, dois controladores *fuzzy* Mamdani ($C1$ and $C2$) são projetados para gerar as velocidades linear ($LVel$) e angular ($AVel$) que são, posteriormente, convertidas em ações de controle a serem ponderadas e aplicadas às rodas direita e esquerda do robô, denominadas u_{D1} e u_{E1} , respectivamente. $C1$ e $C2$ são controladores do tipo PD-*Fuzzy* e estão de acordo com o desenvolvimento apresentado por Passino e Yurkovich (1998). A variável de entrada $Erro_D$ representa a distância entre as posições do robô e do seu objetivo, enquanto $\Delta Erro_D$ é a variação deste erro em uma amostra. De forma similar, $Erro_\alpha$ e $\Delta Erro_\alpha$ referem-se ao ângulo entre a frente do robô e sua posição objetivo. $Erro_D$ e $Erro_\alpha$ são ilustrados na Figura 8.

O controlador *fuzzy* $C3$, componente do módulo “Desvio de obstáculos”, é usado para promover o desvio de obstáculos estáticos (componentes do ambiente) e dinâmicos (a estrutura dos robôs). As intensidades dos três sensores de proximidade (esquerda (E), frente (F) e direita (D) — Figura 4) são usadas como entradas do controlador. Sua base de regras é desenvolvida para movimentar o robô em uma direção oposta ao obstáculo detectado, de acordo com a maior intensidade detectada. As saídas consistem nas ações de controle u_{D2} e u_{E2} , a serem tratadas pelo módulo de tomada de decisão e aplicadas às rodas do robô.

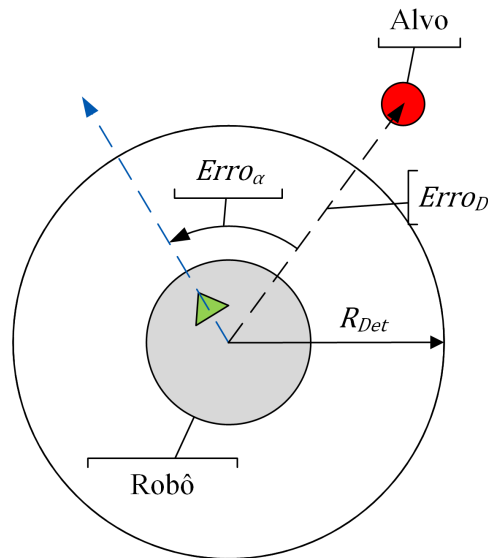


Figura 8: Ilustração das variáveis de entrada do sistema *fuzzy* para o seguimento de rota.

Fonte: Autoria própria.

No terceiro módulo, denominado “Tomada de decisão”, um sistema *fuzzy* é usado para ponderar as ações de controle obtidas dos outros dois módulos (u_{D1} , u_{E1} , u_{D2} e u_{E2}) e determinar as ações de controle resultante (u_D e u_E) que, de fato, são aplicadas às rodas do robô (R_E e R_D da Figura 4, em *rad/seg*).

A ponderação mencionada representa a decisão se um obstáculo detectado realmente implica em uma iminência de colisão. Para isso, o parâmetro $\tau \in [0, 1]$ é gerado por uma função *fuzzy* não-linear, $g(x) : U \subset \mathfrak{R}^2 \rightarrow \mathfrak{R}$ (WANG, 1997). A distância entre as posições atual do robô e do seu objetivo, $Dist_{RA}$, e a intensidade do sensor que apontar a maior iminência de colisão, $Sensor$, são consideradas como entradas para esta função, ambas normalizadas de acordo com o alcance de detecção do robô.

As funções de pertinência desta função não-linear são pseudo-trapezoidais definidas em \mathfrak{R} , como apresentada na Equação (2), onde $[a, d] \in \mathfrak{R}$; $a \leq b \leq c \leq d$ e $a < d$. Todas as variáveis $x \in U$ são definidas no universo de discurso $U = [0, 1]$.

$$\mu_A(x; a, b, c, d) = \begin{cases} \frac{x-a}{b-a}, & \text{se } x \in [a, b) \\ 1, & \text{se } x \in [b, c] \\ \frac{x-d}{c-d}, & \text{se } x \in (c, d] \\ 0, & \text{se } x \in \mathfrak{R} - (a, d) \end{cases} \quad (2)$$

As entradas x_1 ($Dist_{RA}$) e x_2 ($Sensor$) possuem as mesmas três funções de pertinência,

como mostradas na Figura 9 e descritas pelos parâmetros (x_i, a, b, c, d) : $\mu_{A_i^1} = (x_i, 0, 0, 0.2, 0.4)$, $\mu_{A_i^2} = (x_i, 0.2, 0.4, 0.4, 0.6)$ e $\mu_{A_i^3} = (x_i, 0.4, 0.6, 1, 1)$, onde $i = 1, 2$. A função de saída ($\tau \in [0, 1]$) possui quatro funções de pertinência, como apresentadas na Figura 10: $\mu_{B^1} = (0, 0, 0.2, 0.4)$, $\mu_{B^2} = (0.2, 0.4, 0.4, 0.6)$, $\mu_{B^3} = (x_\tau; 0.4, 0.6, 0.6, 0.8)$ e $\mu_{B^4} = (x_\tau; 0.6, 0.8, 1, 1)$.

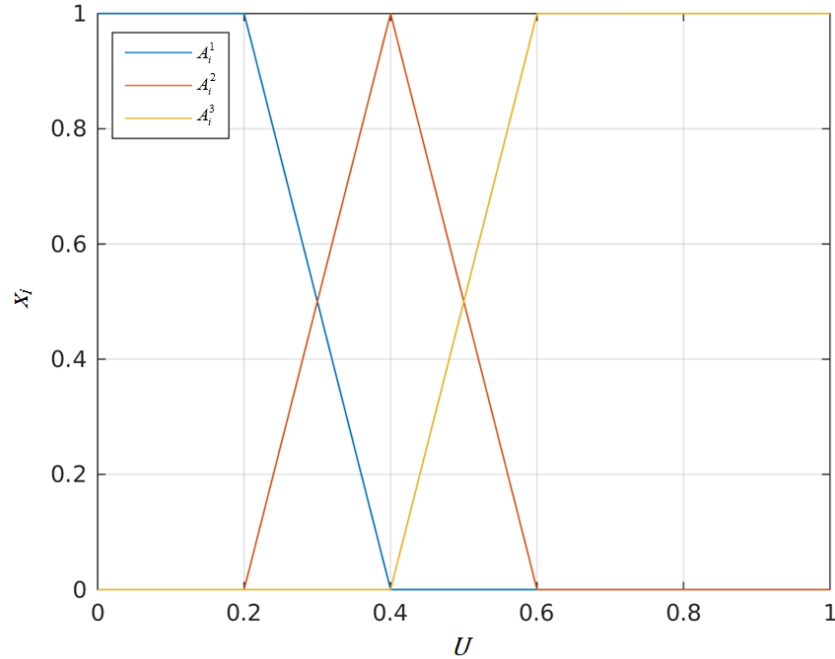


Figura 9: Funções de pertinência das entradas do interpolador fuzzy.

Fonte: Autoria própria.

A base de regras é composta por 9 sentenças, como apresentadas na Tabela 2, onde a_i^{j1} e b^{j2} ($i = 1, 2$, $j1 = 1, 2, 3$ e $j2 = 1, 2, 3, 4$) são os centros das funções de pertinência das entradas (A_1 e A_2) e da saída (B).

Tabela 2: Base de regras do sistema fuzzy para tomadas de decisão.

$Dist_{RA} \setminus Sensor$	a_2^1	a_2^2	a_2^3
a_1^1	b^4	b^2	b^1
a_1^2	b^4	b^3	b^2
a_1^3	b^4	b^3	b^2

Para uma melhor compreensão, três exemplos linguísticos desta base de regras são mostrados a seguir:

1. se $Dist_{RA}$ é a_1^1 (objetivo distante) e $Sensor$ é a_2^1 (obstáculo distante), então τ é b^4 ;
2. se $Dist_{RA}$ é a_1^1 (objetivo distante) e $Sensor$ é a_2^2 (obstáculo mais próximo que o objetivo), então τ é b^2 ;

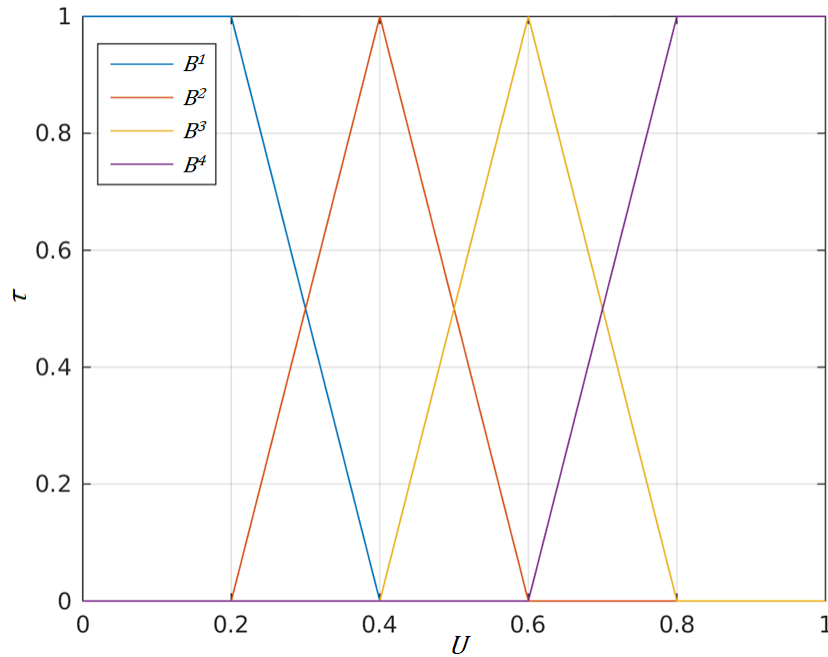


Figura 10: Funções de pertinência da saída do interpolador *fuzzy*.

Fonte: Autoria própria.

3. se $Dist_{RA}$ é a_1^1 (objetivo distante) e $Sensor$ é a_3^2 (obstáculo muito próximo do robô), então τ é b^1 .

De uma forma geral, a base de regras definida neste sistema é desenvolvida para garantir o seguinte comportamento:

- τ possui um valor “alto” quando o obstáculo está mais distante do robô que a posição do objetivo;
- τ possui um valor “baixo” quando o objetivo está mais distante que o obstáculo detectado;
- τ é “médio” quando ambas as distâncias são equiparadas.

A saída τ deste sistema *fuzzy*, usando o mecanismo de inferência pelo produto e defuzzificação por centro de gravidade, é definida na Equação (3), onde $g(\cdot)$ resulta das regras ativadas. Esta função *fuzzy* não-linear gera a superfície de resposta apresentada na Figura 11.

$$\tau(x_1, x_2) = \frac{\sum_{i1=1}^3 \sum_{i2=1}^3 g\left(\mu_{A_1^{i1}}(x_1) \mu_{A_2^{i2}}(x_2)\right)}{\sum_{i1=1}^3 \sum_{i2=1}^3 \left(\mu_{A_1^{i1}}(x_1) \mu_{A_2^{i2}}(x_2)\right)} \quad (3)$$

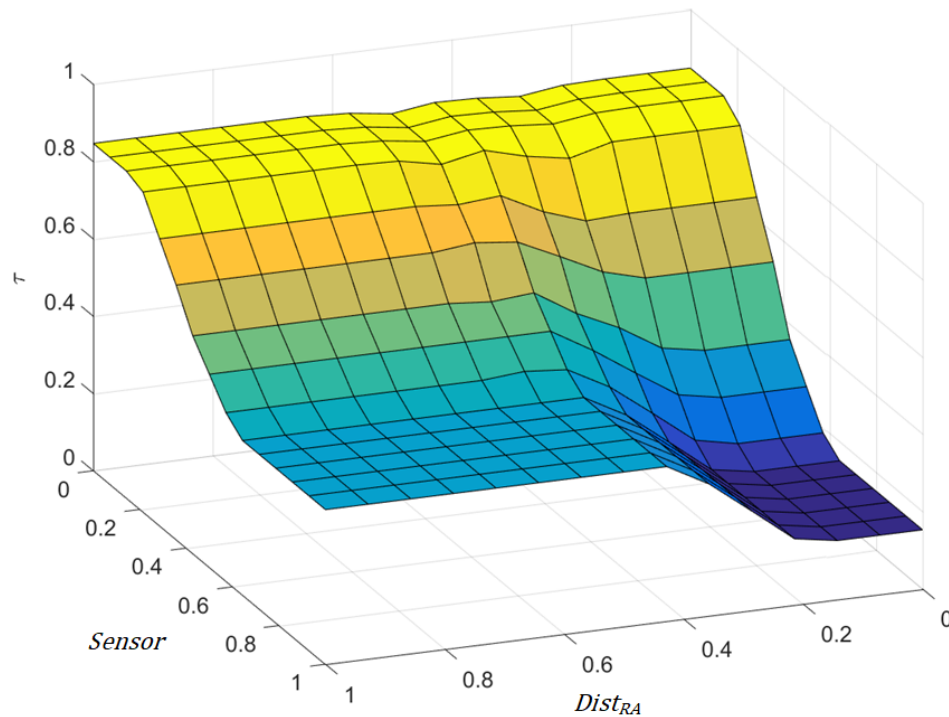


Figura 11: Superfície de resposta referente ao sistema *fuzzy* para a tomada de decisão.

Fonte: Autoria própria.

Após a obtenção de τ , a ponderação apresentada na Equação (4) é aplicada e as ações de controle u_D e u_E são geradas.

$$\begin{aligned} u_D &= (u_{D1} \cdot \tau) + [u_{D2} \cdot (1 - \tau)] \\ u_E &= (u_{E1} \cdot \tau) + [u_{E2} \cdot (1 - \tau)] \end{aligned} \quad (4)$$

As ações de controle calculadas pela Equação (4) movimentam o robô em direção ao seu objetivo, definido na etapa de planejamento da navegação.

3.1.5 SISTEMA DE AUTOLOCALIZAÇÃO DOS ROBÔS

Cada robô considerado neste trabalho é capaz de estimar a sua localização no ambiente a partir de dados obtidos de sua odometria. Entretanto, acidentes topográficos no ambiente, características do solo que promovem a derrapagem de suas rodas e caminhos que requerem um longo período de navegação provocam erros cumulativos em sistemas de localização baseados em odometria (HATTORI et al., 2016; LEE et al., 2016; SIEGWART et al., 2011). Durante a navegação de um robô, estes erros acumulados correspondem a uma incerteza acerca de sua posição no ambiente e, conseqüentemente, a rota de fato executada pode divergir da planejada.

De uma forma geral, as literaturas da área de robótica móvel afirmam que a dinâmica do acúmulo de erros na odometria pode ser modelada como uma elipse crescente, como exemplificado na Figura 12. A posição real do robô pode estar localizada em qualquer ponto no interior desta geometria (SIEGWART et al., 2011). Se os erros acumulados resultam em uma elipse de dimensões amplas, a real posição do robô (geralmente, o centro de sua geometria) é completamente diferente da calculada pelo seu sistema de navegação, que é baseado em um sistema global de coordenadas com uma referência fixa.

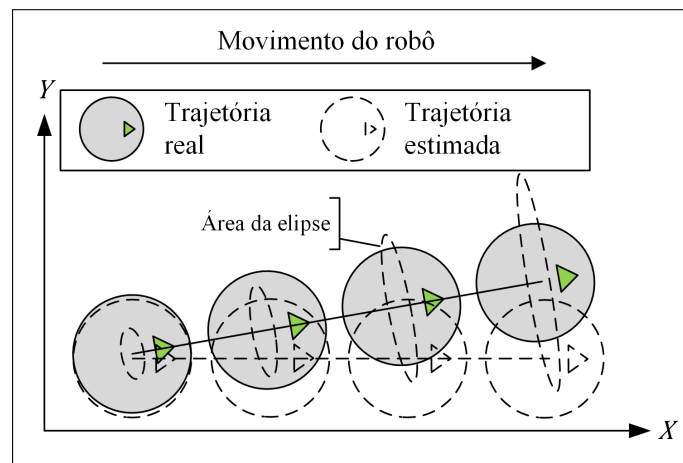


Figura 12: Ilustração da propagação do erro em sistemas de odometria.

Fonte: Autoria própria.

Neste trabalho, o acúmulo de erros advindos da odometria, representado pela elipse crescente na Figura 12, é modelado de acordo com o desenvolvimento apresentado em (SIEGWART et al., 2011) e resumido conforme segue.

Geralmente, a posição (p) de um robô móvel em um ambiente bidimensional (coordenadas x , y e orientação θ) é representada pela Equação (5). Para um robô móvel com acionamento diferencial, a atualização de sua posição no ambiente (p') em função da rotação de suas rodas é dada pela Equação (6), onde: Δs_d e Δs_e correspondem à rotação das rodas direita e esquerda, respectivamente; e b é a distância entre as rodas direita e esquerda.

$$p = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (5)$$

$$p' = f(x, y, \theta, \Delta s_d, \Delta s_e) = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \frac{\Delta s_d + \Delta s_e}{2} \cos\left(\theta + \frac{\Delta s_d - \Delta s_e}{2b}\right) \\ \frac{\Delta s_d + \Delta s_e}{2} \text{sen}\left(\theta + \frac{\Delta s_d - \Delta s_e}{2b}\right) \\ \frac{\Delta s_d - \Delta s_e}{b} \end{bmatrix} \quad (6)$$

Em seguida, um modelo para os erros de integração de p é estabelecido em (SIEGWART et al., 2011) para se obter a matriz de covariância $\Sigma_{p'}$ da posição estimada do robô baseada nos dados de odometria (Δs_d e Δs_e) e, para isso, assume-se que a matriz de covariância Σ_p é conhecida no início de sua navegação. Conforme o robô se movimenta (incrementos de Δs_d e Δs_e), obtém-se a matriz de covariância Σ_Δ apresentada na Equação (7), onde k_d e k_e são constantes de erros associadas às rodas direita e esquerda, respectivamente, que representam seus parâmetros não determinísticos, tais como suas imperfeições construtivas, os erros nos *encoders*, as situações de derrapagens e acidentes topográficos no ambiente (interação das rodas com o ambiente), por exemplo.

$$\Sigma_\Delta = \text{covar}(\Delta s_d, \Delta s_e) = \begin{bmatrix} k_d |\Delta s_d| & 0 \\ 0 & k_e |\Delta s_e| \end{bmatrix} \quad (7)$$

Após as manipulações matemáticas detalhadas em (SIEGWART et al., 2011) e com base nas Equações (6) e (7), a matriz de covariância $\Sigma_{p'}$ pode ser escrita como mostrado na Equação (8), onde $\Delta_{d,e} = \begin{bmatrix} \Delta s_d & \Delta s_e \end{bmatrix}^T$.

$$\Sigma_{p'} = \nabla_p f \cdot \Sigma_p \cdot \nabla_p f^T + \nabla_{\Delta_{d,e}} f \cdot \Sigma_\Delta \cdot \nabla_{\Delta_{d,e}} f^T \quad (8)$$

Finalmente, usando a Equação (6), duas matrizes são desenvolvidas, $F_p = \nabla_p f$ e $F_{\Delta_{d,e}} = \nabla_{\Delta_{d,e}} f$:

$$F_p = \nabla_p f = \nabla_p (f^T) = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta s \cdot \text{sen}(\theta + \Delta\theta/2) \\ 0 & 1 & \Delta s \cdot \cos(\theta + \Delta\theta/2) \\ 0 & 0 & 1 \end{bmatrix} \quad (9)$$

$$F_{\Delta_{d,e}} = \begin{bmatrix} \frac{1}{2} \cos\left(\theta + \frac{\Delta\theta}{2}\right) - \frac{\Delta s}{2b} \text{sen}\left(\theta + \frac{\Delta\theta}{2}\right) & \frac{1}{2} \cos\left(\theta + \frac{\Delta\theta}{2}\right) + \frac{\Delta s}{2b} \text{sen}\left(\theta + \frac{\Delta\theta}{2}\right) \\ \frac{1}{2} \text{sen}\left(\theta + \frac{\Delta\theta}{2}\right) + \frac{\Delta s}{2b} \cos\left(\theta + \frac{\Delta\theta}{2}\right) & \frac{1}{2} \text{sen}\left(\theta + \frac{\Delta\theta}{2}\right) - \frac{\Delta s}{2b} \cos\left(\theta + \frac{\Delta\theta}{2}\right) \\ \frac{1}{b} & -\frac{1}{b} \end{bmatrix} \quad (10)$$

onde Δs e $\Delta \theta$ são definidos nas Equações (11) e (12), respectivamente:

$$\Delta s = \frac{\Delta s_d + \Delta s_e}{2} \quad (11)$$

$$\Delta \theta = \frac{\Delta s_d - \Delta s_e}{b} \quad (12)$$

O desenvolvimento matemático apresentado é utilizado para representar a evolução do acúmulo de erros presente na odometria (ilustrada na Figura 12), considerada em cada robô deste trabalho.

No contexto dos problemas de planejamento de rota, assim como abordados nesta tese, a execução de um caminho que diverge do planejado significa que o robô pode não se dirigir aos alvos atribuídos a ele. Neste trabalho, esta situação é classificada como “navegação cega” e o robô deve possuir mecanismos que lhe permitam executar procedimentos alternativos para se localizar, recalculando sua rota e se redirecionar ao seu objetivo assim que possível. Tais procedimentos devem ser realizados de forma *on-line*, e serão apresentados conforme as estratégias de planejamento de rota são propostas nos capítulos seguintes.

3.2 CONFIGURAÇÃO DOS EXPERIMENTOS

De um modo simplificado, as características do ambiente e do SMR apresentados neste capítulo são implementadas em ambientes simulado e real, de acordo com a configuração ilustrada na Figura 13. Em especial, para a implementação em ambiente real, um sistema baseado em realidade aumentada é desenvolvido para possibilitar a execução das habilidades de sensoriamento e comunicação dos robôs, requeridas por cada estratégia de planejamento de rota propostas neste trabalho, de forma virtual. Assim, os diversos aspectos de comunicação entre robôs (comunicação direta) e com o ambiente (comunicação indireta) podem ser testados e validados sob circunstâncias de robôs reais com capacidades limitadas.

As informações do ambiente (posição dos robôs/alvos/obstáculos) e dos dispositivos sensoriais dos robôs são publicadas em tópicos específicos do *framework* ROS. O *software* Matlab é utilizado para implementar as estratégias propostas de planejamento de rota em simulações e *scripts* em C++ são utilizados para os experimentos com robôs reais, de acordo com as informações disponíveis nos tópicos ROS. As respostas calculadas também são publicadas em tópicos ROS e são recebidas pelo ambiente de simulação (V-REP) ou real para a atuação do SMR.

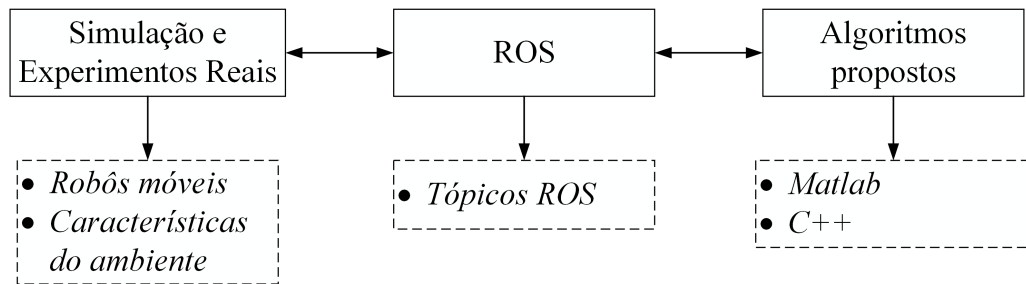


Figura 13: Configuração dos experimentos de simulação e com robôs reais.

Fonte: Autoria própria.

Os detalhes adicionais acerca das implementações mencionadas são descritas nos capítulos 4, 5 e 6, conforme são apresentadas as estratégias propostas nesta tese.

4 PLANEJAMENTO DE ROTAS PARA TAREFAS DE VISITAR ALVOS CONHECIDOS EM UM AMBIENTE SEMI-DESCONHECIDO

Com base nas características apresentadas no capítulo 3, considera-se neste capítulo a configuração de um ambiente em que o sistema multirrobô (SMR) conhece os pontos de interesse (alvos) e desconhece a posição dos obstáculos.

Um cenário típico desta configuração é ilustrado na Figura 14, onde 5 robôs estão em sua posição inicial, as linhas tracejadas representam um exemplo de rotas calculadas previamente sem o conhecimento dos obstáculos, os alvos são numerados de 1 a 20, e os 12 obstáculos estáticos (cilindros cinzas) estão em posições desconhecidas. Seguindo as rotas de cada robô, pode-se observar que algumas posições são destacadas por retângulos cinzas (C1 a C5), que representam os riscos de colisão não previsíveis para a navegação. Quando os robôs atingirem estas regiões, eles devem possuir habilidades que os permitam detectar os obstáculos, recalculando suas rotas para evitar a colisão e retomar o seu caminho calculado anteriormente.

Um sistema de navegação do SMR para tal configuração deve ser capaz de calcular a rota de todos os robôs e lidar com os problemas de colisão com obstáculos estáticos e dinâmicos (os outros robôs), assim como amenizar as situações de incertezas na localização dos robôs no ambiente.

Para a solução deste problema, um planejador híbrido de rotas é proposto neste capítulo, composto por um módulo *off-line* (executado antes da navegação) para a geração de rotas otimizadas utilizando um algoritmo genético, e outro *on-line* (executado de forma independente por cada robô), utilizado durante a navegação para recalculando as rotas obtidas pela etapa *off-line*, devido ao desvio de obstáculos e a perda momentânea de localização. Neste caso, devido ao prévio conhecimento da posição de todos os alvos, não é necessário que os robôs desempenhem a tarefa de exploração do ambiente e, portanto, a habilidade de comunicação indireta, conforme descrita na seção 3.1.3, não é considerada neste capítulo. Estes módulos combinados com o controlador *fuzzy* de navegação, mostrado na Figura 7, compõem o sistema autônomo completo de navegação para o ambiente avaliado neste capítulo.

Os módulos de planejamento *off-line* e *on-line* são descritos nas seções 4.1 e 4.2,

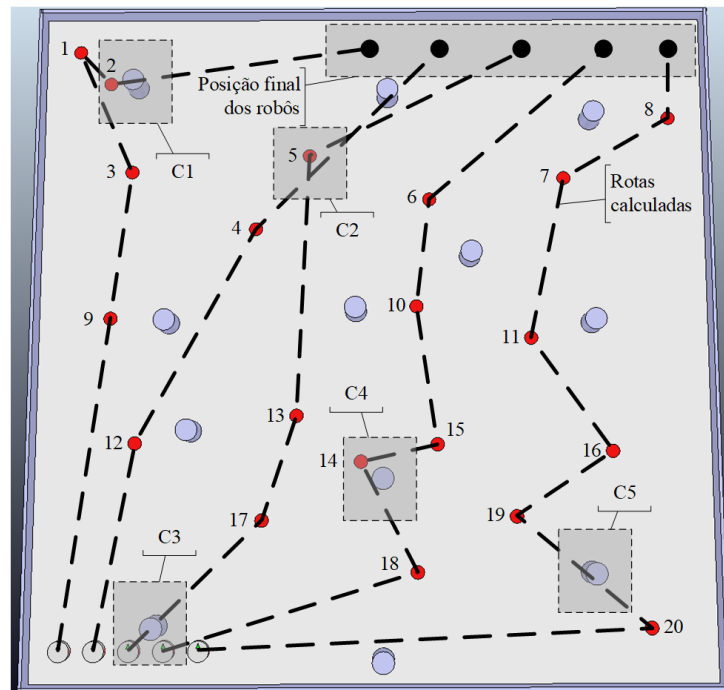


Figura 14: Exemplo de solução para a configuração em que a posição dos alvos é conhecida e dos obstáculos é desconhecida.

Fonte: Autoria própria.

respectivamente, enquanto os resultados de validação deste planejador híbrido são apresentados na seção 4.3.

4.1 PLANEJAMENTO *OFF-LINE* DE ROTAS BASEADO EM ALGORITMO GENÉTICO

O objetivo do módulo *off-line* de planejamento de rotas consiste em calcular uma lista de caminhos, composta por um conjunto de alvos em sequência, e atribuí-las individualmente a cada robô do sistema. Esta otimização visa obter a menor distância possível entre as posições inicial e final dos robôs, enquanto todos os alvos são visitados pelo menos uma vez.

Este problema de otimização é modelado como uma variação do problema do caixeiro viajante múltiplo (MTSP), que é uma variação do conhecido problema do caixeiro viajante (*Traveling Salesman Problem (TSP)*) (CARTER; RAGSDALE, 2006). No MTSP, m ($m > 1$) caixeiros devem visitar n ($n > m$) cidades, por meio de subrotas atribuídas a cada um. Em analogia, caixeiros são robôs e cidades são os alvos neste trabalho.

Para uma melhor compreensão desta modelagem, as seguintes variáveis são definidas:

- (i) o sistema multirrobô é composto por n_R robôs (agentes). R_i representa o i -ésimo agente

deste sistema ($i = 1, 2, \dots, n_R$);

- (ii) n_A alvos devem ser visitados durante a navegação e eles pertencem ao conjunto S . A_j é identificado como o j -ésimo alvo deste conjunto ($j = 1, 2, \dots, n_A$);
- (iii) o subconjunto S_{R_i} representa os alvos atribuídos ao robô R_i pelo planejador, onde $S = S_{R_1} \cup S_{R_2} \cup \dots \cup S_{R_{n_R}}$

Para ilustração, a solução exemplificada na Figura 14 apresenta $n_R = 5$, $n_A = 20$, $S = \{1, 2, \dots, 20\}$ e os subconjuntos atribuídos aos robôs são: $S_{R_1} = \{1, 2, 3, 9\}$, $S_{R_2} = \{4, 12\}$, $S_{R_3} = \{5, 13, 17\}$, $S_{R_4} = \{6, 10, 14, 15, 18\}$ e $S_{R_5} = \{7, 8, 11, 16, 19, 20\}$. Estes subconjuntos compõem as rotas dos robôs, formados pelas linhas pretas tracejadas, que devem ser seguidas por eles.

A partir destas definições, um algoritmo genético é desenvolvido para obter os subconjuntos $S_{R_1} \cup S_{R_2} \cup \dots \cup S_{R_{n_R}}$, com os seguintes objetivos: (i) minimizar a distância a ser navegada por cada robô; (ii) distribuir igualmente os alvos entre os subconjuntos (o mais uniforme possível), sem sobrecarregar um robô específico; e (iii) evitar desvios excessivos de um caminho auxiliar, definido como a linha reta entre as posições inicial e final de cada robô sem passar por alvos (a menor rota possível para alcançar a posição final, partindo da posição inicial).

A codificação do cromossomo é baseada nas abordagens de Yuan et al. (2013) e Carter e Ragsdale (2006). Cada cromossomo é composto por duas partes: na “Parte 1” tem-se a lista completa dos alvos do ambiente (conjunto S) e cada gene representa um alvo A_j ($j = 1, 2, \dots, n_A$). A sequência dos genes constitui uma rota. Cada gene da “Parte 2” representa um robô e sua posição determina o número (identificação) correspondente, ou seja, o primeiro gene da “Parte 2” representa o robô R_1 , o segundo é o R_2 , até que o gene n_R representa o robô R_{n_R} . O valor dos alelos para cada gene da “Parte 2” corresponde ao número de genes da “Parte 1” que compõem a rota atribuída a cada robô. Deste modo, as rotas (subconjuntos de genes) são atribuídas em sequência.

Um indivíduo (um cromossomo) modelado por esta representação é exemplificado na Figura 15, onde $S = \{1, 2, \dots, 18\}$ e $n_R = 4$. Neste exemplo, a “Parte 2” indica que o robô R_1 deve visitar os primeiros 5 alvos da “Parte 1”, portanto a rota atribuída a R_1 é $S_{R_1} = \{3, 2, 10, 1, 11\}$; R_2 irá visitar os próximos 7 alvos, ou seja, $S_{R_2} = \{12, 4, 6, 5, 13, 8, 7\}$; e assim por diante.

O operador de cruzamento utilizado para a recombinação dos indivíduos consiste no método *Two-Chromosome Crossover* (TCX) (YUAN et al., 2013), aplicável à codificação

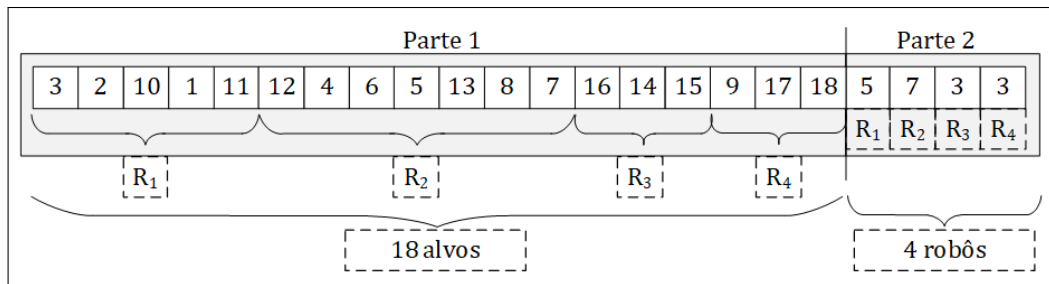


Figura 15: Exemplo numérico de codificação de um cromossomo.

Fonte: Autoria própria.

de cromossomo apresentada na Figura 15. Este método de cruzamento é sintetizado em 4 passos para que, a partir de dois cromossomos progenitores (P1 e P2), gerar um cromossomo descendente (D1). Os passos são listados a seguir e são exemplificados na Figura 16, em que 9 alvos devem ser distribuídos entre 3 robôs (um segundo descendente pode ser obtido com a repetição destes procedimentos e adotando P2 como referência no passo 2):

1. selecionar o par de cromossomos progenitores, P1 e P2 (Figura 16 (a));
2. selecionar em P1, de forma aleatória, um segmento de genes para cada robô (genes em vermelho na Figura 16 (b));
3. selecionar os genes de P2 que correspondem aos genes não selecionados em P1 (genes em azul na Figura 16 (c));
4. gerar um descendente (D1) de acordo com as seleções efetuadas nos passos 2 e 3 (combinação dos genes em vermelho e em azul na Figura 16 (d)).

Todos os operadores do algoritmo genético proposto são resumidos na Tabela 3, cujos principais detalhes podem ser encontrados em (AGGARWAL et al., 2014; SASTRY et al., 2014; YUAN et al., 2013).

Tabela 3: Operadores do algoritmo genético para o planejador de rotas *off-line*.

Parâmetros	Especificação
Seleção para cruzamento	Torneio com 3 candidatos
Cruzamento	TCX
Probabilidade de mutação	5%
Mutação	<i>Swap</i>

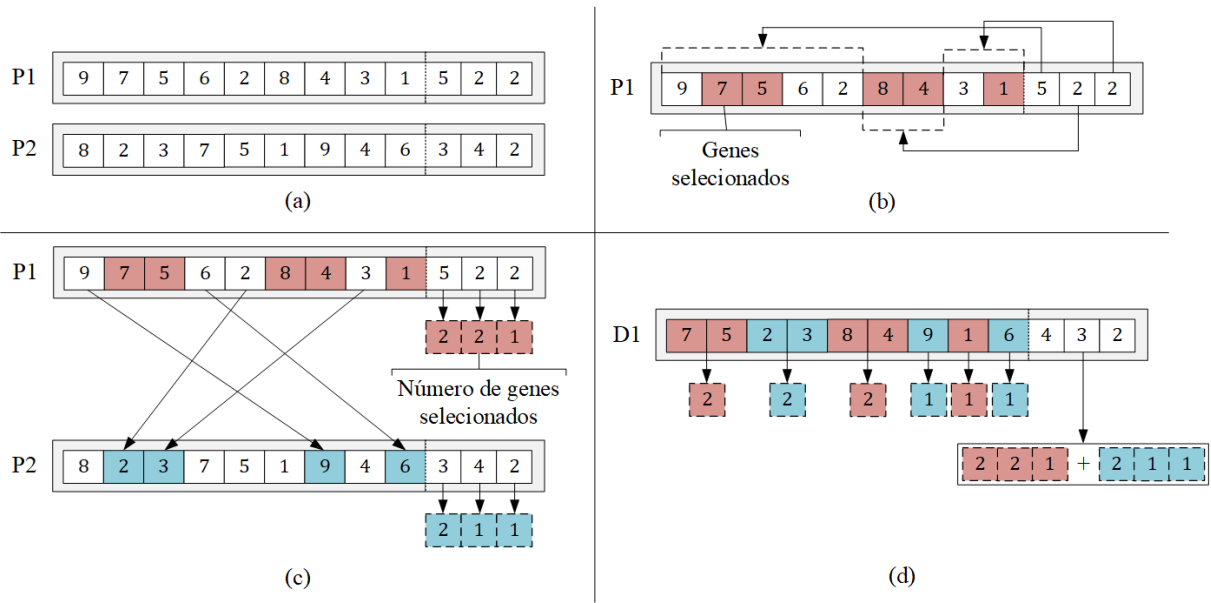


Figura 16: Procedimentos para o operador de cruzamento TCX: (a) cromossomos progenitores (P1 e P2) selecionados; (b) seleção aleatória dos genes de P1; (c) seleção dos genes de P2 e (d) combinação e obtenção de um cromossomo descendente (D1).

Fonte: Adaptado de (YUAN et al., 2013).

Uma função de aptidão (*fitness*) é desenvolvida para viabilizar os objetivos de otimização previamente descritos, que são: minimizar a distância a ser percorrida por cada robô, evitar desvios excessivos da rota auxiliar (um vetor entre as posições inicial e final), e equalizar o número de alvos a serem visitados pelos robôs. Para isso, as seguintes variáveis são definidas e ilustradas na Figura 17 (etapas de A a D), em que 3 objetivos (dois alvos e a posição final) são atribuídos a um robô e a menor rota possível é representada pela linha tracejada:

- (i) $SP_{\alpha k}$ é o ângulo entre o vetor de referência no eixo x e o vetor que interliga as posições atual e a final desejada do robô, no instante k . Portanto, este ângulo se altera conforme o robô se movimenta;
- (ii) NA_{Ri} é o número de alvos contidos no subconjunto S_{Ri} ;
- (iii) $Dist_{Rij}$ é a distância a ser navegada pelo robô R_i da sua posição atual ao seu j -ésimo objetivo;
- (iv) $Dist_{Ri}$ é a distância total a ser navegada pelo robô R_i :

$$Dist_{Ri} = \sum_{j=1}^{NA_{Ri}+1} Dist_{Rij} \quad (13)$$

(v) $Dist_{Total}$ é a distância total a ser navegada pelo sistema multirrobo, composta pela soma de todas as distâncias calculadas individualmente pela Equação (13);

(vi) R_{Dist}^{Ri} é a relação entre $Dist_{Ri}$ e $Dist_{Total}$:

$$R_{Dist}^{Ri} = \frac{Dist_{Ri}}{Dist_{Total}} \quad (14)$$

(vii) $Erro_{\alpha}^{Rij}$ é o desvio angular, em graus, entre o vetor com origem à frente do robô e o vetor que interliga sua posição atual à final. Este erro representa o quanto o robô R_i desvia-se de sua rota de referência (menor rota possível, que interliga as posições inicial e final do robô) para visitar o objetivo j . Nesse caso, o erro máximo é de 180° ;

(viii) $R_{Erro_{\alpha}}^{Ri}$ é a soma do erro normalizado (por 180°) de $Erro_{\alpha}^{Rij}$ para todo j :

$$R_{Erro_{\alpha}}^{Ri} = \sum_{j=1}^{NA_{Ri}} \frac{Erro_{\alpha}^{Rij}}{180^\circ} \quad (15)$$

(ix) R_{NA}^{Ri} é a razão entre o número de alvos atribuídos ao robô R_i , NA_{Ri} , e o total de alvos a serem atribuídos no sistema multirrobo, n_A :

$$R_{NA}^{Ri} = \frac{NA_{Ri}}{n_A} \quad (16)$$

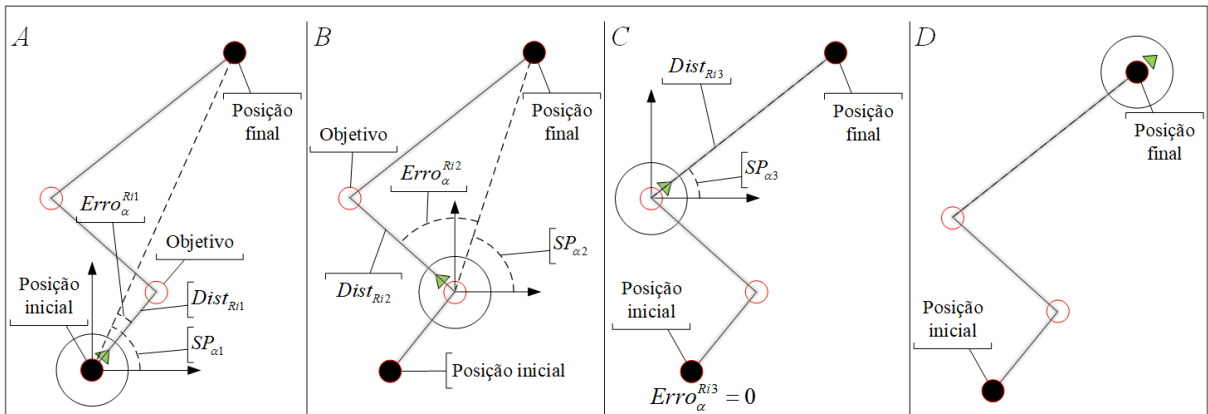


Figura 17: Ilustração das variáveis definidas para a função de aptidão (*fitness*) do algoritmo genético. *A* é a situação inicial; *B* é a situação em que o primeiro alvo foi visitado; *C* é o momento em que o robô visita o seu segundo alvo; e em *D* o robô alcança sua posição final.

Fonte: Autoria própria.

As variáveis R_{Dist}^{Ri} , $R_{Erro_{\alpha}}^{Ri}$ e R_{NA}^{Ri} representam os múltiplos objetivos do problema de otimização previamente discutidos. Assim, esta otimização corresponde à maximização da

função de aptidão (*fit*) definida na Equação (17). Além disso, observou-se durante as execuções do algoritmo que a relação de compromisso entre estes objetivos de otimização prejudica a distribuição de rotas conforme aumenta-se a complexidade do problema (tamanho do ambiente, número de robôs e alvos), isto é, o planejador apresenta dificuldades em manter um equilíbrio aceitável entre minimizar o tamanho das rotas (R_{Dist}^{Ri}), os desvios da rota auxiliar ($R_{Erro\alpha}^{Ri}$) e a equalização do número de alvos a ser visitado por cada robô (R_{NA}^{Ri}). Dessa forma, a variável K foi inserida na Eq (17) com o intuito de ponderar a influência de R_{Dist}^{Ri} na solução.

$$fit = \frac{1}{Dist_{Total} \cdot \sum_{i=1}^{nR} \left(K \cdot R_{Dist}^{Ri} + R_{Erro\alpha}^{Ri} + R_{NA}^{Ri} \right)} \quad (17)$$

Após a execução do algoritmo genético proposto, o subconjunto de alvos (“Parte 1” da Figura 15) do melhor indivíduo é distribuído a cada robô como uma lista de tarefas, de acordo com as indicações da “Parte 2”. Em sequência, a navegação é iniciada com todos os robôs rastreando seus objetivos (alvos e posição final). Durante a navegação, cada robô é independentemente guiado por seu controlador *fuzzy* e por seu módulo de planejamento *on-line*. A etapa de planejamento *off-line* é encerrada a partir deste momento.

4.2 PLANEJAMENTO ON-LINE DE ROTAS

Cada robô possui um módulo de planejamento *on-line*, responsável por executar procedimentos locais durante a navegação para a correção das rotas calculadas anteriormente pelo planejador *off-line*. Seu principal objetivo é redirecionar o robô a sua rota planejada sempre que a rota navegada divergir da calculada. Neste trabalho, as seguintes situações podem causar tais desvios de rotas: desvio de obstáculos e erros na odometria.

Quanto ao desvio de obstáculos, dado que a posição dos objetos estáticos e dinâmicos (a estrutura dos outros robôs) é desconhecida ao planejador *off-line*, todos os robôs devem detectá-los localmente durante a navegação (pelos sensores de proximidade – veja Figura 4) e evitar a colisão para preservar sua integridade física. Essas manobras de prevenção de colisão implicam em erros de seguimento da rota calculada que devem ser corrigidos assim que possível.

Para a prevenção de colisões e correção de sua rota, o robô verifica se o obstáculo mais próximo está à esquerda ou à direita de sua rota planejada, por meio de seus sensores de proximidade, e calcula um caminho secundário, através da rotação do vetor que interliga as posições atual do robô e seu próximo objetivo. Essa rotação corresponde ao ângulo

$\pm\alpha_{desvio}$, resultante das ações de controle combinadas e obtidas do sistema de navegação *fuzzy* apresentado na Figura 7. Este comportamento é exemplificado na Figura 18, onde o robô R1 detecta o robô R2 como um obstáculo e calcula a correção de sua rota, obtida por meio da inserção de um objetivo auxiliar, a partir do qual a rota planejada é substituída por uma auxiliar, composta de 2 segmentos: o primeiro interliga a posição atual do robô e o objetivo auxiliar, enquanto o segundo interliga este último passo até o objetivo anteriormente planejado.

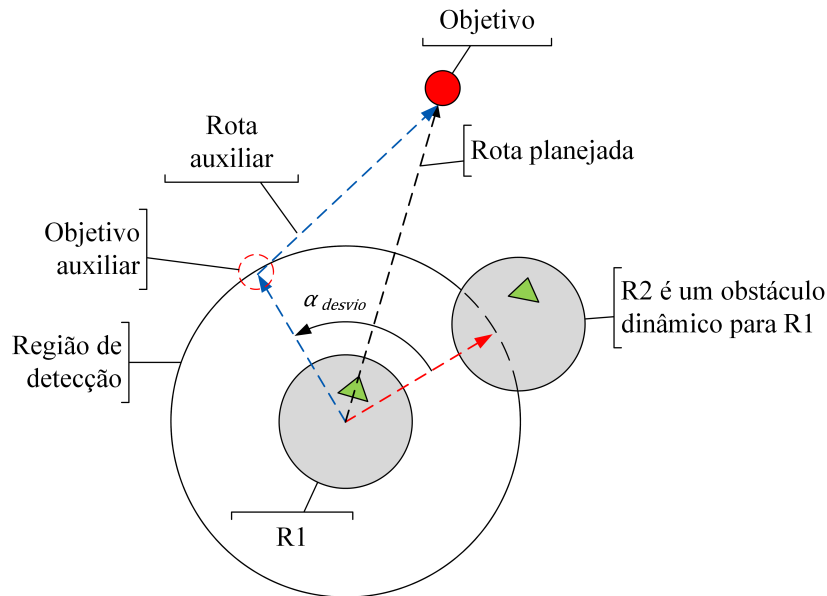


Figura 18: Manobra para correção de rota devido à presença de obstáculos.

Fonte: Autoria própria.

O outro caso que requer uma correção *on-line* de rota refere-se às navegações por longos caminhos, nas quais os erros de odometria se acumulam devido às características do ambiente e derrapagem das rodas dos robôs, como descrito na seção 3.1.5. Tais erros são modelados como uma elipse crescente (Figura 12) e quanto maior é sua área, maior é a incerteza acerca da real posição atual do robô. Uma situação crítica acontece neste trabalho quando a área desta elipse excede um valor de limiar K_{NC} . Neste caso, considera-se que o robô perdeu sua referência de localização no ambiente e está inapto para se mover em direção aos alvos atribuídos a ele. Assim, o robô apenas se move em linha reta e desvia de obstáculos, se necessário. Este comportamento é definido como “navegação cega” e esta condição permanece até que o robô consiga se autolocalizar, ou até que o robô encontre, por coincidência, seu próximo objetivo (um alvo ou sua posição final) e então consiga retomar sua rota original.

O método de autolocalização implementado neste módulo de planejamento *on-line* utiliza dados exteroceptivos, obtidos de interações locais, para executar um algoritmo de triangulação (LEE et al., 2016; ESTEVES et al., 2003). Este procedimento permite que o

robô calcule sua posição no ambiente, saia da condição de navegação cega, e assuma uma nova rota em direção ao seu próximo objetivo.

No algoritmo de triangulação, os outros robôs e os alvos são considerados como pontos de apoio (*landmarks*) incertos e corretos, respectivamente. Os alvos são considerados pontos de trabalho com localização conhecida, dotados de dispositivos para comunicação. Quando três ou mais *landmarks* estão no interior da área de comunicação do robô (Figura 6), eles compartilham informações. Os robôs envolvidos informam suas posições estimadas e as respectivas incertezas associadas (área da elipse), enquanto os alvos detectados informam sua posição exata.

Com as informações de posição obtidas, um robô sob a condição de navegação cega é capaz de executar o procedimento de autolocalização. Neste trabalho, é sempre necessário que pelo menos um alvo esteja envolvido neste procedimento, dado que não há incertezas acerca de sua posição no ambiente. Portanto, suas coordenadas (x, y) são usadas como pontos de referência para a autolocalização. Além deste ponto, a seguinte regra de prioridade é considerada para a composição dos demais *landmarks*:

- (i) uma formação triangular composta apenas por alvos é sempre prioritária;
- (ii) quando apenas um ou dois alvos são detectados, outros robôs podem complementar a formação triangular como *landmarks* incertos, e aqueles que possuem a menor incerteza em sua posição (a menor área da elipse) são escolhidos;
- (iii) robôs sob navegação cega não podem ser usados para a autolocalização.

Estas prioridades são ilustradas por meio da Figura 19, onde o robô R1 executa a autolocalização. Nas partes (a) e (b), há duas opções para a composição triangular (formações em azul ou em vermelho) e, de acordo com as regras definidas, o triângulo em azul é escolhido para as duas situações de detecção. Na parte (c), a condição de navegação cega de R2 impede que R1 o use como um *landmark* para a autolocalização.

Após a escolha dos *landmarks*, o robô estima sua posição e atualiza a área de sua elipse (incerteza) nas seguintes condições: (i) se apenas alvos são usados na triangulação, a área da elipse será nula, o que significa que o robô sabe sua exata posição; e (ii) se um ou mais robôs estão envolvidos na triangulação, a área da elipse resultante herda o maior valor entre as áreas associadas aos robôs usados como *landmarks* incertos.

Um fluxograma do funcionamento do planejador *on-line* de rotas é apresentado na Figura 20, onde a correção da rota é destacada à direita. Ao lado esquerdo, a navegação cega permanece se o robô não encontrar os *landmarks* requeridos para o algoritmo de triangulação.

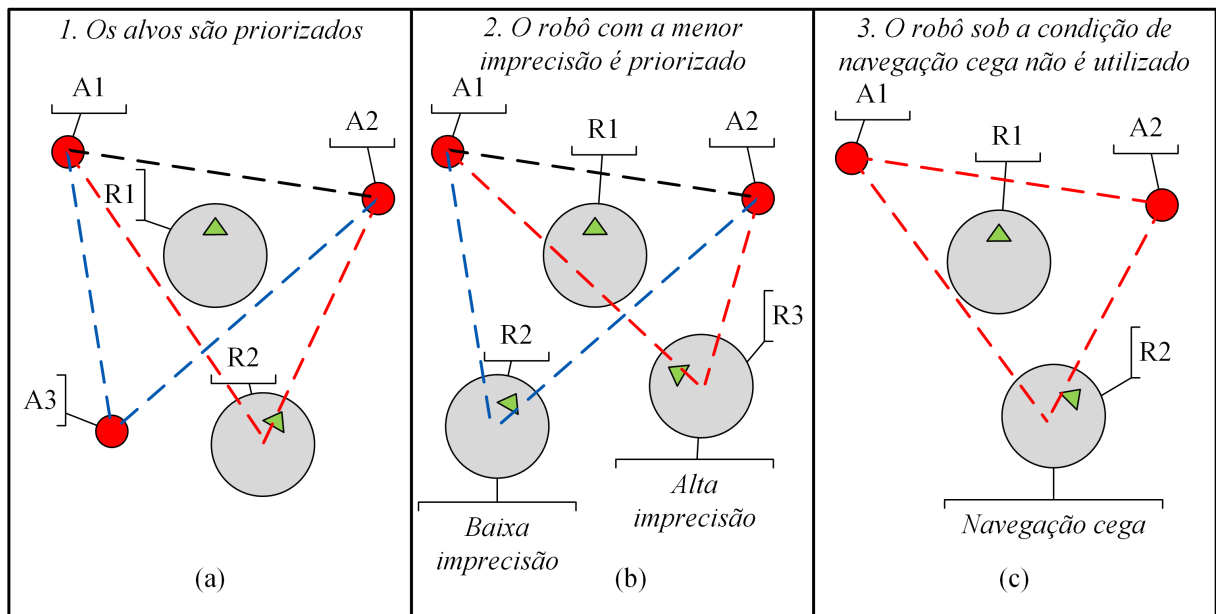


Figura 19: Regra de prioridade para o algoritmo de triangulação: (a) O landmark A3 possui prioridade sobre o robô R2. (b) O robô com a menor imprecisão (R2) é priorizado ao invés de R3. (c) Robôs sob a condição de navegação cega não são usados neste procedimento.

Fonte: Autoria própria.

4.3 RESULTADOS SIMULADOS

Nesta seção, a estratégia de planejamento de rota proposta é validada em três etapas, por meio de experimentos simulados.

Inicialmente, o planejador *off-line* descrito na seção 4.1 é aplicado em diferentes cenários com diversas configurações (número de robôs, número de alvos a serem visitados, entre outros). Neste contexto, é analisado se as rotas calculadas compõem caminhos factíveis (os menores possíveis) a serem navegados por cada robô, assim como se a distribuição de alvos entre eles é equalizada.

O desempenho do planejador *off-line* é avaliado por meio de 20 execuções do algoritmo genético para cada cenário simulado, ou seja, 20 execuções com inicializações aleatórias para cada conjunto fixo de alvos e robôs. *Box-plots* são considerados para análises estatísticas dos resultados e validação da distribuição de rotas e alvos.

Na segunda etapa, o planejador *on-line* (descrito na seção 4.2) é validado. Dada uma lista de rotas calculadas, verifica-se se os robôs são aptos a recalculá-las dinamicamente após os desvios ocasionados pelos erros de odometria ou por evitar colisão com obstáculos e, conseqüentemente, retomar a rota calculada anteriormente. Também é analisado

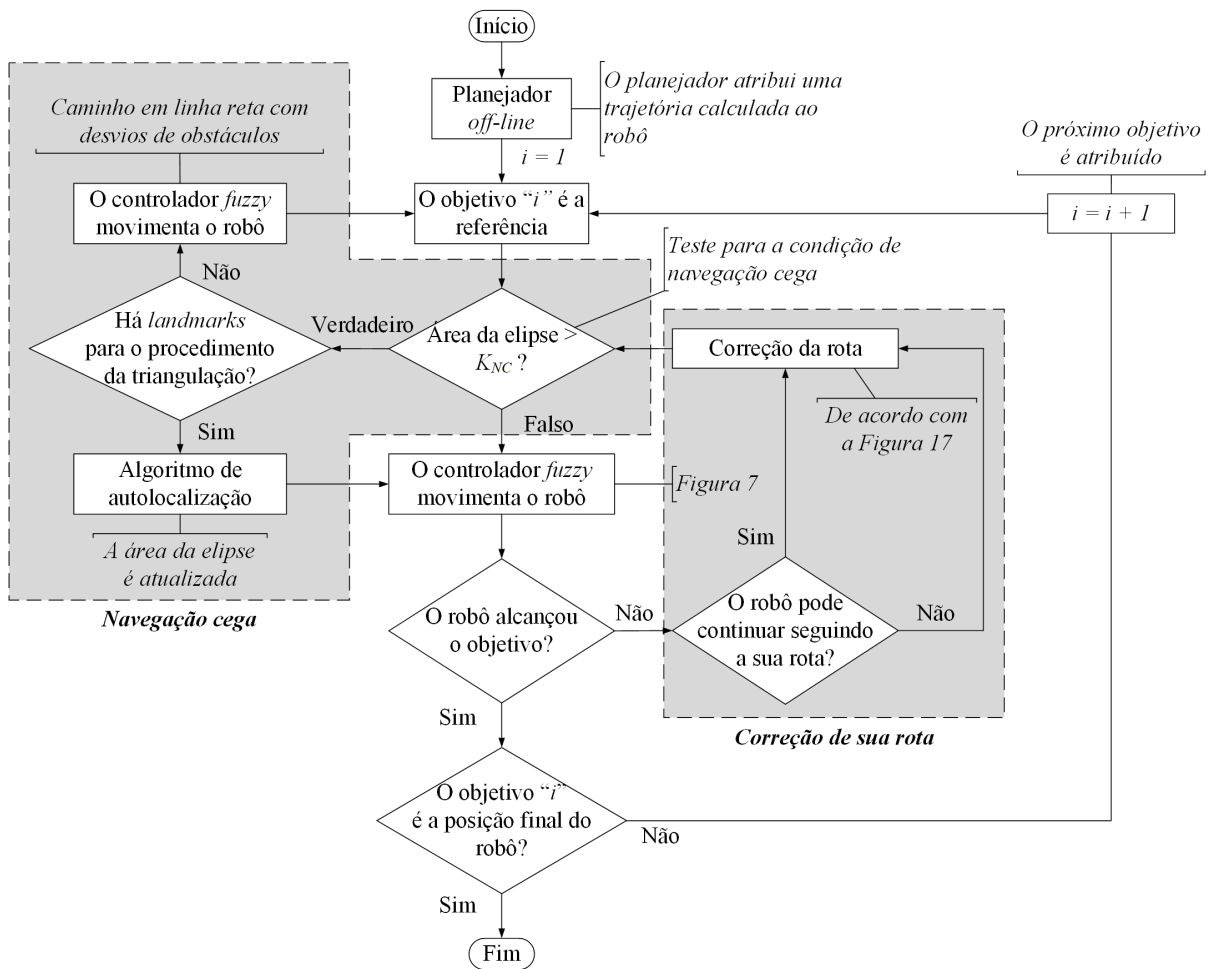


Figura 20: Procedimentos executados pelo planejamento *on-line* de rotas.

Fonte: Autoria própria.

o comportamento dos robôs quando executam o algoritmo de autocalização para corrigir o problema da navegação cega. Esta etapa de avaliação é finalizada ao confirmar se os módulos de planejamento *off-line* e *on-line* compõem, juntamente, um sistema de navegação capaz de levar os robôs a partir de suas posições iniciais, a desviar de obstáculos estáticos e dinâmicos, a visitar todos os alvos conhecidos e a alcançar os pontos finais pré-determinados.

Na terceira e última parte deste capítulo, é apresentado um estudo comparativo incluindo outros dois planejadores *off-line* encontrados na literatura (HAN et al., 2017), que também são baseados em algoritmos genéticos. O primeiro usa um algoritmo genético padrão, enquanto o outro usa um AG com operador de cruzamento dedicado.

O ambiente e os robôs móveis são simulados no *software* V-REP e estão de acordo com a configuração apresentada na Figura 13. São considerados todos os requisitos do sistema básico de navegação descrito no capítulo 3. *Encoders* são acoplados às rodas dos robôs para

fins de localização e possibilitando o cálculo da área da elipse.

4.3.1 VALIDAÇÃO DO PLANEJAMENTO *OFF-LINE* DE ROTAS

O planejador *off-line* proposto é validado por meio de 4 cenários simulados e com dois diferentes tamanhos de ambiente e número de robôs e alvos, mantendo uma razão fixa de 4 alvos por cada robô ($\frac{n_A}{n_R} = 4$) em todos os casos. As características de cada cenário são apresentadas na Tabela 4, onde *SL* corresponde ao tamanho da rota em linha reta que interliga as posições inicial e final de cada robô. Isto significa que o menor caminho para cada robô tem o mesmo comprimento. O cenário 4 é mostrado na Figura 21 para ilustração.

Tabela 4: Características dos cenários simulados para a validação do planejador *off-line* de rotas.

Cenários	Tamanho do ambiente (m)	n_A	n_R	<i>SL</i> (m)
1	10 x 10	20	5	11,32
2	10 x 10	40	10	9,37
3	14 x 14	60	15	14,22
4	14 x 14	80	20	13,36

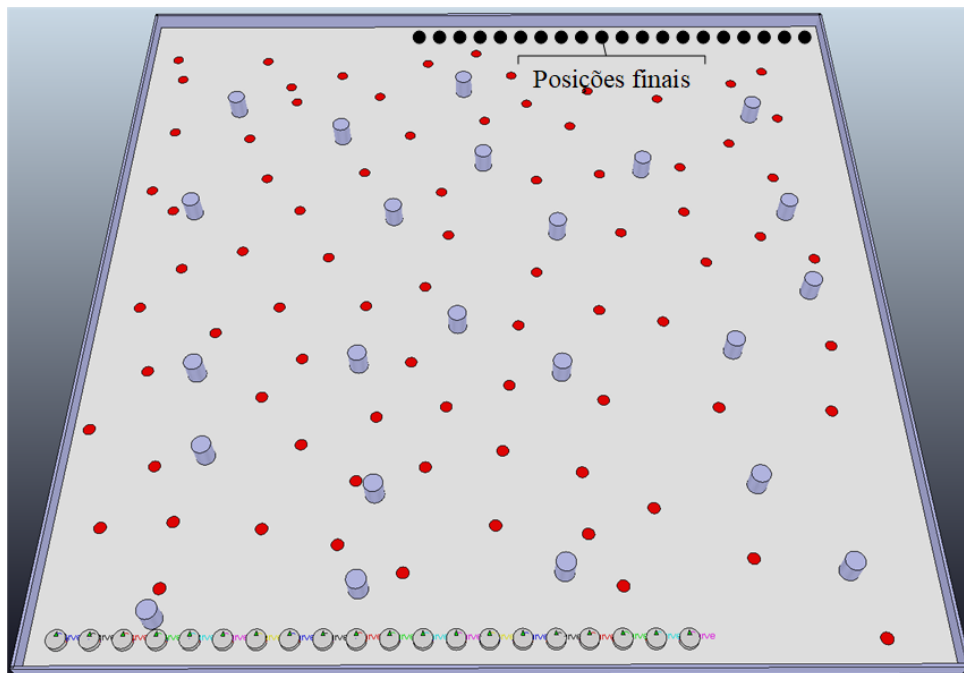


Figura 21: Cenário 4 simulado no V-REP: 20 robôs e 80 alvos.

Fonte: Autoria própria.

Vinte execuções do planejamento *off-line* (algoritmo genético) são considerados para cada cenário da Tabela 4 e a melhor solução obtida entre elas é usada como o conjunto de rotas

planejadas. Os principais parâmetros usados nestas simulações, tais como a população inicial, o critério de parada e o ganho K (Equação (17)), são os mesmos para as 20 execuções e são apresentados na Tabela 5.

Tabela 5: Parâmetros usados no algoritmo genético.

Parâmetro	Cenário 1	Cenário 2	Cenário 2	Cenário 4
População inicial	300	600	2500	4000
Critério de parada (iterações)	500	500	500	500
Ganho K	7	7	25	25

A análise dos resultados e a validação consideram dois objetivos: o primeiro visa analisar as distâncias planejadas, enquanto o outro se concentra na distribuição de alvos entre todos os robôs. Ambos os objetivos são considerados para o mesmo tamanho de ambiente, ou seja, uma comparação entre os cenários 1 e 2 e entre 3 e 4.

Para a análise das distâncias planejadas, os *box-plots* apresentados na Figura 22 sintetizam o desempenho do planejador nos cenários 1 e 2. Em relação a maioria dos dados da distribuição apresentada (intervalo interquartil), é possível observar que o planejador é capaz de atribuir menores distâncias aos robôs, conforme aumenta a densidade dos alvos espalhados no ambiente, como pode ser visto do cenário 1 para o 2. Por exemplo, todos os robôs do cenário 1 apresentam uma rota cujo tamanho é em torno de 14 m , indicando um diferença de apenas 3 m do tamanho da rota usada como referência, que interliga as posições inicial e final dos robôs (coluna SL da Tabela 4). O mesmo ocorre no cenário 2, onde apenas 3 robôs (R2, R5 e R6) recebem caminhos com tamanho superior a 14 m . Além disso, os valores de mediana dos *box-plots* confirmam esta redução, ou seja, estes valores variam de 12,5 a 16,5 no cenário 1 ($SL = 11,32 m$) e de 10,5 a 14 no cenário 2 ($SL = 9,37 m$), com destaque ao robô R3, devido menor dispersão entre as distâncias atribuídas a ele.

Ao contrário desta análise, uma maior variabilidade nas distâncias das rotas atribuídas pode ser observada nos cenários 3 e 4, para os quais existem um número maior de robôs e alvos a serem visitados, como apresentado na Figura 23. Nesse caso, alguns robôs recebem caminhos curtos enquanto outros devem desenvolver longas rotas. Estes *box-plots* também apresentam diversos dados discrepantes (cruz vermelha), indicando soluções com caminhos muito longos. De fato, aumentando o tamanho do ambiente e o número de alvos (n_A), enquanto a mesma razão $\frac{n_A}{n_R} = 4$ é mantida, aumentam-se as possibilidades de solução, como é esperado para um problema de otimização combinatória similar ao MTSP (YUAN et al., 2013; CARTER; RAGSDALE, 2006).

Considerando a melhor solução adotada para cada cenário, a qual apresenta o maior

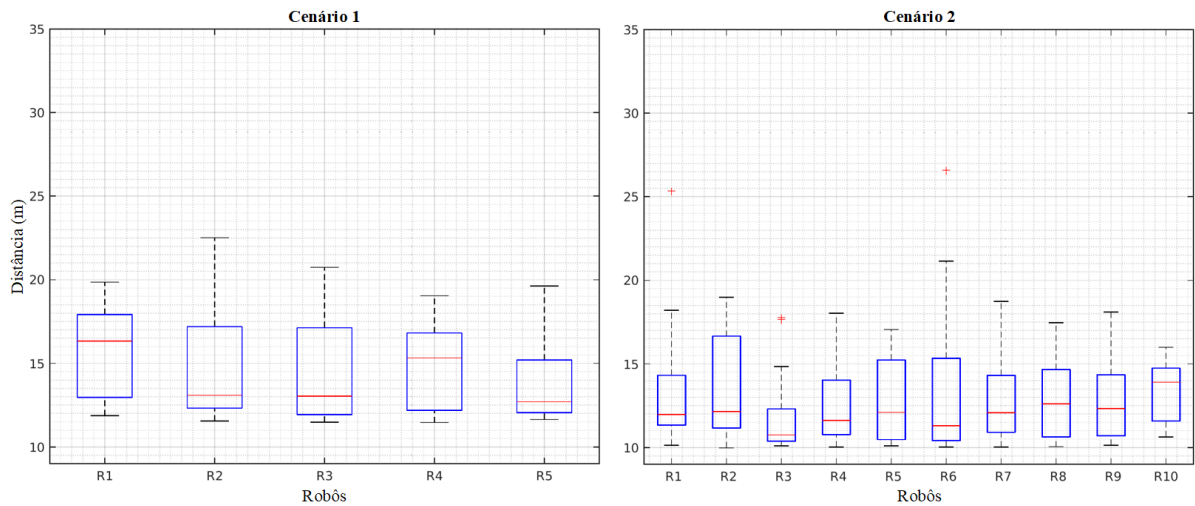


Figura 22: Distâncias planejadas: distribuição das rotas nos cenários 1 e 2.

Fonte: Autoria própria.

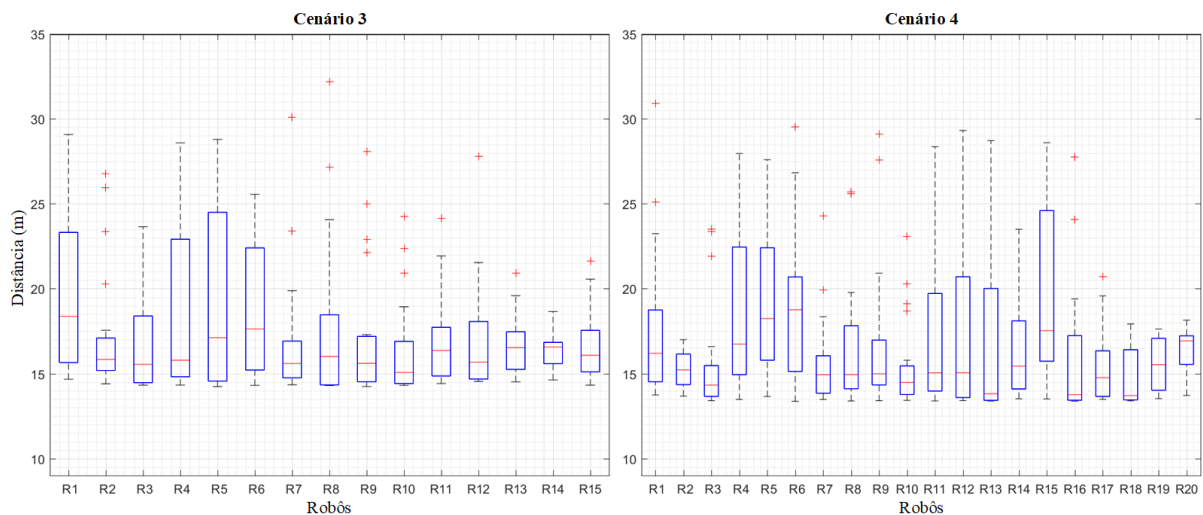


Figura 23: Distâncias planejadas: distribuição das rotas nos cenários 3 e 4.

Fonte: Autoria própria.

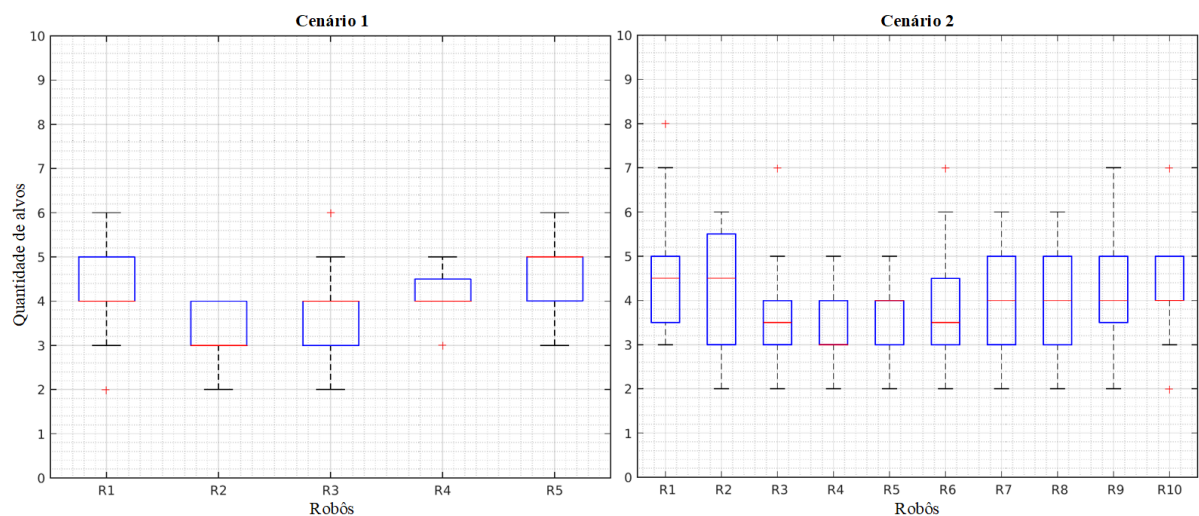
valor da função de aptidão dentre as 20 execuções, os principais índices estatísticos das distâncias planejadas entre todos os robôs são apresentados na Tabela 6. São mostrados os valores de distâncias mínima (Min), máxima (Máx), média, mediana e o desvio padrão.

Em relação a distribuição de alvos entre os robôs, os resultados são mostrados nos *box-plots* das Figuras 24 e 25. Para todos os cenários, a quantidade ideal de alvos por robô é $\frac{n_A}{n_R} = 4$. Assim como ocorreu para as distâncias atribuídas aos robôs, uma distribuição de alvos satisfatória é alcançada nos cenários 1 e 2 (Figura 24) e uma variabilidade na solução é observada nos cenários 3 e 4, com ocorrência de pontos discrepantes (Figura 25). Estas

Tabela 6: Resultados estatísticos para as distâncias planejadas da melhor solução.

Índices	Cenário 1	Cenário 2	Cenário 2	Cenário 4
Min	11,78	10,04	14,30	13,39
Máx	16,33	14,75	25,42	27,60
Média	13,87	11,77	17,09	16,65
Mediana	13,09	11,10	16,20	14,85
Desv. Padrão	2,02	1,68	3,25	4,11

distribuições apresentam uma variação de 3 a 5 alvos nos cenários 1 e 2, de 2 a 7 no cenário 3 e de 2 a 6 no cenário 4.

**Figura 24: Distribuição de alvos entre os robôs nos cenários 1 e 2.**

Fonte: Autoria própria.

É importante destacar que a posição dos alvos no ambiente influencia diretamente a atribuição de alvos, como pode ser notado por meio do exemplo apresentado na Figura 14, onde os alvos das bordas (4 à esquerda e 6 à direita) são melhores atribuídos aos robôs R1 e R5, devido às suas respectivas posições iniciais.

A melhor solução (o total das distâncias planejadas e a distribuição dos alvos) obtida para cada cenário é sintetizada na Tabela 7, onde C1 a C4 referem-se aos cenários de 1 a 4. Como esperado, a distância total aumenta conforme o número de alvos, robôs e tamanho do ambiente também aumentam. Além disso, o número de alvos atribuídos aos robôs varia em torno da razão de referência ($\frac{n_A}{n_R} = 4$) em todos os cenários simulados. Também é possível observar que diversos robôs são associados a distâncias planejadas próximas aos seus respectivos valores de referência SL , enquanto o número de alvos atribuídos a eles é mantido em torno de 4, como ocorre para os robôs R3 e R4 do cenário 1 (exatos 4 alvos), por exemplo.

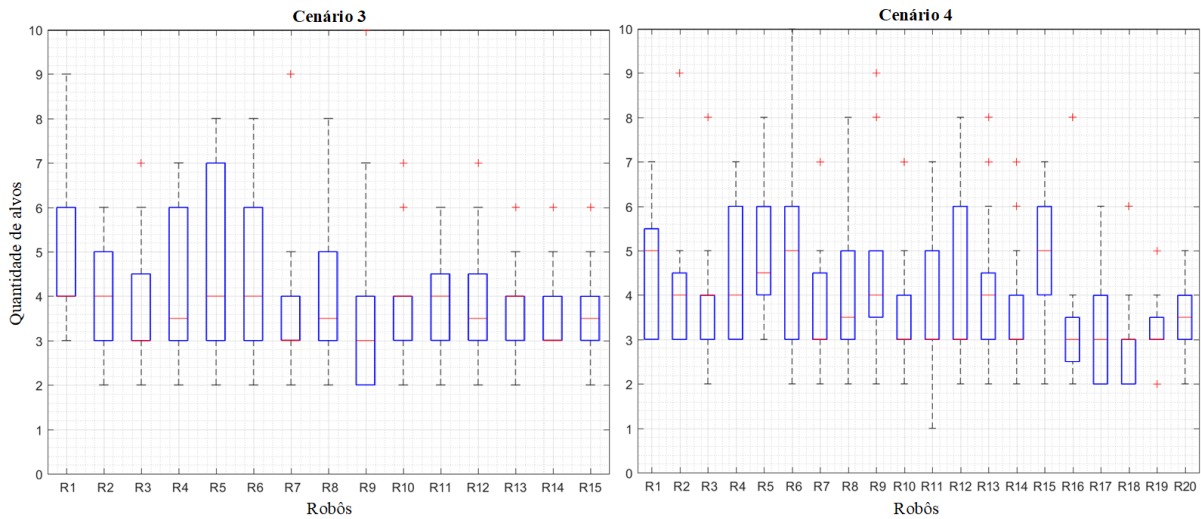


Figura 25: Distribuição de alvos entre os robôs nos cenários 3 e 4.

Fonte: Autoria própria.

Os resultados apresentados nesta seção mostram que o planejador *off-line* é capaz de planejar a rota de todos os robôs do sistema, visando uma distribuição uniforme de distâncias e alvos a serem visitados por eles, mesmo quando há um aumento no tamanho do ambiente, quantidade de robôs e alvos.

Após esta etapa, as rotas otimizadas são atribuídas a cada robô para que a navegação (execução das rotas calculadas) seja iniciada. Todos os robôs devem seguir e recalculas suas rotas, se necessário, de forma a cumprir suas tarefas e finalizar nos seus pontos finais pré-determinados.

4.3.2 DESEMPENHO DO SISTEMA DE NAVEGAÇÃO

Nesta seção é avaliado o desempenho do sistema de navegação proposto, composto pelo módulo de planejamento *on-line*. Três pontos são analisados: (i) a factibilidade das rotas calculadas pelo planejador *off-line* na presença de obstáculos desconhecidos e erros na odometria dos robôs; (ii) a eficácia do planejador *on-line* para redirecionar os robôs às suas rotas calculadas quando ocorre um desvio; (iii) a efetividade do sistema de navegação em desempenhar os objetivos do sistema multirrobo, ou seja, desenvolver um caminho satisfatório entre as posições inicial e final, ambas previamente conhecidas, enquanto fornece condições aos robôs para visitarem pontos de passagens obrigatórias (alvos) no ambiente.

Para esta validação, cada robô do SMR recebe como conhecimento *a priori* (antes de iniciar a navegação) apenas suas rotas calculadas pelo planejador *off-line*, composta pela

Tabela 7: Distâncias planejadas e distribuição de alvos obtidas na melhor solução.

Robôs	Distâncias planejadas				Distribuição de alvos			
	C1	C2	C3	C4	C1	C2	C3	C4
R1	16,33	14,65	23,82	16,72	3	5	6	5
R2	13,09	11,99	14,42	14,74	4	5	3	4
R3	11,78	10,41	18,67	21,91	4	3	4	5
R4	11,90	11,09	15,85	13,51	4	4	4	3
R5	16,23	11,10	25,42	27,60	5	3	5	8
R6	-	10,66	18,53	20,54	-	4	4	6
R7	-	12,92	14,87	13,48	-	5	3	2
R8	-	10,04	14,30	13,41	-	2	3	4
R9	-	10,13	15,06	14,25	-	4	3	4
R10	-	14,75	14,72	19,10	-	5	4	5
R11	-	-	16,32	13,39	-	-	5	3
R12	-	-	14,54	13,56	-	-	3	3
R13	-	-	16,86	25,54	-	-	5	6
R14	-	-	16,76	15,67	-	-	3	3
R15	-	-	16,20	14,11	-	-	4	3
R16	-	-	-	13,44	-	-	-	3
R17	-	-	-	13,72	-	-	-	2
R18	-	-	-	16,99	-	-	-	5
R19	-	-	-	16,42	-	-	-	3
R20	-	-	-	14,95	-	-	-	3
Total	69,33	117,75	256,33	333,04	20	40	60	80

lista ordenada de alvos atribuídos a ele e suas posições inicial e final. As rotas de cada robô são mostradas na Figura 26, com destaque ao robô R8, utilizado para a validação dos pontos mencionados. A posição dos obstáculos estáticos e das paredes do ambiente são desconhecida para todos os robôs.

A factibilidade das rotas (item (i)) é medida pela diferença entre as distâncias planejadas (DP) e navegadas (DN), em percentual. Esse cálculo representa o quanto um robô específico, ou todo o sistema, desviou da rota recebida do planejador *off-line*. O índice DP é obtido pela solução do algoritmo genético (planejador *off-line*), enquanto DN é calculado após o término da navegação. Ambos os índices podem ser analisados sob duas perspectivas: (a) individual: DP e DN de um robô específico; ou (b) coletiva: DP e DN de todo o sistema multirrobô, ou seja, a soma de todas as distâncias individuais. Esta diferença (Dif) é calculada pela Equação (18).

$$Dif(\%) = \left[\left(\frac{DN}{DP} \right) - 1 \right] \cdot 100 \quad (18)$$

Um valor de limiar para a navegação cega, $K_{NC} = 0,6$, é admitido para um robô navegar

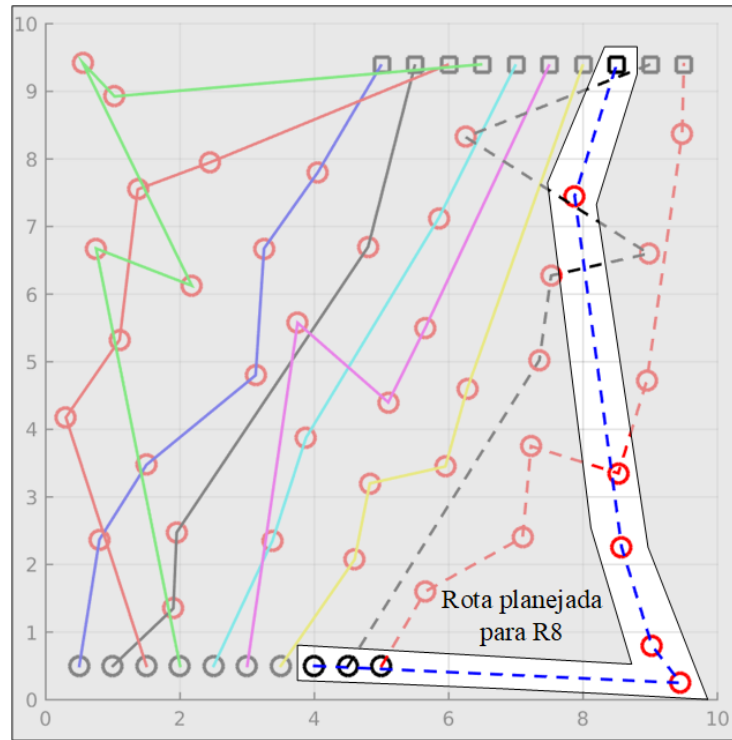


Figura 26: Rotas planejadas para um cenário com $n_A = 40$ e $n_R = 10$.

Fonte: Autoria própria.

no ambiente, o que representa o máximo grau de incerteza acerca de sua posição que o robô pode reter. Se a área da elipse do robô é maior que este limiar (aproximadamente 10 vezes a área ocupada pela estrutura do robô), então o robô se encontra na condição de navegação cega. Além disso, dois diferentes tamanhos da região de detecção dos robôs ($R_{Det} = 1,5 m$ e $R_{Det} = 3 m$ – Figura 6) são considerados para todos os robôs nas simulações, de tal forma a verificar o quanto a habilidade de detecção, para a finalidade de autolocalização proposta, influencia do desempenho da navegação (item (ii)). A escolha de $R_{Det} = 1,5 m$ corresponde a um raio de comunicação 10 vezes maior que a medida de raio da estrutura do robô.

As distâncias planejadas e navegadas são apresentadas na Tabela 8 para cada condição de região de detecção dos robôs. Os valores da coluna *Dif* são calculados de acordo com a Equação (18).

As rotas navegadas no simulador V-REP e com $R_{Det} = 1,5 m$ são apresentadas na Figura 27, onde os caminhos executados por cada robô são diferenciados por cores. O sistema multirrobô alcançou todos os objetivos (alvos e a posição final) atribuídos a cada robô. Comparando tais rotas desempenhadas com as mostradas na Figura 26, cujos valores de DP e DN estão na Tabela 8, é possível observar que a maioria dos robôs navegou conforme planejado, com especial destaque aos robôs R7 (cor amarela em ambas as figuras) e R5 (ciano). Os

Tabela 8: Resultado das distâncias planejadas e navegadas para um ambiente com $n_A = 40$ e $n_R = 10$.

Robôs	DP	$R_{Det} = 1,5 m$		$R_{Det} = 3 m$	
		DN	<i>Dif</i> (%)	DN	<i>Dif</i> (%)
R1	10,42	11,35	8,93	11,32	8,64
R2	10,25	10,85	5,85	10,92	6,54
R3	12,49	13,13	5,12	12,80	2,48
R4	17,69	19,82	12,04	19,32	9,21
R5	10,00	10,10	1,00	10,12	1,20
R6	12,47	12,71	1,92	13,06	4,73
R7	10,52	10,54	0,19	11,42	8,56
R8	14,97	31,50	110,42	16,04	7,15
R9	14,27	14,79	3,64	15,29	7,15
R10	11,80	12,04	2,03	12,05	2,12
Total	124,88	146,83	17,58	132,34	5,97

dois robôs citados desenvolveram exatamente a rota planejada, exceto por um breve desvio de obstáculo. Os desempenhos apresentados confirmam a efetividade do sistema de navegação proposto (item (iii)).



Figura 27: Rotas navegadas com $R_{Det} = 1,5 m$.

Fonte: Autoria própria.

Entretanto, esta comparação também mostra que o robô R8 (rota em azul, destacada em ambas as figuras) realiza desvios excessivos, especialmente no interior da região destacada (quadrado tracejado) da Figura 27. Esta rota irregular é causada pela condição de navegação cega, como pode ser verificada pela análise da evolução da área da elipse deste robô, na Figura 28. Esta condição ocorre frequentemente devido a pequena quantidade de *landmarks*

(alvos e/ou outros robôs) na área destacada, o que dificulta o processo de autolocalização. Um outro fator agravante é o tamanho da região de detecção considerada neste experimento e, conseqüentemente, os três alvos localizados no interior do quadrado tracejado (círculos vermelhos) não estão próximos o suficiente para servir como suporte à triangulação. Neste experimento, R8 consegue se localizar em uma outra região do ambiente (acima da área destacada) para então retornar e visitar os alvos da região em análise.

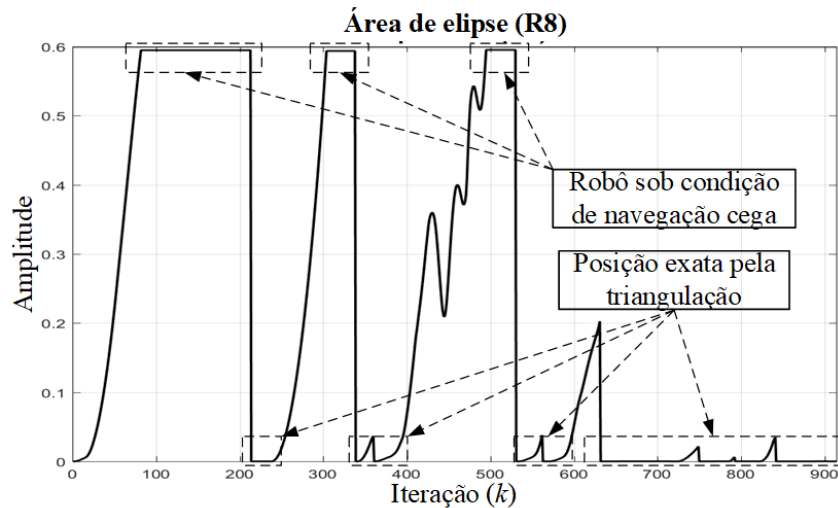


Figura 28: Área da elipse de R8.

Fonte: Autoria própria.

A evolução da variável τ para tomadas de decisão, apresentada na Figura 29, também confirma a situação descrita: $\tau = 0$ quando o robô está sob a condição de navegação cega e, apenas após a execução da triangulação, τ possui um valor alto. Nos outros momentos, os valores de τ decrescem devido a presença de obstáculos (paredes do ambiente), o que requer manobras de desvio.

As ações de controle resultante, aplicadas às rodas do robô (u_D e u_E) e ponderadas pela variável τ , são apresentadas na Figura 30. Quando ocorre a navegação cega e não há iminências de colisão, ambas as rodas do robô recebem a mesma ação de controle ($u_D = u_E = 0,5$) e o robô se move em um caminho reto até encontrar os *landmarks* necessários para sua autolocalização. O longo período sob navegação cega implica em um acréscimo de DN em torno de 110,42% em comparação a DP, conforme é mostrado na Tabela 8 para o robô R8.

De uma forma geral, exceto pela navegação de R8, todos os robôs desta simulação apresentaram um desempenho satisfatório na adaptação de suas rotas, apresentando um acréscimo máximo de DN em relação a DP de 12,04 % (R4 – verde) e mínimo de 0,19 % (R7 – amarelo), ambos também expostos na Tabela 8.

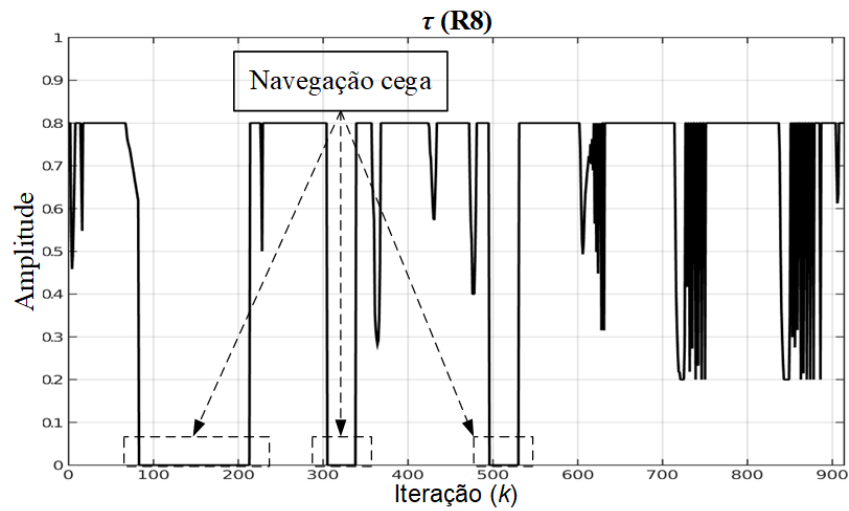


Figura 29: Evolução da variável τ para tomadas de decisão.

Fonte: Autoria própria.

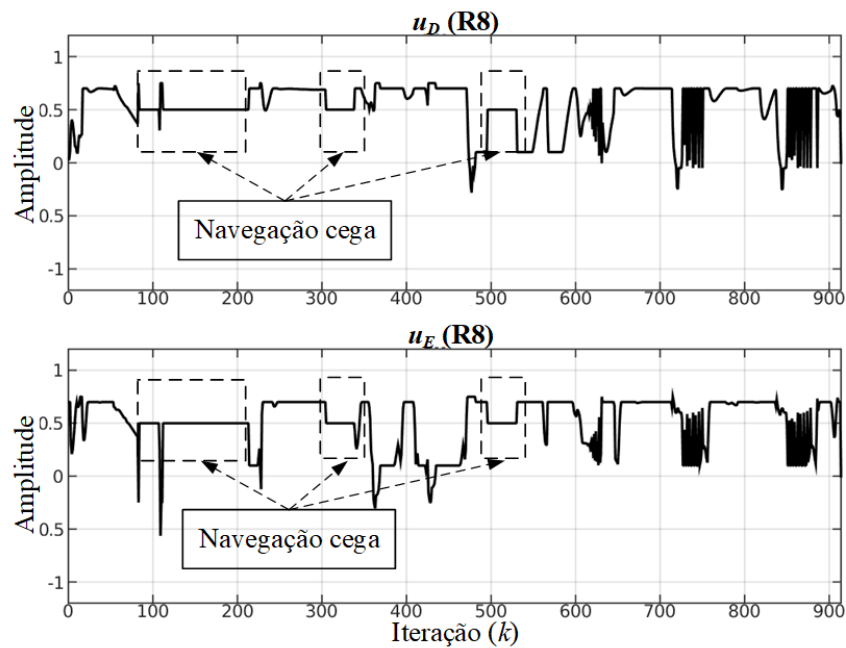


Figura 30: Ações de controle aplicadas ao robô R8.

Fonte: Autoria própria.

Em uma segunda análise, a região de detecção dos robôs é alterada para $R_{Det} = 3 m$, representando uma capacidade do robô em detectar *landmarks* mais distantes. As rotas navegadas sob esta condição são apresentadas na Figura 31.

Ao contrário da simulação anterior ($R_{Det} = 1,5 m$), esta ampliação da capacidade de detecção permite ao robô R8 detectar simultaneamente os três alvos no interior da área

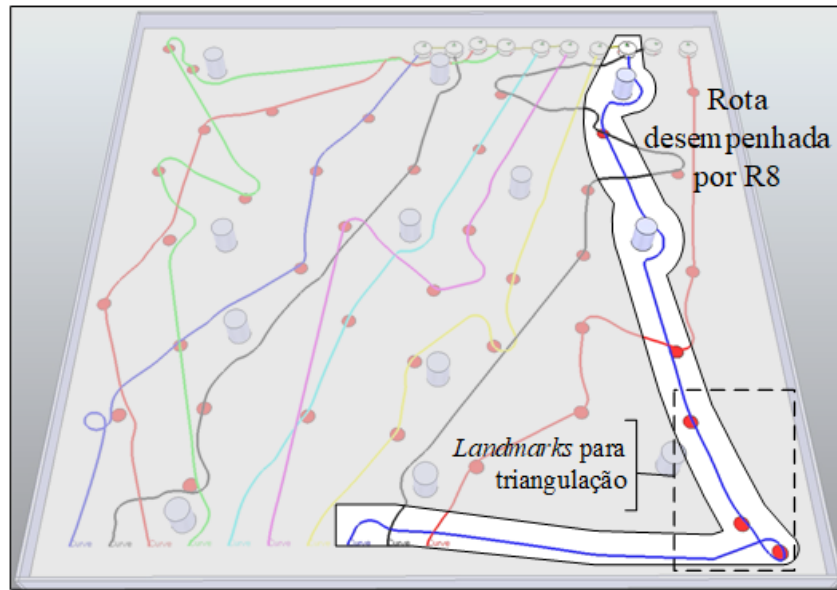


Figura 31: Rotas navegadas com $R_{Det} = 3 m$.

Fonte: Autoria própria.

destacada (quadrado tracejado) e então realizar o procedimento de autolocalização pela triangulação mais rapidamente. Por esta razão, o robô R8 não apresenta dificuldades para desempenhar sua navegação. Este comportamento implica em um decréscimo de 110,42 % (com $R_{Det} = 1,5 m$) para 7,15 % (com $R_{Det} = 3 m$), como mostrado na Tabela 8 e comparado com a simulação anterior.

Embora os robôs R2, R5, R6, R7, R9 e R10 apresentem um pequeno acréscimo em DN quando $R_{Det} = 3 m$, há uma significativa redução no total de DN em relação a DP ($Diff(\%)$), ou seja, de 17,58 % quando $R_{Det} = 1,5 m$ para 5,97 % quando $R_{Det} = 3 m$.

De fato, uma região de comunicação mais ampla (R_{Det}) corresponde a um maior conhecimento sobre o ambiente a ser navegado, a que se reflete no desempenho do planejador *on-line* proposto, induzindo o sistema multirrobô a seguir as rotas calculadas anteriormente sem desvios significativos.

4.3.3 ESTUDOS COMPARATIVOS

Nesta seção, o planejador genético de rotas proposto é comparado com a abordagem apresentada por Han et al. (2017). Neste trabalho comparativo, o planejamento de rotas de múltiplos *Automated Guided Vehicles* (AGV) é executado por um algoritmo genético com aprimoramentos no operador de cruzamento. O principal objetivo do planejador é, de um mesmo ponto inicial, calcular rotas para que todos os AGVs visitem diversos pontos de trabalho

(alvos) de uma forma otimizada e retornem às suas posições iniciais. Este problema é similar ao investigado nesta tese, em que há um grupo de robôs, composto por n_R AGVs (análogos aos robôs móveis) e n_A alvos de posições conhecidas. Além disso, a otimização descrita pelos autores visa minimizar a distância máxima associada a um AGV e também a soma de todas as distâncias planejadas.

Um ambiente de dimensões $10\text{ m} \times 10\text{ m}$, 5 robôs móveis ($n_R = 5$) em um mesmo ponto inicial e 39 alvos ($n_A = 39$) são considerados para este estudo comparativo. A posição de todos os robôs e alvos está de acordo com a descrição dos autores. O planejador genético *off-line*, como proposto nesta tese, é usado para planejar todas as rotas. Além dos resultados apresentados do método proposto pelos autores, eles também demonstram resultados obtidos de um algoritmo genético com o operador de cruzamento padrão, o qual também é usado para esta comparação. Os resultados são concentrados em distâncias planejadas e número de alvos atribuídos a cada robô e estão disponíveis na Tabela 9, organizados em ordem decrescente para facilitar a visualização dos valores mínimo e máximo de cada variável (distâncias planejadas e número de alvos). A coluna “A” se refere aos resultados da estratégia de planejamento proposta nesta tese, enquanto “B” e “C” são relacionadas ao algoritmo genético com e sem especialização do operador de cruzamento, respectivamente (HAN et al., 2017).

Tabela 9: Estudo comparativo referente ao planejador *off-line* de rotas.

Robôs	Distâncias planejadas			Número de alvos		
	A	B	C	A	B	C
R1	23,32	28,14	40,48	9	12	21
R2	22,61	24,24	20,17	9	10	8
R3	21,28	21,28	13,01	9	9	4
R4	20,25	14,91	11,49	8	4	4
R5	11,45	12,86	1,51	4	4	2
Total	98,91	101,43	86,66	39	39	39

Em relação às distâncias planejadas, é possível observar que o planejador proposto nesta tese “A” apresenta o menor valor da distância máxima atribuída a um robô (comparando as distâncias planejadas para R1 na Tabela 9), o que significa uma redução em torno de 17% e 42% em relação aos resultados de “B” e “C”, respectivamente. Além disso, o tamanho das rotas calculadas pelo planejador “A” são semelhantes para todos os robôs, exceto para R5, com uma variação de 20,25 m para R4 a 23,32 m para R1. Essa variação compreende de 14,91 m a 28,14 m para a abordagem “B” e de 11,49 m a 40,48 m para o planejador “C”, também desconsiderando o curto caminho atribuído ao robô R5. Em complemento, a soma de todas as distâncias planejadas entre os robôs (valor total) da abordagem “A” é menor em relação a “B” e maior a “C”.

O número de alvos atribuído a cada robô, que representa a carga de trabalho de cada um, também é melhor distribuído nos resultados de “A” (coluna “Número de alvos” na Tabela 9). Pela estratégia proposta nesta tese (“A”), é atribuído a mesma quantidade (9 alvos) para os robôs R1, R2 e R3, um alvo a menos ao robô R4 (8 alvos) e apenas 4 alvos à R5 (robô com a menor rota a ser desempenhada). Por outro lado, os resultados obtidos pelas abordagens “B” e “C” mostram uma disparidade na quantidade máxima de alvos associados ao robô R1 (12 e 21 alvos, respectivamente), assim como para o robô R5 (4 e 2 alvos, respectivamente).

Em uma análise geral, os resultados apresentados nesta seção confirmam a factibilidade da estratégia proposta de planejamento *off-line*, principalmente quanto à redução da distância a ser navegada por cada robô e também no equilíbrio da distribuição de alvos entre eles. Os resultados comparativos mostram que a estratégia proposta supera as abordagens apresentadas por Han et al. (2017), em todas as métricas comparadas, exceto pelo valor da distância total a ser percorrida pelo SMR.

Portanto, a estratégia de planejamento de rotas para tarefas de visitar alvos conhecidos em um ambiente semi-desconhecido, como proposta neste capítulo, é validada por meio de experimentos que confirmam a capacidade de atribuir caminhos otimizados aos robôs e também fornecer um módulo *on-line* de navegação, para que o SMR seja capaz de lidar com os elementos não previstos na etapa de planejamento *off-line*, tais como o desvio de obstáculos desconhecidos e incertezas na localização dos robôs.

5 PLANEJAMENTO DE ROTAS PARA TAREFAS DE EXPLORAÇÃO DE AMBIENTES

O problema considerado neste capítulo herda as características descritas em (PALMIERI et al., 2018) e (RANGO et al., 2018), dado que os obstáculos estáticos e os pontos de interesse (alvos) estão todos em posições desconhecidas. O desconhecimento da posição destes elementos impede que as rotas sejam computadas previamente por um planejador *off-line* e, assim, o SMR deve possuir as habilidades de exploração, navegação e cooperação para o planejamento de rota *on-line* dos robôs, a fim de que todos os alvos sejam encontrados e os robôs se dirijam em segurança (sem colisão) ao ponto final desejado.

Uma ilustração de um cenário abordado neste capítulo é apresentada na Figura 32: os 5 robôs devem explorar o ambiente em busca dos alvos (círculos vermelhos) e planejar suas rotas localmente, de acordo com as suas detecções. As regiões em cinza representam áreas ainda não exploradas (desconhecidas), enquanto as regiões mais claras representam as áreas conhecidas dos robôs, isto é, já explorada por pelo menos um robô. Embora esta aplicação seja inspirada em um problema de busca por minas terrestres (RANGO et al., 2018), a tarefa de exploração e coleta de alvos pode representar também os problemas que requerem a busca por pontos de reabastecimentos, nós de comunicação, pontos de interesse para a confecção de um mapa do ambiente, entre outros.

A partir do problema apresentado, uma estratégia *on-line* e bioinspirada para planejamento de rotas é proposta neste capítulo, baseada em tarefas cooperativas, comunicação direta e indireta por feromônios artificiais e outros mecanismos de exploração. Cada robô executa o seu próprio planejamento, de acordo com dados obtidos de sensoriamento local, e não há uma coordenação centralizada do SMR, o que caracteriza uma estratégia distribuída. Neste caso, assume-se que todos os robôs sabem a sua posição no ambiente a todo momento da navegação e, portanto, os problemas de incertezas na localização dos robôs, conforme descritos na seção 3.1.5, não são considerados neste capítulo.

O objetivo do planejamento *on-line* proposto é permitir que os robôs tomem decisões autônomas para, de um ponto inicial conhecido, explorar o ambiente de forma cooperativa em

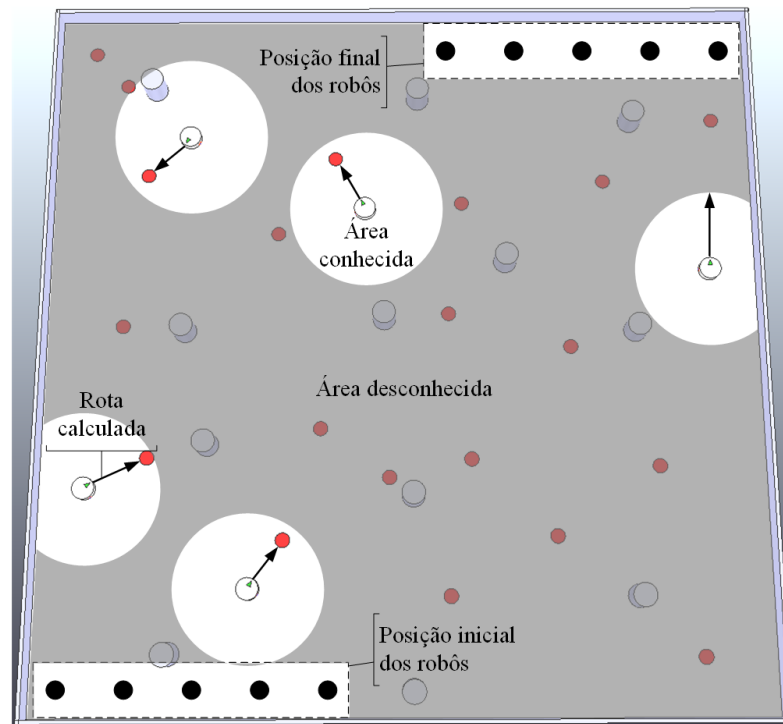


Figura 32: Exemplo de um cenário em que a posição dos alvos e dos obstáculos no ambiente é desconhecida.

Fonte: Autoria própria.

busca dos alvos, de tal forma que os robôs colem aproximadamente a mesma quantidade de alvos (distribuição de alvos) e toda a área do ambiente seja visitada por pelo menos um robô. Assim que esta tarefa é cumprida, os robôs devem se dirigir a um ponto final pré-determinado. Neste problema, apenas as posições inicial e final compõem o conhecimento *a priori* dos robôs.

Este capítulo está organizado da seguinte forma: na seção 5.1 são apresentados os mecanismos de comunicação direta e indireta dos robôs, necessários para o desenvolvimento do planejador *on-line*. Em seguida, a estratégia de planejamento desenvolvida é descrita na seção 5.2. Os resultados simulados, utilizados para a validação desta proposta, são apresentados na seção 5.2, incluindo um estudo comparativo com outros trabalhos encontrados na literatura.

5.1 REQUISITOS DE COMUNICAÇÃO

A estratégia proposta neste capítulo requer que os robôs possuam todas as habilidades de navegação descritas no capítulo 3 (habilidades de detecção/comunicação e locomoção, exceto a autolocalização, como explicada anteriormente). Para a locomoção, o controlador *fuzzy* será o mesmo do capítulo 4 e, necessita-se ainda de um aprimoramento no mecanismo de comunicação direta para o compartilhamento de informações.

5.1.1 COMUNICAÇÃO DIRETA E COMPARTILHAMENTO DE INFORMAÇÕES

O compartilhamento de informações através da comunicação direta ocorre sempre que um ou mais robôs estejam próximos o suficiente para desempenhar esta tarefa, ou seja, no interior da região de detecção/comunicação R_{Det} (definido no capítulo 3). Na estratégia proposta neste capítulo, os robôs envolvidos nesta interação compartilham a sua lista de alvos detectados e já visitados (informação sobre a posição destes alvos), com o objetivo de estabelecer um comportamento cooperativo na distribuição de alvos.

Uma característica transitiva é adicionada ao compartilhamento de informações mencionado, com a seguinte funcionalidade: a detecção de um alvo por um robô pode ser compartilhada diretamente para outro robô na sua região de detecção e, este segundo robô repete a informação a um terceiro membro da sua região de detecção, que pode estar distante do primeiro (fora da região de detecção), inviabilizando assim uma comunicação direta entre estes.

Um exemplo deste compartilhamento de informações é ilustrado na Figura 33, onde: (i) o alvo A1 é, de fato, detectado pelo robô R1, compartilhado com R2 por comunicação direta e replicado para R3 via transitividade (R2 informa R3 por comunicação direta); (ii) A3 é detectado por R2 e R3 e compartilhado com R1 por meio de R2; (iii) A4 é detectado por R3, que compartilha com R2 por meio de comunicação direta e com R1 por transitividade; e (iv) A2 não é detectado por nenhum dos robôs e, portanto, o SMR não tem a informação de sua existência.

5.1.2 COMUNICAÇÃO INDIRETA E ESTIGMERGIA

A habilidade de comunicação indireta por meio de feromônios artificiais, como descrita no capítulo 3, é usada nesta proposta de planejamento de rota como um mecanismo que permite a exploração do ambiente pelo SMR.

O principal objetivo do uso dos feromônios artificiais nesta estratégia é marcar temporariamente o caminho percorrido por um robô no ambiente, de tal forma que outro membro do SMR possa identificá-la e assim ter o conhecimento de que outro robô já explorou esta determinada área, sem a necessidade de ocorrer a comunicação direta. Para isso, o lançamento de feromônio no ambiente ocorre no interior de uma região circular, denominada $Area_F$, cujo centro coincide com o da estrutura do robô, como pode ser observado na Figura 34. A intensidade do sinal lançado é forte no centro desta área e decai conforme se aproxima das bordas da região circular, assumindo valores no intervalo de 0 (sinal fraco ou nulo) a 1 (sinal

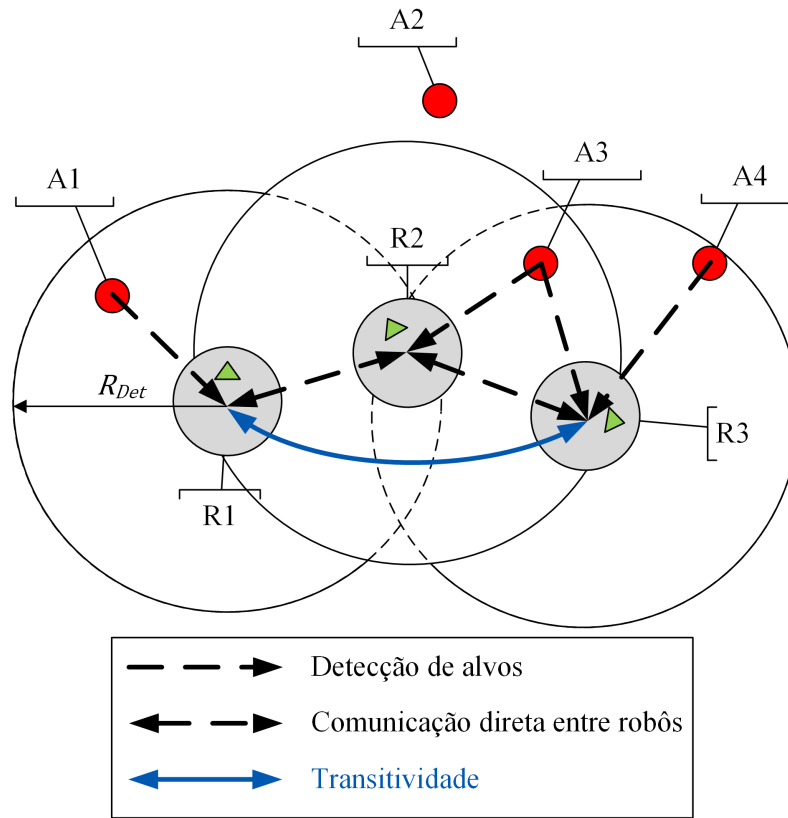


Figura 33: Ilustração da dinâmica de compartilhamento de informações.

Fonte: Autoria própria.

forte). A evaporação é representada pelo esmaecimento da cor vermelha, que representa a trilha deixada pelo robô em movimento.

Dessa forma, a Equação (1) é reescrita conforme Equação (19), onde $Dist_F$ é a distância entre o ponto (x, y) a ser atualizado e o centro do robô que lançou o feromônio e $Range_F$ é o raio da região circular de lançamento do feromônio no ambiente.

$$\sigma_{x,y}(k) = \begin{cases} (1 - \rho) \cdot \sigma_{x,y}(k-1) + \left[1 - \left(\frac{Dist_F}{Range_F} \right) \right], & \text{se } Dist_F \leq Range_F \\ (1 - \rho) \cdot \sigma_{x,y}(k-1), & \text{do contrario} \end{cases} \quad (19)$$

A partir da Equação (19), a movimentação de um robô resulta na trilha de feromônios mostrada na Figura 34. Esta figura mostra a evolução do robô nos instantes: $k = 0$, $k = 1$ e $k = 2$. Considerando que nenhum outro robô passou por esta área ($\sigma_{x,y}^{inicial} = 0$), no instante $k = 0$, $\sigma_{x,y}(0) > 0$ em todos os pontos (x, y) que estão no interior de $Area_F$ e $\sigma_{x,y}(0) = \sigma_{x,y}^{inicial} = 0$ para as outras coordenadas. O robô se movimenta e lança feromônio em uma nova área, no instante $k = 1$, enquanto os sinais lançados anteriormente (caminho executado de $k = 0$ a $k = 1$)

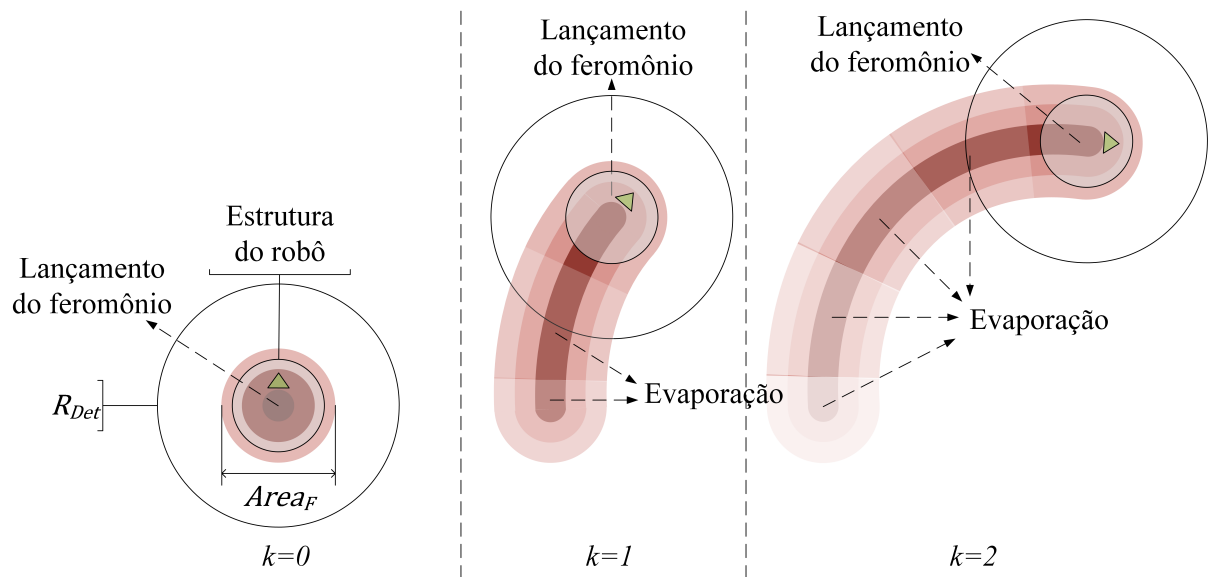


Figura 34: Trilha de feromônios.

Fonte: Autoria própria.

sufrem evaporação, de acordo com a Equação (19). Uma nova área é marcada em $k = 2$ e todos os sinais anteriores continuam sendo evaporados, mais uma vez nas áreas marcadas de $k = 0$ a $k = 1$ e pela primeira vez de $k = 1$ a $k = 2$. Esta evolução ocorre durante a navegação de todos os robôs móveis do SMR.

A intensidade do feromônio lançado no ambiente, e percebido pelos robôs, caracterizam o traço de informação por estigmergia. Quando um sinal forte é detectado em uma determinada região ($\sigma_{x,y}(k) \approx 1$), significa que um outro robô passou por esta área e ainda está próximo. Do contrário, quando o sinal detectado é fraco ($\sigma_{x,y}(k) \approx 0$), o robô que registrou este caminho pode estar distante desta região. Um exemplo desta detecção é ilustrada na Figura 35.

Tais informações são úteis para a estratégia de planejamento *on-line* proposta neste capítulo, já que os robôs podem ter informações de áreas exploradas por outros robôs sem a necessidade de haver, de fato, uma comunicação direta entre os membros do SMR. Dessa forma, os robôs entendem que as áreas já foram exploradas e decidem sobre suas novas rotas.

5.2 PLANEJADOR DE ROTAS *ON-LINE* E BIOINSPIRADO

Com base nas habilidades apresentadas, o planejador proposto é executado individualmente por cada robô e uma decisão acerca de sua tarefa de exploração é tomada em

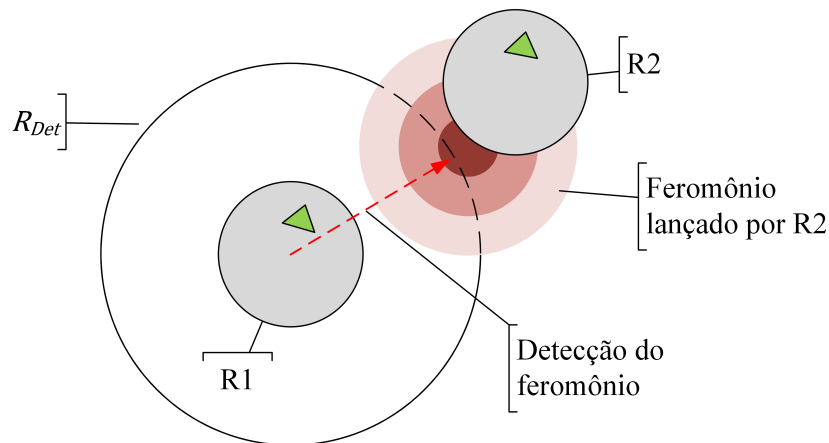


Figura 35: Comunicação indireta: o robô R1 detecta o feromônio lançado por R2.

Fonte: Autoria própria.

uma abordagem distribuída, durante a navegação. O processo de tomada de decisão de um robô envolve as seguintes ações: visitar um alvo detectado pelo seu próprio sistema sensorial, ou por informação compartilhada de um outro robô (D_1); explorar o ambiente em busca de alvos ainda não encontrados (D_2); ou se dirigir ao seu ponto final para finalizar sua navegação (D_3). O cálculo da rota, resultante de uma decisão (D_1 ou D_2 ou D_3), pode ser modificado para evitar áreas já exploradas por outros robôs (D_4), ou seja, áreas com a presença de feromônios. Neste caso, a correção de rota é suave. No entanto, afim de evitar colisão com obstáculos, uma rota pode ser fortemente modificada (D_5).

O planejador é implementado em duas camadas hierárquicas, que englobam as decisões deliberativas D_1 , D_2 , D_3 e D_4 na primeira camada e, a decisão reativa D_5 , juntamente com o controle de movimentação, na segunda camada. A primeira tem o objetivo de calcular a rota a ser navegada pelo robô, de acordo com o resultado da tomada de decisão entre D_1 , D_2 e D_3 , e também realizar os ajustes devido à detecção de feromônios (D_4). Tais procedimentos combinam heurísticas cooperativas e algoritmo ACO. A segunda camada corresponde ao controlador *fuzzy*, descrito na seção 3.1.4, que recebe a rota calculada na camada superior e a implementa ou a corrige, em face de uma iminência de colisão (D_5). Quando o robô manobra para evitar a colisão com os obstáculos detectados, a rota planejada é suspensa e retomada assim que a iminência de colisão for amenizada.

5.2.1 CAMADA DE PLANEJAMENTO INTELIGENTE: DECISÕES DELIBERATIVAS

A camada de planejamento inteligente é desenvolvida com base no conhecimento heurístico acerca do comportamento desejado do robô na presença de obstáculos, alvos, sinais de feromônios e/ou outros robôs. A rota é planejada em duas etapas, como sintetizado na Figura 36: (i) um objetivo (posição (x,y)) é obtido da ativação de D_1 , ou D_2 ou D_3 (apenas uma das decisões é ativada) e um caminho é gerado na forma de um vetor que interliga a atual posição do robô e seu objetivo; e (ii) o caminho gerado pode ou não ser ajustado devido a presença de feromônio ao longo do vetor calculado (D_4).

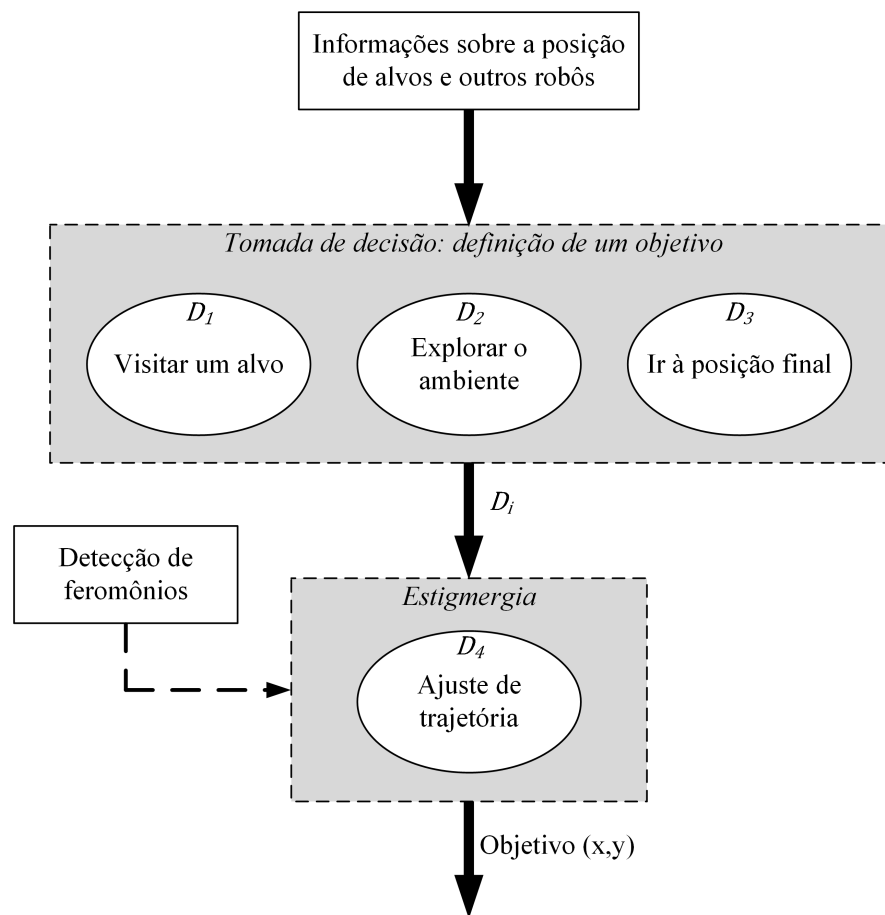


Figura 36: Tomada de decisão: tarefas e planejamento de rota.

Fonte: Autoria própria.

É importante enfatizar que a correção de rota realizada na segunda etapa é diferente da ação reativa D_5 para a prevenção de colisão, que pode mudar completamente o objetivo do robô. D_4 ajusta a rota resultante de D_1 , D_2 ou D_3 e assegura um desvio necessário de áreas já exploradas por outros robôs, mas mantém a rota o mais próximo possível do objetivo calculado. Por esta razão, o desvio de feromônio é executado no nível de planejamento, e não no nível do

controlador *fuzzy*.

As decisões são ativadas de acordo com um conjunto de variáveis heurísticamente idealizadas (h_i , $i = 1, 2, 3, 4$). Variáveis auxiliares que envolvem a distância já navegada pelo robô, a quantidade de alvos já visitados e a intensidade de feromônio em uma posição do ambiente também são consideradas para tomar essas decisões. Um valor binário é usado para representar a ativação de uma decisão ($D_i = 0$: a decisão i não é ativada; $D_i = 1$: a decisão i é ativada).

De uma forma geral, é desejável que cada robô visite os alvos detectados no interior de sua região de detecção, portanto D_1 possui prioridade sobre D_2 e D_3 . Entretanto, há um impasse quando dois ou mais robôs detectam o mesmo alvo (identificação do alvo por transitividade ou de fato detectado pelo robô). Nesse caso, é aplicada uma política que visa uma distribuição uniforme de alvos visitados entre os robôs. O robô que visitou a menor quantidade de alvos é favorecido. O algoritmo heurístico utilizado para a atribuição de alvos a um robô é apresentado no diagrama da Figura 37, onde os principais comentários (em itálico) são incluídos para uma melhor compreensão. A variável auxiliar $h_1 \in \{0, 1\}$ é um indicador se o robô avaliado encontrou e teve um alvo atribuído a ele pelo algoritmo apresentado na Figura 37. Quando $h_1 = 1$, a decisão D_1 é tomada.

A tarefa de exploração (decisão D_2) envolve a análise de três situações: o robô não possui um alvo associado a ele ($h_1 = 0$); está longe de sua posição final; e visitou poucos alvos até o momento. A decisão D_2 tem prioridade sobre D_3 e é associada à variável auxiliar $h_2 \in [0, 1]$, definida na Equação (20), onde: $\gamma \in [0, 1]$ é um fator de ponderação variável, η é a relação entre o número de alvos visitados pelo robô e seu limite de carga (Lim_{alvos}) e $h_3 \in [0, 1]$ é distância linear normalizada entre as posições atual ($P_k(x, y)$) e final do robô ($P_{final}(x, y)$), dadas na Equação (21).

$$h_2 = (1 - \gamma) \left[1 - (1 - h_3)^{1 - \eta} \right] \quad (20)$$

$$h_3 = \frac{\|P_k(x, y) - P_{final}(x, y)\|}{\|P_{final}(x, y)\|} \quad (21)$$

O fator γ é introduzido na Equação (20) para prevenir que o robô permaneça explorando por um tempo excessivo, principalmente quando não há mais alvos a serem encontrados. Seu valor inicial é zero e é incrementado por γ^+ toda vez que a decisão D_2 é ativada. Além disso, a quantidade de alvos visitados, de acordo com η na Equação 20, contribui com um decaimento para a ativação de D_2 .

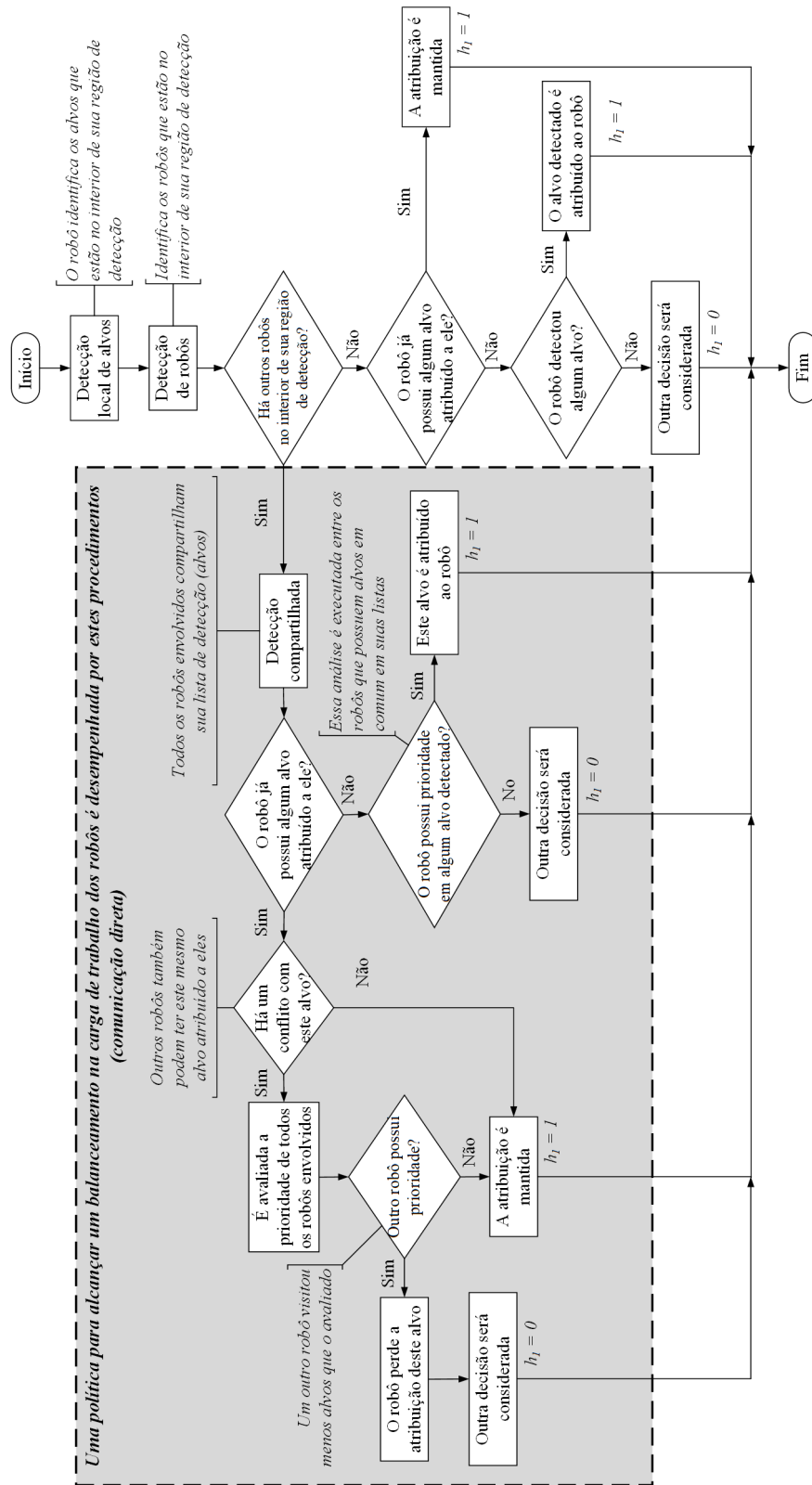


Figura 37: Fluxograma do algoritmo heurístico de atribuição de alvos.

Fonte: Autoria própria.

Quando um robô decide explorar o ambiente (ativação de D_2), os seguintes procedimentos são executados: (i) verifica-se se há outros robôs no interior de sua região de detecção, mais precisamente se outros robôs são detectados à esquerda ou à direita de uma rota linear entre a sua posição atual até o ponto final de sua navegação; (ii) se há um ou mais robôs no lado direito, uma rota secundária é calculada pelo robô, permitindo-lhe explorar uma área a sua esquerda. Nesse caso, o vetor que interliga as posições atual e final do robô é rotacionado por $+\alpha_{Exp} \text{ rad}$, e então seu novo objetivo será a coordenada apontada por este vetor rotacionado. O mesmo comportamento é desempenhado quando outros robôs são detectados a sua esquerda, entretanto o ângulo de rotação é $-\alpha_{Exp} \text{ rad}$; (iii) se há outros robôs detectados em ambos os lados, o robô avaliado segue em direção a sua posição final. Os robôs memorizam suas detecções sempre que outros robôs são detectados; e (iv) se não há outros robôs detectados, a última memória de detecção é verificada e a decisão é tomada de acordo com esta informação. Se o robô não detectar quaisquer outros robôs durante a navegação, ele segue em direção a sua posição final, como descrito no passo (iii).

Este mecanismo de decisão para a exploração do ambiente é ilustrado na Figura 38. Os robôs R1 e R3 tomam a decisão de acordo com o passo (ii), a fim de explorar as áreas a esquerda e a direita, respectivamente; R2 segue o passo (iii); e a decisão de R4 depende de sua memória de detecção. Quando R1 calcula sua rota secundária (linha azul tracejada), o conhecimento sobre o tamanho total do ambiente sob exploração é definido pela área cinza retangular (Área 1). O mesmo ocorre para o robô R3, que deve explorar a área retangular verde (Área 2). Além disso, quanto mais perto o robô está de sua posição final, menor é a área a ser explorada, desde que esta decisão seja tomada.

A decisão D_3 é tomada quando D_1 e D_2 não são ativadas. Nesse caso, o robô se move em direção a sua posição final. Ao contrário da decisão de explorar o ambiente, é desejado que um robô tome esta decisão quando ele estiver próximo de sua posição final e/ou quando ele já visitou uma quantidade suficiente de alvos durante a navegação.

Assim que uma das decisões descritas é tomada, uma tarefa é atribuída: visitar um alvo detectado (D_1), explorar uma área (D_2) ou se dirigir à posição final (D_3). Com isto, um objetivo (x, y) é definido e uma rota é calculada (uma linha reta que interliga a posição atual do robô e o objetivo pretendido).

Após a atribuição de tarefa, o robô analisa se será necessária uma manobra para o desvio temporário do objetivo definido anteriormente (decisão D_4), devido à presença de feromônios (calculado pela Equação (19)) no interior de sua região de detecção, que significa uma área já explorada por outro robô.

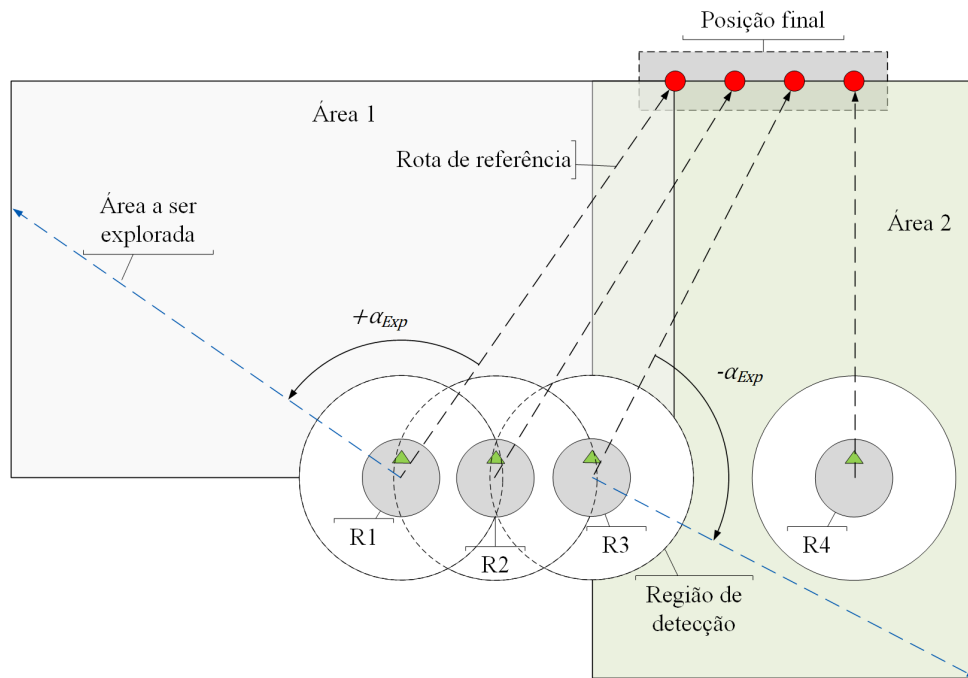


Figura 38: Ilustração da decisão de explorar o ambiente.

Fonte: Autoria própria.

Para a correção da rota, o robô verifica se a região com a maior intensidade de feromônio está à esquerda ou à direita de seu caminho planejado e executa um procedimento similar ao cálculo da área a ser explorada na decisão D_2 . Neste caso, uma rota secundária é calculada pela rotação do vetor que interliga as posições atual e do objetivo do robô, podendo ser um alvo, uma área a ser explorada ou sua posição final. Esta rotação corresponde a um ângulo $\pm\alpha_F$, proporcional à intensidade do feromônio detectado, como exemplificado na Figura 39, onde o robô R1 detecta o feromônio lançado por R2 e executa uma correção de rota. Este comportamento representa a região com feromônios como um campo repulsivo.

Entretanto, a decisão sobre a correção da rota calculada também depende da distância entre as posições atual e final do robô, representado pela variável h_3 na Equação (21). Quando um robô está longe de sua posição final e um sinal de feromônio é detectado, a variável auxiliar $h_4 \in \{0, 1\}$ indica se o robô detectou um sinal de feromônio.

A tomada de decisão na camada deliberativa é de fato realizada na Equação (22), a partir da análise das variáveis auxiliares mencionadas anteriormente, em que ζ_{Exp} e ζ_F são limiares de decisão que regem a importância relativa entre as tarefas D_2 e D_3 , e D_4 , respectivamente.

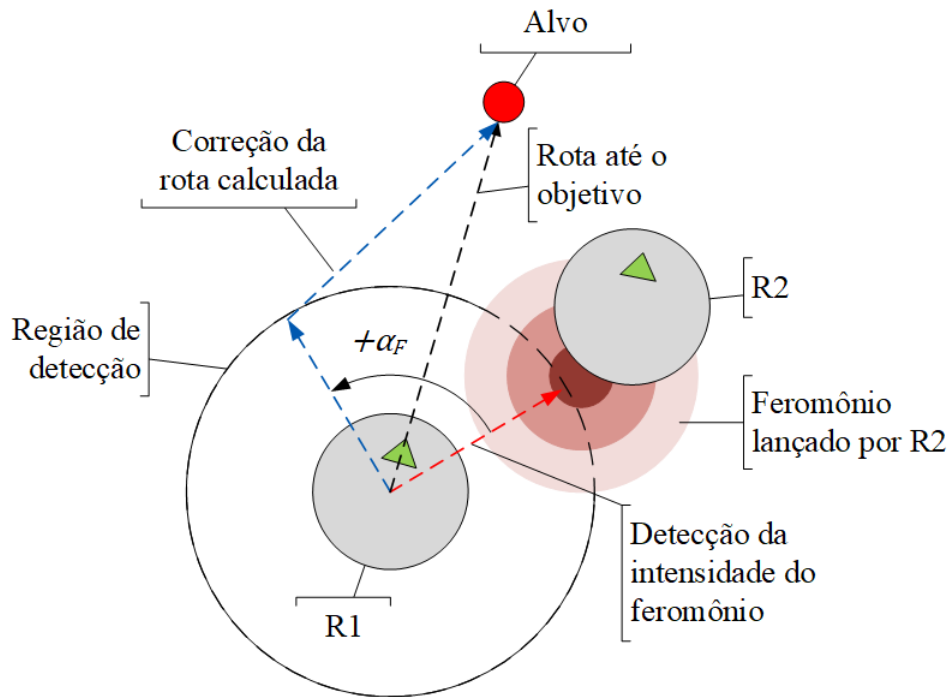


Figura 39: Correção da rota devido a presença de feromônio.

Fonte: Autoria própria.

$$\begin{aligned}
 D_1 &= \begin{cases} 1, & \text{se } h_1 = 1 \\ 0, & \text{caso contrário} \end{cases} \\
 D_2 &= \begin{cases} 1, & \text{se } (1 - h_1) h_2 \geq \zeta_{Exp} \\ 0, & \text{caso contrário} \end{cases} \\
 D_3 &= \begin{cases} 0, & \text{se } 0 \leq (1 - h_1) (1 - h_2) < \zeta_{Exp} \\ 1, & \text{caso contrário} \end{cases} \\
 D_4 &= \begin{cases} 1, & \text{se } h_3 \cdot h_4 \geq \zeta_F \\ 0, & \text{caso contrário} \end{cases}
 \end{aligned} \tag{22}$$

Convém ressaltar que a decisão D_4 não modifica o objetivo (coordenada (x,y)) a ser visitado pelo robô, que é resultado das decisões D_1 , D_2 ou D_3 (escolha da tarefa), ela apenas corrige a rota calculada até este ponto. Portanto, o resultado final deste planejamento deliberativo é um objetivo (alvo, área a ser explorada, ou posição final) e uma rota com um ou dois segmentos (causada pela presença de feromônios) para alcançar esta coordenada.

5.2.2 CAMADA REATIVA: DECISÕES DE NAVEGAÇÃO

As decisões de navegação são tomadas pelo sistema *fuzzy* que, nesta estratégia, corresponde a camada reativa dos múltiplos robôs. Baseado na rota planejada, este sistema (descrito na seção 3.1.4) promove o movimento dos robôs em direção aos seus objetivos enquanto evita colisão com obstáculos estáticos e com outros robôs.

O comportamento de desvio de obstáculos, calculado pelo sistema de navegação, corresponde à decisão reativa D_5 . Esta manobra de segurança se sobrepõe nas decisões tomadas pela camada deliberativa (D_1 a D_4), ou seja, mesmo se o robô já completou o ajuste de rota ocasionado pela detecção de feromônios, ele ainda pode fazer um ajuste adicional, que se refere à manobra de desvio de obstáculos, necessária para preservar sua estrutura física.

5.3 RESULTADOS SIMULADOS

Nesta seção, o planejador de rotas *on-line* é validado por meio de experimentos de simulação. Inicialmente, o planejador é testado com somente 5 robôs. As tomadas de decisão da primeira camada (decisões deliberativas) e os controladores de navegação para a decisão reativa (segunda camada) são validados. Também é analisado se a integração entre as duas camadas é suficiente para que um robô seja capaz de planejar satisfatoriamente uma rota para navegar no ambiente de um ponto inicial a um ponto final, enquanto desvia de obstáculos e visita os alvos desconhecidos. Um segundo conjunto de experimentos é realizado para analisar o comportamento global do sistema multirrobô, de acordo com métricas definidas para a avaliação do custo de exploração, eficácia, eficiência, entre outras. Estes indicadores são usados para avaliar o desempenho do SMR em quatro cenários diferentes, com diferentes configurações acerca do número de robôs, tamanho do ambiente, quantidade de alvos e obstáculos estáticos. A análise mencionada fornece um melhor entendimento sobre o comportamento cooperativo induzido pela estratégia proposta de planejamento de rotas. O desempenho do SMR é também analisado em cenários em que todos os alvos são concentrados em algumas áreas específicas do ambiente. Finalmente, são apresentados estudos comparativos com abordagens similares da literatura.

5.3.1 VALIDAÇÃO DO PLANEJAMENTO DE ROTAS BIOINSPIRADO

Um pequeno ambiente ($10\text{ m} \times 10\text{ m}$) com apenas 5 robôs e 20 alvos é usado para validar o mecanismo de decisão e, individualmente, o planejamento de rotas. A representação simulada deste ambiente, por meio do *software* V-REP, é apresentada na Figura 40. Este

ambiente de pequenas proporções auxilia na análise dos caminhos desempenhados por cada robô durante a navegação, assim como as decisões consideradas por cada um.

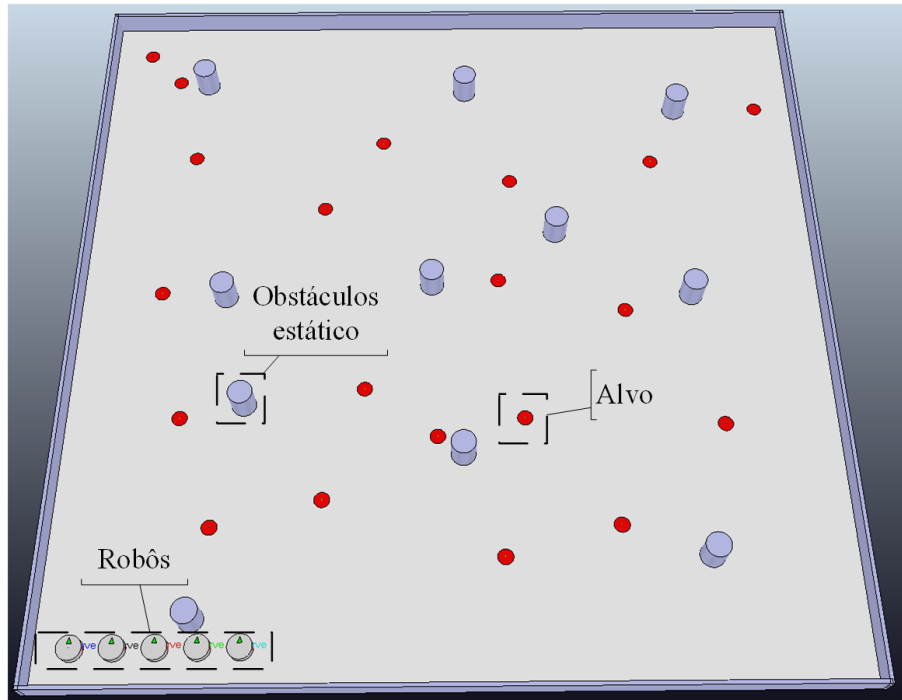


Figura 40: Ambiente no simulador V-REP.

Fonte: Autoria própria.

Os principais parâmetros usados nas simulações são apresentados na Tabela 10. As características estruturais dos robôs são determinadas pelo *hardware* descrito no capítulo 3. Os valores de limiar de decisão (ζ_{Exp} e ζ_F) são ajustados de acordo com a influência desejada na navegação. Os outros parâmetros são heurísticamente ajustados após diversos experimentos. Por uma questão de clareza, os obstáculos estáticos são omitidos nas figuras de resultados, entretanto, estes estão presentes em todas as simulações e influenciam na rota dos robôs.

Inicialmente, os mecanismos de decisão para as tarefas “visitar um alvo” (D_1) e “dirigir-se à posição final” (D_3) são avaliados, ou seja, o planejador de rotas é executado sem a decisão de explorar o ambiente (D_2) e de evitar sinais de feromônio (D_4). Tais considerações são necessárias para validar o funcionamento da camada reativa, implementada pelo sistema *fuzzy*. Os resultados obtidos são apresentados na Figura 41, onde os círculos pretos são as posições iniciais, os vermelhos são os alvos, os quadrados pretos são as posições finais e as rotas executadas por cada robô são diferenciadas por cor. Além disso, as distâncias navegadas por cada robô, associadas à sua respectiva rota, são mostradas na caixa superior direita da figura.

Como esperado, todos os robôs alcançaram suas respectivas posições finais após

Tabela 10: Parâmetros da simulação.

Parâmetro	Valor
Região de detecção (R_{Det})	1,5 m
Região de lançamento do feromônio ($Area_F$)	0,3 m (diâmetro)
Taxa de evaporação do feromônio (ρ)	0,15
Limite de alvos por robô (Lim_{alvos})	8
Incremento no fator de desconto γ (γ^+)	$\gamma^+ = 0,015$ e $\gamma^+ = 0,03$
Limiar para a ativação de D_2 (ζ_{Exp})	$\zeta_{Exp} = 0,15$
Ângulo de rotação na decisão D_2 ($\pm\alpha_{Exp}$)	$\pm\alpha_{Exp} = \pi/2$ rad
Limiar para a ativação de D_4 (ζ_F)	$\zeta_F = 0,3$
Ângulo de rotação na decisão D_4 ($\pm\alpha_F$)	$\pm\alpha_F = \pi/2$ rad

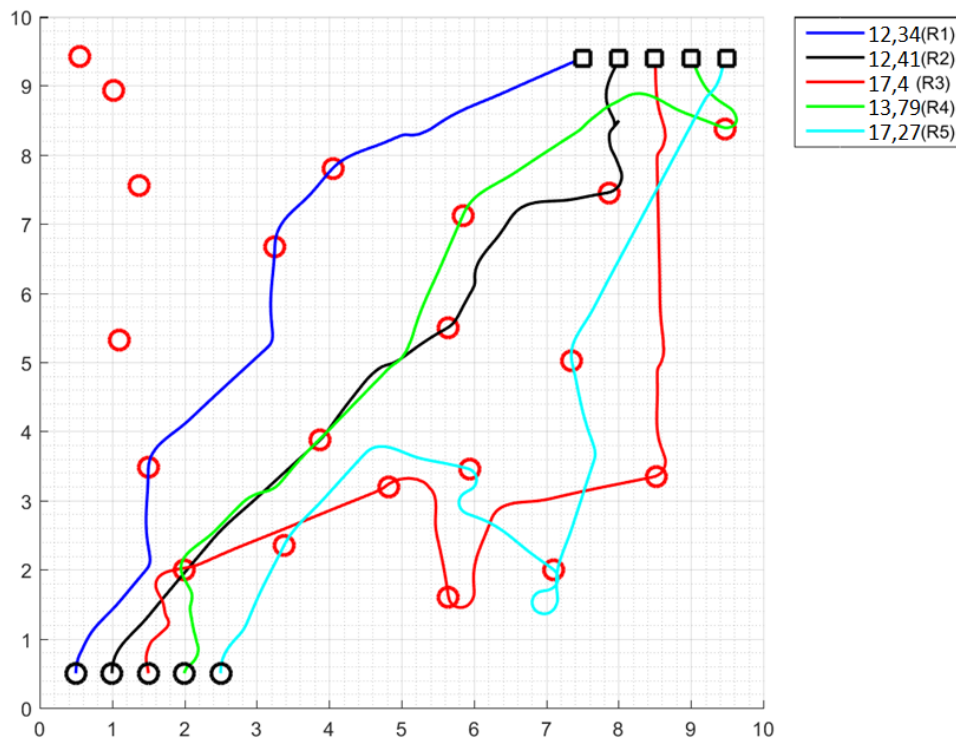


Figura 41: Rotas executadas por 5 robôs no cenário apresentado na Figura 40, sem a tarefa de exploração do ambiente e sem a deposição de feromônios.

Fonte: Autoria própria.

detectarem e visitarem a maioria dos alvos e não colidiram com os obstáculos estáticos e/ou com outros robôs. Em especial, os robôs R1 (azul), R2 (preto) e R4 (verde) desempenharam uma rota sem excessos de desvios da referência (vetor que interliga as posições inicial e final), quando comparados com os robôs R3 (vermelho) e R5 (ciano). Os desvios nas rotas de R3 e R5 são efetuados devido as decisões de evitar colisão entre eles. Além disso, os robôs R2 e R4

executaram rotas similares, procurando por alvos em áreas redundantes. Isso ocorre devido a ausência do comportamento repulsivo entre os robôs quando os sinais de feromônio são detectados e também da decisão de explorar o ambiente. Como consequência, quatro alvos (região superior esquerda) não foram detectados e, portanto, não visitados.

Os resultados da Figura 41 mostram, assim como ocorreu no capítulo 4, que os controladores *fuzzy* são capazes de mover os robôs de forma segura (sem colisões) através do ambiente, visitar os alvos encontrados durante a navegação e alcançar a posição final.

O segundo grupo de experimentos objetiva avaliar o mecanismo de decisão relacionado às tarefas de exploração (D_2) e de correção de rotas para evitar áreas já exploradas por outros robôs (D_4). Como os robôs desconhecem o tamanho do ambiente (apenas conhecem suas posições inicial e final), a tarefa de exploração é afetada pelo valor acumulado no fator de desconto γ (Equação (22)), incrementado pelo parâmetro fixo γ^+ cada vez que o robô decide explorar o ambiente.

Como consequência, quando o ambiente é amplo e γ^+ possui um valor elevado, o robô irá rejeitar precocemente a decisão de explorar o ambiente, deixando áreas não exploradas, onde podem haver alvos que não serão visitados. Por outro lado, se o ambiente é pequeno e γ^+ possui um valor baixo, o robô irá repetir esta decisão com frequência, mesmo quando ele alcançar as bordas (paredes) do ambiente. Por isso, há uma relação de compromisso entre o tamanho do ambiente e o valor da variável γ^+ , de forma a garantir um conjunto de decisões que leve o SMR como um todo a completar sua navegação como o esperado, isto é, visitar todos os alvos existentes no ambiente.

A relação de compromisso mencionada é destacada pela comparação das rotas apresentadas na Figura 42, partes (a) e (b), que correspondem a utilização de $\gamma^+ = 0,015$ e $\gamma^+ = 0,03$, respectivamente. A área destaca pelo retângulo cinza na parte (b), na região superior esquerda do ambiente, engloba os 4 alvos que não são encontrados pelos robôs quando $\gamma^+ = 0,03$, indicando que a decisão D_3 sobrepôs D_2 e assim R1 decide se dirigir ao seu ponto final, abandonando a exploração da área mencionada. As rotas da parte (b) são similares as da Figura 41, quando a decisão D_2 não é considerada.

Além disso, existe uma região com alta intensidade de feromônios lançados no ambiente pelos robôs, no canto inferior esquerdo das Figuras 42 (a) e (b). Esta região coincide com o início da simulação, e como não há influência de outros pontos de interesse (alvos e obstáculos), será usada para avaliar a influência do feromônio na navegação. Em ambas as simulações (partes (a) e (b) da Figura 42), os robôs apresentam um comportamento repulsivo entre eles enquanto estão no interior da área mencionada, e a seguinte situação ocorre:

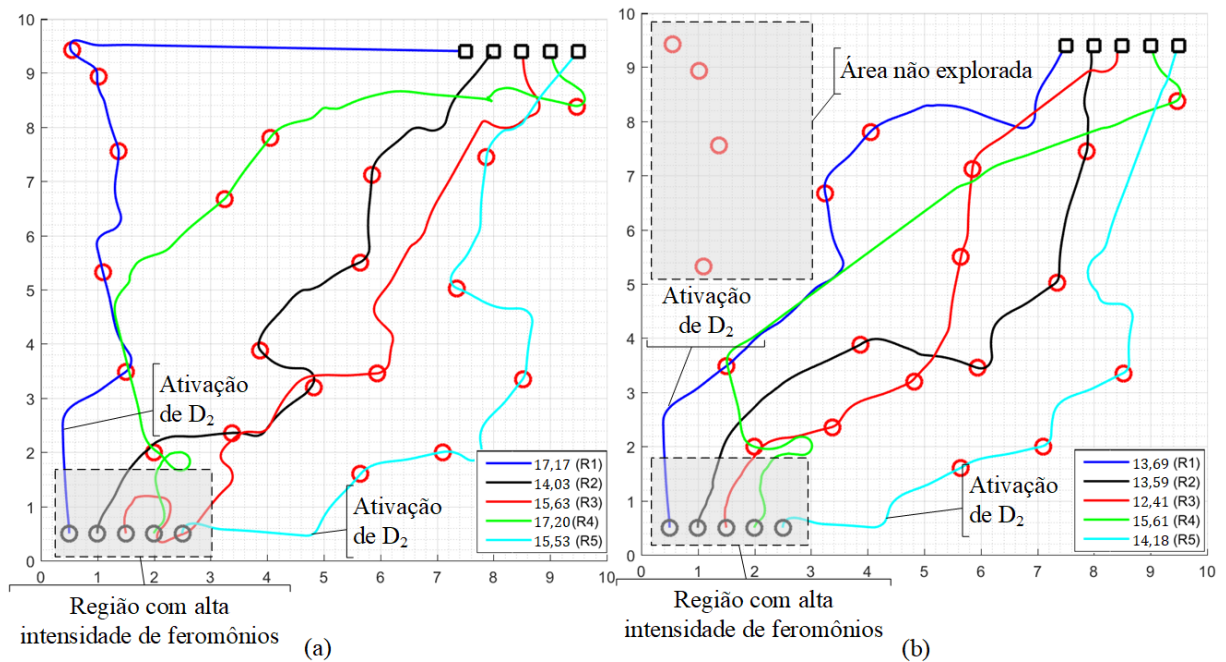


Figura 42: Rotas dos 5 robôs: (a) $\gamma^+ = 0,015$ e (b) $\gamma^+ = 0,03$.

Fonte: Autoria própria.

- o robô R1 (azul) decide explorar a região a sua esquerda, porém ele deve evitar a colisão com a parede do ambiente (detectada pelo sensor esquerdo de proximidade) e, de forma simultânea, evitar a trilha de feromônio lançada por R2 (preto). Devido à parede como obstáculo estático, o robô R1 apresenta o mesmo comportamento nas partes (a) e (b), enquanto está no interior da região inicial com feromônios;
- o robô R2 decide explorar a região a sua direita (devido a detecção de R1 a sua esquerda) e também evitar as áreas de feromônio deixadas por R3 (vermelho). Na parte (a), R3 realiza uma manobra abrupta a sua direita e então se distancia de R2, amenizando os sinais de feromônio deixados no ambiente. Isso faz com que R2 se aproxime dos alvos que estão localizados mais ao centro do ambiente, que posteriormente serão disputados com R3. Já, na parte (b) desta figura, R2 se mantém com a decisão de explorar a região a sua direita e ajusta sua rota devido a presença de feromônios de R3 de elevada intensidade;
- o robô R3, na parte (a), decide evitar a colisão com R2 e também as áreas com feromônios deixadas por R4 (verde), resultando em uma manobra abrupta a sua direita. Isso ocorre pois a decisão reativa (evitar colisão) altera completamente o objetivo advindo da camada deliberativa. Após isso, R3 se mantém na decisão de explorar a região direita do ambiente, substituindo a atuação de R4. Tal situação é diferente na parte (b), pois R4 toma uma distância maior de R3, admitindo uma manobra amenizada;

- R5 (ciano) realiza rotas semelhantes em ambas as figuras pois, diferente de R1, ele está distante dos limites do ambiente (paredes) e então se distancia de R4 assim que a navegação é iniciada, devido a sua decisão de explorar a região direita do ambiente.

Após este evento, o mecanismo de decisão dos robôs diverge entre as duas simulações em todos os outros momentos da navegação. R2 (preto) e R3 (vermelho) desenvolvem rotas próximas em ambas as simulações, e este comportamento afeta suas detecções de alvos e a decisão de atribuição deles. Na parte (b), a decisão de R1 (azul) em mover-se em direção ao seu ponto final (devido ao valor de γ^+ , como explicado anteriormente), ao invés de continuar explorando o ambiente (como ocorre na parte (a)), leva a uma disputa de alvos com o robô R4 (verde). Como resultado, R4 visita apenas dois alvos durante a simulação (b), diferente do que ocorre na parte (a), em que R4 visita 4 alvos.

Comparando as distâncias navegadas pelos robôs nas duas simulações (caixa no canto inferior direito das Figuras 42 (a) e (b)), o caso (a) apresenta uma distância média de aproximadamente 15,91 m, a máxima de 17,2 m (desempenhada por R4) e a menor de 14,03 m (navegada por R2). No caso (b), a distância média é de 13,9 m, enquanto as distâncias maior e menor são associadas aos robôs R4 (15,61 m) e R3 (12,41 m), respectivamente. Em ambos os casos, a diferença do tamanho da rota média é inferior a 2 m para todos os robôs. Portanto, embora o desempenho do SMR seja diferente nos casos analisados, devido as diferentes tomadas de decisão durante a navegação, os valores das distâncias navegadas por cada robô indicam que o comportamento cooperativo em alcançar uma distribuição uniforme de tarefas se mantém frente as mudanças de γ^+ .

Além disso, a área total explorada pelos robôs, em ambos os casos analisados, é apresentada na Figura 43. As áreas são computadas pela união das regiões exploradas por cada robô, de acordo com sua região de detecção. Os resultados correspondem à exploração de 95,75 % do ambiente com $\gamma^+ = 0,015$ (caso (a)) e 82,92 % com $\gamma^+ = 0,03$ (caso (b)). Este fato confirma que $\gamma^+ = 0,015$ é suficiente para promover uma exploração satisfatória deste ambiente, sem um expressivo aumento da distância média navegada, assegurando uma visita a todos os alvos.

Após esta explanação acerca do compromisso de escolha do parâmetro de incremento γ^+ com a área explorada resultante, o cenário com $\gamma^+ = 0,015$ é usado para analisar outras características do planejador proposto.

Os robôs R1 (azul) e R3 (vermelho) são usados para analisar a evolução das tomadas de decisão autônomas durante a navegação mostrada na Figura 42 (a). Nesta simulação, R1

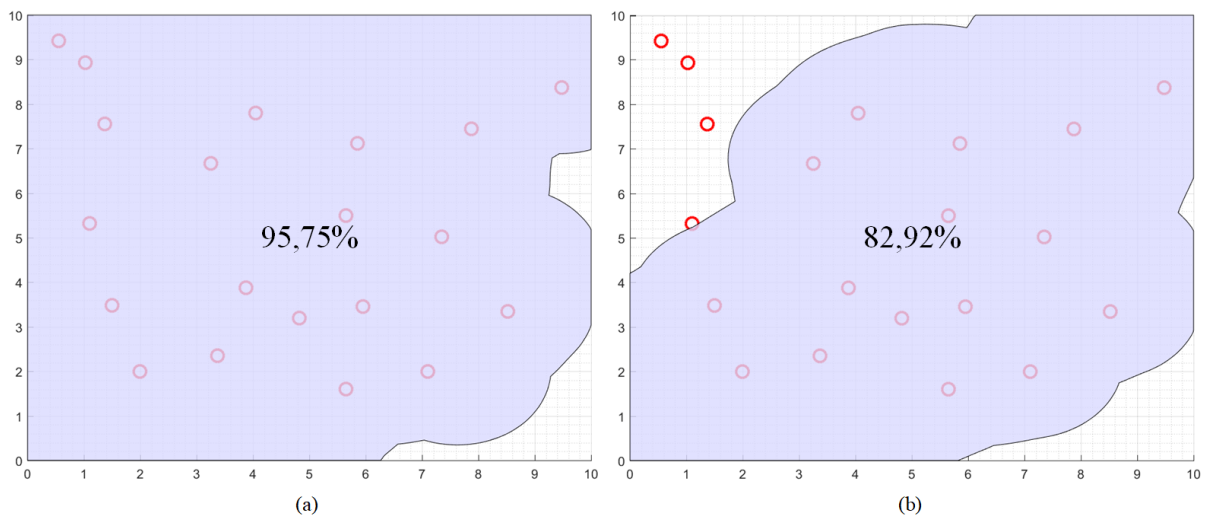


Figura 43: Área total explorada: (a) $\gamma^+ = 0,015$ e (b) $\gamma^+ = 0,03$.

Fonte: Autoria própria.

detecta outros robôs e/ou sinais de feromônios apenas ao seu lado direito, enquanto R3 detecta em ambos os lados.

A evolução das variáveis auxiliares, de h_1 a h_4 , e as decisões tomadas (D_1 a D_4) por R1 e R3, de acordo com a Equação (22), são mostradas nas Figuras 44 e 45, respectivamente. Os valores da variável h_1 e h_2 são usados para uma melhor compreensão da ativação de D_1 , D_2 e D_3 , enquanto a evolução de $h_3 \cdot h_4$ implica na ativação de D_4 . Nestas figuras, a parte (a) se refere às decisões deliberativas D_1 , D_2 e D_3 e a parte (b) indica se foi necessário uma correção da rota calculada (decisão D_4). No gráfico inferior da parte (a), os valores de “Decisão” (eixo y) 1, 2 e 3 representam a ativação de D_1 , D_2 e D_3 , respectivamente.

Em geral, embora a sequência das decisões (atribuição de tarefas durante a navegação) dos dois robôs seja similar, as rotas desempenhadas por cada um divergem. Por exemplo, comparando as Figuras 44 (a) e 45 (a), exceto para as iterações de 100 a 130, os robôs R1 e R3 executam as mesmas tarefas, entretanto a rota de R3 indica que este robô desempenha mais manobras para evitar sinais de feromônios que R1. Esta situação pode ser observada pela frequente necessidade de correção da rota calculada (ativação da decisão D_4), como mostrado na Figura 45 (b). Especialmente no início da simulação (até a iteração 80 na Figura 45), R3 é forçado a fazer um desvio abrupto devido a sua posição no arranjo inicial entre todos os robôs, fato confirmado pelas manobras realizadas no canto inferior esquerdo da Figura 42 (a).

A correção da rota calculada devido a iminência de colisão com obstáculos estáticos e dinâmicos também é avaliada nesta seção. A evolução da variável τ , referente aos robôs

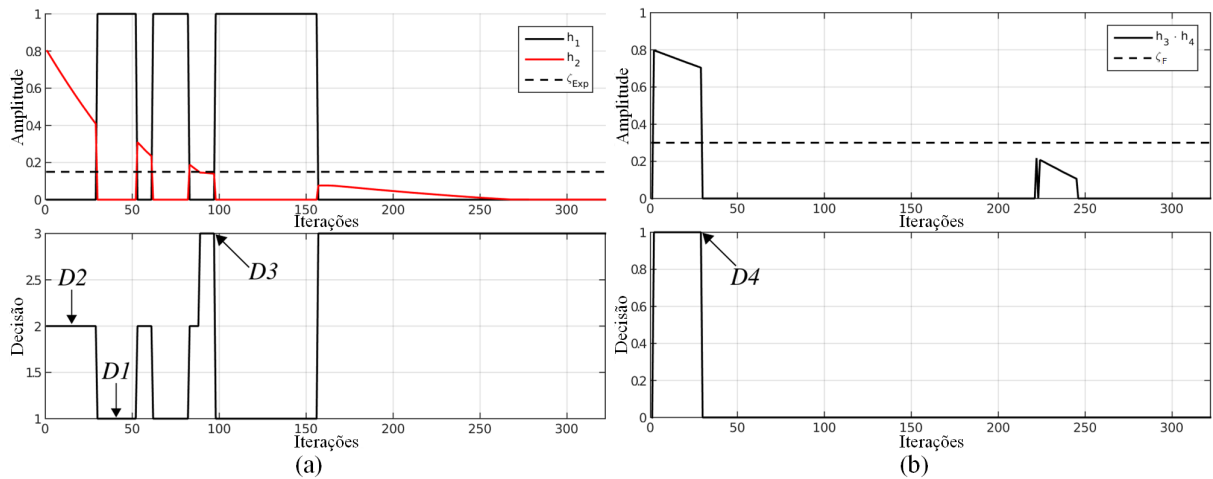


Figura 44: Decisões de R1 com $\gamma^+ = 0,015$: (a) decisão da tarefa a ser realizada e (b) correção da rota calculada.

Fonte: Autoria própria.

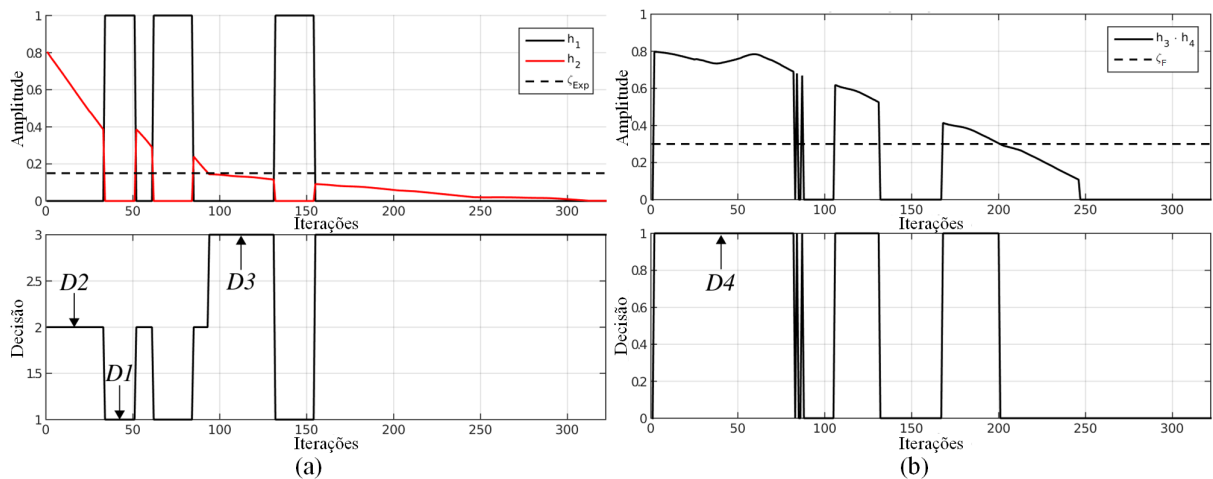


Figura 45: Decisões de R3 com $\gamma^+ = 0,015$: (a) decisão da tarefa a ser realizada e (b) correção da rota calculada.

Fonte: Autoria própria.

R1 e R3, é apresentada na Figura 46, onde $\tau \geq 0,8$ indica que não há colisão a vista e $\tau = 0$ corresponde ao término da simulação. Desta figura, pode-se observar que R3 executou desvios excessivos, especialmente no início da navegação, como mencionado anteriormente, assim como próximo de encerrar sua navegação, quando R3 desvia de R4 e R5 para alcançar sua posição final (final das rotas da Figura 42 (a)). Todos estes desvios implicam em um atraso para finalizar sua tarefa ($\tau = 0$), quando comparado com R1 na Figura 46.

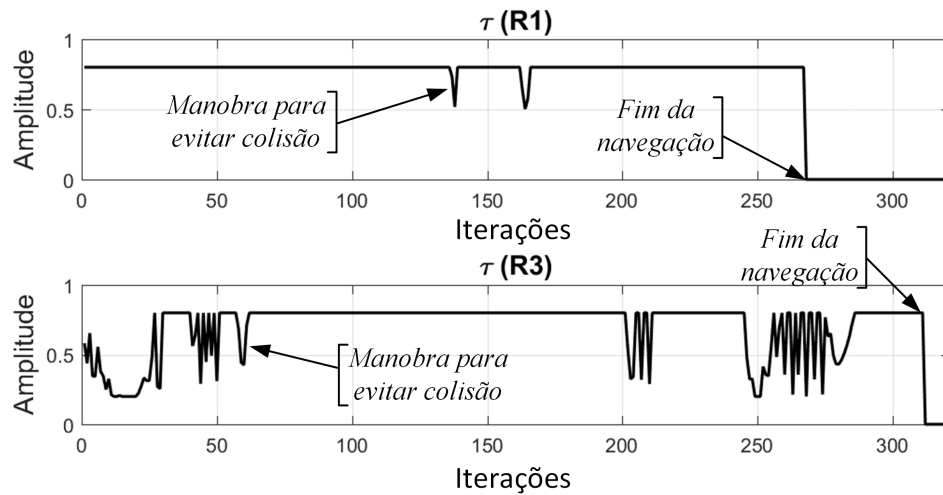


Figura 46: Decisões de R1 e R3 para evitar colisões (τ).

Fonte: Autoria própria.

5.3.2 DESEMPENHO GLOBAL DO PLANEJAMENTO DE ROTAS

Após a análise individual do efeito causado pelo mecanismo de tomada de decisão proposto e da estratégia de planejamento de rotas no comportamento dos robôs, é avaliado o desempenho global do SMR, de forma a comprovar se o planejador é capaz de cumprir com os requerimentos de cooperação, autonomia e objetivos.

A análise de desempenho é executada por meio de 4 diferentes cenários simulados, como é dado na Tabela 11. Tais cenários refletem diferentes complexidades do problema estudado, em que dimensões do ambiente, quantidade de alvos, número de robôs e obstáculos estáticos são alteradas. Em todos os casos simulados, os alvos são igualmente espaçados ao longo do ambiente e os demais parâmetros de simulação são os mesmos apresentados na Tabela 10, exceto por $R_{Det} = 1\text{ m}$ e γ^+ , cujos valores são dados na Tabela 11.

Tabela 11: Parâmetros das simulações referentes a validação global do planejador.

Parâmetro	Valor			
	Cenário 1	Cenário 2	Cenário 3	Cenário 4
Dimensões do ambiente	10 m x 10 m	20 m x 20 m	10 m x 10 m	20 m x 20 m
Número de robôs	6	12	9	6
Número de alvos	25	100	25	100
Número de obstáculos estáticos	12	35	12	35
γ^+	0,005	0,002	0,005	0,002

O desempenho do SMR é inicialmente avaliado por meio de métricas que envolvem o custo de exploração, eficácia e eficiência, relacionadas ao tamanho da área explorada e

ao número de alvos visitados pelos robôs. Posteriormente, o desempenho de cada robô é detalhado, permitindo verificar se a distribuição de tarefas entre eles é balanceada. Em seguida, é apresentada uma breve discussão acerca do custo computacional do planejador de rotas proposto. As métricas usadas para as avaliações mencionadas são as seguintes:

- custo da exploração de área (C_{Area}): esta métrica é calculada pela razão entre a soma das distâncias desenvolvidas por todos os robôs ($Dist_{Total}$) e a área total explorada (AE), obtida pela união de todas as áreas exploradas por cada robô, dadas pelas Equações (23), (24) e (25):

$$C_{Area} = \frac{Dist_{Total}}{AE} \quad (23)$$

$$Dist_{Total} = \sum_{i=1}^{n_R} Dist_{Ri} \quad (24)$$

$$AE = \bigcup_{i=1}^{n_R} A_i \quad (25)$$

onde $Dist_{Ri}$ é a distância navegada pelo robô i , n_R é o número de robôs que compõem o SMR e A_i é a área explorada pelo robô i , de acordo com sua região de detecção;

- custo para visitar os alvos (C_{Alvos}): é definido como uma relação entre a soma de todas as distâncias navegadas (Equação (24)) e o número total de alvos visitados durante a navegação ($A_{Visitados}$), como apresentado na Equação (26);

$$C_{Alvos} = \frac{Dist_{Total}}{A_{Visitados}} \quad (26)$$

- eficácia na exploração de área ($Eficacia_{Area}$): indica a porção da área do ambiente, $Area_{Amb}$, que foi explorada pelo SMR. Esta eficácia é calculada pela Equação (27), onde $Area_{Amb}$ é obtida pelas dimensões do ambiente, disponíveis na Tabela 11, enquanto AE é calculada pela Equação (25). Esta métrica reflete o desempenho da aplicação da decisão deliberativa D_2 ;

$$Eficacia_{Area} = \frac{AE}{Area_{Amb}} \quad (27)$$

- eficácia para visitar alvos ($Eficacia_{Alvos}$): de forma similar à $Eficacia_{Area}$, esta métrica é apresentada na Equação (28) e considera a relação entre a quantidade de alvos visitados

pelo SMR ($A_{Visitados}$) e a quantidade total de alvos inseridos no ambiente (A_{Total}), também obtido da Tabela 11;

$$Eficacia_{Alvos} = \frac{A_{Visitados}}{A_{Total}} \quad (28)$$

- eficiência na exploração de área ($Eficiencia_{Area}$): esta métrica complementa esta análise e mede o desempenho dos robôs nas manobras para evitar áreas já exploradas por outros, devido à detecção de sinais de feromônios e aplicação da decisão D_4 . Este indicador é calculado pela Equação (29).

$$Eficiencia_{Area} = \frac{AE}{AE + AE_{sob}} \quad (29)$$

onde AE_{sob} é a soma das áreas exploradas por mais de um robô (áreas sobrepostas).

Um alto valor de $Eficiencia_{Area}$, ocasionado por um baixo valor de AE_{sob} , indica que há poucas áreas sobrepostas. Entretanto, o tamanho da região de detecção implica na ocorrência destas sobreposições no início e no fim da navegação, devido a proximidade e ao alinhamento dos robôs nas posições inicial e final. Assim, a decisão D_4 apenas promove um resultado efetivo conforme os robôs navegam longe destas áreas. Essas situações de sobreposição nos pontos mencionados não são descartadas durante o cálculo da eficiência e, como consequência, esta métrica não alcança seu valor ótimo ($Eficiencia_{Area} = 1$) nos resultados apresentados nesta seção.

Espera-se que o SMR alcance a maior eficácia (explorar a maior área do ambiente e visitar a maior quantidade de alvos possível), enquanto mantém o menor custo e a maior eficiência possível. Ainda, é desejável que a área explorada e o número de alvos visitados por cada robô sejam balanceadas.

Duas situações são consideradas para esta etapa de validação. Na primeira, “Caso 1”, os robôs executam o planejador completo (todas as decisões, de D_1 a D_5). Para o “Caso 2”, não é utilizada a decisão de evitar sinais de feromônio (D_4), mantendo-se somente as três tarefas: visitar um alvo (D_1), explorar uma área (D_2) e dirigir-se até a posição final (D_3), além da habilidade de desviar de obstáculos (D_5).

Para cada cenário simulado, as métricas são computadas após a navegação dos robôs e os resultados são apresentados por meio dos histogramas na Figura 47. Os *box-plots* da Figura 48 mostram a distribuição de tarefas e o respectivo balanceamento entre os robôs. O custo computacional necessário para o SMR atingir seu objetivo em todos os cenários, medido pelo

número de iterações do algoritmo, é apresentado na Tabela 12. A respeito da decisão reativa de evitar colisão com obstáculos (D_5), nenhuma colisão ocorreu durante os experimentos, o que valida o comportamento conferido pelo sistema *fuzzy*.

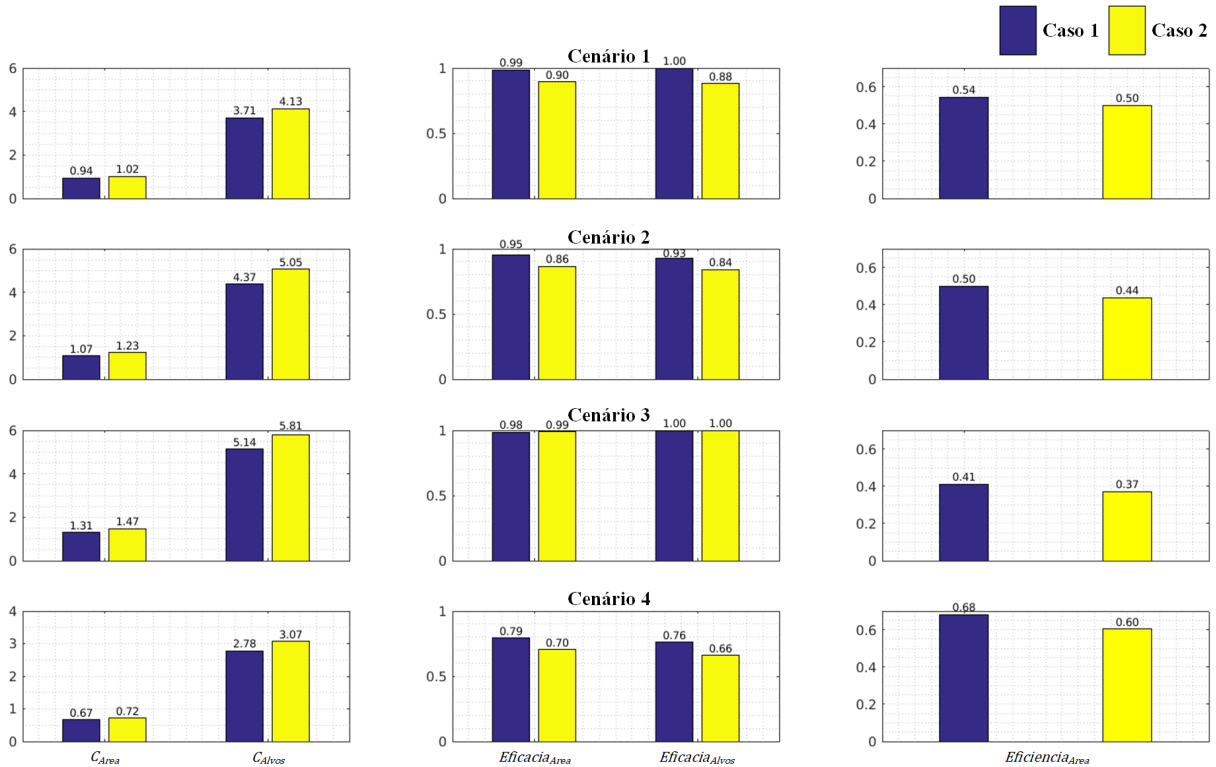


Figura 47: Resultado das métricas.

Fonte: Autoria própria.

Tabela 12: Número de iterações (k).

	Cenário 1		Cenário 2	
	Caso 1	Caso 2	Caso 1	Caso 2
Iterações (k)	267	294	689	663
	Cenário 3		Cenário 4	
	Caso 1	Caso 2	Caso 1	Caso 2
Iterações (k)	251	311	833	796

Sob uma visão geral, a Figura 47 indica que o planejador de rotas bioinspirado (Caso 1) supera o planejador que não considera a detecção de feromônios (Caso 2): ambos os custos (C_{Area} e C_{Alvos}) são menores para todos os cenários; a eficácia é maior, exceto para o cenário 3, em que ambos os casos são praticamente iguais; e a eficiência é maior para o caso 1 em todos os cenários.

Pela comparação de todos os *box-plots* apresentados na Figura 48, é possível observar que a distribuição de alvos entre os robôs é mais dispersa quando a decisão de desvio de

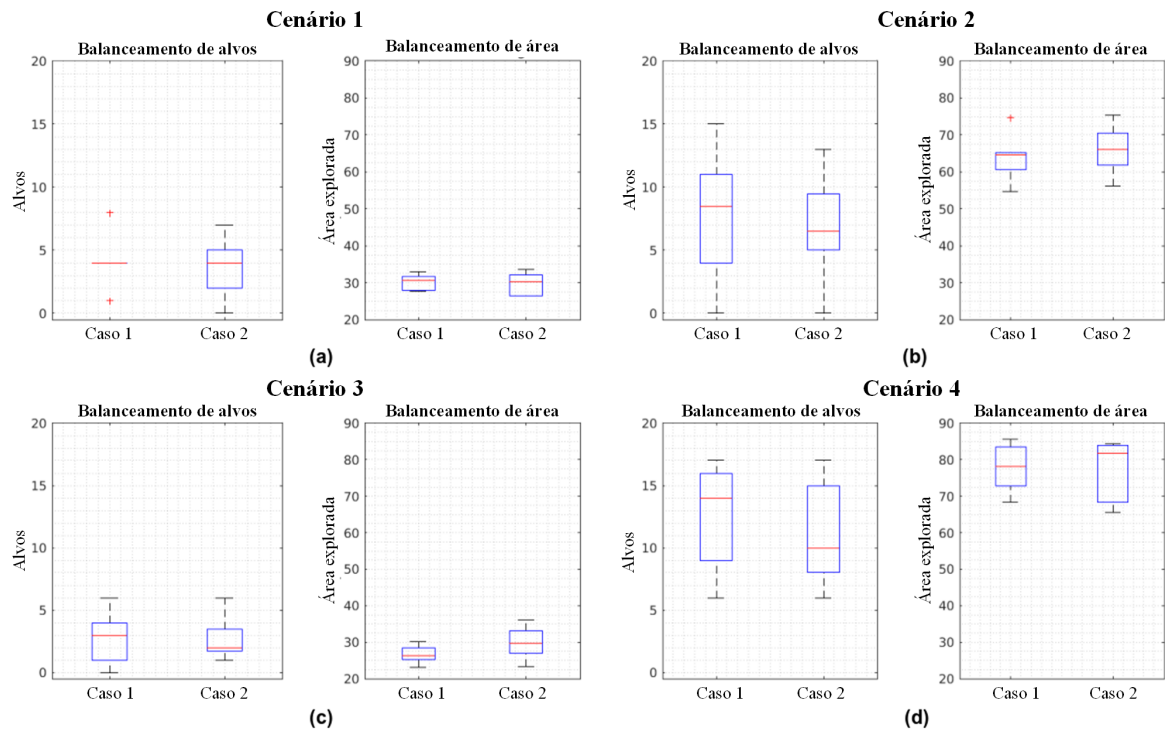


Figura 48: Balanceamento da distribuição de tarefas.

Fonte: Autoria própria.

feromônios é considerada. Provavelmente, em consequência desta dispersão, o SMR alcança mais alvos do que no caso 2, como pode ser notado pelos maiores valores de mediana em todos os *box-plots* para o caso 1. Contudo, considera-se nesta análise que ambos os casos apresentam um satisfatório balanceamento na distribuição de alvos entre os robôs. Por outro lado, a distribuição de tarefas relacionadas à exploração de áreas do caso 1 é superior ao caso 2 em todos os cenários, como pode ser observado nos *box-plots* de balanceamento de área.

Em relação ao custo computacional (Tabela 12), as iterações crescem conforme aumenta-se a complexidade do problema (dimensões do ambiente, número de robôs e alvos). Este crescimento é mais severo com o mecanismo bioinspirado (lançamento e detecção de feromônio), especialmente para os ambientes com maiores dimensões (ambientes 2 e 4), pois a tomada de decisão para o desvio e correção da rota calculada, devido à detecção de feromônios, é executada mais vezes durante rotas mais extensas.

O impacto das dimensões do problema no desempenho do SMR pode ser analisado pela comparação entre os cenários 1 e 2, na Figura 47. Quando as características do cenário (dimensões do ambiente, quantidade de alvos e número de robôs) são aumentadas, como considerado para o cenário 2, ambos os custos aumentam e todos os indicadores de eficácia e eficiência diminuem. Para o caso 1, $C_{Área}$ e C_{Alvos} aumentam 13 % e 18 %, respectivamente,

enquanto ocorre um incremento de aproximadamente 20 % em ambos os custos quando o desvio de feromônio não é utilizado (caso 2). A queda do desempenho, segundo as métricas de eficácia e eficiência para áreas exploradas, resulta em aproximadamente 4 %, enquanto o SMR deixa de visitar apenas 7 alvos no cenário 2 do caso 1 ($Eficacia_{Alvos}$). Para o caso 2, a eficácia na exploração de áreas ($Eficacia_{Area}$) também apresenta uma queda de 4 %, entretanto isso implica em uma perda de eficiência de aproximadamente 7 % e também 16 alvos deixaram de ser visitados.

Das análises apresentadas, pode-se concluir que o crescimento das dimensões do problema prejudica o desempenho do SMR, entretanto o uso do mecanismo bioinspirado proposto ameniza as perdas quando tais acréscimos são considerados.

É possível observar nos *box-plots* apresentados na Figura 48, como as tarefas são compartilhadas entre os robôs conforme as dimensões do problema aumentam. Inicialmente, para o cenário 1 – caso 1, todos os robôs visitam o mesmo número de alvos (4), exceto dois robôs, destacados pelos valores discrepantes (“+” em vermelho) no *box-plot* referente ao balanceamento de alvos (parte (a)). Em contraste, uma maior dispersão de dados é observada quando a decisão D_4 não é utilizada (caso 2), mesmo apresentando o mesmo valor de mediana em ambos os casos. Uma distribuição satisfatória também é atingida em relação ao balanceamento de área explorada, em que todos os robôs exploraram aproximadamente $30 m^2$ no caso 1 e há um ligeiro aumento na dispersão do caso 2.

Com o aumento da área a ser explorada e do número de alvos a serem visitados (cenário 2), o planejador alcança um balanceamento de alvos aceitável em ambos os casos, de acordo com a Figura 48, parte (b). Entretanto, o tamanho da área explorada por cada robô no caso 1 é aproximadamente $65 m^2$, exceto um que robô que explorou $75 m^2$ (destacado como o ponto discrepante “+” em vermelho). Para o caso 2, a área explorada por cada robô varia de $60 m^2$ a $70 m^2$.

A próxima análise considera o cenário 3, que possui as mesmas dimensões do cenário 1 e um acréscimo na quantidade de robôs (Tabela 11). Os valores de eficácia apresentados na Figura 47 indicam que a área do ambiente é completamente coberta e, praticamente, todos os alvos são visitados em ambos os casos (com e sem a decisão D_4). Entretanto, quando comparado com o cenário 1, há uma perda de eficiência ($Eficiencia_{Area}$) em torno de 13 %, também em ambos os casos, devido ao excesso de robôs no ambiente e, portanto, a uma quantidade maior de áreas sobrepostas.

Em relação à distribuição de tarefas, os robôs excedentes no cenário 3 implicam em um aceitável balanceamento de áreas exploradas, quando os cenários 1 e 3 são comparados na

Figura 48. Entretanto, a distribuição de alvos é degradada. Por exemplo, ainda que o valor de mediana para o caso 1 resulte em 3 alvos por robô, há um robô que não visitou nenhum alvo, enquanto outro alcançou 6 alvos.

Em seguida, ao comparar os cenários 1 e 3 na Tabela 12, observa-se que o tempo computacional decai para o caso 1 e aumenta para o caso 2. Este desempenho superior (diminuição do número necessário de iterações) é resultante dos desvios de feromônios que ocorrem durante a navegação com mais frequência, devido à quantidade excessiva de robôs. Nessas condições o mecanismo da decisão D_4 promove um espalhamento dos robôs pelo ambiente, evitando que manobras de desvios entre os robôs sejam necessárias e permitindo que eles alcancem suas posições finais mais rapidamente.

A última etapa desta análise considera a comparação entre os cenários 2 e 4 (Figura 47). De acordo com as características do cenário 4 (Tabela 11), é avaliado o desempenho do SMR frente a uma redução significativa na quantidade de robôs, mantendo o mesmo número de alvos e considerando um cenário com amplas dimensões. Os custos apresentados para esta ocasião são os menores entre todas as simulações, enquanto apresenta a maior eficiência. Entretanto, o desempenho do SMR resulta em uma perda significativa de eficácia, quando os cenários 2 e 4 são comparados: no caso 1, esta perda é de aproximadamente 20 % para alvos e 25 % para áreas exploradas; no caso 2, há uma redução em torno de 30 % para alvos e 35 % para áreas exploradas. A ausência de robôs implica em poucos momentos de sobreposição de áreas exploradas, que favorece ao aumento da eficiência, embora os robôs não explorem o ambiente por completo. As amplas dimensões do ambiente e a grande quantidade de alvos, combinadas com um pequeno número de robôs para explorar a área, justificam os custos baixos. Além disso, a utilização de um grupo pequeno de robôs também implica em um incremento no custo computacional, como pode ser visto por meio da comparação entre os cenários 2 e 4 na Tabela 12.

No cenário 4, há uma pequena diferença entre os resultados das métricas dos casos 1 e 2 (Figura 47) e os *boxplots* são similares (Figura 48 (d)). Estes fatos indicam que o mecanismo bioinspirado não é frequentemente ativado no caso 1. Uma vez que os robôs apresentam uma maior dispersão nestes cenários, poucas rotas são sobrepostas, exceto no início e no fim da navegação. Além disso, o longo tempo necessário para navegar até a posição final favorece a evaporação dos sinais de feromônios e, portanto, a decisão D_4 não é frequente entre os robôs.

Após esta análise dos resultados, pode-se concluir que o planejador de rotas proposto é capaz de cumprir os requerimentos de cooperação, autonomia e de objetivos do SMR. A introdução de um mecanismo bioinspirado aprimora o desempenho da navegação, e isso pode

ser observado com o aumento da eficácia para concluir as tarefas e também reduzir seus custos. Um desempenho satisfatório é assegurado por uma escolha apropriada da quantidade de robôs no SMR e considerando as características do problema, tais como o tamanho da região de detecção, dimensões do ambiente e número de alvos desconhecidos.

5.3.3 CASOS ESPECIAIS

Em todas as simulações apresentadas até o momento, os alvos são posicionados igualmente espaçados no ambiente. Esta condição nem sempre é real nas aplicações de SMR. Portanto, dois diferentes cenários são considerados para o estudo de casos especiais, em que os alvos são concentrados em uma região restrita do ambiente. No primeiro, denominado cenário 5, os alvos são concentrados no centro do ambiente, exatamente no meio dos caminhos que interligam as posições inicial e final dos robôs. No segundo, cenário 6, os alvos são alocados no canto inferior direito do ambiente, que corresponde a uma área que frequentemente permaneceu inexplorada nas simulações anteriores, como pode ser visto na Figura 43, por exemplo.

As características dos cenários mencionados são as mesmas do cenário 1, apresentado anteriormente na Tabela 11, exceto por ter 13 alvos inseridos no ambiente com posição desconhecida. Os cenários 5 e 6 são apresentados na Figura 49 (a) e (b), respectivamente. Os resultados acerca do número de alvos visitados e da área explorada pelos robôs são apresentados nas Tabelas 13 e 14, respectivamente.

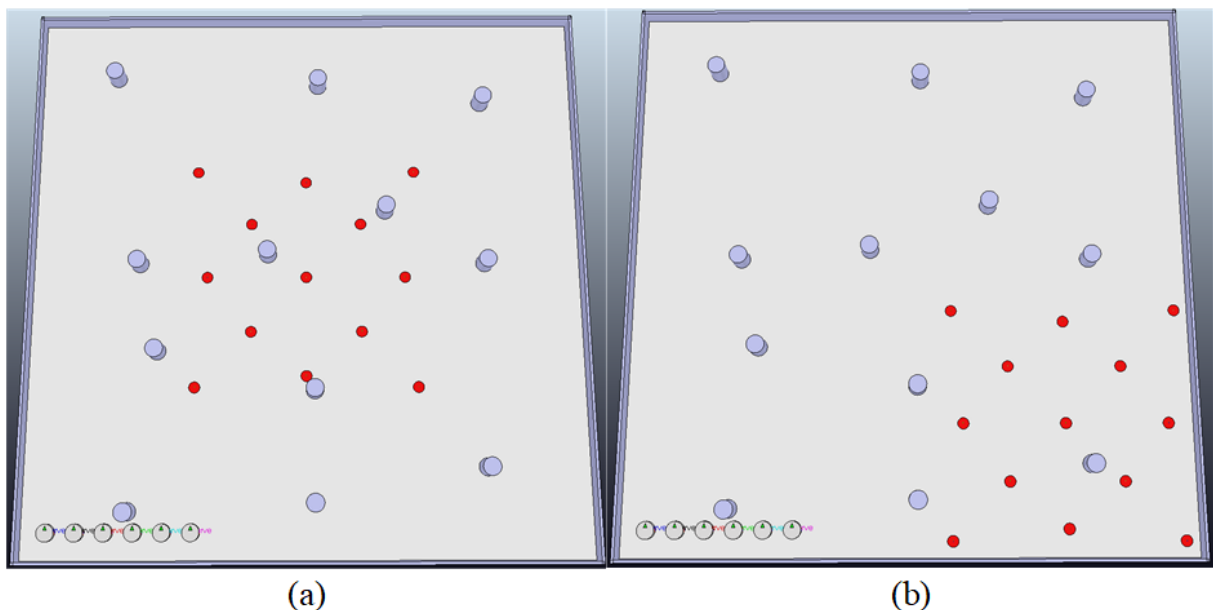


Figura 49: Ilustração dos casos especiais: (a) Cenário 5 e (b) Cenário 6.

Fonte: Autoria própria.

Tabela 13: Número de alvos visitados nos casos especiais.

Robôs	Cenário 5		Cenário 6	
	Caso 1	Caso 2	Caso 1	Caso 2
R1	0	0	0	0
R2	1	0	0	0
R3	3	3	0	0
R4	4	4	0	1
R5	4	0	3	3
R6	0	0	7	7
Eficácia	0,92	0,69	0,77	0,85

Tabela 14: Áreas exploradas nos casos especiais (m^2).

Robôs	Cenário 5		Cenário 6	
	Caso 1	Caso 2	Caso 1	Caso 2
R1	25,12	25,34	25,04	24,92
R2	27,41	32,76	26,82	30,36
R3	28,11	26,39	28,91	28,99
R4	27,11	29,82	26,53	28,68
R5	28,03	34,45	28,59	27,36
R6	24,50	24,46	27,53	28,27
Eficácia	0,91	0,93	0,96	0,97

É possível notar que a concentração de alvos no centro do cenário 5 não impede que os robôs explorem o ambiente. A eficácia de exploração é alta, em torno de 92 % em ambos os casos (planejador com e sem a decisão D_4). Entretanto, a situação não é a mesma para o número de alvos visitados. O uso do mecanismo para o desvio de feromônios (caso 1) leva os robôs a detectarem estes sinais em ambos os lados de sua rota (esquerda e direita), em adição a detecção da presença real de outros robôs, e então seguir uma rota que intercepta o meio do ambiente, o que favorece a detecção e atribuição dos alvos. Este comportamento foi discutido anteriormente, na seção 5.3.1, quando as rotas (Figura 42) dos robôs R2 (preto) e R3 (vermelho) foram analisadas. Por outro lado, este fato não é observado no caso 2, que resulta em uma eficácia de 69 % em relação aos alvos visitados.

Como esperado, os alvos inseridos no cenário 5 são distribuídos entre os robôs R2, R3, R4 e R5 no caso 1, enquanto R1 e R6 priorizaram explorar as regiões esquerda e direita do ambiente, conforme notado pelo número de alvos visitados, presente na Tabela 13. Apesar disso, a área explorada por cada robô é similar entre eles, como pode ser observado na Tabela 14.

Quando os alvos são concentrados no canto inferior direito, como apresentado no cenário 6, o desempenho de exploração é similar ao cenário 5 para ambos os casos (eficácia

> 90 %). Em relação aos alvos visitados, esta tarefa é atribuída somente aos robôs R5 e R6 do caso 1, encarregados de explorar esta região específica. Estes robôs deixaram de identificar e visitar outros 3 alvos, o que resulta em uma perda de eficácia (77 %). Para o caso 2, 3 robôs (R4, R5 e R6) visitam 11 alvos com uma melhor eficácia (85 %), como pode ser visto na Tabela 13.

Os resultados apresentados nestes casos especiais mostram que a implementação completa da estratégia de planejamento de rota proposta (caso 1) promove uma distribuição de tarefas equilibrada (número de alvos visitados e área explorada por cada robô), independente da distribuição espacial dos alvos. Por exemplo, no cenário 5, os alvos foram distribuídos entre 4 robôs (R2, R3, R4 e R5), enquanto que apenas os robôs R3 e R4 visitaram alvos no caso 2. Para as áreas exploradas, os resultados variam de 24,50 a 28,11 m^2 no caso 1, enquanto este intervalo é de 24,56 a 34,45 m^2 no caso 2. No cenário 6, apesar do caso 2 apresentar um melhor desempenho na distribuição de visita aos alvos, o caso 1 apresenta uma melhor distribuição de área explorada.

5.3.4 ESTUDOS COMPARATIVOS

O planejador de rotas proposto neste trabalho é comparado com outras abordagens presentes na literatura. Em (PALMIERI et al., 2018), um grupo de robôs navegam sob a influência de feromônios artificiais e tomam decisões de acordo com um mecanismo adaptativo. Como discutido no capítulo 2, seção 2.6, a solução apresentada pelos autores é similar à abordagem proposta neste capítulo e, portanto, ambas são comparadas. Além disso, um *benchmark* para avaliar estratégias de buscas para um enxame de robôs é apresentado por Couceiro et al. (2014), o qual é utilizado para comparar o desempenho da estratégia proposta neste capítulo com as seguintes outras abordagens, também apresentadas em (COUCEIRO et al., 2014): *Robotic Darwinian Particle Swarm Optimization* (RDPSO), *Extended Particle Swarm Optimization* (EPSO), *Physically-embedded Particle Swarm Optimization* (PPSO), *Glowworm Swarm Optimization* (GSO) e *Aggregations of Foraging Swarm* (AFS).

5.3.4.1 ESTUDO COMPARATIVO 1

Inicialmente, o planejador proposto neste capítulo é comparado com o trabalho de (PALMIERI et al., 2018), onde o ambiente discreto é um *grid* com 50 x 50 células, e o mecanismo de exploração é avaliado em diversos cenários, com alterações no número de robôs e alvos, inseridos em posições aleatórias. Cada robô no interior deste *grid* possui uma região de sensoriamento de 4 células ao seu redor, ou seja, um robô é capaz de detectar outros robôs, alvos e sinais de feromônios quando estão no interior desta região.

Para este estudo comparativo, um ambiente quadrado de $25\text{ m} \times 25\text{ m}$, com obstáculos e alvos, é representado no simulador V-REP. Este ambiente contínuo é dividido em pequenas células ($0,5\text{ m} \times 0,5\text{ m}$) e todos os parâmetros de simulação são equivalentes ao trabalho a ser comparado. O desempenho de exploração do ambiente é avaliado de acordo com variações no número de robôs (15, 25 e 40) e na quantidade de alvos (10, 15 e 20), com posições desconhecidas. Os autores Palmieri et al. (2018) apresentam resultados obtidos da variação de um peso de ajuste utilizado em sua estratégia e, somente a sintonia que resulta em dados mais balanceados é considerada nesta análise. O percentual de células não exploradas em cada experimento está disponível na Tabela 15, onde “A” é a estratégia proposta neste capítulo e “B” é o desempenho obtido de (PALMIERI et al., 2018).

Tabela 15: Percentual de células não exploradas.

Número de alvos	Número de robôs					
	15		25		40	
	A	B	A	B	A	B
10	8 %	6 %	0,39 %	4,5 %	2,66 %	3,5 %
15	3 %	7 %	0 %	5 %	3 %	3,5 %
20	2,1 %	18 %	0 %	5,5 %	2,61 %	4,5 %

Dos resultados apresentados, o método proposto A supera a estratégia B em todos os cenários, exceto para a situação de 15 robôs e 10 alvos. O melhor desempenho de A é alcançado com 25 robôs, independente do número de alvos considerados, o que significa um satisfatório balanceamento nas tarefas do SMR: exploração do ambiente (sob influência de feromônios), detecção e visita aos alvos ou seguir para a posição final. Por outro lado, a estratégia B apresenta maiores valores percentuais de células não exploradas para o mesmo número de robôs.

Em contraste, o experimento com 15 robôs resulta no pior desempenho para ambas as estratégias. Especialmente para A, o mecanismo bioinspirado não é frequentemente ativado, pois o tamanho da região de sensoriamento dos robôs não é o suficiente para promover as interações locais frente as dimensões de ambiente utilizadas.

É importante destacar que a estratégia B possui características particulares que não são consideradas no planejamento proposto neste capítulo. Como exemplo, os autores definem que o objetivo dos robôs é procurar e desarmar as minas terrestres, que podem explodir a qualquer momento e atingir um robô, desabilitando-o da navegação. Esta dinâmica de manipulação das minas afetam o desempenho da estratégia comparada e, provavelmente, explicam os desempenhos inferiores apresentados na Tabela 15.

5.3.4.2 ESTUDO COMPARATIVO 2

O segundo estudo comparativo se refere ao trabalho de Couceiro et al. (2014), onde um ambiente com amplas dimensões ($2000 m^2$) e com obstáculos estáticos é considerado para avaliar as seguintes abordagens de exploração por um SMR: RDPSO, AFS, GSO, PPSO e EPSO. Estas estratégias são usadas em tarefas de exploração para propósitos de mapeamento. Os robôs se movimentam pelo ambiente, sem uma posição final determinada, com o principal objetivo de identificar se uma célula do ambiente está ocupada ou não. Durante esta navegação, os robôs são capazes de desempenhar uma comunicação direta entre eles, de acordo com uma região de comunicação, assim como o parâmetro R_{Det} considerado nesta tese. A partir de um valor de referência ($R_{Det} = 3 m$), uma severa limitação de 30 % nesta capacidade de se comunicar ($R_{Det} = 0,9 m$) é aplicada para avaliar a influência deste parâmetro na tarefa de exploração.

De acordo com as características descritas, um ambiente simulado no V-REP com $2000 m^2$ (aproximadamente $44,7 m \times 44,7 m$) e 75 obstáculos estáticos, com posições aleatórias e desconhecidas, é adotado para esta comparação. O desempenho de todas as estratégias comparadas é avaliado com três configurações do SMR (10, 20 e 30 robôs) e dois tamanhos para a região de comunicação (3 m e 0,9 m).

A métrica *Area Under the Curve* (AUC), resultado da razão entre as áreas exploradas e o número de iterações, é usada para analisar a probabilidade do SMR em explorar completamente o ambiente. A AUC normalizada é calculada pela Equação (30), onde: N_{Iter} é o número de iterações e $Eficacia_{Area}(k)$ (computada na Equação (27)) é a medida de eficácia no momento k .

$$AUC = \frac{1}{N_{Iter}} \sum_{k=0}^{N_{Iter}} Eficacia_{Area}(k) \quad (30)$$

A AUC para cada experimento é mostrado na Figura 50, onde “Caso 1” se refere à $R_{Det} = 3$, “Caso 2” à $R_{Det} = 0,9$, A = estratégia proposta neste capítulo, B = RDPSO, C = AFS, D = GSO, E = PPSO e F = EPSO.

Para um tamanho fixo da região de comunicação, o aumento do número de robôs contribui para um aumento no valor de AUC para todas as estratégias comparadas, como é apresentado nas partes (a) e (b) da Figura 50.

A respeito dos tamanhos da região de comunicação considerados, é possível observar na Figura 50 (a) que A supera todos os métodos quando $R_{Det} = 3 m$. Entretanto, quando

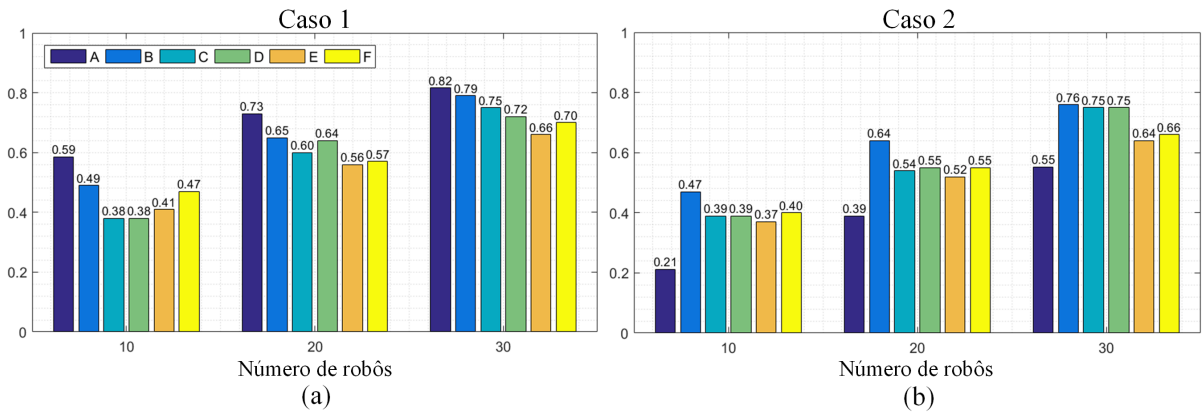


Figura 50: AUC: (a) $R_{Det} = 3$ e (b) $R_{Det} = 0,9$. A = estratégia proposta neste capítulo, B = RDPSO, C = AFS, D = GSO, E = PPSO e F = EPSO.

Fonte: Autoria própria.

aplica-se a limitação de 30 % nesta característica, o desempenho de A é mais afetado que as outras estratégias, de acordo com o que é apresentado na Figura 50 (b). Como discutido nas seções anteriores, os robôs do SMR proposto tem dificuldades em desempenhar a exploração em ambientes de dimensões muito ampla, principalmente quando a região de detecção ou o número de robôs são reduzidos, pois eles necessitam de mais tempo (N_{Iter} é elevado na Equação (30)) para detectar a presença de outros robôs próximos e também sinais de feromônios lançados no ambiente.

Os estudos comparativos apresentados nesta seção confirmam a factibilidade da estratégia bioinspirada de planejamento de rotas *on-line* proposta neste capítulo. É possível observar e comparar seu desempenho em ambientes de dimensões amplas, com um elevado número de robôs e sob restrições na região de comunicação/detecção.

Além disso, o estudo também serviu para mostrar a dependência do sistema de navegação proposto com seus parâmetros relacionados ao tamanho da região de detecção, limiar de decisão para a tarefa de exploração e o uso de feromônios artificiais.

6 UMA ARQUITETURA BASEADA EM REALIDADE AUMENTADA PARA A VALIDAÇÃO DO SISTEMA MULTIRROBÔ PROPOSTO

Uma arquitetura baseada em um sistema de localização visual é proposta neste capítulo para validar as estratégias de planejamento de rota descritas nos capítulos 4 e 5. Esta arquitetura permite construir um ambiente em realidade aumentada, onde é possível reproduzir experimentos com SMRs submetidos a restrições na comunicação e na capacidade de sensoreamento dos robôs.

De uma forma geral, os robôs, alvos e obstáculos considerados nos experimentos são associados a identificadores visuais do tipo ARTag (SAGITOV et al., 2017), cujas posições são definidas por meio de imagens capturadas por uma câmera e então disponíveis como tópicos ROS. A partir disso, esta arquitetura possibilita a utilização de robôs móveis reais de estrutura simples nos experimentos, mesmo que apresentem diversas limitações em suas habilidades de detecção e comunicação, pois os dispositivos periféricos responsáveis pela execução das habilidades dos robôs (descritos ao longo dos capítulos 3, 4 e 5) são virtualmente reproduzidos em um ambiente de realidade aumentada, em função das medidas de distância entre as coordenadas obtidas dos marcadores ARTags. Assim, apenas a capacidade de locomoção em um ambiente plano e um protocolo simples de comunicação com o *framework* ROS são necessários aos robôs.

Neste capítulo, robôs do tipo Sphero (SPHERO, 2018; MIMRÁNEK; SMUTNÝ, 2016) são usados para a validação do sistema de navegação desenvolvido, considerando o ambiente real mencionado.

A descrição desta arquitetura está organizada da seguinte forma: inicialmente, as principais características dos robôs Sphero são apresentados na seção 6.1. Em seguida, uma descrição completa da arquitetura experimental é apresentada ao longo da seção 6.2. As estratégias abordadas nos capítulos 4 e 5 são implementadas neste ambiente e os resultados são apresentados na seção 6.3.

6.1 DESCRIÇÃO DOS ROBÔS SPHERO

Um time composto por robôs Sphero (Figura 51) é usado para validar as estratégias de planejamento de rota desenvolvidas nesta tese. Este robô possui um *drive* de acionamento diferencial para sua locomoção, é revestido por uma casca de plástico no formato esférico e é equipado com comunicação *Bluetooth*, meio usado para receber os comandos de movimentos.



Figura 51: Robô Sphero.

Fonte: (SPHERO, 2018).

Sua interface é originalmente idealizada para promover a comunicação entre o robô e um dispositivo móvel (telefone celular), que opera como um controle remoto, permitindo o seu acionamento manual. Embora esta interface permita que o robô envie e receba dados, não há um meio de reprogramar o seu *hardware*.

Além disso, este robô não possui dispositivos periféricos que o permita detectar obstáculos ao seu redor. Ao invés disso, o robô Sphero possui apenas um sensor de colisão, que requer contato para registrar tal ocorrência.

Um pacote ROS denominado `sphero_swarm` é disponibilizado por terceiros para permitir a comunicação entre os robôs Sphero e um computador, o que possibilita enviar comandos de velocidade e orientação aos robôs por meio de tópicos ROS específicos.

Os robôs Sphero foram utilizados para esta validação devido às sua disponibilidade no laboratório LASCA. Tais robôs possuem severas limitações em habilidades de sensoriamento e comunicação, o que destaca a flexibilidade de utilização desta arquitetura experimental. Neste sentido, os robôs utilizados são capazes de executar virtualmente todas as habilidades

estabelecidas nos capítulos 3, 4 e 5, relacionadas às estratégias de planejamento de rotas apresentadas anteriormente.

6.2 DESCRIÇÃO DA ARQUITETURA EXPERIMENTAL BASEADA EM REALIDADE AUMENTADA

A configuração completa da arquitetura desenvolvida é composta por um ambiente real retangular, cujas dimensões suportam a inclusão de múltiplos robôs de pequeno porte e outros objetos (alvos e obstáculos), uma câmera USB, posicionada acima do ambiente, e um computador com pacotes ROS. Uma visão geral deste arranjo é apresentada na Figura 52.

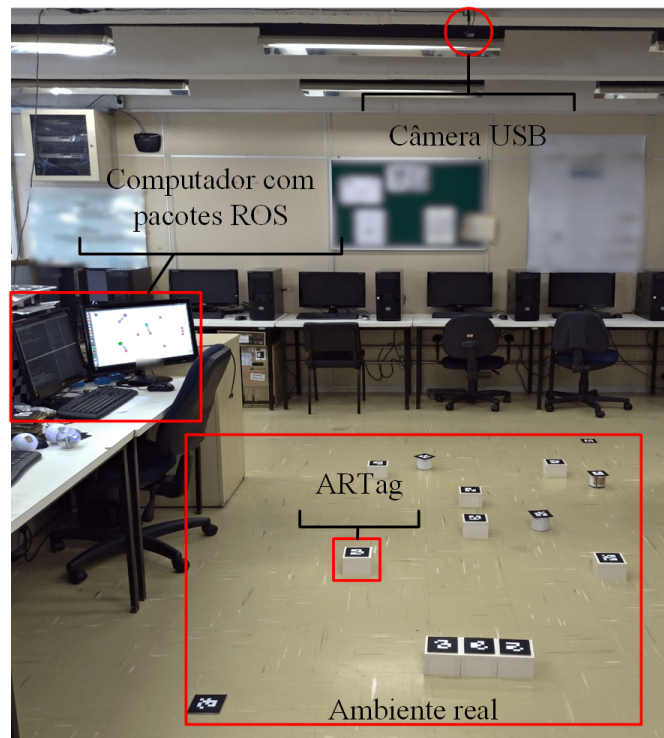


Figura 52: Ambiente real e seus dispositivos.

Fonte: Autoria própria.

Um experimento conduzido neste ambiente requer que todos os objetos identificáveis (robôs, alvos e obstáculos) possuam marcadores visuais ARTags, que são detectados pela câmera USB e sua imagem é processada pelo computador com os pacotes ROS apropriados. Após esta aquisição de imagem, um ambiente virtual é construído por meio de pacotes ROS específicos (a serem explicados posteriormente), a fim de reproduzir as configurações necessárias para os algoritmos de planejamento de rota a serem validados. O computador utilizado nesta configuração é o agente central que executa o algoritmo genético, referente à

estratégia híbrida (capítulo 4), enquanto os módulos de planejamento *on-line*, abordados em ambas as estratégias apresentadas (capítulos 4 e 5), são reproduzidos virtualmente por meio de nós ROS individuais (no computador), a serem detalhados nas próximas seções. Como última etapa, as ações de controle resultantes são enviadas aos robôs por meio de um protocolo *Bluetooth*. Este fluxo de operação é apresentado na Figura 53 e ocorre continuamente, exceto a etapa relacionada aos procedimentos *off-line*.

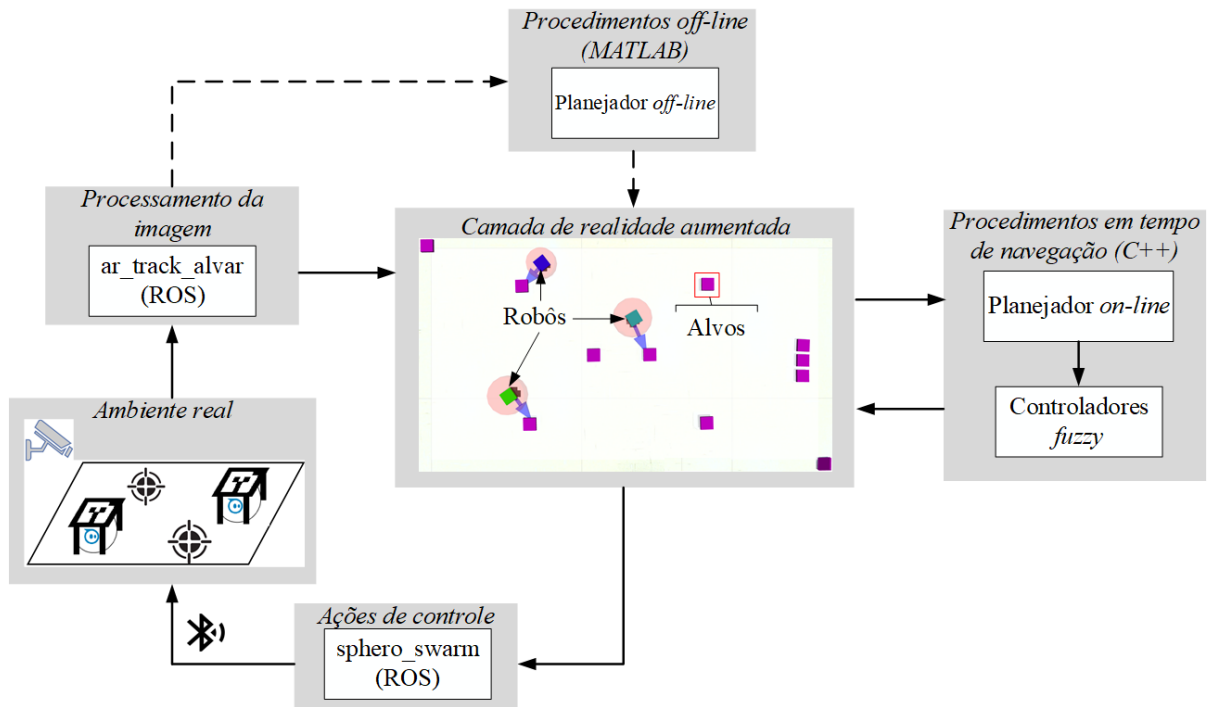


Figura 53: Diagrama do fluxo de operação da arquitetura experimental.

Fonte: Autoria própria.

Os principais detalhes acerca da representação dos objetos (robôs, alvos e obstáculos) neste ambiente e da metodologia considerada para a implementação das estratégias de planejamento, incluindo as habilidades de detecção e comunicação, são apresentados conforme segue.

6.2.1 REPRESENTAÇÃO DOS OBJETOS NO AMBIENTE REAL

Todos os robôs, alvos e obstáculos são representados por meio de ARTags (marcadores visuais) inseridos no ambiente, cujo exemplo é apresentado na Figura 54 (a). Um marcador ARTag é formado por um padrão bicolor que corresponde a um identificador único e pode ser detectado por um pacote ROS denominado `ar_track_alvar` (WIKI, 2018a), por meio de uma imagem digital, obtida da câmera USB posicionada na parte superior do ambiente real.

Conforme os marcadores são detectados, este pacote publica cada nome identificador e sua respectiva posição e orientação (*pose*) em tópicos ROS específicos. Neste trabalho, cada ARTag possui uma dimensão de $94\text{ mm} \times 94\text{ mm}$.

Devido a forma esférica dos Sphero, uma estrutura auxiliar foi desenvolvida para posicionar corretamente os marcadores em cada robô, sem impedir a sua capacidade de rotação e garantir que a ARTag fique sempre visível à câmera. Dessa forma, a posição do marcador no ambiente se torna a do próprio robô. Esta estrutura é apresentada na Figura 54 (b).

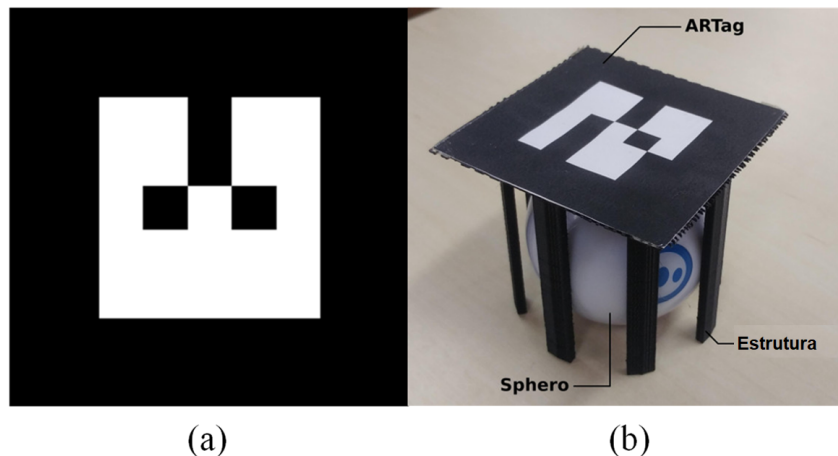


Figura 54: Sistema de localização dos robôs: (a) marcador ARTag e (b) suporte do marcador.

Fonte: Autoria própria.

A coordenada padrão de referência para o pacote `ar_track_alvar` é o centro da imagem detectada (coordenada (0,0)). Por conveniência, este ponto de referência é redirecionado a outros marcadores ARTags, cuja localização é fixa nas extremidades do ambiente e interpretados como limites do cenário. Todos os outros elementos associados aos demais marcadores visuais (alvos e obstáculos) possuem sua posição definida de acordo com este sistema de referência. Este procedimento facilita a implementação virtual das habilidades de detecção e comunicação dos robôs durante os experimentos, pois permite uma interpretação visual dessas habilidades (trilha de feromônios, região de detecção, entre outras).

Para proporcionar uma melhor visualização dos experimentos conduzidos neste ambiente, um sistema de realidade aumentada é desenvolvido para representar as ARTags em uma camada virtual, possibilitando a adição de características específicas que facilitam a visualização pelo usuário. Este procedimento é realizado por meio do pacote ROS `RViz` (WIKI, 2018c). Assim, é possível distinguir todos os marcadores inseridos no ambiente, como um robô de outros elementos (outros robôs, alvos e obstáculos), e mostra as características relacionadas a navegação dos robôs, tais como a execução das habilidades (região de detecção/comunicação),

a rota até um alvo detectado, os sinais de feromônio lançados no ambiente, entre outras.

Para propósitos de ilustração, a configuração experimental apresentada na Figura 52 tem sua representação virtual mostrada na Figura 55, onde todos os robôs (quadrados em azul, verde claro e verde escuro) calculam suas rotas em direção aos alvos detectados e lançam feromônios no ambiente.

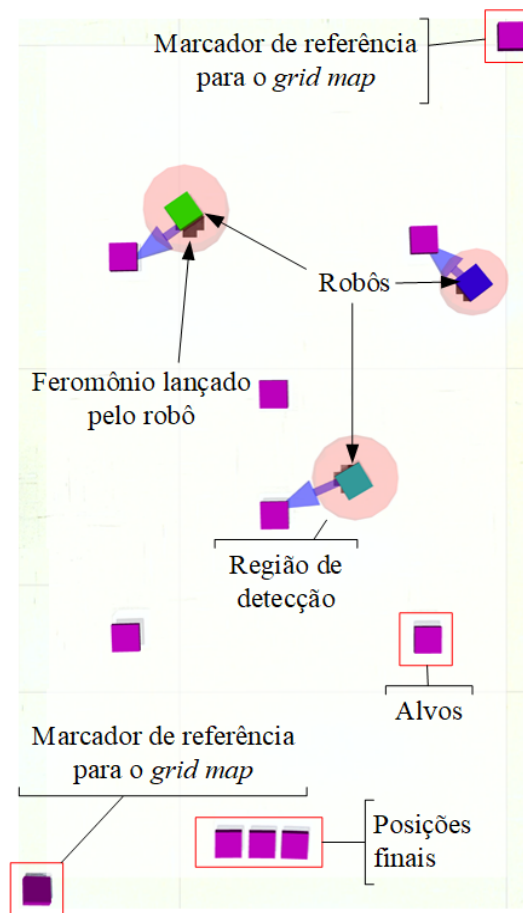


Figura 55: Ambiente da Figura 52 em realidade aumentada.

Fonte: Autoria própria.

A partir das informações apresentadas, todos os dados necessários para a implementação das habilidades dos robôs (detecção e comunicação) e das estratégias de planejamento de rotas consideradas nesta tese estão disponível em tópicos ROS específicos.

6.2.2 METODOLOGIA DE IMPLEMENTAÇÃO DAS ESTRATÉGIAS

Para a implementação das estratégias de planejamento de rota, é necessário que todos os requisitos do sistema básico de navegação (capítulo 3) e as habilidades desenvolvidas nos capítulos 4 e 5 sejam executadas para cada robô utilizado nos experimentos.

Portanto, o comportamento distribuído entre os robôs é alcançado por meio de *scripts* individuais em C++, executados em diferentes nós ROS e associados a cada membro do SMR. Os *scripts* são desenvolvidos de acordo com o *framework* ActionLib (SANTOS et al., 2017), onde o paradigma cliente-servidor é intermediado pelo ROS e permite que os robôs processem as tarefas de uma forma preemptiva e por meio de mensagens padronizadas.

Cada *script* é dividido em três etapas de execução, conforme ilustrado na Figura 56: (i) a partir dos dados do ambiente (posição dos marcadores ARTags), obtidos da imagem capturada pela câmera, subscrever e publicar mensagens em tópicos ROS relacionados às habilidades de detecção/comunicação; (ii) executar a estratégia de planejamento; e (iii) publicar as ações de controle resultantes.

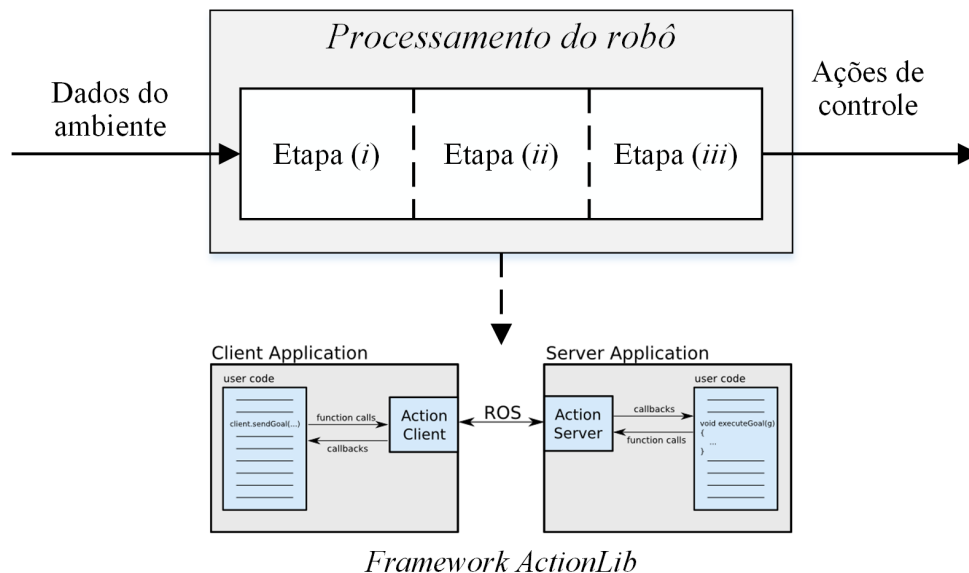


Figura 56: Processamento dos *scripts* dos robôs.

Fonte: Adaptado de (SANTOS et al., 2017).

Na etapa (i), os robôs acessam a posição de todas as ARTags, que são detectadas pela câmera (alvos e/ou robôs) e publicadas usando o pacote ROS `ar_track_alvar`. Entretanto, apenas as ARTags que são detectadas pelos robôs (posicionadas no interior da região de detecção dos robôs) são consideradas para a próxima etapa do algoritmo. Assume-se que os objetos não detectados (fora da região de detecção) são completamente desconhecidos aos robôs.

Este primeiro procedimento da etapa (i) destaca a importância da região de detecção para obter o conhecimento do ambiente. Para a estratégia híbrida de planejamento de rota (capítulo 4), todos os objetos localmente detectados pela leitura das ARTags são considerados

como obstáculos e oferecem um risco de colisão, ou seja, os robôs devem desviar dos alvos atribuídos a ele após coletá-los, desviar dos alvos que não constam em sua lista de tarefas e desviar de outros robôs (obstáculos dinâmicos). Por outro lado, na estratégia de planejamento bioinspirada em feromônios artificiais (capítulo 5), as ARTags detectadas desta maneira devem ser diferenciadas entre alvos e robôs. Quando representar um alvo, ele pode ser considerado para a visita ou se tornar um obstáculo estático (se o alvo não for atribuído a um robô), de acordo com as decisões tomadas localmente pelo SMR, enquanto as ARTags associadas aos robôs são sempre obstáculos dinâmicos.

Para que ocorra esta interação, cada robô publica duas listas em tópicos ROS individuais: uma é composta pelas ARTags que representam outros robôs, denominada L1, e a segunda contém a posição de todos os outros elementos detectados (alvos e/ou obstáculos), denominada L2. Essas listas podem ser acessadas pelos robôs envolvidos no processo de alocação de tarefas (política de negociação de alvos). Estes procedimentos servem como base para que a comunicação direta seja executada entre os robôs, permitindo-lhes compartilhar informações, incluindo o aspecto transitivo, descrito no capítulo 5 (Figura 33).

A comunicação direta estabelecida é exemplificada na Figura 57. Neste caso, três robôs (R1, R2 e R3) publicam suas listas (L1 e L2) em seus respectivos tópicos ROS, representado pelas linhas tracejadas em azul. Como R2 é detectado por R1, e vice-versa, eles são capazes de compartilhar suas listas: R1 e R2 acessam as listas publicadas pelo outro, representado pelas linhas tracejadas em vermelho. O robô R3 apenas publica suas listas e, por não detectar nenhum outro robô (e também não ser detectado), sua lista de detecções não pode ser acessadas por outros robôs.

Para a detecção de feromônios (mecanismo do planejamento bioinspirado), a posição de um robô no ambiente é convertida para a posição equivalente em uma célula de um *grid map* criado. Assim, cada robô faz a leitura das células ao seu redor, ou seja, que estão no interior de sua região de detecção e toma suas decisões de acordo com a intensidade detectada de feromônio.

A respeito da estratégia bioinspirada em feromônios artificiais, conforme descrição no capítulo 5, os sinais para a comunicação indireta são representados neste ambiente real por meio de um *grid map*, executado pelos pacotes ROS `OccupancyGrid` e `costmap_2d` (WIKI, 2018b). Esta *grid* utiliza o sistema de referência definido anteriormente e possui uma resolução de $0,02\text{ m/cel}$. Cada célula desta *grid* é utilizada para armazenar a intensidade dos sinais de feromônio, permitindo a sua leitura por meio de tópicos ROS apropriados.

Em seguida, na etapa (ii), os dados obtidos das detecções são usados para executar a

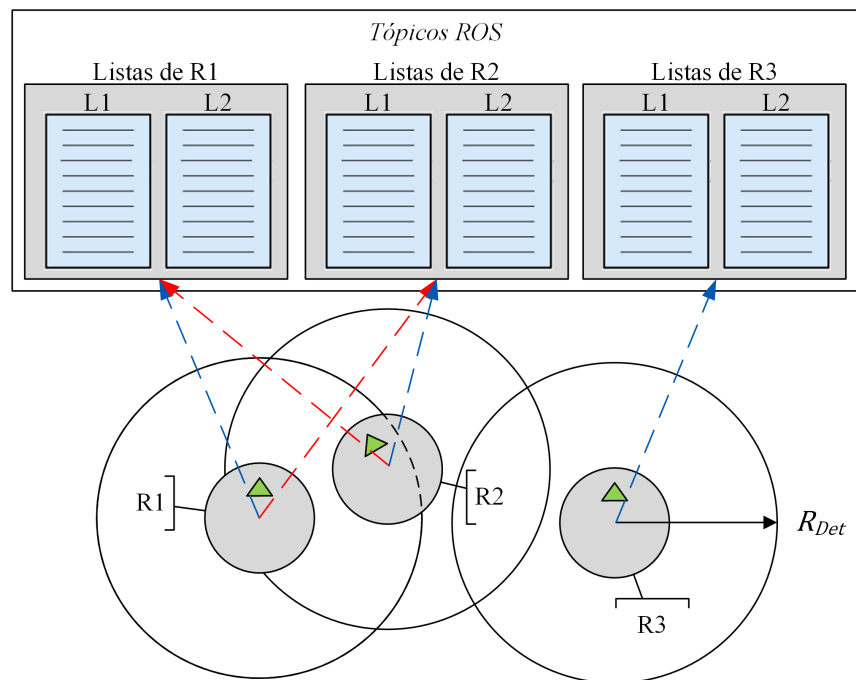


Figura 57: Exemplo da comunicação direta na arquitetura proposta: compartilhamento de informações entre robôs.

Fonte: Autoria própria.

estratégia de planejamento de rota, como descritas nos capítulos 4 e 5. Para a estratégia híbrida, devido as limitações de tamanho do ambiente real, assume-se que cada robô sabe sua posição atual no ambiente em todo instante e, portanto, a navegação cega não ocorre nos experimentos conduzidos com os robôs reais. Entretanto, os robôs frequentemente derrapam devido ao seu formato esférico, o que causa desvios nas rotas calculadas. Tais erros são corrigidos assim que são detectados, usando os mesmos procedimentos dos casos de navegação cega. Em relação ao planejamento bioinspirado em feromônios, todas as decisões de navegação (D_1 , D_2 , D_3 , D_4 e D_5 – Figura 36) são consideradas nesta implementação. O sistema de locomoção *fuzzy*, com todos os seus controladores e decisões (Figura 7), é implementado individualmente em todos os *scripts* dos robôs.

Finalmente, na etapa (*iii*), as ações de controle resultante são publicadas nos tópicos ROS relacionados ao mecanismo de locomoção dos robôs, por meio do uso do pacote ROS `sphero_swarm`.

6.3 RESULTADOS EXPERIMENTAIS

A arquitetura baseada em realidade aumentada, desenvolvida neste capítulo, é usada para implementar as estratégias de planejamento de rotas propostas nesta tese. Os resultados obtidos são apresentados nesta seção em duas partes: na “Parte 1” são descritos os resultados da estratégia híbrida de planejamento; e na “Parte 2” os resultados da estratégia bioinspirada em feromônios. Para uma melhor clareza da movimentação das ARTags durante a navegação dos robôs, as imagens do sistema de realidade aumentada são omitidas na “Parte 1”, em que apenas *frames* de uma filmagem do ambiente real são apresentados. As telas de realidade aumentada são mostradas na “Parte 2”.

6.3.1 PARTE 1: IMPLEMENTAÇÃO DA ESTRATÉGIA HÍBRIDA DE PLANEJAMENTO DE ROTAS

A estratégia de planejamento de rotas considerada nesta seção é validada por meio de dois cenários. O primeiro objetiva mostrar as habilidades de navegação de um único robô, isto é, as habilidades de seguir uma rota calculada previamente, desviar de obstáculos, calcular uma nova rota localmente e retomar a original assim que possível. O segundo considera 4 robôs (R1 a R4) e 6 alvos (A1 a A6) para validar o desempenho completo de navegação do SMR.

No primeiro caso avaliado, o planejamento calculado previamente (*off-line*) consiste em uma rota linear, da posição inicial de um robô até o seu objetivo. A evolução deste experimento é apresentado na Figura 58, a qual é dividida entre os três principais *frames* de um vídeo digital gravado pela câmera, entre os instantes $k = 20$ a $k = 45$. No *frame* 1, o robô R está localizado à esquerda, o objetivo à direita e o obstáculo entre os dois elementos do *frame*. A linha tracejada em azul é a rota planejada de maneira *off-line*, sem a informação do obstáculo, enquanto a linha pontilhada em vermelho é o caminho que, de fato, o robô executou. No *frame* 2, R realiza uma manobra para evitar a colisão com o obstáculo e, no *frame* 3, ele finalmente alcança a posição definida.

As ações de controle e a evolução da variável de decisão τ , aplicadas ao robô R, são apresentadas nas Figuras 59 (a) e (b), respectivamente. Os sinais de controle enviados ao robô Sphero são compostos por um vetor de velocidade (x, y) , denominado “CA_x” e “CA_y”.

No momento $k = 25$, como visto na Figura 59 (b), R detecta o obstáculo e a variável de decisão τ apresenta uma mudança abrupta. Um comportamento equivalente ocorre no vetor de velocidade (ações de controle), como mostrado na parte (a), o que implica em um desvio temporário da rota planejada frente ao obstáculo inesperado. Conforme o robô se aproxima de

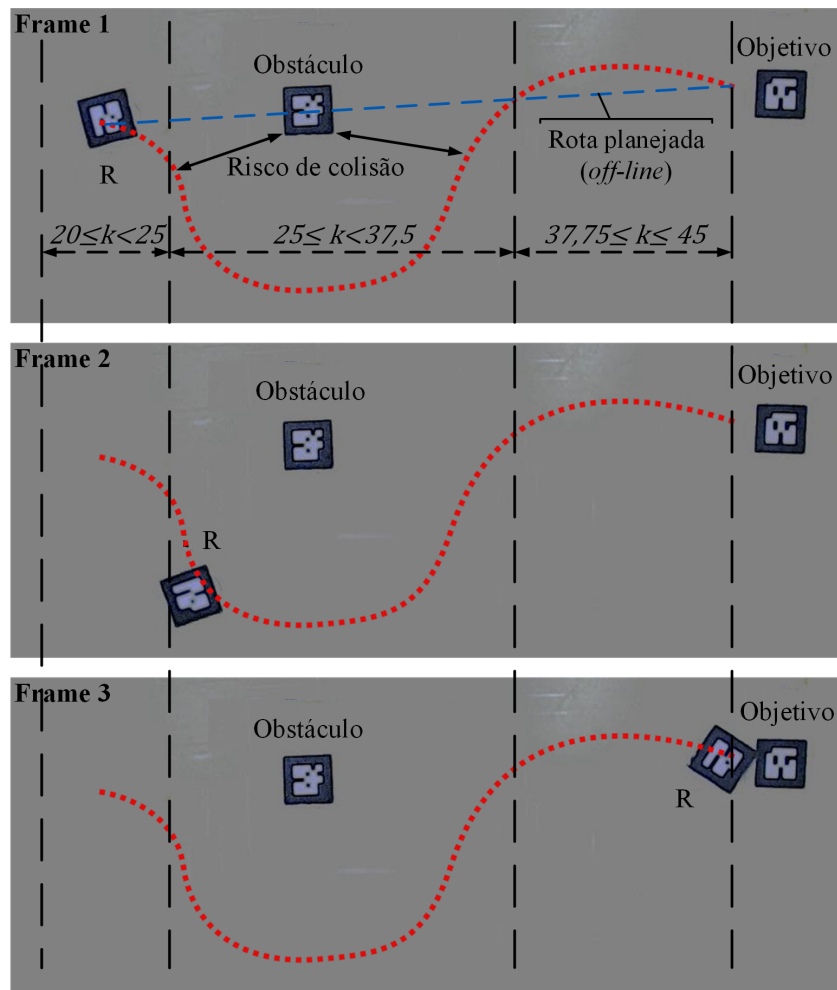


Figura 58: Frames de execução do primeiro cenário da parte 1.

Fonte: Autoria própria.

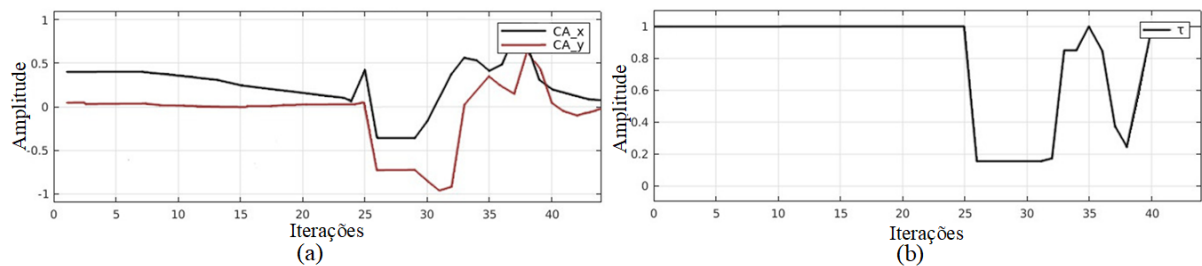


Figura 59: Desempenho local de um robô: (a) ações de controle e (b) evolução dinâmica de τ .

Fonte: Autoria própria.

sua rota original, no intervalo $35 < k < 37,5$, o obstáculo afeta o cálculo de τ novamente, uma vez que ele ainda está no interior da região de detecção do robô, entretanto a presença deste obstáculo é logo ignorada, pois não oferece risco de colisão para o restante de seu caminho. O comportamento descrito confirma que o sistema de navegação proposto desempenha como

esperado.

Neste experimento, nota-se um desvio importante da rota planejada, devido principalmente à derrapagem do robô durante sua movimentação. No entanto, o sistema de locomoção *fuzzy* é eficiente em levar o robô até a posição desejada.

Todas as habilidades de navegação podem ser observadas neste primeiro experimento: R detecta o obstáculo e calcula sua distância até ele; R é capaz de se autolocalizar no ambiente por meio do sistema de localização visual (ARTag); o sistema de locomoção da Figura 7 é usado para seguir uma rota calculada e evitar colisão com obstáculos.

No segundo cenário avaliado, o experimento considera um grupo composto por 4 robôs que devem visitar 6 alvos. Os robôs iniciam a navegação à esquerda do ambiente, cujas dimensões são de $1,5\text{ m} \times 2,5\text{ m}$, e devem concluir a tarefa no lado direito do cenário. As rotas calculadas pelo planejador genético podem ser observadas na Figura 60 e os 5 principais *frames* de um vídeo digital obtido pela câmera USB são mostrados na Figura 61. Para uma melhor compreensão, apenas as rotas navegadas por R1 (vermelho) e R4 (verde), as respectivas rotas planejadas (linha azul tracejada) e as posições finais (quadrados em preto) são destacados nos *frames* de execução.

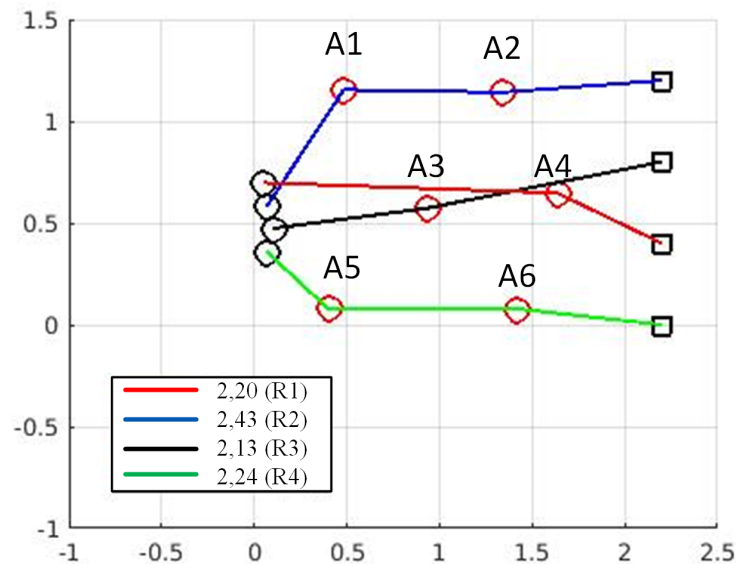


Figura 60: Rotas planejadas para os robôs R1, R2, R3 e R4.

Fonte: Autoria própria.

De acordo com as rotas planejadas, R1 deve visitar apenas o alvo A4 e, portanto desviar de A3, considerado como um obstáculo durante a navegação. De forma similar, R4 deve visitar os alvos A5 e A6. Na Figura 61, o *frame* 1 mostra a configuração inicial do experimento, quando

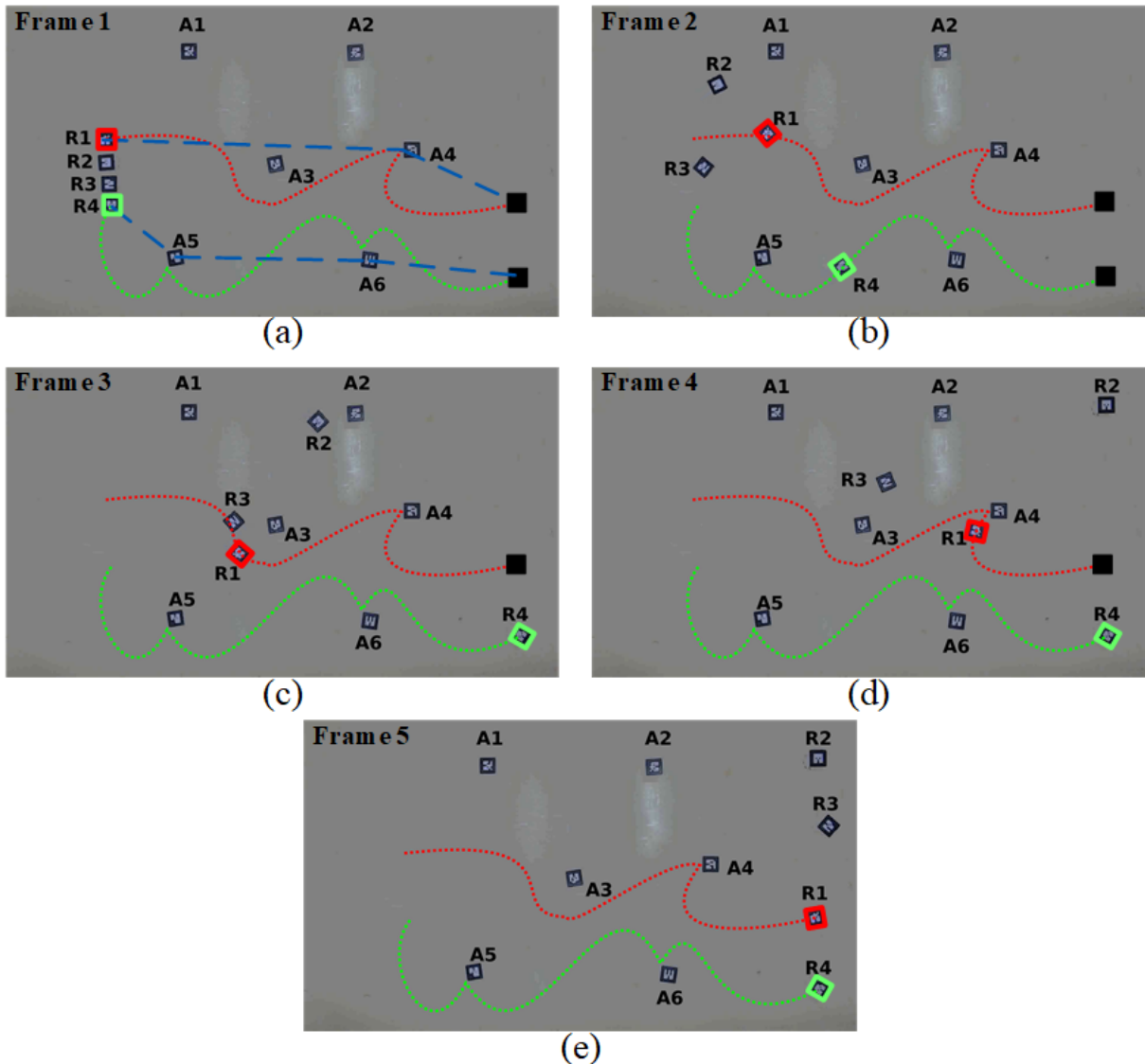


Figura 61: *Frames* de execução do segundo cenário da parte 1.

Fonte: Autoria própria.

todos os robôs estão alinhados na parte esquerda do ambiente. No *frame 2*, R4 retoma sua rota calculada após visitar A5, enquanto R1 se dirige ao alvo A4. Há uma aproximação entre os robôs R1 e R3 no *frame 3* e então é iniciada uma manobra de desvio entre eles. Como resultado deste comportamento, os caminhos de R1 e R3 se cruzam antes do planejado (ver Figura 60). Das rotas planejadas pelo algoritmo genético, os caminhos em vermelho e preto se cruzam para que R1 possa visitar o alvo A4, pois o planejamento *off-line* não considera as manobras de desvio entre os robôs. No *frame 4*, R1 visita A4 e se dirige para sua posição final. Finalmente, todos os robôs alcançam suas posições finais no lado direito do ambiente, e o experimento é finalizado no *frame 5*.

Durante este experimento, R4 desempenha suas tarefas mais rápido que R1, conforme é observado nos *frames* 3, 4 e 5. Isso ocorre por diversas razões: (i) o erro causado pelas derrapagens durante a navegação, combinado com a frequência de amostragem da câmera e do pacote ROS utilizado, promove um atraso no processo de obtenção da localização das ARTags; (ii) embora todos os robôs sejam do mesmo modelo, as respectivas velocidades dependem da versão do *firmware* e também do nível de bateria em cada robô; (iii) o ambiente não é totalmente controlado, portanto fatores externos e inesperados (claridade e sombras no interior da área experimental) podem influenciar na localização das ARTags; entre outros.

Os dois cenários apresentados nesta seção, embora simples, validam o planejador híbrido de rotas e comprovam a viabilidade da inserção de habilidades de comunicação e detecção virtuais, através da realidade aumentada.

6.3.2 PARTE 2: IMPLEMENTAÇÃO DA ESTRATÉGIA BIOINSPIRADA EM FEROMÔNIOS

Nesta seção, o sistema de realidade aumentada é utilizado e a navegação do SMR é analisada em dois cenários. No primeiro, utiliza-se apenas um robô, 4 alvos e uma posição final pré-definida. Ao contrário dos experimentos abordados na seção 6.3.1, o robô não possui o conhecimento da posição dos alvos e deve visitá-los conforme os encontra durante a navegação, de acordo com a sua capacidade de detecção. Em seguida, deve se dirigir à sua posição final. No segundo cenário, são analisadas as implementações virtuais das habilidades de comunicação dos robôs, composto pelo compartilhamento de informações por meio dos mecanismos de comunicação direta e indireta (feromônios artificiais). Nesse caso, são utilizados 3 robôs e 6 alvos.

A configuração inicial do primeiro cenário de experimentos é apresentado na Figura 62 (a), onde há o robô R (quadrado verde), 4 alvos com posições desconhecidas pelo robô (A1 a A4) e uma determinada posição final (quadrados rosa). O sistema de realidade aumentada, executado pelo pacote *RViz*, auxilia na identificação da região de detecção do robô (círculo que envolve a ARTag), enquanto a seta azul indica o atual objetivo do robô.

Conforme descrito no capítulo 5, o robô deve visitar a maior quantidade de alvos possível, evitar colisão com obstáculos e terminar a navegação na posição final. Os alvos são representados como obstáculos estáticos conforme são visitados pelo robô. Os *frames* apresentados ao longo das Figuras 62 (a) a (f) mostram a evolução da navegação do robô até que ele alcance sua posição final.

Enquanto o robô (R) está em sua posição inicial (Figura 62 (a)), ele não detecta alvos

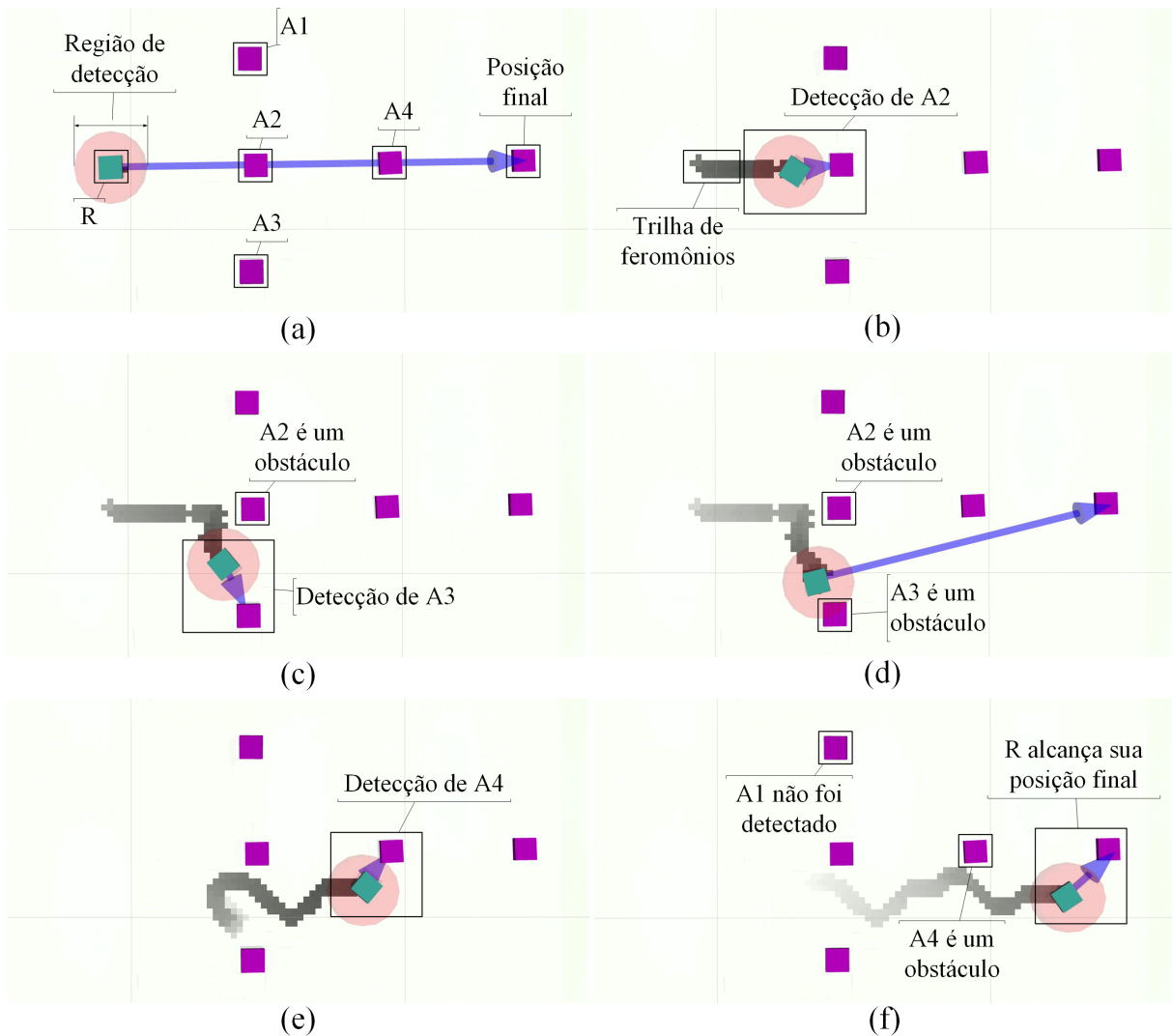


Figura 62: Frames de execução do primeiro cenário da parte 2.

Fonte: Autoria própria.

e, portanto seu objetivo (rota calculada) é dirigir-se a posição final. Conforme R segue sua rota, A2 é detectado e uma manobra em sua direção é planejada, conforme mostrado na Figura 62 (b). Após R visitar seu primeiro alvo (Figura 62 (c)), A2 é considerado como um obstáculo estático e o robô deve desempenhar uma manobra de desvio pelo seu lado direito para evitar esta colisão. O robô detecta o alvo A3, como consequência deste desvio, e segue para visitá-lo. Após estes eventos, na Figura 62 (d), R calcula sua próxima rota em direção a posição final, pois nenhum alvo é detectado novamente. Durante a execução deste caminho, A4 é encontrado e R recalcula sua caminho para visitar este alvo, conforme é apresentado na Figura 62 (e). Nessa ocasião, também é possível observar por meio da trilha de feromônio deixada por R que ele desvia de A2, visitado anteriormente. Porém, os sinais de feromônios lançados por R neste experimento não influenciam em suas decisões, apenas facilitam a visualização de sua rota desempenhada,

por meio do sistema de realidade aumentada. Finalmente, a Figura 62 (e) mostra que R evita colisão com A4 após visitá-lo e finaliza a sua navegação na posição final pré-determinada. A1 não foi detectado durante esta navegação e, portanto o robô não visitou este alvo.

A evolução da variável de tomada de decisão τ , relacionada a sequência de *frames* apresentada ao longo das Figuras 62 (a) a (f), é apresentada na Figura 63. Os momentos em que R desempenha manobras abruptas para evitar colisão com obstáculos ($\tau = 0$) são destacados por meio dos retângulos tracejados. Estas alterações abruptas ocorrem logo quando o robô visita um determinado alvo e, então ele já se transforma em um obstáculo, ainda no momento em que o robô está na posição alcançada por esta aproximação. Por outro lado, quando R desvia de A2 pela segunda vez, como destacado na Figura 62 (e), esta detecção de obstáculo ocorre com uma distância segura e, portanto τ apresenta valores intermediários e sem alterações abruptas. A influência do valor de τ nas ações de controle aplicadas ao robô Sphero pode ser observada na Figura 64.

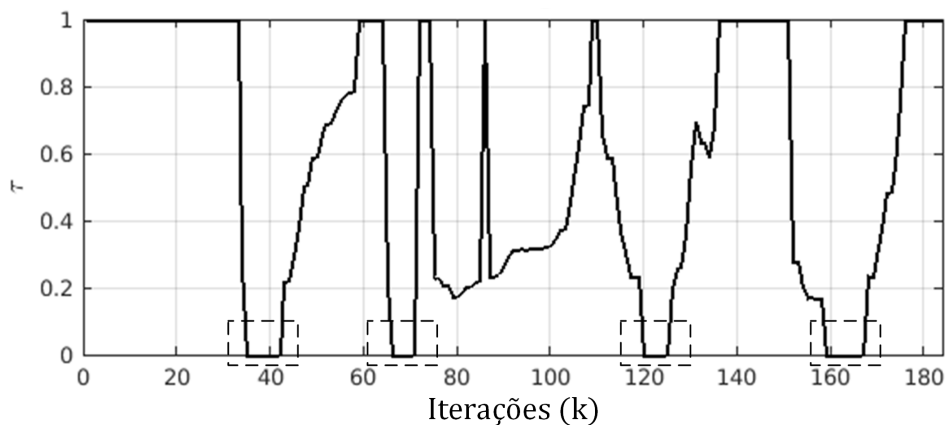


Figura 63: Evolução da variável de tomada de decisão τ : experimentos da parte 2.

Fonte: Autoria própria.

Os resultados da navegação neste primeiro cenário mostram a capacidade da arquitetura proposta em fornecer virtualmente as habilidades necessárias ao robô Sphero, que apresenta severas limitações sensoriais.

No segundo cenário, múltiplos robôs Sphero compõem um SMR e o comportamento de estigmergia para a exploração cooperativa é avaliado. São implementados todos os mecanismos de comunicação direta e indireta requeridos para esta estratégia de planejamento, como descritos nos capítulos 3 e 5, assim como decisões de navegação sintetizadas na Figura 36. Este experimento possui sua configuração inicial apresentada na Figura 65 (a), onde 3 robôs (R1, R2 e R3) devem navegar no interior de um ambiente com 6 alvos de localização desconhecida (A1

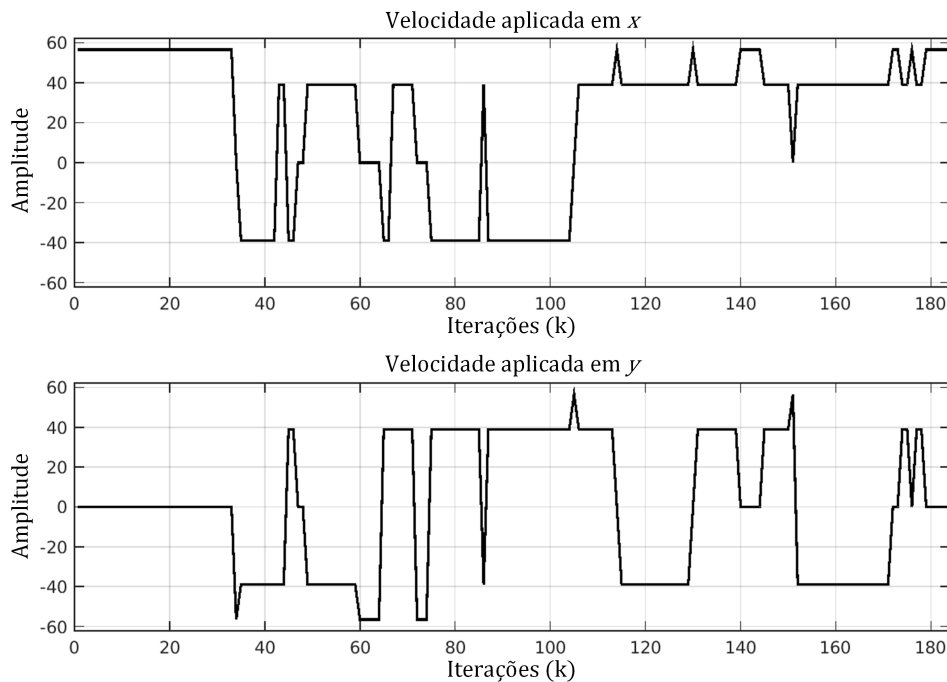


Figura 64: Ações de controle aplicada ao robô R: experimentos da parte 2.

Fonte: Autoria própria.

a A6) e alcançar as posições finais pré-determinadas para concluir os objetivos de navegação. Os resultados desta exploração são apresentados ao longo dos *frames* de execução da Figura 65, de (a) a (i).

Inicialmente, uma alta concentração de feromônios lançados na posição inicial de cada robô implica em um comportamento de dispersão entre eles, como observado nas rotas calculadas (setas em azul) da Figura 65 (a). Neste caso, todos os robôs se comunicam de forma direta para o compartilhamento da decisão tomada por cada um e então ocorre o seguinte comportamento no SMR: R1 e R3 tomam a decisão de explorar o ambiente (ativação de D_2) e calculam caminhos em direções opostas a detecção de R2; e R2 mantém sua rota calculada em direção à sua posição final, pois ele detecta a presença de robôs em ambos os lados (R1 e R3). A decisão D_4 (desvio de feromônios) também é ativada para todos os robôs.

Após este comportamento inicial, os robôs tomam distâncias entre eles e não há mais a detecção de outros robôs mencionadas anteriormente. Por isso, todos calculam suas rotas em direção a posição final (decisão D_3), como apresentado na Figura 65 (b), pois nenhum alvo foi detectado até este momento.

A primeira detecção ocorre na Figura 65 (c), onde R2 detecta A1 e, portanto a decisão

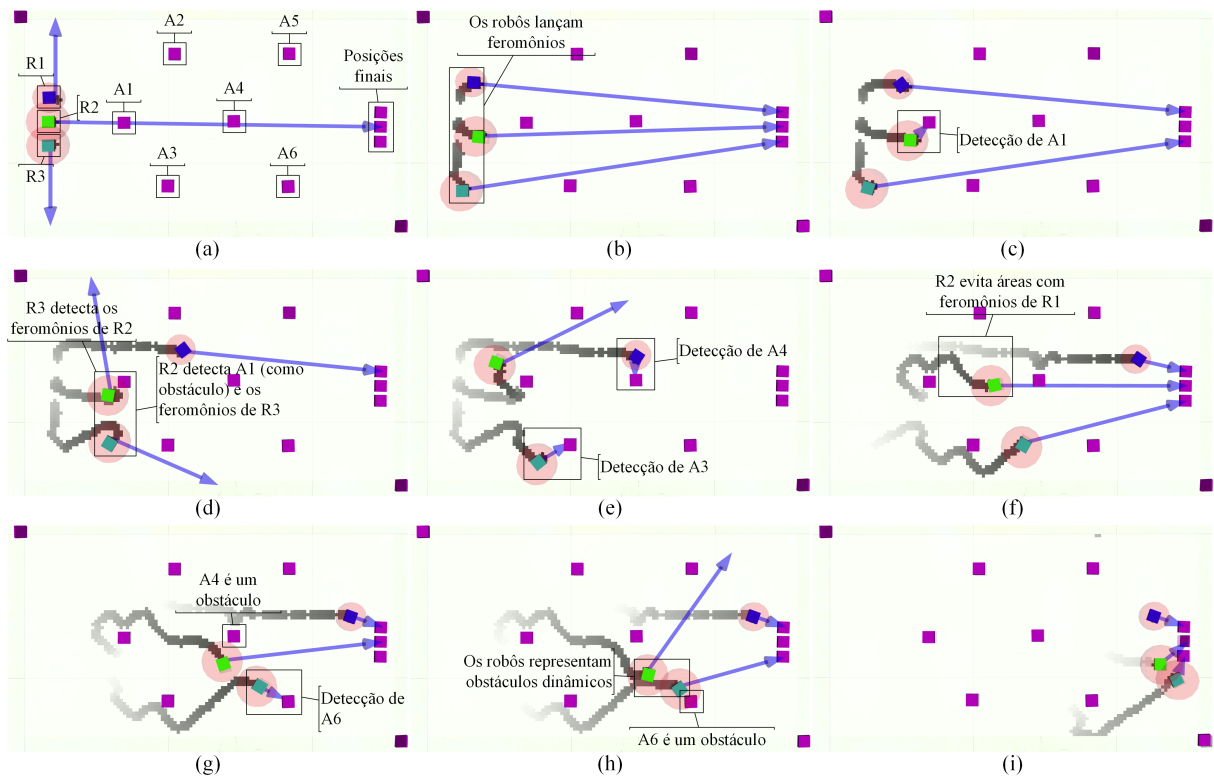


Figura 65: Frames de execução do segundo cenário da parte 2.

Fonte: Autoria própria.

D_1 é ativada. R1 e R3 mantêm suas rotas calculadas anteriormente.

Na Figura 65 (d), os sinais de feromônio influenciam as decisões de R2 e R3. Especialmente para R2, dois eventos importantes ocorrem de forma simultânea: R2 deve evitar colisão com A1 enquanto detecta os sinais de feromônio lançados por R3. Dado que não há outro alvo detectado por R2, a decisão D_3 é ativada para calcular uma rota até sua posição final, enquanto a decisão D_4 ajusta este caminho para evitar as áreas exploradas por R3. Entretanto, devido ao elevado risco de colisão entre R2 e A1, a decisão reativa (D_5) para evitar colisão com obstáculos também é ativada e predomina na rota resultante de R2. Por outro lado, R3 detecta os sinais de feromônio lançados por R2 e ajusta sua rota para tomar distância desta área detectada. R1 não detecta A2 durante este *frame* da gravação e mantém seu caminho até a posição final.

Na Figura 65 (e), R1 detecta A4, R2 continua realizando a manobra de desvio mencionada e R3 detecta A3.

Após os robôs R1 e R3 visitarem A4 e A3, respectivamente, a Figura 65 (f) mostra que estes robôs retomam as rotas até a posição final, enquanto a trilha de feromônios deixada por R2 confirma um ajuste de rota pela decisão D_4 , devido a detecção de feromônios lançados por

R1 e a decisão D_3 é novamente ativada após este desvio.

Na Figura 65 (g), R1 alcança sua posição final e não detecta A5. Simultaneamente, R2 desempenha uma manobra para evitar colisão com A4, visitado anteriormente por R1, e R3 detecta A6.

Após R3 visitar A6, este robô deve lidar com a aproximação de R2 (obstáculo dinâmico) e também com a presença de A6 (obstáculo estático), como é mostrado na Figura 65 (h). Como uma consequência desta situação, a Figura 65 (i) mostra que R3 decide desempenhar uma rota pela sua direita para então alcançar sua posição final.

Finalmente, todos os robôs alcançam suas posições finais na Figura 65 (i). As decisões deliberativas (Figura 36) promovem o comportamento de estigmergia apropriado durante a navegação do SMR. Dois alvos (A2 e A5), que se encontram na região superior do ambiente, não foram detectados e, portanto, não são visitados pelos robôs. Para que todos os alvos fossem visitados, seria necessário aumentar o número de robôs no SMR ou aumentar sua área de detecção.

Os experimentos visualizados pelo sistema de realidade aumentada permite confirmar o funcionamento básico dos mecanismos de detecção e comunicação (direta e indireta), requisitados pela estratégia de planejamento de rota bioinspirada, antes de uma possível aquisição ou prototipagem dos mesmos. Toda a evolução dos robôs, durante a navegação, pôde ser analisada por meio da arquitetura desenvolvida.

7 CONCLUSÕES

Nesta tese são investigados problemas que envolvem o planejamento de rota e a navegação autônoma de um sistema multirrobo. A partir de cenários com diferentes características e requisitos, duas estratégias são apresentadas. Um sistema *fuzzy* de navegação também é desenvolvido para conferir aos robôs a capacidade de seguir as rotas calculadas pelos planejadores propostos e desviar de obstáculos para preservar sua integridade física.

Inicialmente, um ambiente semi-desconhecido é considerado, onde os pontos de passagem obrigatórios (alvos) são conhecidos previamente mas os obstáculos são desconhecidos. Assim, um planejador híbrido é idealizado, com etapas *off-line* (executada antes de iniciar a navegação dos robôs) e *on-line* (cada robô executa localmente), com o objetivo de garantir que todos os alvos sejam visitados de forma cooperativa e que os robôs devem desviar dos obstáculos durante a navegação.

A etapa *off-line* é inspirada no problema do caixeiro viajante múltiplo e solucionado por um algoritmo genético, que atribui as rotas aos robôs. Entretanto, as rotas obtidas podem oferecer caminhos que comprometem uma navegação segura dos robôs, devido a presença de obstáculos estáticos e dinâmicos (outros robôs se movimentando) no ambiente, todos com posições desconhecidas. Com isso, os robôs devem executar manobras de desvio de obstáculos, realizando caminhos diferentes dos planejados. Para solucionar estes problemas, um planejador *on-line* é desenvolvido e executado por cada robô, com os seguintes objetivos: (i) quando o robô desviar de sua rota original, mas ainda possuir o conhecimento de sua posição no ambiente, calcular um novo caminho até o seu próximo objetivo; e (ii) quando há incertezas em sua posição, devido a erros de odometria, executar o procedimento de autolocalização baseado no algoritmo da triangulação.

O planejamento executado *off-line* é validado por meio de quatro cenários simulados, onde o número de alvos e robôs variam. Os resultados mostram que esta abordagem é capaz de distribuir as rotas entre os robôs, de tal forma que nenhum deles fique sobrecarregado, ou seja, com a garantia de um equilíbrio nos valores de distância navegadas e/ou no número de alvos

atribuídos a cada um deles.

O planejador *on-line* também é validado com experimentos simulados. Todos os robôs são capazes de seguir as rotas calculadas pelo planejador *off-line* e recalculá-las quando necessário. Entretanto, o desempenho desta etapa *on-line* é diretamente influenciada pelas habilidades dos robôs. O tamanho da região de detecção, relacionada à habilidade de comunicação direta, por exemplo, é decisiva para uma satisfatória autolocalização, isto é, uma ampla capacidade de detecção implica em amplas chances de encontrar os *landmarks* necessários para executar o algoritmo da triangulação. Por outro lado, quando um robô navega sob a condição de navegação cega, seu desempenho é afetado devido à excessiva procura pelos *landmarks* para conseguir se localizar e retomar sua rota planejada.

Também é apresentado um estudo comparativo com outros dois planejadores encontrados na literatura, ambos baseados em algoritmo genético. Os resultados comparativos confirmam a eficiência da estratégia proposta, principalmente quando a navegação dos robôs objetiva uma redução na distância a ser navegada por cada membro do SMR, assim como uma distribuição equalizada de alvos entre os robôs.

Em seguida, um segundo problema é considerado, que corresponde a tarefa de exploração de ambientes em que as posições dos alvos e dos obstáculos são desconhecidas. Nesse caso, é proposto um planejador de rotas *on-line* e bioinspirado, com execução distribuída. Esta estratégia fornece um mecanismo de tomada de decisão que permite aos múltiplos robôs explorarem o ambiente cooperativamente em busca dos alvos e finalizarem a navegação em um ponto final pré-definido. Basicamente, o planejador proposto confere a um time de robôs, dotados de habilidades de sensoriamento e comunicação limitadas, um sistema de tomadas de decisão baseadas em conhecimentos locais do ambiente e compartilhamento de informações. Por meio das comunicações direta e indireta desempenhada pelos robôs, o SMR alcança uma distribuição de tarefas equilibrada e um comportamento de estigmergia para a exploração.

Resultados de simulação são apresentados para a análise individual do comportamento dos robôs e também das ações cooperativas do SMR como um todo. O desempenho global do SMR é avaliado baseado em métricas de custo, eficiência e eficácia, considerando diferentes situações para as dimensões do ambiente, número de robôs e alvos.

As análises do comportamento individual mostram que os robôs são capazes de tomar decisões autônomas para explorar áreas ainda não visitadas por outros membros do SMR e também decidir cooperativamente sobre a distribuição de tarefas entre eles.

O comportamento cooperativo do SMR foi analisado em seis diferentes cenários, onde

os alvos são igualmente espaçados em quatro deles, enquanto os outros dois apresentam uma distribuição de alvos concentrada em áreas específicas do ambiente (centro e canto inferior direito). As métricas calculadas indicam a factibilidade do planejador proposto. Para a maioria dos cenários avaliados, o SMR cumpriu seus objetivos de explorar a superfície do ambiente e encontrar/visitar o maior número de alvos possível.

O desenvolvimento do planejador foi baseada na suposição de que todos os robôs conhecem sua posição no ambiente durante toda a simulação. Assim, eles sempre são capazes de calcular a distância entre eles e sua posição final. Entretanto, esta suposição nem sempre é possível em algumas aplicações e/ou para alguns tipos de robôs. Além disso, o desempenho do SMR é afetado pela escolha de parâmetros, tais como o número de robôs no SMR, o tamanho da região de detecção, da área de liberação de feromônios e também do parâmetro que pondera a decisão entre continuar explorando o ambiente ou seguir para a sua posição final. Uma escolha insatisfatória destes parâmetros pode resultar em uma perda de eficácia e/ou eficiência, especialmente para ambientes com dimensões maiores.

A utilização real das estratégias desenvolvidas requer um time de robôs com habilidades que envolvam detecção e identificação de objetos (alvos, outros robôs e sinais de feromônios lançados por eles), um mecanismo para lançar feromônios no ambiente e um sistema de localização que permite os robôs determinarem sua posição durante toda a exploração. Esta realidade nem sempre é possível, pois exigiria robôs sofisticados e dotados de vários dispositivos de *hardware* e de capacidade de processamento embarcado.

Assim, para possibilitar a experimentação real de um SMR formado por robôs simples, uma arquitetura baseada em realidade aumentada foi desenvolvida, que suporta experimentos envolvendo múltiplos robôs, alvos e obstáculos. Todas as habilidades requeridas para a execução das estratégias de planejamento (detecção e comunicação direta/indireta) são conferidas virtualmente aos membros do SMR, viabilizando o uso de robôs reais que possuam severas limitações em seus periféricos de detecção e comunicação.

Como exemplo de utilização desta arquitetura experimental, um time composto por robôs Spheros foi considerado nestes experimentos. Estes robôs possuem um formato esférico e contam somente com as habilidades de locomoção em um ambiente plano e de comunicação via *Bluetooth* com o computador para receber comandos.

Os resultados com os robôs reais confirmam a factibilidade de ambas as estratégias avaliadas. Todos os robôs foram capazes de seguir uma rota definida, desviar de obstáculos e alcançar os requisitos de cooperação, específicos para cada implementação. Entretanto, dado que a arquitetura utilizada para os experimentos é baseada em um sistema de localização

visual (marcadores ARTags identificados por uma câmera), o algoritmo de autolocalização pela triangulação não foi considerado na implementação da estratégia híbrida. Por outro lado, os robôs Sphero desenvolvem sua locomoção com derrapagens que os fazem divergir da rota planejada, assim como ocorre quando um robô está sob a condição de navegação cega (elevada incerteza sobre sua localização) e, assim, esta situação também é corrigida com os procedimentos de ajuste de rota. Uma outra situação ocorre quando perde-se temporariamente a leitura das ARTags e, conseqüentemente, a posição de todos os elementos inseridos no cenário, devido a sombras próximas ao ambiente de experimento.

Portanto, conclui-se que as estratégias de planejamento de rota propostas nesta tese atendem os objetivos geral e específicos definidos anteriormente, ou seja, conferem a múltiplos robôs inseridos em ambientes semi-desconhecidos, que possuem limitações de comunicação/detecção, um sistema autônomo de navegação com capacidade de planejamentos *off-line* (com ajustes de rota) e *on-line*. Pôde-se perceber também que o desempenho dos planejadores indica possíveis utilizações em tarefas que vão além da visita de alvos, como por exemplo a exploração para fins de mapeamento de um ambiente desconhecido, um sistema de logística para visita a pontos de trabalho em uma indústria, a busca e identificação de recursos importantes para o SMR, entre outras.

7.1 TRABALHOS FUTUROS

Para os trabalhos futuros da estratégia híbrida de planejamento de rota, pretende-se analisar a factibilidade do planejador *off-line* considerando outras restrições relacionadas a intersecção de rotas e condições de navegação segura. Novos experimentos mais realísticos devem ser considerados para promover outras características de navegação, tais como: controle de formação, transporte cooperativo de cargas, formação líder-seguidor, entre outras.

Em relação ao planejador bioinspirado, os trabalhos futuros serão focados nos seguintes pontos: (i) sintonia adaptativa dos parâmetros de decisão para mitigar os efeitos de uma escolha insatisfatória de valores; e (ii) métodos de autolocalização para viabilizar a suposição de que os robôs sempre conhecem sua posição no ambiente.

Para os experimentos com robôs reais, pretende-se utilizar uma área experimental que possua uma dimensão maior da que foi utilizada nesta tese, a fim de possibilitar a implementação da estratégia de localização e outras abordagens de planejamento de rota. Além disso, pretende-se realizar experimentos que incluam robôs e obstáculos reais e simulados, simultaneamente em um mesmo ambiente, por meio do sistema de realidade aumentada.

REFERÊNCIAS

- ABADI, D. N. M.; KHOOBAN, M. H. Design of optimal mamdani-type fuzzy controller for nonholonomic wheeled mobile robots. **Journal of King Saud University-Engineering Sciences**, Elsevier, v. 27, n. 1, p. 92–100, 2015.
- AGGARWAL, S.; GARG, R.; GOSWAMI, P. A review paper on different encoding schemes used in genetic algorithms. **International Journal on Advanced Research in Computer Science and Software Engineering**, v. 4, p. 496–600, 2014.
- ARVIN, F. et al. Investigation of cue-based aggregation in static and dynamic environments with a mobile robot swarm. **Adaptive Behaviour**, v. 24, n. 2, p. 102–118, 2016.
- BAYINDIR, L. A review of swarm robotics tasks. **Neurocomputing**, v. 172, p. 292–321, 2016.
- BECHAR, A.; VIGNEAULT, C. Agricultural robots for field operations. part 2: Operations and systems. **Biosystems Engineering**, v. 153, p. 110 – 128, 2017. ISSN 1537-5110.
- BRAMBILLA, M. et al. Swarm robotics: a review from the swarm engineering perspective. **Swarm Intelligence**, Springer US, v. 7, n. 1, p. 1–41, 2013.
- CAI, Y.; YANG, S. X. A pso-based approach with fuzzy obstacle avoidance for cooperative multi-robots in unknown environments. **International Journal of Computational Intelligence and Applications**, v. 15, n. 01, 2016.
- CALVO, R. et al. A bioinspired coordination strategy for controlling of multiple robots in surveillance tasks. **International Journal on Advances in Software**, v. 5, p. 146–165, 2012.
- CARTER, A. E.; RAGSDALE, C. T. A new approach to solving the multiple traveling salesperson problem using genetic algorithms. **European Journal of Operational Research**, v. 175, n. 1, p. 246 – 257, 2006.
- CASTILLO, O. et al. A new approach for dynamic fuzzy logic parameter tuning in ant colony optimization and its application in fuzzy control of a mobile robot. **Applied Soft Computing**, v. 28, p. 150 – 159, 2015. ISSN 1568-4946.
- COLARES, R. G.; CHAIMOWICZ, L. A novel distance cost approach for multi-robot integrated exploration. In: **2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR)**. 2015. p. 192–197.
- COLARES, R. G.; CHAIMOWICZ, L. The next frontier: Combining information gain and distance cost for decentralized multi-robot exploration. In: **Proceedings of the 31st Annual ACM Symposium on Applied Computing**. 2016. (SAC '16), p. 268–274. ISBN 978-1-4503-3739-7.
- COUCEIRO, M. S. et al. Benchmark of swarm robotics distributed techniques in a search task. **Robotics and Autonomous Systems**, v. 62, n. 2, p. 200–213, 2014.

- DAS, P.; BEHERA, H.; PANIGRAHI, B. A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning. **Swarm and Evolutionary Computation**, v. 28, p. 14 – 28, 2016. ISSN 2210-6502.
- DORIGO, M.; BIRATTARI, M.; STÜTZLE, T. Ant colony optimization – artificial ants as a computational intelligence technique. **IEEE Computational Intelligence Magazine**, v. 1, p. 28–39, 2006.
- ESTEVEZ, J. S.; CARVALHO, A.; COUTO, C. Generalized geometric triangulation algorithm for mobile robot absolute self-localization. In: **2003 IEEE International Symposium on Industrial Electronics**. 2003. v. 1, p. 346–351.
- EVERSHAM, J. D.; RUIZ, V. F. Experimental analysis of the reynolds flocking model. **Paladyn**, v. 2, n. 3, p. 145–155, 2011.
- FUJIMORI, A. et al. Autonomous flight control system of quadrotor and its application to formation control with mobile robot. **IFAC-PapersOnLine**, v. 51, n. 22, p. 343 – 347, 2018. ISSN 2405-8963. 12th IFAC Symposium on Robot Control SYROCO 2018.
- GAZI, V.; PASSINO, K. M. **Swarm Stability and Optimization**. 1st. ed. USA: Springer Publishing Company, Incorporated, 2011. ISBN 364218040X, 9783642180408.
- GERKEY, B. P.; MATARIĆ, M. J. A formal analysis and taxonomy of task allocation in multi-robot systems. **The International Journal of Robotics Research**, v. 23, n. 9, p. 939–954, 2004.
- HAN, Z. et al. Multi-agv path planning with double-path constraints by using an improved genetic algorithm. **PLOS ONE**, Public Library of Science, v. 12, n. 7, p. 1–16, 07 2017.
- HATTORI, K. et al. Generalized measuring-worm algorithm: high-accuracy mapping and movement via cooperating swarm robots. **Artificial Life and Robotics**, Springer, v. 21, n. 4, p. 451–459, 2016.
- HECKER, J. P.; MOSES, M. E. Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms. **Swarm Intelligence**, v. 9, n. 1, p. 43–70, 2015.
- HONG, C.; PARK, C. W.; KIM, J. H. Evolutionary dual rule-based fuzzy path planner for omnidirectional mobile robot. In: **2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)**. 2016. p. 767–774.
- HÖNIG, W. et al. Mixed reality for robotics. In: **2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**. 2015. p. 5382–5387.
- JADBABAIE, A.; LIN, J.; MORSE, A. S. Coordination of groups of mobile autonomous agents using nearest neighbor rules. **IEEE Transactions on Automatic Control**, v. 48, n. 6, p. 988–1001, 2003. ISSN 0018-9286.
- JOSE, K.; PRATHIHAR, D. K. Task allocation and collision-free path planning of centralized multi-robots system for industrial plant inspection using heuristic methods. **Robotics and Autonomous Systems**, v. 80, p. 34 – 42, 2016. ISSN 0921-8890.
- KHAMIS, A.; ELGINDY, A. Minefield mapping using cooperative multirobot systems. **Journal of Robotics**, v. 2012, p. 1–17, 2012.

- KHAMIS, A.; HUSSEIN, A.; ELMOGY, A. Multi-robot task allocation: A review of the state-of-the-art. In: KOUBÂA, A.; DIOS, J. Martínez-de (Ed.). **Cooperative Robots and Sensor Networks 2015**. Cham: Springer International Publishing, 2015. p. 31–51. ISBN 978-3-319-18299-5.
- KISDI, A.; TATNALL, A. R. L. Future robotic exploration using honeybee search strategy: Example search for caves on mars. **Acta Astronautica**, v. 68, n. 11–12, p. 1790 – 1799, 2011.
- KUYUCU, T.; TANEV, I.; SHIMOHARA, K. Superadditive effect of multi-robot coordination in the exploration of unknown environments via stigmergy. **Neurocomputing**, Elsevier B.V., v. 148, p. 83–90, 2015. ISSN 0925-2312.
- LADOSZ, P.; OH, H.; CHEN, W.-H. Trajectory planning for communication relay unmanned aerial vehicles in urban dynamic environments. **Journal of Intelligent & Robotic Systems**, 2017. ISSN 1573-0409.
- LARSEN, L. et al. Path planning of cooperating industrial robots using evolutionary algorithms. **Procedia Manufacturing**, v. 17, p. 286 – 293, 2018. ISSN 2351-9789.
- LEE, S. K.; FEKETE, S. P.; MCLURKIN, J. Structured triangulation in multi-robot systems: Coverage, patrolling, voronoi partitions, and geodesic centers. **International Journal of Robotics Research**, v. 35, n. 10, p. 1234–1260, 2016.
- LI, B. et al. Centralized and optimal motion planning for large-scale agv systems: A generic approach. **Advances in Engineering Software**, v. 106, p. 33 – 46, 2017. ISSN 0965-9978.
- LIMA, D. A.; OLIVEIRA, G. M. A cellular automata ant memory model of foraging in a swarm of robots. **Applied Mathematical Modelling**, v. 47, p. 551 – 572, 2017. ISSN 0307-904X.
- MAC, T. T. et al. Heuristic approaches in robot path planning: A survey. **Robotics and Autonomous Systems**, v. 86, p. 13 – 28, 2016. ISSN 0921-8890.
- MEHRJERDI, H.; SAAD, M.; GHOMMAM, J. Hierarchical fuzzy cooperative control and path following for a team of mobile robots. **IEEE/ASME Transactions on Mechatronics**, v. 16, n. 5, p. 907–917, Oct 2011.
- MILLARD, A. G. et al. Ardebug: An augmented reality tool for analysing and debugging swarm robotic systems. **Frontiers in Robotics and AI**, v. 5, p. 87, 2018. ISSN 2296-9144.
- MIMRÁNEK, J.; SMUTNÝ, P. Control of robotic ball using mobile device. In: **2016 17th International Carpathian Control Conference (ICCC)**. 2016. p. 499–504.
- NIETO-GRANDA, C.; ROGERS, I. J. G.; CHRISTENSEN, H. I. Coordination strategies for multi-robot exploration and mapping. **The International Journal of Robotics Research**, v. 33, n. 4, p. 519–533, 2014.
- OLFATI-SABER, R.; FAX, J. A.; MURRAY, R. M. Consensus and cooperation in networked multi-agent systems. **Proceedings of the IEEE**, v. 95, n. 1, p. 215–233, 2007. ISSN 0018-9219.
- PALMIERI, N. et al. Self-adaptive decision-making mechanisms to balance the execution of multiple tasks for a multi-robots team. **Neurocomputing**, v. 306, p. 17 – 36, 2018. ISSN 0925-2312.

- PARKER, L. E. Multiple mobile robot teams, path planning and motion coordination. In: MEYERS, R. A. (Ed.). **Encyclopedia of Complexity and Systems Science**. New York, NY: Springer New York, 2009. p. 5783–5800. ISBN 978-0-387-30440-3.
- PARKER, L. E.; RUS, D.; SUKHATME, G. S. Multiple mobile robot systems. In: SICILIANO, B.; KHATIB, O. (Ed.). **Springer Handbook of Robotics**. Cham: Springer International Publishing, 2016. p. 1335–1384. ISBN 978-3-319-32552-1.
- PASSINO, K. M.; YURKOVICH, S. **Fuzzy control**. California, USA: Addison-Wesley, 1998.
- PETITTI, A. et al. Decentralized motion control for cooperative manipulation with a team of networked mobile manipulators. In: **2016 IEEE Int. Conf. on Robotics and Automation**. 2016. p. 441–446.
- PUIG, D.; GARCIA, M.; WU, L. A new global optimization strategy for coordinated multi-robot exploration: Development and comparative evaluation. **Robotics and Autonomous Systems**, v. 59, n. 9, p. 635 – 653, 2011. ISSN 0921-8890.
- PURNAMADAJA, A. H.; RUSSELL, R. A. Pheromone communication in a robot swarm: necrophoric bee behaviour and its replication. **Robotica**, Cambridge University Press, v. 23, n. 6, p. 731–742, 2005.
- QUIGLEY, M. et al. Ros: an open-source robot operating system. In: **ICRA Workshop on Open Source Software**. 2009.
- RANGO, F. D. et al. Swarm robotics in wireless distributed protocol design for coordinating robots involved in cooperative tasks. **Soft Computing**, v. 22, n. 13, p. 4251–4266, 2018. ISSN 1433-7479.
- REINA, A. et al. Ark: Augmented reality for kilobots. **IEEE Robotics and Automation Letters**, v. 2, n. 3, p. 1755–1761, 2017. ISSN 2377-3766.
- REINA, A. et al. Augmented reality for robots: Virtual sensing technology applied to a swarm of e-pucks. In: **2015 NASA/ESA Conference on Adaptive Hardware and Systems (AHS)**. 2015. p. 1–6.
- ROBOTICS, C. **V-REP - virtual robot experimentation platform**. 2017. [Online; acesso em 15 de julho de 2018]. Disponível em: <<http://www.coppeliarobotics.com/>>.
- RUBENSTEIN, M.; CORNEJO, A.; NAGPAL, R. Programmable self-assembly in a thousand-robot swarm. **Science**, American Association for the Advancement of Science, v. 345, n. 6198, p. 795–799, 2014. ISSN 0036-8075.
- RUIZ, D.; BACCA, B.; CAICEDO, E. Control strategy based on swarms algorithms to cooperative payload transport using a non-holonomic mobile robots group. **IEEE Latin America Transactions**, v. 14, n. 2, p. 445–456, 2016.
- SABATTINI, L.; SECCHI, C.; FANTUZZI, C. Multi-robot systems implementing complex behaviors under time-varying topologies. **European Journal of Control**, v. 38, p. 73 – 87, 2017. ISSN 0947-3580.
- SAGITOV, A. et al. Effects of rotation and systematic occlusion on fiducial marker recognition. **MATEC Web of Conferences**, v. 113, p. 02006, 01 2017.

SANTOS, H. B. et al. Control of mobile robots using actionlib. In: KOUBAA, A. (Ed.). **Robot Operating System (ROS): The Complete Reference (Volume 2)**. Cham: Springer International Publishing, 2017. p. 161–189. ISBN 978-3-319-54927-9.

SASTRY, K.; GOLDBERG, D. E.; KENDALL, G. Genetic algorithms. In: BURKE, E. K.; KENDALL, G. (Ed.). **Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques**. Boston, MA: Springer US, 2014. p. 93–117. ISBN 978-1-4614-6940-7.

SCHLOTTFELDT, B. et al. Anytime planning for decentralized multirobot active information gathering. **IEEE Robotics and Automation Letters**, v. 3, n. 2, p. 1025–1032, April 2018. ISSN 2377-3766.

SIEGWART, R.; NOURBAKSHI, I. R.; SCARAMUZZA, D. **Introduction to Autonomous Mobile Robots**. 2nd. ed. USA: The MIT Press, 2011.

SPHERO. **Sphero robot**. 2018. [Online; acesso em 15 de maio, 2018]. Disponível em: <<https://www.sphero.com/sphero>>.

SPURNY, V. et al. Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles. **Journal of Field Robotics**, v. 36, n. 1, p. 125–148.

SUNDAR, K.; RATHINAM, S. Algorithms for heterogeneous, multiple depot, multiple unmanned vehicle path planning problems. **Journal of Intelligent & Robotic Systems**, Springer Netherlands, p. 1–14, 2016.

TANG, B.; ZHU, Z.; LUO, J. Hybridizing particle swarm optimization and differential evolution for the mobile robot global path planning. **International Journal of Advanced Robotic Systems**, v. 13, n. 3, p. 86, 2016.

WANG, L.-X. **A Course in Fuzzy Systems and Control**. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997. ISBN 0-13-540882-2.

WANG, Q. et al. Distributed multi-robot formation control in switching networks. **Neurocomputing**, v. 270, p. 4 – 10, 2017. ISSN 0925-2312.

WIKI, R. **ar_track_alvar**. 2018. [Online; acesso em 15 de julho de 2018]. Disponível em: <http://wiki.ros.org/ar_track_alvar/>.

WIKI, R. **costmap_2d**. 2018. [Online; acesso em 26 de julho de 2018]. Disponível em: <http://wiki.ros.org/costmap_2d>.

WIKI, R. **RViz**. 2018. [Online; acesso em 15 de julho de 2018]. Disponível em: <<http://wiki.ros.org/rviz>>.

YAN, Z.; JOUANDEAU, N.; CHERIF, A. A. A survey and analysis of multi-robot coordination. **International Journal of Advanced Robotic Systems**, v. 10, n. 12, p. 399, 2013.

YANGUAS-ROJAS, D.; MOJICA-NAVA, E. Exploration with heterogeneous robots networks for search and rescue. **IFAC-PapersOnLine**, v. 50, n. 1, p. 7935 – 7940, 2017. ISSN 2405-8963. 20th IFAC World Congress.

YU, J. et al. Neural network-based region reaching formation control for multi-robot systems in obstacle environment. **Neurocomputing**, v. 333, p. 11 – 21, 2019. ISSN 0925-2312.

YUAN, S. et al. A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms. **European Journal of Operational Research**, v. 228, n. 1, p. 72–82, 2013.

ZARZHITSKY, D.; SPEARS, D. F.; SPEARS, W. M. Distributed robotics approach to chemical plume tracing. In: **2005 IEEE/RSJ International Conference on Intelligent Robots and Systems**. 2005. p. 4034–4039. ISSN 2153-0858.

ZHOU, Y. et al. Collision and deadlock avoidance in multirobot systems: A distributed approach. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, v. 47, n. 7, p. 1712–1726, 2017. ISSN 2168-2216.

ZHOU, Y. et al. A distributed approach to robust control of multi-robot systems. **Automatica**, v. 98, p. 1–13, 2018.