CENTRO UNIVERSITÁRIO FEI

RODRIGO DA ROSA

POSICIONAMENTO DE ROBÔ EM SITUAÇÃO DE EMERGÊNCIA QUE ENVOLVE MULTIDÃO EM AMBIENTE FECHADO

São Bernardo do Campo

2019

RODRIGO DA ROSA

POSICIONAMENTO DE ROBÔ EM SITUAÇÃO DE EMERGÊNCIA QUE ENVOLVE MULTIDÃO EM AMBIENTE FECHADO

Tese apresentada ao Centro Universitário FEI, como parte dos requisitos necessários para obtenção do título de Doutor em Engenharia Elétrica. Orientado pelo Prof. Dr. Flavio Tonidandel.

São Bernardo do Campo

2019

ROSA, RODRIGO DA.
POSICIONAMENTO DE ROBÔ EM SITUAÇÃO DE EMERGÊNCIA QUE ENVOLVE MULTIDÃO EM AMBIENTE FECHADO / RODRIGO DA ROSA. SÃO BERNARDO DO CAMPO, 2019.
217 p. : il.

Tese - Centro Universitário FEI.
Orientador: Prof. Dr. FLAVIO TONIDANDEL.

1. ROBÔ AUTÔNOMO TERRESTRE. 2. SITUAÇÕES DE EMERGÊNCIA. 3. CLASSIFICAÇÃO CELULAR. 4. AMBIENTE FECHADO. 5. MULTIDÃO. I. TONIDANDEL, FLAVIO, orient. II. Título.

Elaborada pelo sistema de geração automática de ficha catalográfica da FEI com os dados fornecidos pelo(a) autor(a).

centro universitário FEI

APRESENTAÇÃO DE TESE
ATA DA BANCA EXAMINADORA

Programa de Pós-Graduação Stricto Sensu em Engenharia Elétrica

Doutorado

PGE-10

Aluno: Rodrigo da Rosa **Matrícula:** 515102-2

Título do Trabalho: Interação humano-robô em situações de emergência que envolvem multidão em ambientes fechados.

Área de Concentração: Inteligência Artificial Aplicada à Automação e Robótica

Orientador: Prof. Dr. Flavio Tonidandel

Data da realização da defesa: 25/06/2019

ORIGINAL ASSINADA

Avaliação da Banca Examinadora

A banca examinadora solicitou alteração do título da Tese para: "Posicionamento de robô em situação de emergência que envolve multidão em ambiente fechado"._____

São Bernardo do Campo, / / .

MEMBROS DA BANCA EXAMINADORA	
Prof. Dr. Flavio Tonidandel	Ass.: _____
Prof. Dr. Plinio Thomaz Aquino Junior	Ass.: _____
Prof. Dr. Reinaldo Augusto da Costa Bianchi	Ass.: _____
Prof. Dr. Fernando Santos Osório	Ass.: _____
Prof. Dr. Antonio Henrique Pinto Selvatici	Ass.: _____

A Banca Examinadora acima-assinada atribuiu ao aluno o seguinte:

APROVADO ☒ REPROVADO ☐

VERSÃO FINAL DA TESE

ENDOSSO DO ORIENTADOR APÓS A INCLUSÃO DAS RECOMENDAÇÕES DA BANCA EXAMINADORA

_____

Aprovação do Coordenador do Programa de Pós-graduação

_____

Prof. Dr. Carlos Eduardo Thomaz

À Daiane, Mariana e Gustavo.

AGRADECIMENTOS

Ao longo destes anos de pesquisa, Deus me ajudou a superar grandes desafios para concluir este trabalho. Além disso, colocou pessoas especiais em meu caminho, que direta ou indiretamente, contribuíram para esta realização pessoal. Agradeço ao Senhor por tudo o que fez e faz por mim.

Agradeço à minha família (Daiane, Mariana e Gustavo) pela compreensão e apoio que sempre demonstraram. Minha esposa e filhos, vocês são minhas riquezas.

Ao prof. Dr. Flavio Tonidandel, meu orientador, pela atenção e paciência dedicadas durante todo o desenvolvimento deste trabalho. Muito obrigado, professor.

Agradeço ao prof. Dr. Plinio Thomaz Aquino Junior pelo auxílio em momentos importantes. Agradeço também aos demais professores com quem tive contato.

Ao colega Douglas De Rizzo Meneghetti.

Ao Centro Universitário FEI pela oportunidade de realizar minha pesquisa com o benefício da bolsa de estudos por mérito acadêmico.

“Nada temos que recear quanto ao futuro, a menos que esqueçamos a maneira em que o Senhor nos tem guiado, e os ensinos que nos ministrou no passado.”

Ellen Gould White

RESUMO

As necessidades e desejos do dia-a-dia fazem com que as pessoas se aglomerem em determinados ambientes para algum propósito específico. Em muitos casos, a aglomeração se dá em áreas externas como ruas, avenidas e parques e, em outros, em áreas internas como *shoppings centers*, supermercados, estações de trem ou de ônibus e restaurantes.

Os robôs móveis autônomos terrestres estão, cada vez mais, sendo inseridos nos ambientes internos para auxiliarem seres humanos em tarefas como limpeza, transporte de materiais hospitalares, serviços hoteleiros, coleta e entrega de pedidos de refeições em restaurantes, atendimento e guia de passageiros em aeroportos, entre outros.

Entretanto, muitas são as razões que podem fazer com que uma multidão necessite evacuar o ambiente fechado. Pode ser motivada por um alarme, pela ocorrência de um incêndio ou pela informação de risco de detonação de uma bomba, por exemplo. Em situações como estas, o robô pode provocar acidentes ou ainda obstruir algum acesso, dependendo de suas ações de posicionamento antes ou durante a emergência.

Este trabalho contribui nos casos em que a multidão necessite evacuar um ambiente fechado com o menor risco possível de ter o robô atrapalhando este objetivo. A teoria proposta prevê que o robô decompõe o mapa do ambiente em células de tamanho e formato fixos e classifique cada uma delas com valores que determinam, a partir de dados de um simulador de multidão, a intensidade de passagem de pessoas (agentes), auxiliando-o a identificar regiões onde esta intensidade seja baixa, para estacionar.

Foram realizadas simulações, projetando ambientes fechados semelhantes à ambientes reais e construindo mapas capazes de orientar o robô a decidir a melhor região de segurança para estacionar. Os resultados demonstraram que a teoria de classificação celular, por meio de dados de um simulador de multidão, contribui para as ações do robô de identificação da célula objetivo, planejamento de trajetória e posicionamento em regiões do mapa onde a quantidade de passagem de pessoas é mínima.

Palavras-chave: Robô autônomo terrestre. Situações de emergência. Classificação celular. Ambiente fechado. Multidão.

ABSTRACT

The day-to-day needs and desires cause people to crowd in certain environments for some specific purpose. In many cases, the crowd occurs in external areas such as streets, avenues and parks, and in others, in internal areas such as shopping centers, supermarkets, train or buses stations and restaurants.

The ground mobile autonomous robots are increasingly being inserted into the indoor environments to assist human beings in tasks such as cleaning, transportation of hospital materials, hotel services, collection and delivery of meals requests in restaurants, attendance and passenger guides in airports , among others.

However, many are the reasons that can cause a crowd to need to evacuate an enclosed environment. It can be caused by an alarm, by the occurrence of a fire or by the risk of a bomb detonation, for example. In situations like these, the robot can cause accidents or even obstruct some access, depending on its actions of positioning before or during the emergency.

This work contributes in cases where the crowd needs to evacuate a closed environment with the lowest possible risk of having the robot disrupt this goal. The proposed theory predicts that the robot decomposes the map of the environment into cells of fixed size and format and classify each one of them with values that determine, from data of a crowd simulator, the estimated amount of people (agents) in each cell, helping it to identify regions where this number of people is low, to park.

Simulations were carried out, designing closed environments similar to real environments and constructing maps capable of guiding the robot to decide the best safety region to park. The results showed that the cell classification theory, through data from a crowd simulator, contributes to the actions of the robot of identification of the target cell, planning of trajectory and positioning in regions of the map where the flow of passage of people is minimal.

Keywords: Ground autonomous robot. Emergency situations. Cellular classification. Indoor environment. Crowd.

LISTA DE ILUSTRAÇÕES

Figura 1 – Robôs do tipo móveis com rodas. 36
Figura 2 – Robôs do tipo humanoides. 37
Figura 3 – Robôs do tipo com várias pernas. 38
Figura 4 – Aplicação dos robôs. 40
Figura 5 – Tarefas que precisam ser resolvidas por um robô para adquirir modelos precisos do ambiente. 43
Figura 6 – Exemplo de grade de ocupação para planejamento de trajetória. 46
Figura 7 – Valores do custo *f(n)* para as demais direções que o robô pode assumir. . . 47
Figura 8 – Trajetória do robô até a célula objetivo. 47
Figura 9 – Perfil de probabilidade de ocupação para um sensor ideal. 49
Figura 10 – Exemplo de autômato celular bidimensional. 52
Figura 11 – Exemplo de vizinhanças de autômato celular bidimensional. 53
Figura 12 – Condições de limite em Autômato Celular (AC)s finitos. 55
Figura 13 – Tela inicial do Oasys® MassMotion®. 59
Figura 14 – Característica de *Activities*. 60
Figura 15 – Característica da aba *Simulation & Analysis*. 60
Figura 16 – Etapas de construção dos mapas utilizados pelo robô. 75
Figura 17 – Exemplo de aplicação do método de decomposição exata e do grafo de conectividade. 78
Figura 18 – Exemplo de aplicação do método de decomposição aproximada. 79
Figura 19 – Exemplo de mapa com decomposição celular. 81
Figura 20 – Exemplo de mapa de ambiente com decomposição celular e classificação das células. 83
Figura 21 – Exemplo de valores de coordenadas de cinco agentes extraídos de um simulador de multidão convertidos em quantidades de acessos em cada célula. 88
Figura 22 – Mapa de Intensidade (MI)s com as classificações dos exemplos de níveis de emergência da tabela 2. 91
Figura 23 – Exemplo de MI gerado pelo robô, com o status de cada célula representado por cores. 93

Figura 24 – Exemplo de simulações, obtido do simulador de multidão, considerando alteração no mapa de intensidade com base na obstrução de um dos acessos do ambiente. 95
Figura 25 – Exemplos de sentidos de movimento da multidão e coordenadas da vizinhança do robô. 96
Figura 26 – Exemplo do modelo de vizinhança estendida de Moore, com nível 2. . . . 99
Figura 27 – Definição da célula objetivo. 100
Figura 28 – Apresentação da célula objetivo definida pelo robô, com a influência de regiões Região de Risco (RR)s próximas à célula de Região Livre (RL) mais próxima. 101
Figura 29 – Células a serem desqualificadas no sentido ED. 102
Figura 30 – Escolha da célula objetivo quando os custos são iguais para mais de uma célula livre. 103
Figura 31 – Desqualificação de células que não podem ser a célula objetivo. 104
Figura 32 – Exemplo de simulação, de mapa de intensidade e de dados obtidos por meio do simulador. 108
Figura 33 – Robôs cujas características são consideradas para os resultados. 111
Figura 34 – Exemplo de mapa de um centro de exposições. 112
Figura 35 – Células objetivo em cada posição do robô no centro de exposições. 114
Figura 36 – Situação onde duas células podem ser definidas como a célula objetivo. . . 115
Figura 37 – Medidas entre centros de duas células vizinhas. 117
Figura 38 – Esquema de trajetória criada pelo algoritmo A* para a situação em que o robô está na posição A, conforme figura 35a. 117
Figura 39 – Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo, no centro de exposições. 118
Figura 40 – Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo. . 118
Figura 41 – Trajetória seguida pelo robô nos outros casos do centro de exposições. . . . 119
Figura 42 – Exemplo de mapa de um salão de festas. 121
Figura 43 – MI ou Mapa de Obstrução (MO) do salão de festas com sobreposição da representação do mapa. 122
Figura 44 – Células objetivo para as posições A e B do robô, no salão de festas. 123
Figura 45 – Células objetivo para as posições C e D do robô, no salão de festas. 124
Figura 46 – Trajetória seguida pelo robô utilizando a técnica A*, no salão de festas. . . 125

Figura 47 – Exemplo de mapa de uma Universidade. 128
Figura 48 – Células objetivo em cada posição do robô na Universidade. 129
Figura 49 – Trajetória seguida pelo robô utilizando a técnica A*, na Universidade. . . . 130
Figura 50 – Exemplo de mapa de uma biblioteca. 133
Figura 51 – MI da biblioteca com sobreposição da representação do mapa. 134
Figura 52 – Células objetivo em cada posição do robô na biblioteca. 135
Figura 53 – Trajetória seguida pelo robô utilizando a técnica A*, na biblioteca. 137
Figura 54 – Exemplo de mapa de um apartamento. 140
Figura 55 – MI do apartamento com sobreposição da representação do mapa do apartamento. 141
Figura 56 – Células objetivo em cada posição do robô no apartamento. 142
Figura 57 – Trajetória seguida pelo robô utilizando a técnica A*, no apartamento. . . . 143
Figura 58 – Representação de célula parcialmente ocupada por um obstáculo (parede). 145
Figura 59 – Decomposição em células de tamanhos variados. 146
Figura 60 – Exemplo de mapa de um shopping center. 148
Figura 61 – MI do shopping center com sobreposição da representação do mapa. 149
Figura 62 – Células objetivo em cada posição do robô no shopping center. 150
Figura 63 – Trajetória seguida pelo robô utilizando a técnica A*, no shopping center. . 152
Figura 64 – Exemplo de mapa de um supermercado. 154
Figura 65 – MI do supermercado com sobreposição da representação do mapa. 155
Figura 66 – Células objetivo em cada posição do robô no supermercado. 156
Figura 67 – Trajetória seguida pelo robô utilizando a técnica A*, no supermercado. . . 158
Figura 68 – Exemplos de mapas de um posto de saúde. 161
Figura 69 – MI do posto de saúde com sobreposição da representação do mapa. 162
Figura 70 – Células objetivo em cada posição do robô no posto de saúde. 163
Figura 71 – Trajetória seguida pelo robô utilizando a técnica A*, no posto de saúde. . . 165
Figura 72 – Exemplo de mapa do piso térreo de um hotel. 168
Figura 73 – MI do hotel com sobreposição da representação do mapa. 169
Figura 74 – Células objetivo em cada posição do robô no hotel. 170
Figura 75 – Trajetória seguida pelo robô utilizando a técnica A*, no hotel. 171
Figura 76 – Caso de exceção once um acesso está obstruído. 176
Figura 77 – Sensores sonar em um Pioneer® 3DX. 205
Figura 78 – Sensores a laser. 206

Figura 79 – Sensores de infravermelho. 207
Figura 80 – Estrutura de sensores Bumpers. 208
Figura 81 – Sensor de visão Sick®. 209
Figura 82 – Microsoft® Kinect®. 210
Figura 83 – Exemplo de deformação de luz estruturada infravermelha padrão em objetos de uma cena. 211
Figura 84 – Sensores de temperatura. 212
Figura 85 – Exemplos de aplicações de atuadores. 217

LISTA DE TABELAS

Tabela 1 – Possíveis estados e respectivas cores. 83
Tabela 2 – Exemplo de níveis de emergência com base na classificação das áreas para geração do MI. 90
Tabela 3 – Valores de referência para classificação das células do MI. 92
Tabela 4 – Exemplos de ambientes fechados e seus respectivos mapas. 107
Tabela 5 – Velocidade de cada robô. 111
Tabela 6 – Tipos de sensores e exemplos. 204
Tabela 7 – Tipos de atuadores, funções e exemplos. 216

LISTA DE ALGORITMOS

Algoritmo 1 – Algoritmo que calcula o número de ocorrências em cada coordenada (x,y) do mapa. 89
Algoritmo 2 – Algoritmo que normaliza os valores das células, conforme padrão do Mapa do Ambiente (MA). 93
Algoritmo 3 – Algoritmo que desqualifica algumas células para serem objetivo. 104

LISTA DE ABREVIATURAS

AC	Autômato Celular
IEEE	Institute of Electrical and Electronics Engineers
IHM	Interface Homem-Máquina
IR	Infrared
LIDAR	Light Detection and Ranging
MA	Mapa do Ambiente
MCL	Monte Carlo Localization
MI	Mapa de Intensidade
MN	Mapa de Navegação
MO	Mapa de Obstrução
QF	Quantidade Fornecida
QH	Quantidade por Hipótese
QL	Quantidade Lida
RA	Região de Atenção
RL	Região Livre
RR	Região de Risco
SFM	Social Force Model
SLAM	Simultaneous Localization and Mapping

SUMÁRIO

1 INTRODUÇÃO 29
1.1 OBJETIVO 30
1.2 JUSTIFICATIVA 31
1.3 METODOLOGIA 31
1.4 ESTRUTURA DO TRABALHO 32
2 CONCEITOS FUNDAMENTAIS 33
2.1 ROBÓTICA 33
2.1.1 Tipos de Robôs 34
2.1.1.1 Estrutura geométrica e locomoção 35
2.1.1.2 Robôs Inteligentes 38
2.1.2 Aplicações dos robôs 39
2.2 MAPEAMENTO E LOCALIZAÇÃO SIMULTÂNEOS - SLAM 42
2.3 PLANEJAMENTO DE TRAJETÓRIA - A* 45
2.4 GRADES DE OCUPAÇÃO 48
2.5 SIMULAÇÃO DE MULTIDÃO - MODELOS 50
2.5.1 Modelo de Simulação Macroscópica 50
2.5.2 Modelo de Simulação Microscópica 50
2.6 AUTÔMATOS CELULARES 51
2.7 FERRAMENTAS DE SIMULAÇÃO 55
2.7.1 Ferramentas de simulação de robôs sem multidão 56
2.7.2 Ferramentas de simulação de multidão sem robôs 57
2.7.2.1 Oasys® MassMotion® 58
2.8 CONCLUSÃO DO CAPÍTULO 60
3 REVISÃO DA LITERATURA 63
3.1 ROBÔS MÓVEIS AUTÔNOMOS 63
3.1.1 Robôs móveis autônomos na multidão em ambientes fechados 65
3.1.2 Robôs móveis autônomos em situações de emergência 67
3.1.3 Robôs móveis autônomos em ambientes fechados - uso de autômatos celulares 68
3.2 SIMULAÇÃO DE MULTIDÃO E AUTÔMATOS CELULARES 71
3.3 CONCLUSÃO DO CAPÍTULO 72

4 CLASSIFICAÇÃO CELULAR 75
4.1 DECOMPOSIÇÃO DO MAPA DO AMBIENTE EM CÉLULAS 77
4.1.1 Método da decomposição exata 77
4.1.2 Método da decomposição aproximada 78
4.2 MAPA DO AMBIENTE - MA 80
4.3 MAPA DE INTENSIDADE - MI 84
4.3.1 Parâmetros da simulação 86
4.3.1.1 Quantidade de agentes 86
4.3.1.2 Partida da multidão 86
4.3.1.3 Destino da multidão 87
4.3.2 Classificação celular no MI 87
4.4 MAPA DE OBSTRUÇÃO - MO 94
4.5 MAPA DE NAVEGAÇÃO - MN 97
4.5.1 Definição da célula objetivo 98
4.5.1.1 Célula objetivo onde não há impedimento de acesso 100
4.5.1.2 Célula objetivo onde há impedimento de acesso ou há sentido único de evacuação 101
5 RESULTADOS E DISCUSSÕES 107
5.1 CENTRO DE EXPOSIÇÕES 112
5.1.1 Definição da célula objetivo no centro de exposições 113
5.1.2 Definição da trajetória do robô no centro de exposições 116
5.1.3 Simulação com os robôs no centro de exposições 120
5.2 SALÃO DE FESTAS 121
5.2.1 Definição da célula objetivo no salão de festas 123
5.2.2 Definição da trajetória do robô no salão de festas 124
5.2.3 Simulação com os robôs no salão de festas 126
5.3 UNIVERSIDADE 127
5.3.1 Definição da célula objetivo na Universidade 128
5.3.2 Definição da trajetória do robô na Universidade 129
5.3.3 Simulação com os robôs na Universidade 131
5.4 BIBLIOTECA 133
5.4.1 Definição da célula objetivo na biblioteca 135
5.4.2 Definição da trajetória do robô na biblioteca 136

5.4.3 Simulação com os robôs na biblioteca 138
5.5 APARTAMENTO 139
5.5.1 Definição da célula objetivo no apartamento 141
5.5.2 Definição da trajetória do robô no apartamento 143
5.5.3 Particularidade do ambiente apartamento 144
5.5.4 Simulação com os robôs no apartamento 146
5.6 CENTRO COMERCIAL - SHOPPING CENTER 148
5.6.1 Definição da célula objetivo no shopping center 150
5.6.2 Definição da trajetória do robô no shopping center 151
5.6.3 Simulação com os robôs no shopping center 152
5.7 SUPERMERCADO 154
5.7.1 Definição da célula objetivo no supermercado 156
5.7.2 Definição da trajetória do robô no supermercado 157
5.7.3 Simulação com os robôs no supermercado 159
5.8 POSTO DE SAÚDE 161
5.8.1 Definição da célula objetivo no posto de saúde 162
5.8.2 Definição da trajetória do robô no posto de saúde 164
5.8.3 Simulação com os robôs no posto de saúde 166
5.9 HOTEL 167
5.9.1 Definição da célula objetivo no hotel 169
5.9.2 Definição da trajetória do robô no hotel 171
5.9.3 Simulação com os robôs no hotel 172
5.10 ANÁLISE GERAL DOS RESULTADOS 174
5.10.1 Casos de exceção 175
6 CONCLUSÕES E TRABALHOS FUTUROS 177
REFERÊNCIAS 180
APÊNDICE A – SENSORES 203
APÊNDICE B – ATUADORES 215

1 INTRODUÇÃO

Ambientes fechados são espaços físicos limitados, podendo ser ambientes de trabalho (empresas, indústrias), de estudo (escolas, colégios, faculdades, universidades), de comércio (supermercados, livrarias), de lazer ou diversão (cinemas, teatros, casas de espetáculos, shoppings), entre outros. As características atuais das sociedades desenvolvidas e em desenvolvimento fazem com que um elevado número de pessoas passe a maior parte do seu dia em locais como estes (QUADROS, 2008; PITARMA; MARQUES; FERREIRA, 2016; LADOU; HARRISON, 2016). A tendência global para a industrialização e urbanização leva as pessoas a cada vez mais viverem e trabalharem em ambientes fechados. Elas nascem em hospitais, são criados em casas ou apartamentos e trabalham em edifícios de escritórios ou fábricas (KELLEY; GILBERT, 2013).

Em ambientes fechados sempre há riscos de situações de emergência que causam nas pessoas uma busca imediata por evacuar o cenário em que se encontram como, por exemplo, incêndio, desabamento de estruturas, ameaça de bombas, entre outros. Nestes casos, acidentes poderão acontecer sem que haja tempo suficiente para evitar danos físicos e até mesmo psicológicos.

A segurança em ambientes assim é um requisito fundamental, uma vez que a ocorrência de acidentes pode trazer consequências para vidas humanas. Desta forma, é de grande valia a existência de soluções computacionais que auxiliem forças operacionais na atuação em uma emergência em ambientes fechados (NUNES et al., 2016).

A robótica é uma das áreas das ciências exatas e tecnológicas que mais se desenvolveu nos últimos anos. A utilização de robôs para a realização de tarefas com o intuito de auxiliar e/ou substituir os humanos em atividades diárias, vem se tornando cada vez mais frequente e, com isso, novas aplicações em diversas áreas vêm surgindo a cada dia (SOUTO; MAIA; SOUZA, 2014).

A robótica pode ser aplicada em muitos campos, como indústria automotiva, engenharia, projetos militares, laboratórios médicos, medicina, energia nuclear, agricultura e serviços ao cliente (GURAVAIAH; DANIEL, 2015).

Em muitos casos (CHATTERJEE, 2016; JIANG, C. et al., 2016; VEMULA; MUELLING; OH, 2017), os robôs são inseridos em locais, abertos ou fechados, onde há grande concentração de pessoas (multidão) para realização de tarefas específicas.

Os robôs móveis autônomos terrestres já fazem parte do cotidiano das pessoas em muitos ambientes fechados, podendo interagir com elas, auxiliando-as de diversas maneiras. Entre elas, há robôs que ajudam equipes de segurança a realizarem o patrulhamento das áreas do ambiente ou ainda podem trabalhar com sistemas de gerenciamento de armazém, empilhadeiras autônomas e transportadores, para organizar e transportar estoques (RBR, 2019a). Há também os robôs utilizados em empresas hoteleiras para fornecerem serviços de quarto a pedido de seus hóspedes. O robô navega pelo hotel entregando os objetos solicitados pelos clientes, evitando obstáculos e pessoas (CHUNG; CAKMAK, 2018). Outros robôs realizam serviços básicos em tribunais de justiça, sendo capaz de memorizar e explicar leis e regulamentos, oferecendo conveniência aos advogados e cidadãos (RBR, 2019b).

Existem estudos sendo realizados para inserir carrinhos de compras (robôs) em supermercados para auxiliar os consumidores, transportando seus produtos durante o período de compras, além de calcular o caminho mais curto com base nos itens da lista que o consumidor deseja comprar (ALVES; LINHARES; SOUZA, 2018).

Em situações de emergência, que motivam as pessoas a evacuarem o ambiente fechado, há sempre o risco de acidentes, principalmente quando a multidão é tomada pelo pânico. Há estudos que propõem robôs inseridos na multidão e que a auxilia em situações como estas. São capazes de conduzir grupos de seres humanos para saídas menos congestionadas (BOUKAS et al., 2015; TANG et al., 2016; SAKOUR; HU, 2017), reduzindo assim o efeito afunilamento nos acessos.

Entretanto, outro aspecto pouco aplicado e de grande relevância para situações de emergência é o posicionamento do robô, considerando que a velocidade com que a multidão se movimenta para evacuar o ambiente é, em tese, maior do que a dele. Partindo deste pressuposto, um robô que não esteja programado para decidir sobre o lugar mais adequado para ficar estacionado durante a saída das pessoas, pode ser um fator extra de causa de acidentes, contrariando assim, os motivos pelos quais os robôs têm sido inseridos em tais ambientes, ou seja, para realizar tarefas de auxílio aos seres humanos.

1.1 OBJETIVO

O objetivo desta tese é propor uma teoria de classificação celular por meio de simulador de multidão, em ambientes fechados, para definição de célula objetivo, visando o posicionamento adequado do robô autônomo em situações de emergência.

1.2 JUSTIFICATIVA

Uma vez que os robôs sejam capazes de planejar trajetórias e, a partir disso, movimentar-se em um ambiente fechado com multidão, um grande desafio se refere ao seu posicionamento em situações em que emergências surgem, situações estas que levam as pessoas a evacuarem o local, muitas vezes tomadas pelo pânico.

As pesquisas referentes a trabalhos que propõem o posicionamento do robô autônomo em situações de emergência, de modo que ele não atrapalhe a evacuação da multidão, revelaram que são poucos os estudos realizados para esta finalidade.

O desenvolvimento deste trabalho é justificado pela necessidade do robô decidir, de maneira autônoma, em qual área do ambiente fechado irá estacionar no momento em que houver a ocorrência de uma situação de emergência, onde as pessoas necessitarão evadir o espaço. Tal comportamento se dá a partir de mapas de intensidade do ambiente, gerados por meio de dados coletados de um simulador de multidão, usados para classificar células quando ocorre a decomposição celular deste ambiente fechado. As técnicas para o mapeamento do ambiente, decomposição celular, simulação de multidão, geração dos mapas, definição da célula objetivo e planejamento de trajetória são executadas pelo próprio robô.

1.3 METODOLOGIA

A primeira etapa do desenvolvimento desta tese foi identificar o problema a ser estudado e, a partir disso, elencar um conjunto de assuntos associados às possíveis contribuições para ele.

A fundamentação teórica foi realizada identificando diversos trabalhos desenvolvidos sobre assuntos importantes para dar sustentação à teoria proposta neste projeto. Estes assuntos estão relacionados aos robôs móveis autônomos terrestres na multidão em ambientes fechados, em situações de emergência, usando autômatos celulares, além da abordagem sobre simulação de multidão e autômatos celulares.

Para os testes, foram criados nove ambientes no simulador de multidão MassMotion® e para cada um deles, configurados os parâmetros para a simulação, definidas as quantidades de agentes e executadas as simulações. Em seguida, foram coletados os dados de passagem dos agentes em cada uma das posições de coordenadas x e y do ambiente. Estes dados foram

utilizados no MATLAB® para geração dos mapas de intensidade e de obstrução, necessários para a identificação das regiões de baixa, média e alta passagem de agentes pelo ambiente.

Para finalizar, foram realizadas as discussões sobre os resultados obtidos de modo a avaliar a eficiência da teoria proposta.

1.4 ESTRUTURA DO TRABALHO

Esta tese é composta por 6 capítulos, descritos a seguir.

O capítulo 1 apresenta a introdução deste trabalho, visando mostrar ao leitor o problema que esta tese se propõe a contribuir.

O capítulo 2 apresenta conceitos fundamentais sobre assuntos como robótica, técnicas de mapeamento e localização simultâneos, planejamento de trajetória, grades de ocupação, modelos de simulação de multidão, autômatos celulares e ferramentas de simulação.

O capítulo 3 apresenta a revisão da literatura que contribui para dar sustentação a teoria desenvolvida neste trabalho.

O capítulo 4 apresenta a teoria de classificação celular desenvolvida para contribuir com a solução do problema introduzido nesta tese.

O capítulo 5 apresenta os testes realizados, bem como os resultados e discussões sobre cada um deles.

O capítulo 6 apresenta as conclusões e proposta de trabalhos futuros, com base nas pesquisas e nos testes deste trabalho.

2 CONCEITOS FUNDAMENTAIS

Este capítulo tem como objetivo apresentar alguns temas fundamentais para dar o embasamento geral do trabalho proposto. Sobre o tema *robótica*, são apresentados os mais variados tipos de robôs, bem como suas aplicações no cotidiano das pessoas e companhias. Sobre o tema *SLAM (Mapeamento e Localização Simultâneos)* é feita uma breve descrição desta técnica que é tão importante para a navegação do robô em ambientes fechados, criando mapas que determinam a localização de obstáculos, além de identificar sua própria localização no contexto do ambiente. Em seguida, o tema *grades de ocupação* conceitua este assunto que é fundamental para a decomposição do ambiente fechado em células, necessário para compor a base deste trabalho. O tema *simulação de multidão* visa mostrar a importância da simulação de evacuação de ambientes fechados e apresenta alguns modelos de simulação que podem ser utilizados para este fim. Entre os modelos de simulação, a técnica de *Autômatos Celulares* é apresentada com maiores detalhes. Por fim, o tema *ferramentas de simulação* mostra algumas ferramentas de simulação, dividindo-as em ferramentas de simulação de robôs sem multidão e de multidão sem robôs, descrevendo com maiores detalhes a ferramenta de simulação escolhida para este trabalho.

2.1 ROBÓTICA

Robótica é o ramo da tecnologia que lida com o design, construção, operação e aplicação de robôs, bem como sistemas de computador para seu controle, feedback sensorial e processamento de informações. Estas tecnologias lidam com máquinas automatizadas que podem ocupar o lugar de seres humanos em ambientes perigosos ou processos de fabricação, podendo ainda se assemelharem a humanos na aparência, comportamento e/ou cognição (GURAVAIAH; DANIEL, 2015).

A palavra robótica é derivada da palavra robô, que foi introduzida ao público pelo escritor checo Karel Čapek em sua peça teatral de ficção científica R.U.R. (Rossum's Universal Robots), apresentada em 1921 (GURAVAIAH; DANIEL, 2015). Robô é uma palavra que significa *trabalho forçado*, *serviço obrigatório*, *trabalho pesado* (ABDULKAREEM et al., 2016).

Isaac Yudovich Asimov (1920 - 1992), bacharel em química pela Columbia University, foi um escritor de obras de ficção científica, nascido em Petrovichi, Rússia, porém criado em

Nova Iorque a partir dos três anos de idade. Cunhou o termo robótica e, para ele, ferramentas criadas para a conveniência humana diária tendem a apresentar elementos de segurança como, por exemplo, o isolamento em fiações elétricas e a válvula em panelas de pressão. Para ele, os robôs certamente seriam construídos com a preocupação de prevenir ferimentos aos seres humanos (BIZONY, 2015).

As características fictícias de suas publicações tinham viés mais racional e humano e menos fantasioso. Uma de suas mais conhecidas obras é "I, Robot" (Eu, Robô), uma compilação de nove histórias publicadas separadamente entre 1940 e 1950 (BILANCIA, 2014). Este material foi base para a produção do filme de mesmo nome, lançado em 2004 pela 20th Century Fox® (BILANCIA, 2014).

As três leis de Asimov, de sua publicação "I, Robot", são (ASIMOV, 1956 apud BILANCIA, 2014):

a) Um robô não pode causar dano a um ser humano ou, por omissão, permitir que um ser humano venha a sofrer.
b) Um robô deve obedecer às ordens dadas a ele por seres humanos, exceto quando tais ordens entrarem em conflito com a Primeira Lei.
c) Um robô deve proteger sua própria existência, desde que tal proteção não entre em conflito com a Primeira ou Segunda Lei.

O paradigma subjacente das leis de Asimov dizia respeito à autonomia do robô. Operando independentemente de qualquer intervenção humana, o robô tem a capacidade física e intelectual de tomar suas próprias decisões morais com base em seu conhecimento e racionalidade internamente mantidos (MCBRIDE; HOFFMAN, 2016).

As próximas seções descreverão brevemente os variados tipos de robôs existentes, dando destaque aos robôs móveis autônomos terrestres, uma vez que estes fazem parte do contexto de simulação deste trabalho.

2.1.1 Tipos de Robôs

Os robôs são projetados para auxiliar os seres humanos em diversas áreas e de diversas maneiras, conforme as necessidades vão surgindo. Existem variados tipos de robôs e nesta seção serão apresentados alguns deles.

2.1.1.1 Estrutura geométrica e locomoção

Em termos de estrutura geométrica e locomoção, os robôs podem ser classificados como (ANGELO, 2007; MCKERROW, 1999 apud TZAFESTAS, 2015):

a) Manipuladores robóticos fixos: consistem de uma sequência de elos que podem ser rígidos ou flexíveis, interconectados por meio de articulações chamadas de juntas. Geralmente são compostos por um braço que assegura mobilidade, um pulso e um efetuador, ou garra, ao qual pode ser acoplada uma determinada ferramenta para realizar a tarefa desejada pelo robô (FERNANDES JUNIOR, 2014 apud NUNES, 2016). A importância dos manipuladores robóticos nas indústrias é principalmente, em função das tarefas repetitivas que eles podem executar, nas montadoras de veículos e de placas eletrônicas, carga e embaladoras de produtos em geral (NUNES, 2016).

b) Robôs voadores ou veículos aéreos não tripulados (UAV, em inglês): robô voador é um termo amplo que pode ser aplicado a plataformas autônomas que viajam pelo ar usando uma variedade de modos de locomoção e abrangem várias ordens de magnitude em tamanho e peso (KLAPTOCZ, 2012). Esses sistemas são amplamente utilizados para aplicações militares, operações de busca e salvamento, pulverização agrícola, filmagem de eventos esportivos ou filmes de quase qualquer ângulo e equipamentos de transporte ou controle. Veículos não tripulados são importantes quando se trata de realizar uma tarefa desejada em um ambiente perigoso e ou inacessível (MAITY et al., 2013).

c) Robôs submarinos: são robôs extensamente aplicados na exploração submarina e na pesquisa ambiental marinha (CHOTIKARN et al., 2010). Para Goulart (2007 apud GOMES, 2011), robôs submarinos são utilizados em estudos científicos sobre habitats oceânicos, localização e resgaste de destroços de navios ou de aviões, sondagens do solo marinho, inspeção e reparo de cabos de telecomunicações subaquáticos, operações de perfuração de petróleo e gás, recuperação de material bélico, destruição de minas anti-navio e prestação de socorro submarino.

d) Robôs terrestres - móveis com rodas: a figura 1 apresenta alguns robôs que se enquadram nesta categoria. Estes robôs movem-se usando dois tipos de rodas: rodas convencionais e rodas especiais (TZAFESTAS, 2013 apud 2015).

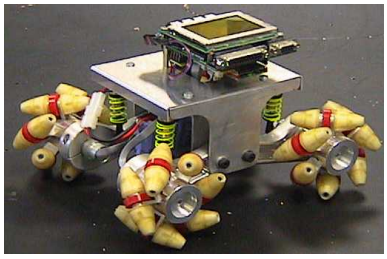
Figura 1 – Robôs do tipo móveis com rodas.

(a) Pioneer 3DX

(b) Pioneer 3AT

(c) triciclo

(d) omnidirecional



(e) Tibi e Dabo

Fontes: (a) (CYBERBOTICS, 2017), (b) (CYBERBOTICS, 2017), (c) (NI, 2017), (d) (ROBOTICS, 2017), (e) (INDEPENDENT, 2017)

As rodas convencionais podem ser: rodas fixas motorizadas, rodas de rodízio e volantes motorizados. As rodas fixas são conduzidas pelos motores montados em posições fixas do veículo. Rodas de rodízios não são alimentadas e podem girar livremente em torno de um eixo perpendicular ao seu eixo de rotação. Os volantes motorizados têm um motor de acionamento para a sua rotação e podem ser orientados em torno de um eixo perpendicular ao seu eixo de rotação. As rodas especiais envolvem rodas universais, rodas mecanum, e rodas esféricas (TZAFESTAS, 2015). O robô Tibi é um modelo humanoide com rodas (CORTÉS, 2014).

e) Robôs terrestres - com locomoção por um par de pernas, por exemplo, bípedes (humanoides): são robôs que imitam o andar humano. Comparados com outros robôs, eles têm maior flexibilidade e maior adaptabilidade aos ambientes (JIANG, Y. et al., 2013). Os robôs bípedes humanoides se assemelham aos seres humanos na aparência e também em seus movimentos (LUO et al., 2014). Eles têm muitas vantagens, como a capacidade de mudar de direção facilmente em áreas estreitas ou mover-se estavelmente em terrenos difíceis. Em comparação com os robôs que possuem quatro pernas ou mecanismo de rodas, o robô bípede é um melhor sistema móvel para robôs humanoides, bem como para outros robôs

(DOAN; HAYASHI; YAMAKITA, 2015). Alguns exemplos de robôs humanoides são apresentados na figura 2.

Figura 2 – Robôs do tipo humanoides.

(a) Lynxmotion Hytec Robonova®

(b) Honda Asimo®

(c) Kondo KHR-1HV®

(d) WowWee® Robosapien X®

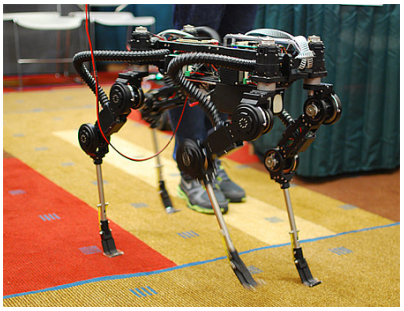
Fontes: (a) (ROBOSAVVY, 2017), (b) (SCIENSE, 2017), (c) (ROBOTICSTODAY, 2017), (d) (WOWWEE, 2017).

f) Robôs com várias pernas: nesta categoria estão os robôs que possuem mais do que duas pernas e a figura 3 apresenta alguns exemplos. Robôs quadrúpedes têm maior possibilidade de escolha de colocação dos pés para manter o equilíbrio estático. Quando comparado aos robôs com seis e oito pernas os quadrúpedes desfrutam de uma estrutura mais simples, porém o desempenho é afetado diretamente quando há uma falha elétrica ou mecânica de uma de suas pernas durante o período de locomoção (OLIVEIRA, 2016). Os robôs hexápodes possuem um par de pernas a mais do que os robôs quadrúpedes. O horizonte de trabalho dos hexápodes é ampliado, pois tanto robôs bípedes quanto quadrúpedes são fortemente afetados quando há uma avaria em uma de suas pernas (OLIVEIRA, 2016). Assim, devido

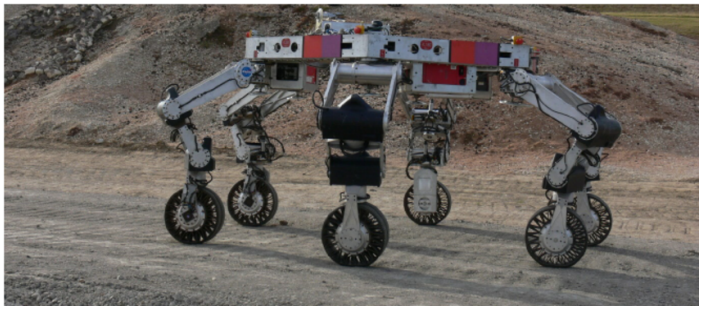
à sua redundância, hexápodes e octópodes podem, após detectar tanto o mau funcionamento quanto a perda total da perna, reconfigurar seu padrão de locomoção e, em certos casos, manter o robô estável, com um decréscimo em sua velocidade de movimentação.

Figura 3 – Robôs do tipo com várias pernas.

(a) Quadrúpede SQ1, da South Korean company SimLab



(b) ATHLETE - NASA



(c) Hexapod Kondo® KMR-M6

(d) Octópode Robugtix® T8X

Fontes: (a) (SPECTRUM, 2017), (b) (NASA, 2017), (c) (GEEKY-GADGETS, 2017; ROBUGTIX, 2017).

A proposta deste trabalho poderá ser aplicada em robôs do tipo móvel terrestre com rodas, humanoides e até mesmo com várias pernas, estando eles em ambiente fechado contendo multidão.

As seções seguintes apresentam conceitos relevantes sobre robôs inteligentes e também sobre as variadas possibilidades de aplicações de robôs.

2.1.1.2 Robôs Inteligentes

De acordo com Tzafestas (2015), os robôs inteligentes, isto é, os robôs que incorporam inteligência artificial e controle semiautônomo ou autônomo inteligente, constituem a classe de

robôs que são capazes de tomar decisão e usam informações multissensoriais sobre seu estado interno e ambiente de espaço de trabalho para gerar e executar planos para a realização de tarefas complexas. Eles também podem monitorar a execução de seus planos, aprender com a experiência passada, melhorar seu comportamento e comunicar-se com um operador humano em linguagem natural ou quase natural. A característica-chave de um robô inteligente é que ele pode executar um repertório de tarefas diferentes em condições que podem não ser conhecidas a priori.

Os principais componentes de qualquer robótica inteligente são: efetuadores (braços, mãos, rodas e pernas com seus atuadores, motores e mecanismos de propulsão), sensores (acústico, alcance, velocidade, força ou toque, visão, entre outros), computadores (controladores locais, supervisores, coordenadores, controladores de execução) e equipamentos auxiliares (ferramentas de ponta, paletes, dispositivos elétricos, plataformas, entre outros) (TZAFESTAS, 1991; JACAK, 2002 apud TZAFESTAS, 2015).

Tzafestas (2015) afirmam que os robôs inteligentes executam autonomamente quatro tarefas:

a) Desvio de obstáculos;
b) Reconhecimento de metas;
c) Planejamento de trajetória/movimento;
d) Localização e mapeamento (com o auxílio de marcos).

Estas tarefas são fundamentais para este trabalho, pois o robô, navegando pelo ambiente, precisará desviar de pessoas e objetos para evitar acidentes, reconhecer as metas para se posicionar adequadamente em caso de emergência e planejar a trajetória para chegar à meta. Para que tudo isso seja possível, necessitará realizar o mapeamento do ambiente e saber sua posição no ambiente a cada instante.

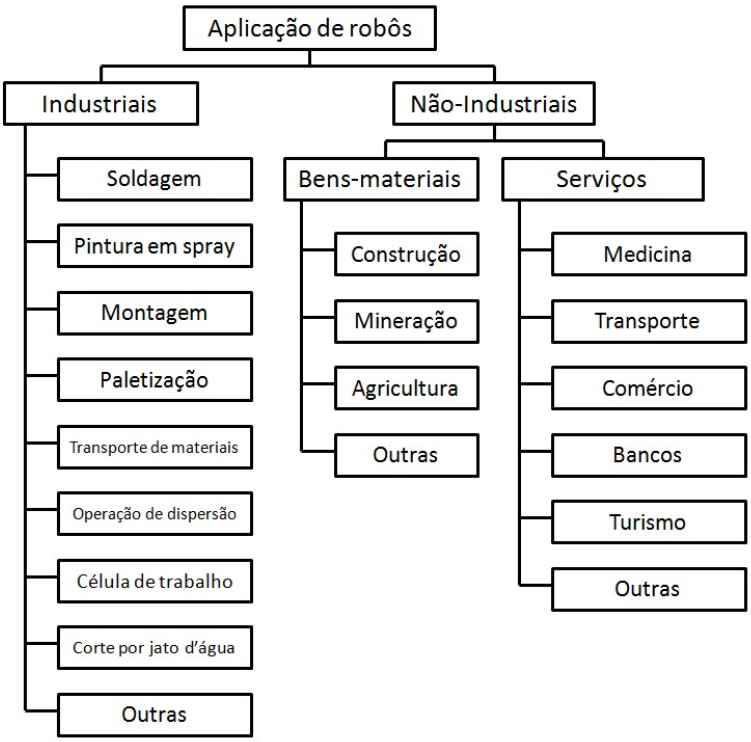
Segundo Dang et al. (2017), os robôs inteligentes estão gradualmente coexistindo com humanos em áreas mais amplas, como robôs de serviço, robôs sociais e robôs de saúde.

2.1.2 Aplicações dos robôs

Os robôs são atualmente utilizados em diversos setores e para diversas tarefas. Em muitos casos os robôs têm realizado atividades que, em outras épocas, eram desempenhadas somente por seres humanos.

Para Dobra (2014), um dos arranjos mais simples de robôs ordenados por aplicação pode ser representado conforme a figura 4.

Figura 4 – Aplicação dos robôs.



Fonte: Autor (adaptada de Dobra (2014)).

Tzafestas (2015) amplia esta classificação, observando as tarefas que os robôs realizam ou podem realizar, conforme segue:

a) Robôs industriais: são também conhecidos como robôs de fábrica. Os robôs móveis são usados na fábrica para manuseio de materiais e transferência de produtos de um lugar para outro para inspeção, controle de qualidade, entre outros. As áreas dominantes da robótica na fábrica são (NOF, 1999 apud TZAFESTAS, 2015): 1) Carregamento e descarregamento de máquina; 2) Manuseio de materiais e fundição; 3) Soldagem e montagem; 4) Maquinação e inspeção; 5) Perfuração, forjamento, entre outros.

b) Robôs de suporte à área de medicina: robôs são utilizados no campo médico para transporte de material hospitalar, segurança e vigilância, inspeção no campo nuclear, sistemas de automação de farmácia, sistemas cirúrgicos integrados e entretenimento. Os sistemas robóticos operatórios representam uma classe particular

de telerrobôs que permitem aos cirurgiões realizarem operações cirúrgicas muito mais precisas e com sucesso do que a cirurgia clássica (SPEICH; ROSEN, 2004 apud TZAFESTAS, 2015).

c) Robôs domésticos: são robôs móveis projetados para realizarem tarefas domésticas como limpeza de pisos e azulejos, limpeza de piscinas e preparo de café. Robôs capazes de ajudar pessoas idosas e pessoas com necessidades especiais também podem ser incluídos nesta classe, embora possam também pertencer à categoria dos robôs de apoio.

d) Robôs de apoio: robótica de apoio, ou assistiva, inclui todos os sistemas robóticos que são desenvolvidos para tentar capacitar as pessoas com deficiências a alcançarem e manterem seu melhor nível físico e/ou funcional social, melhorando a sua qualidade de vida e produtividade de trabalho (KATEVAS, 2001; COOK; HUSSEY, 2002; TZAFESTAS, 1998 apud 2015).

e) Robôs de salvamento: são robôs que atuam no resgate e salvamento de pessoas em locais de desastres que são muito perigosos para a intervenção humana ou que não podem ser alcançados (riscos de temperaturas extremas, níveis radioativos, forças de vento forte, entre outros). Robôs modernos de resgate são flexíveis e duráveis. Muitos deles têm câmeras com rotação de 360 graus, que fornecem imagens de alta resolução e outros sensores que podem detectar a temperatura corporal e roupas coloridas.

f) Robôs espaciais: são robôs utilizados para explorarem regiões espaciais. São profundamente estudados em pesquisas da NASA e de outros lugares (Alemanha e Canadá, por exemplo).

g) Robôs militares: são robôs projetados para a guerra. Em geral, operam em ambientes geopoliticamente sensíveis. Os robôs autônomos de guerra incluem mísseis, veículos aéreos de combate não tripulados, veículos terrestres não-tripulados e veículos submarinos autônomos (WHITE, 2007; ZALOGA, 2008 apud TZAFESTAS, 2015).

h) Robôs de entretenimento: robôs socializados são robôs autônomos inteligentes equipados com sensores, capazes de aprender regras sociais e de desempenho. Eles são capazes de interagir com os seres humanos e possuem uma aparência humanoide, são flexibilidade e adaptáveis. Existem robôs sociais educacionais e de pesquisa em desenvolvimento em vários institutos de pesquisa e robôs comercial-

mente disponíveis que podem escrever, dançar, tocar instrumentos musicais, jogar futebol e interagir emocionalmente.

A teoria proposta neste trabalho poderá ser aplicada em robôs de suporte à área da medicina (quando estes são utilizados para transporte de materiais hospitalares, segurança e vigilância, entre outros), robôs domésticos, robôs de salvamento (quando estes possuírem característica autônoma), robôs de apoio e robôs de entretenimento. Estes robôs são, em geral, aqueles que prestam serviço aos seres humanos.

Os robôs necessitam de sensores e de atuadores para que haja interação com o meio no qual estão inseridos. Os detalhes sobre os tipos e funções dos sensores e atuadores são encontrados nos apêndices A e B.

2.2 MAPEAMENTO E LOCALIZAÇÃO SIMULTÂNEOS - SLAM

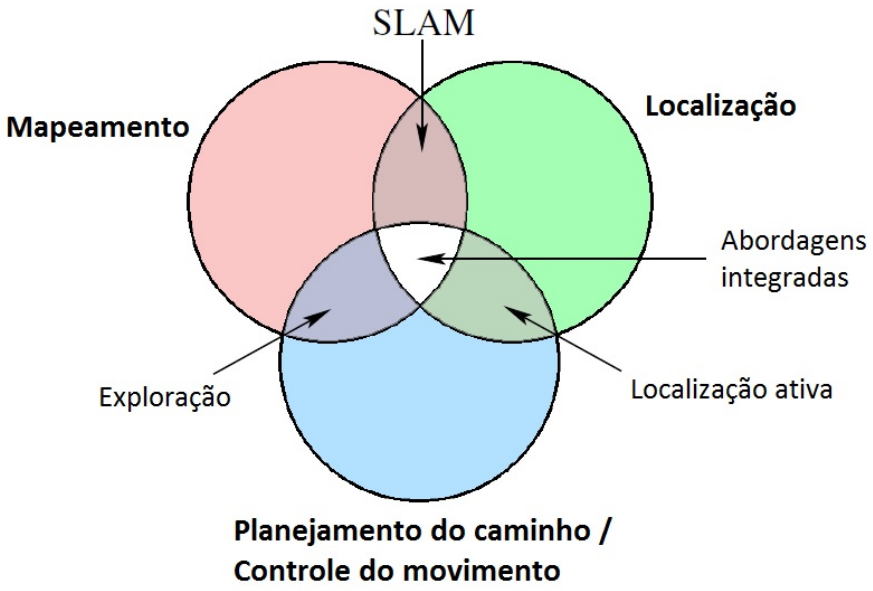
De acordo com Stachniss (2009), os mapas de aprendizagem com sistemas de um robô requerem a solução de três tarefas, que são o mapeamento, a localização e o planejamento do caminho.

Mapeamento é o problema de integrar as informações coletadas com os sensores do robô em uma determinada representação. Os aspectos centrais no mapeamento são a representação do ambiente e a interpretação dos dados do sensor. Localização é o problema de estimar o posicionamento do robô em relação a um mapa. Finalmente, o problema de planejamento de trajeto ou controle de movimento envolve a questão de como guiar eficientemente um veículo para um local desejado ou ao longo de uma trajetória. Essas três tarefas não podem ser resolvidas independentemente uma da outra.

Os robôs precisam também reconhecer os limites de um ambiente para que possam tomar decisões de locomoção em áreas distintas. Alguns fatores podem dificultar ainda mais este processo, pois o mapeamento de um local depende muitas vezes de situações de alteração de iluminação, bloqueio ou objetos em movimento, por exemplo (RAMOS et al., 2012). A aparência visual de locais varia no tempo por causa de mudanças de iluminação (dia e noite, luz artificial ou natural) e por causa das atividades humanas (móveis movimentados, objetos que estão sendo retirados de gavetas) e, assim, as soluções devem ser adaptáveis (PRONOBIS et al.,

2010). A figura 5 apresenta as tarefas de mapeamento, localização e planejamento de trajetória, bem como os problemas associados nas áreas sobrepostas.

Figura 5 – Tarefas que precisam ser resolvidas por um robô para adquirir modelos precisos do ambiente.



Fonte: Autor (adaptada de Makarenko et al. (2002 apud STACHNISS, 2009)).

Robôs móveis devem ser projetados para terem a capacidade de se localizar no espaço e também planejar novos trajetos para sua movimentação, considerando sempre os custos de sua interação com os seres humanos e objetos como, por exemplo, a colisão com estes elementos do ambiente de interação (SHIOMI et al., 2014), ou ainda um posicionamento permanente ou momentâneo em qualquer lugar que venha a atrapalhar o campo visual de alguém ou impedir sua passagem. Tudo deve ser criado, percebido e mapeado em tempo-real (CHENGA et al., 2015).

O termo Simultaneous Localization and Mapping (SLAM) se originou em 1986 na Conferência de Robótica e Automação do Institute of Electrical and Electronics Engineers (IEEE), em São Francisco, com a discussão sobre métodos de localização e mapeamento pelos pesquisadores Hugh Durrant-Whyte, Peter Cheeseman, Jim Crowley, Raja Chatila, Oliver Faugeras e Randal Smith (BAILEY, 2002 apud BIGHETI, 2011). Nesse período, diversos métodos foram propostos para diferentes tipos de ambientes, tendo como resultado métodos robustos para ambientes que são estáticos, estruturados e de tamanho limitado (BIGHETI, 2011).

SLAM pode ser definida como uma técnica onde um robô percorre o espaço de um ambiente e é capaz de detectar pessoas e objetos estáticos ou dinâmicos através de seus sensores, criando um mapa do cenário e gerando a possibilidade de localizar-se no contexto do ambiente (BOAS, 2011).

Chong et al. (2015) entendem que SLAM refere-se a um problema no qual um robô deve construir um mapa que representa seu ambiente espacial, ao mesmo tempo em que mantém dados que permitem a ele localizar-se neste ambiente. O mapa construído é posteriormente utilizado pelo robô para navegação.

O mapeamento de ambientes é responsável por construir um modelo (mapa) do ambiente e armazenar no robô, de modo que o mesmo possa reconhecer espaços vazios e espaços com obstáculos. A Localização determina a configuração do robô, que consiste na posição e na orientação, considerando o modelo gerado na fase do Mapeamento de Ambientes (SOUTO; MAIA; SOUZA, 2014).

O mapeamento é realizado por meio de sensores como câmera, sonar, scanner laser e até mesmo sensores visuais (VALIENTE et al., 2015). Sistemas baseados em laser são a alternativa mais comum de utilização em problemas que envolvem SLAM (CHONG et al., 2015).

Os mapas gerados fornecem informações sobre o meio ambiente e estas informações são usadas pelo robô para saber sua localização (VÁSQUEZ, 2016).

Um exemplo de algoritmo para a localização de um robô móvel, é chamado Monte Carlo Localization (MCL). O MCL é uma versão da família de abordagens probabilísticas, que aplica métodos baseados em amostragem para aproximar as distribuições de probabilidade. O número de amostras é adaptado on-line, invocando grandes conjuntos de amostras apenas quando necessário. Este algoritmo é de fácil implementação em relação a outros existentes (FOX et al., 1999).

Os robôs autônomos móveis terrestres realizam SLAM para que consigam se localizar no ambiente fechado a partir dos mapas que são gerados durante sua navegação pelo espaço. O conhecimento de sua localização é importante para que ele evite colidir com obstáculos e transitar por áreas de risco, onde o movimento de pessoas é mais intenso em uma situação de emergência em que a multidão deverá procurar acessos para evacuar o local. O conhecimento de sua localização também é fundamental para identificar a célula objetivo, que deverá ser utilizada para seu posicionamento no momento da emergência. Deste modo, SLAM é essencial para a aplicação da teoria deste trabalho, porém cenários reais.

2.3 PLANEJAMENTO DE TRAJETÓRIA - A*

Em robótica, o assunto *planejamento do caminho* se preocupa com o problema de como mover um robô de um ponto para outro. A navegação do robô em um ambiente depende do conhecimento que ele tem deste local. Se o robô não tem qualquer descrição do ambiente, ele usa seus sensores para conhecê-lo. No entanto, se o robô tem um mapa preciso do cenário de navegação, ele pode usar muitas técnicas para movimentar-se de maneira ideal. Isso é chamado planejamento de caminho ou navegação global (DUCHON et al., 2014).

Segundo Russell e Norvig (2010), o problema de planejamento de trajetória é encontrar um caminho, de uma configuração para outra, no espaço. Uma abordagem para o planejamento de trajetórias decompõe o espaço livre em um número finito de regiões contíguas, chamadas células. A decomposição celular mais simples consiste em uma grade regularmente espaçada. Armazena-se, então, certos valores para cada célula da grade e considera-se um algoritmo que armazena, para cada célula da grade, o estado exato que foi atingido na primeira pesquisa. Ao propagar informações para células de grade próximas, usa-se esse estado contínuo como base e aplica-se o modelo de movimento contínuo do robô para saltar para as próximas células.

Entre diversas técnicas de planejamento de trajetória, encontra-se a A*. De acordo com Hart, Nilsson e Raphael (1968 apud RUSSELL; NORVIG, 2010), o mecanismo de busca A* (A-estrela) avalia nós, combinando g(n), o custo para alcançar cada nó, e h(n), o custo para ir do nó até a meta. Deste modo:

$$f(n) = g(n) + h(n) \tag{1}$$

onde *f(n)* é igual ao custo estimado da solução de custo mais baixo passando por *n*, uma vez que *g(n)* fornece o custo de caminho desde o nó inicial até o nó *n*, e que *h(n)* é o custo estimado do caminho de custo mais baixo (distância Euclidiana ou de Manhattan, por exemplo) de *n* até o objetivo.

A* será ótima se *h(n)* for uma heurística admissível, ou seja, *h(n)* nunca deve superestimar o custo para alcançar o objetivo. Um exemplo de heurística admissível é a distância em linha reta, usada para ir de um nó até outro, pois uma linha reta é o caminho mais curto entre dois pontos (HART; NILSSON; RAPHAEL, 1968 apud RUSSELL; NORVIG, 2010).

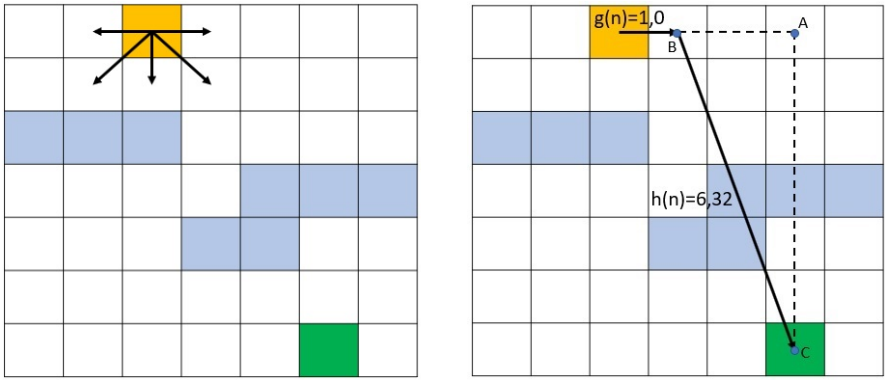
A heurística de distância Euclidiana foi utilizada na teoria deste trabalho, considerando que o robô pode se mover linearmente e também na diagonal. A figura 6 apresenta um exemplo

de grade de ocupação contendo a célula de posição inicial do robô (cor laranja), a célula objetivo (cor verde) e obstáculos (cor azul). O robô escolhe uma das células vizinhas para o primeiro movimento seguindo a técnica de planejamento de trajetórias, A*.

Figura 6 – Exemplo de grade de ocupação para planejamento de trajetória.

(a) Possibilidades de movimentos do robô a partir de sua posição inicial.

(b) Cálculo de *f(n)* para a célula à direita da posição do robô.



Fonte: Autor.

Neste cenário, inicialmente o robô tem cinco opções de movimento, conforme apresentado na figura 6a. Ele irá escolher movimentar-se para a célula vizinha de menor função custo estimado (*f(n)*). A função *g(n)* possui valor 1,0 quando o sentido do movimento é linear e 1,41 quando o sentido é na diagonal até a célula vizinha (considerando que uma célula tem 1 x 1 metro). Na figura 6b, é apresentado o exemplo de cálculo de *f(n)* para a célula vizinha à direita da posição do robô.

A distância Euclidiana h(n) é obtida por meio da equação:

$$h(n) = \sqrt{(x_{fi} - x_{ini}) + (y_{fi} - y_{ini})} \tag{2}$$

onde:

a) x_{ini} é a coordenada x da célula de centro representado pela letra A na figura 6b.

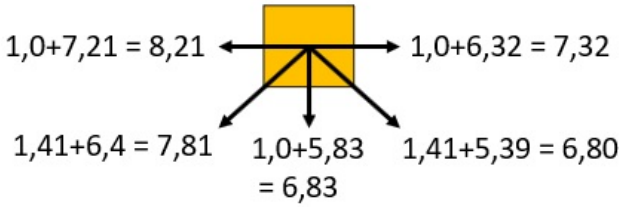
b) x_{fi} é a coordenada x da célula de centro representado pela letra B na figura 6b.

c) y_{ini} é a coordenada y da célula de centro representado pela letra A na figura 6b.

d) y_{fi} é a coordenada y da célula de centro representado pela letra C na figura 6b.

Sendo assim, o custo estimado (*f(n)*) para a célula à direita da célula do robô é: *f(n)* = *g(n)* + *h(n)* = 1,0 + 6,32 = 7,32. Outros custos, referentes às demais direções que o robô pode assumir, estão apresentados na figura 7.

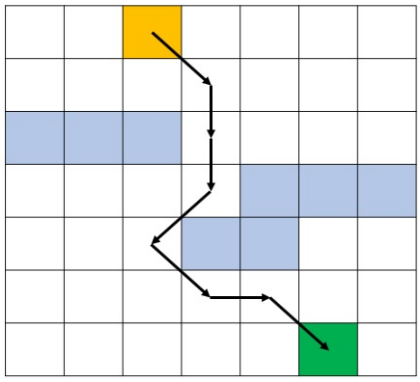
Figura 7 – Valores do custo *f(n)* para as demais direções que o robô pode assumir.



Fonte: Autor.

É possível observar que o menor custo é 6,80 (diagonal). Esta será a direção da célula vizinha escolhida pelo robô para a realização do seu primeiro movimento. Deste modo, a trajetória até a célula objetivo definida pelo robô seguirá conforme ilustrado na figura 8.

Figura 8 – Trajetória do robô até a célula objetivo.



Fonte: Autor.

A compreensão deste tema é importante para este trabalho, pois no capítulo de resultados e discussões (capítulo 5) serão apresentados resultados com base nesta teoria.

2.4 GRADES DE OCUPAÇÃO

Grades de ocupação, introduzidas por Moravec e Elfes Moravec e Elfes (1985 apud MEYERDELIUS; BEINHOFER; BURGARD, 2012), são baseadas em um mosaico regular do ambiente em um número de células retangulares. Armazenam em cada célula desta grade a probabilidade da área correspondente no ambiente estar ocupada por um obstáculo (MEYERDELIUS; BEINHOFER; BURGARD, 2012).

De acordo com Elfes (1989), a grade de ocupação é um campo aleatório multidimensional (tipicamente 2D ou 3D) de espaço nas células, onde cada célula armazena uma estimativa probabilística de seu estado.

A ideia básica da grade de ocupação é representar um mapa multidimensional do ambiente como células uniformemente quadradas ou cúbicas, que possuem variáveis aleatórias indicando a presença de um obstáculo nessa célula. A probabilidade da célula estar ocupada pode ser calculada usando a teoria bayesiana ou a teoria dos conjuntos fuzzy (RIBO; PINZ, 2001 apud ABU-MAHFOUZ; HANCKE, 2018), com base em informações fornecidas por várias fontes (ABU-MAHFOUZ; HANCKE, 2018).

Um mapa de grade de ocupação representa o ambiente por células discretas formando uma grade. Cada célula representa uma área quadrada do ambiente e armazena um valor que indica o estado de ocupação dessa área. Isso geralmente é feito rotulando as células com valores *desconhecidos*, *livres* ou *ocupados* ou com um valor que represente a probabilidade de a célula ser ocupada ou não (ÁLVARES et al., 2004 apud MOJTAHEDZADEH, 2011).

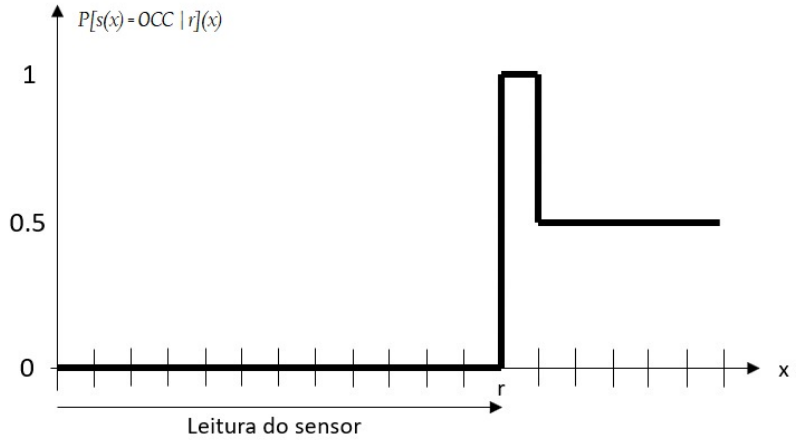
A grade de ocupação é um processo de rede, definida sobre uma rede espacial discreta. As variáveis de estado *s(C)* associadas a uma célula *C* da grade de ocupação são definidas como uma variável aleatória discreta com dois possíveis estados, ocupadas e vazias, classificadas como OCC e EMP, respectivamente (ELFES, 1989). Assim:

$$P[s(C) = OCC] + P[s(C) = EMP] = 1 \tag{3}$$

Para construir um mapa derivado do sensor do robô, as estimativas do estado da célula são obtidas pela interpretação das leituras de intervalo de entrada usando modelos de sensores probabilísticos. Uma medição de faixa de sensor *r* é interpretada usando um modelo de sensor estocástico (estado indeterminado).

Para ilustrar a abordagem, Elfes (1989) apresenta um exemplo como da figura 9 para mostrar o perfil de ocupação para o caso de um sensor de faixa ideal unidimensional. Dado um intervalo de leitura *r*, a célula correspondente tem probabilidade de ocupação 1, enquanto que as anteriores são vazias e têm probabilidade de ocupação 0 e as células posteriores não foram observadas e são desconhecidas, sendo a probabilidade de ocupação definida como 0,5.

Figura 9 – Perfil de probabilidade de ocupação para um sensor ideal.



Fonte: Autor (adaptada de Elfes (1989)).

Mapas de ocupação podem ser gerados por sensores como, por exemplo, sonar (BENICASA, 2012), scanner a laser (SAHDEV; CHEN; TSOTSOS, 2018), de visão monocular (SANTANA, 2011), entre outros.

Sensores como laser podem retornar dados de alcance mais precisos e, consequentemente, permitir a construção de grades mais fiéis ao ambiente (PEDROSA; MEDEIROS; ALSINA, 2019).

Trabalhos como o de Santana, Aires e Medeiros (2016) fazem uso de câmeras para a construção de grades de ocupação, pois de acordo com os autores, são facilmente conectadas ao hardware do robô, fornecem rica informação sobre o ambiente, consomem pouca energia e são relativamente baratas quando comparadas a outros sensores utilizados em robôs.

2.5 SIMULAÇÃO DE MULTIDÃO - MODELOS

De acordo com Sakour e Hu (2017), a simulação de multidão geralmente se refere à representação dos movimentos e comportamentos de um grupo de pessoas e isto é um dos principais desafios no domínio de modelagem e simulação. O uso de simulação por computador para estudar o comportamento de multidões visa economizar tempo e recursos, já que não é necessário um grande número de participantes para realizar experimentos reais. Além disso, o uso de simulação contribui para a coleta de dados de um experimento de maneira mais rápida.

Os autores ainda afirmam que os vários modelos de simulação utilizados para simular uma evacuação podem ser divididos em Modelo Macroscópico e Modelo Microscópico.

2.5.1 Modelo de Simulação Macroscópica

Este tipo de modelo de simulação tem como foco as estruturas físicas do ambiente e seu desenvolvimento, considerando o grupo de pessoas como uma unidade móvel. As características individuais para os pedestres e o tempo de processamento é relativamente pequeno (SAKOUR; HU, 2017). São exemplos de Modelos de Simulação Macroscópica as técnicas de Dinâmica de Fluidos, de Escolha de Rota e de Regressão, descritos com maiores detalhes no trabalho de Sakour e Hu (2017).

2.5.2 Modelo de Simulação Microscópica

Este tipo de modelo tem como foco as características individuais, considerando diferentes fatores que podem afetar os indivíduos da multidão em movimento. É um método mais realista e preciso, porém pode levar a algoritmos complexos por envolver inúmeros objetos e suas interações (SAKOUR; HU, 2017). São exemplos de Modelos de Simulação Microscópica as técnicas de Enfileiramento, de AC, de Força Social (SFM), Baseado em Agentes, descritos com maiores detalhes no trabalho de Sakour e Hu (2017).

Neste trabalho, optou-se pelo Modelo de Simulação Microscópica pelo fato de que as características individuais dos agentes devem ser levadas em conta, uma vez que os dados de posicionamento ao longo do tempo, de cada agente, serão fundamentais para a construção do modelo proposto. A técnica escolhida foi a de AC, pois suas características, no que diz respeito a formato, estado das células, vizinhança e regras de transição, contribuem com a proposta de

teoria desenvolvida. Além disso, a teoria dos ACs é utilizada para gerenciar a dinâmica da multidão, por meio de simulação, em ambientes fechados e, assim, a simulação da dinâmica do comportamento de robôs, inseridos em ambientes fechados, pode apresentar resultados significativos para a evacuação segura da multidão em situações de emergência. A seguir será descrita esta técnica.

2.6 AUTÔMATOS CELULARES

Um autômato celular (AC) é um sistema discreto dinâmico que evolui sob regência de uma regra de transição de estados (WUENSCHE; LESSER, 1993 apud LOZANO, 2017).

De acordo com Muller (2008), ACs são sistemas dinâmicos constituídos por células (sítios), distribuídos em uma rede discreta, onde cada célula desta rede possui um determinado valor, denominado de *estado* da célula.

Os ACs também podem ser definidos como modelos matemáticos compostos por um conjunto de células dispostas no espaço, onde o estado de uma dada célula em um dado tempo depende de regras locais e de sua relação com sua vizinhança (WOLFRAM, 1983 apud RENKAVIESKI et al., 2018). Eles podem ocorrer nas formas unidimensional, formando uma linha, e tridimensional, formando um volume sólido de células (RENKAVIESKI et al., 2018).

A teoria dos ACs começou a ser desenvolvida por John Von Neumann, no final dos anos 1940. Ele idealizava uma teoria contendo aspectos matemáticos e lógicos de modo a contribuir de maneira essencial para a compreensão dos sistemas naturais, chamados de autômatos naturais, bem como a compreensão de computadores analógicos e digitais, estes chamados de autômatos artificiais (NEUMANN, 1966).

Von Neumann contou com a parceria de Stanislaw Ulam, que sugeriu usar um conceito baseado em evoluções de construções gráficas geradas por regras simples chamadas de espaços celulares (cellular spaces), para formular um modelo para uma máquina que fosse plenamente capaz de se autorreproduzir (ULAM, 1962 apud MULLER, 2008). O modelo foi distribuído em um universo discreto, ou seja, um quadrado bidimensional, composto por células, cada parte evoluindo independentemente seguindo regras determinadas (KRAMER, 2014).

Na década de 1970, John Conway propôs um autômato com características binária (dois estados), bidimensional e com regras simples chamado (game of life), um AC que representa

sistemas complexos da evolução da vida por meio de regras matemáticas (BERLEKAMP; CONWAY; GUY, 1982; MULLER, 2008; KRAMER, 2014).

Na década seguinte, Wolfram realizou as primeiras publicações de trabalhos onde foram investigados ACs mais básicos, envolvendo uma rede unidimensional com grau de liberdade binário (WOLFRAM, 1983; 1984; 1986 apud MULLER, 2008).

Como mencionado anteriormente, os ACs representam modelos matemáticos simples, caracterizados por células idênticas contendo variados estados e distribuídas em uma rede n-dimensional. As células interagem no tempo com sua vizinhança, por meio de regras de transição (KRAMER, 2014).

Um AC é composto basicamente por 4 elementos, podendo ser expresso da seguinte maneira:

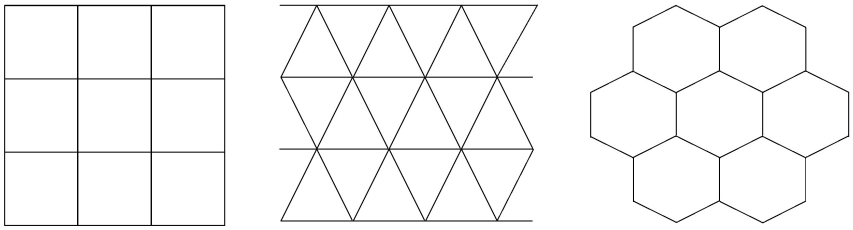
$$AC = (F,E,V,ft)$$

onde:

a) F (Formato): define uma forma geométrica para as células de uma matriz bidimensional. Os formatos mais comuns são os triangulares, quadrados e hexagonais, porém todas as células devem ter o mesmo formato (LEITE; LINS; CERQUEIRA, 2007 apud MELOTTI, 2009). O formato das células deve levar em consideração as características dos objetos que poderão ocupá-las em algum momento. A figura 10, adaptada de Zubeldia (2014), apresenta exemplos dos formatos mais comuns de ACs;

Figura 10 – Exemplo de autômato celular bidimensional.

(a) AC quadrado. (b) AC triangular. (c) AC hexagonal.

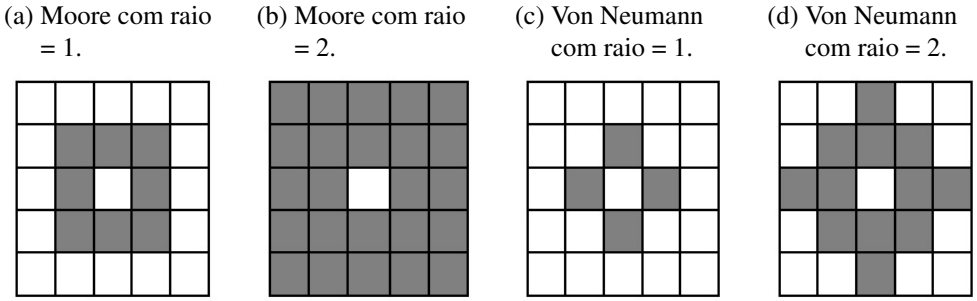


Fonte: Autor (adaptada de Zubeldia (2014)).

b) E (Conjunto finito de estados): define quais são os possíveis estados que uma célula pode assumir no arranjo, em determinado instante. O conjunto de estados pode ser um conjunto de cores (azul, vermelho, branco e preto), uma propriedade (ligado ou desligado), ou ainda, valores numéricos (WEIMAR, 1997 apud GONÇALVES; CENTERO; CANDEIAS, 2011);

c) V (Vizinhança de determinada célula): a vizinhança consiste de uma faixa de vizinhos mais próximos da célula que será atualizada, e dela mesmo. A função das células vizinhas é definir o estado das células no próximo instante (MELOTTI, 2009). Essa vizinhança corresponde a uma determinada seleção de células próximas (podendo eventualmente incluir a própria célula) (WOLFRAM, 2002; RUCKER, 2005 apud SILVA et al., 2008). Para malhas quadradas, que são as mais comuns, utilizam-se basicamente dois tipos de vizinhança, as de Moore e as de Von Neumann (ZUBELDIA, 2014). A figura 11 ilustra estas vizinhanças;

Figura 11 – Exemplo de vizinhanças de autômato celular bidimensional.



Fonte: (ZUBELDIA, 2014).

A vizinhança de Moore considera a influência de todas as células ao redor do ponto de interesse, enquanto que a vizinhança de Von Neumann considera apenas as que estão diretamente ligadas aos lados, em cima e abaixo do ponto. A própria célula (centro) pode formar parte ou não da vizinhança (ZUBELDIA, 2014). A vizinhança de Moore com um raio igual a 1 usa as oito células vizinhas, incluindo as diagonais, enquanto a extensão da vizinhança de Moore com um raio maior que um se estende ao longo da distância além das oito células vizinhas da célula central (GORSEVSKI et al., 2012).

d) (*ft*) (Função de transição, que representa as regras de transição do arranjo): Os estados das células são alterados conforme um conjunto de regras de transição, que

depende da vizinhança (às vezes da própria célula também), ou seja, das células em torno da célula que será atualizada (MELOTTI, 2009). As conexões entre as células, caso existam, servem para que, na simulação, a multidão possa deslocar-se entre elas. Isto é feito através de células com a função específica de transportar os pedestres de uma célula para outra: células de transição (CARNEIRO et al., 2013). O estado futuro de uma célula no tempo (t+1) em relação ao tempo de início (t) é impulsionado pela regra de transição como uma função de (a) estado atual da célula de teste, (b) estados atuais de suas células de vizinhança e (c) regras de transição que conduzem o processo ao longo do tempo (GORSEVSKI et al., 2012).

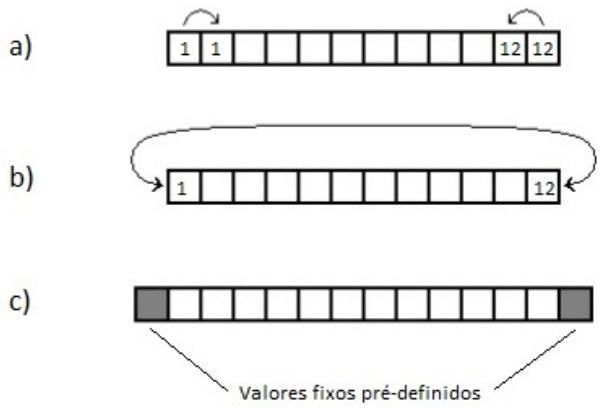
Um problema existente na modelagem da vizinhança se dá no momento em que o robô ocupa uma célula em que há limitações laterais (bordas), podendo estas se configurarem como paredes, por exemplo. Neste caso, um tratamento diferenciado deve ser dado ao modelo, já que o número de vizinhos se torna reduzido devido ao fato de que o robô não terá mais a opção de deslocamento em todos os sentidos da malha, ou seja, sua vizinhança é incompleta.

Para solucionar este ponto, há basicamente três condições de limite das bordas que podem ser implementadas em um modelo, sendo estas (MELOTTI, 2009):

a) Limite reflexivo: é a condição de contorno obtida refletindo-se o autômato em cada borda, simulando uma vizinhança completa.
b) Limite periódico: os limites periódicos são obtidos estendendo-se o autômato, isto é, a última célula terá como um de seus vizinhos a primeira célula e a primeira célula terá como um de seus vizinhos a última célula. Em outras palavras, as células de uma borda tornam-se vizinhas das células da borda oposta.
c) Limite fixo: este limite é obtido simplesmente fixando os estados das células na borda do autômato. Estes estados serão os mesmos em qualquer instante da simulação.

A figura 12 ilustra as condições de limite em ACs finitos. Percebe-se que, em a), os valores das células das extremidades são refletidos para o interior da malha, enquanto que em b), a célula vizinha à esquerda de 1, é a célula 12 e a célula vizinha à direita de 12 é a célula 1. Em c), os valores das extremidades podem conter quaisquer valores que não sofrerão atualização.

Figura 12 – Condições de limite em ACs finitos.



Fonte: Autor (adaptada de Pfeifer et al. (2001)).

Neste trabalho, foi utilizada a técnica de AC bidimensional quadrado, de tamanho fixo. O ambiente fechado simulado é representado como uma matriz, onde os possíveis estados de cada célula assumem valores numéricos que variam de 0 a 1. A vizinhança utilizada foi a de Moore, uma vez que o robô terá a opção de deslocar-se para os lados, para frente ou para traz e também para suas diagonais. A condição de limite escolhida é a de valores fixos, uma vez que os limites são as paredes do ambiente fechado e para estas existe uma classificação específica.

2.7 FERRAMENTAS DE SIMULAÇÃO

Os robôs estão cada vez mais inseridos em ambientes internos lotados, como museus, shoppings e centros de conferência (TSUI et al., 2011 apud AROOR; EPSTEIN; KORPAN, 2018). Antes dessa implantação, o controle de navegação do robô deve ser testado extensivamente, principalmente para evitar colisões. No entanto, projetar e executar testes reais em larga escala é um fator desafiador, pois em um shopping, por exemplo, as possibilidades de testes dependem do horário, dia da semana e eventos agendados irregularmente (AROOR; EPSTEIN; KORPAN, 2018).

Neste sentido, simuladores de multidão podem ser utilizados para realizar os testes que irão prever o comportamento robótico em ambientes fechados antes da implementação do sistema em um cenário real, minimizando assim os desafios no que diz respeito à larga escala de testes que seriam necessários.

Entretanto, há outros desafios a serem superados quanto às ferramentas de simulação, pois a maioria destas ferramentas disponíveis fornece multidões realistas sem robôs (por exemplo, PedSim (PEDSIM, 2018), OpenSteer (REYNOLDS, 1999 apud AROOR; EPSTEIN; KORPAN, 2018), Menge e Continuum (TREUILLE; COOPER; POPOVIĆ, 2006 apud AROOR; EPSTEIN; KORPAN, 2018)) ou robôs realistas sem multidões (por exemplo, Gazebo (KOENIG; HOWARD, 2004 apud AROOR; EPSTEIN; KORPAN, 2018) ou Stage (AROOR; EPSTEIN; KORPAN, 2018)).

Alguns exemplos podem ser observados nas referências a seguir.

2.7.1 Ferramentas de simulação de robôs sem multidão

Gazebo é um simulador de dinâmica 3D usado no desenvolvimento e teste de cenários e aplicações robóticas (KOENIG; HOWARD, 2004 apud OKOLI et al., 2019). Isso permite que pesquisadores e desenvolvedores testem um ou vários robôs rapidamente e avaliem o seu desempenho em condições mais realistas sem danificar o robô real (OKOLI et al., 2019). O simulador Gazebo é um software livre e de código aberto, muito popular na comunidade robótica, que possui funcionalidades como, por exemplo, simulação dinâmica, visualização de gráficos 3D, integração de sensores ao robô, entre outras (OKOLI et al., 2019). Todos os objetos simulados têm massa, velocidade, atrito e vários outros atributos que permitem que eles se comportem de maneira realista quando são empurrados, puxados, derrubados ou carregados (KOENIG; HOWARD, 2004 apud NYMAN; D'ORTO, 2018).

Player/Stage é um ambiente de simulação multi-robô (GERKEY; VAUGHAN; HOWARD, 2003 apud BERGBREITER; PISTER, 2003) projetado na USC Robotics Research Lab. O Player é um servidor de dispositivos que fornece uma interface poderosa e flexível para uma variedade de sensores e atuadores. Os programas de controle do robô podem ser escritos por linguagem de programação e podem ser executados em qualquer computador com conectividade de rede com o robô (USC, 2019). O Stage simula uma população de robôs móveis que se deslocam e detectam um ambiente bidimensional. Diversos modelos de sensores e atuadores estão disponíveis para as interfaces do Stage, portanto, as simulações podem ser facilmente transferidas para o hardware real depois de serem executadas no Stage (BERGBREITER; PISTER, 2003).

Além destes simuladores de robôs sem multidão, há outros exemplos como, o V-Rep (COPPELIAROBOTICS, 2019), Webots (CYBERBOTICS, 2019), ARGoS (ARGOS-SIM, 2019), entre outros.

2.7.2 Ferramentas de simulação de multidão sem robôs

O simulador PedSim é uma biblioteca microscópica de simulação de multidões para pedestres adequado para uso em simulações de multidões (evacuação em ambientes fechados ou simulações externas em grande escala, por exemplo), onde se está interessado em resultados como densidade de pedestres ou tempo de evacuação. A qualidade da trajetória do agente individual é alta o suficiente para criar animações massivas de multidões de pedestres. Ele é baseado na linguagem de programação C++, sem pacotes adicionais, podendo ser executado virtualmente em todos os sistemas operacionais. Os pedestres são visíveis na interface do usuário em tempo real. Usando o arquivo ou a saída baseada em rede, é possível o processamento em lote e em tempo real. A saída do PedSim é normalmente alimentada em um mecanismo de renderização, onde humanos de aparência realista são criados e caminham baseados nas trajetórias geradas por este simulador (PEDSIM, 2002).

O simulador OpenSteer é uma biblioteca de componentes de código aberto usado para a construção de comportamentos de trajetórias para personagens autônomos em jogos e outros tipos de simulações multiagentes, que podem representar personagens (humanos, animais, criaturas alienígenas), veículos (carros, aviões, naves espaciais), entre outros. O OpenSteer é suportado por variados sistemas operacionais, como os da distribuição Linux, Windows e MacOS-X. Ele é destinado ao uso por programadores e não se conecta a ferramentas de produção interativas, nem mesmo suporta a autoria de comportamento por não-programadores (REYNOLDS, 1999).

O simulador Menge é uma estrutura modular de código aberto extensível e multiplataforma para simular o movimento de agentes (pessoas) em uma multidão. O simulador e os comportamentos de agente de alto nível são definidos pelo usuário por meio de uma especificação XML extensível que permite o rápido desenvolvimento de cenários complexos nos quais uma população de agentes heterogêneos participa de diversas atividades e caminha para objetivos distintos. Os principais benefícios desta ferramenta são: fornece uma estrutura simples para misturar e combinar soluções para subproblemas (determinar o que um agente quer fazer, como ele atingirá seu objetivo, como ele responde a desafios imprevistos e, para aplicações vi-

suais, determinar como o seu corpo virtual se move); suporta topologias e ambientes de navegação complexos; trata os agentes como entidades independentes; simula dezenas de milhares de agentes a taxas interativas; simula comportamentos e populações diversos e complexos; fornece interfaces extensíveis simples para analisar XML, gerenciar recursos compartilhados e interagir com outros elementos; oferece um visualizador simples e interativo que exibe parâmetros de agente e simulação em tempo real e suporta a exportação de trajetórias e comportamentos para o formato binário para uso com visualizadores externos e sintetizadores de movimento (CURTIS; BEST; MANOCHA, 2016).

Além destes simuladores que não possuem como foco a implementação de robôs, há também outros exemplos, como uCrowds (UCROWDS, 2019), CrowdSim (CASSOL et al., 2012) e MassMotion, sendo este último descrito com maiores detalhes na subseção seguinte.

2.7.2.1 Oasys® MassMotion®

O MassMotion® é um simulador de multidões. Tal simulador foi escolhido para ser utilizado neste trabalho porque permite a criação de cenários tridimensionais, usando um número definido de indivíduos (chamados de agentes), também tridimensionais, para simular o movimento destes em tais cenários de maneira inteligente, pois são levados em consideração alguns aspectos como, por exemplo, o desvio de obstáculos e a busca pelo melhor trajeto para atingir um objetivo (tempo, menor concentração de agentes, entre outros). Além disso, é capaz de gerar dados das posições que os agentes ocupam no tempo, essenciais para que um robô consiga classificar as células do ambiente de acordo com as intensidades de ocupação celular identificadas neste simulador.

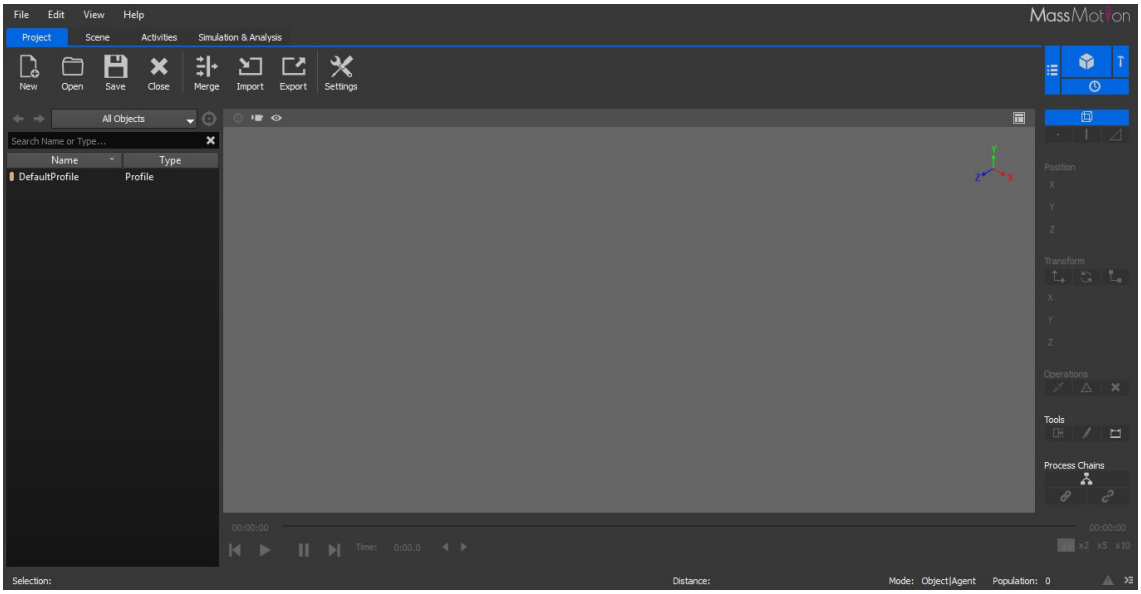
Os ambientes podem ser criados conforme a estrutura de prédios reais, incluindo níveis de elevação (andares), elevadores, escadas e escadas rolantes, por exemplo. Além disso, ele conta com opções de gráficos, mapeamento e tabelas que permitem a realização de análises detalhadas dos efeitos da simulação.

O MassMotion® é compatível com ferramentas de criação de objetos tridimensionais como o 3DS Max®, Autodesk® AutoCAD® e Vectorworks® (OASYS, 2018).

Quanto às licenças, podem ser para disponibilidade individual, ideal para instalações em um único computador, ou para disponibilidade compartilhada, possibilitando que várias máquinas façam uso de uma única licença compartilhada adquirida (OASYS, 2018).

A figura 13 apresenta a tela inicial do Oasys® MassMotion®.

Figura 13 – Tela inicial do Oasys® MassMotion®.



Fonte: Autor.

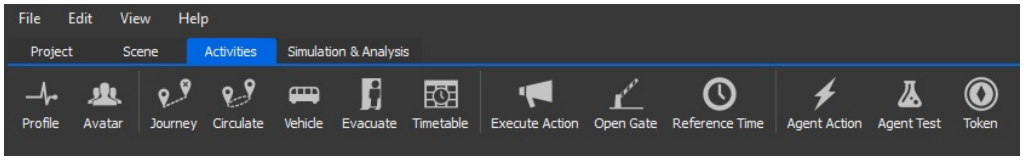
As principais funcionalidades deste simulador são:

a) Project: São disponibilizadas opções de gerenciamento do projeto como, por exemplo, a configuração de projetos e exportação arquivos em softwares de renderização 3D para melhorar a qualidade da simulação em ambientes mais realistas. Neste cenário, são apresentados os objetos que compõem o ambiente já modelado.

b) Scene: São encontradas opções dos objetos de modelagem do cenário como, por exemplo, obstáculos (paredes, caixas, entre outros), escadas, escadas rolantes, pisos, rampas e acessos. É possível ainda escolher opções que poderão definir, em certa medida, o comportamento dos agentes, para que sejam melhor distribuídos no cenário ou tenham restrições de acesso a determinados perímetros ou zonas do ambiente.

c) Activities: As opções desta aba permitem configurar as ações dos agentes no ambiente modelado. É possível definir as origens e destinos por onde deverão caminhar até a evacuação, trajetos por onde irão circular; atribuir tarefas para os agentes em horários específicos, simular horários de chegada de trens, de voos, de ônibus, entre outros. A figura 14 apresenta as funcionalidades gerais de Activities.

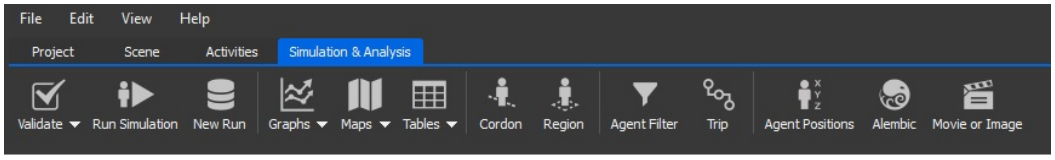
Figura 14 – Característica de *Activities*.



Fonte: Autor.

d) Simulation & Analysis: São encontradas opções de validação e execução do projeto como um todo. É possível também, gerar gráficos, tabelas e mapas para realização de análise do comportamento dos agentes durante toda a circulação pelo ambiente modelado. Os dados referentes aos posicionamentos de cada agente em relação ao tempo de circulação podem ser obtidos. A figura 15 apresenta as funcionalidades gerais de Simulation & Analysis.

Figura 15 – Característica da aba *Simulation & Analysis*.



Fonte: Autor.

Neste trabalho, grande parte destas funcionalidades e suas configurações foram utilizadas para a modelagem do ambiente fechado e simulação de evasão dos agentes, de modo que os dados coletados a partir destes resultados se tornaram elementos essenciais para a aplicação dos testes de comportamento do robô, também em ambiente simulado, com base na proposta anteriormente descrita. Entretanto, em situações reais, o robô deve ter um simulador de multidão embutido nele, tornando-o capaz de realizar a simulação de evacuação da multidão, bem como coletar os dados e aplicá-los para classificação das células no mapa.

2.8 CONCLUSÃO DO CAPÍTULO

Este capítulo apresentou os conceitos fundamentais para o embasamento teórico deste trabalho. Permitiu mostrar que entre os mais variados tipos de robôs, esta teoria pode ser apli-

cada em robôs autônomos móveis terrestres, que atuam em ambientes hospitalares, domésticos, de resgate e também apoio e entretenimento.

Uma vez definido o robô a ser utilizado, faz-se necessário os conceitos relacionados a SLAM para que o robô realize o mapeamento do ambiente fechado e identifique sua própria localização nele. Este princípio também é importante para o planejamento de sua trajetória, dando condições para ele realizar a sua navegação pelo ambiente.

O mapa gerado pelo robô necessita ser decomposto em células, contendo cada uma delas um valor numérico de status que mostra se a célula está vazia (0) ou ocupada (1) por algum obstáculo. No contexto deste trabalho, ainda um valor de nível intermediário (0.5) é utilizado para demonstrar uma região de atenção ao robô, conceito este que será apresentado no capítulo 4.

Entre os modelos de simulação de multidão encontra-se a técnica dos ACs, que permite definir o formato das células do ambiente (quadrado para este trabalho), além do conjunto de estados (representado pelas cores azul, amarela e vermelha quando os status são, respectivamente, 0, 0.5 e 1), da vizinhança (de Moore) e das regras de transição (fará uso de um algoritmo de planejamento de trajetória como, por exemplo, A*).

Muitas são as ferramentas de simulação existentes, algumas que aplicam robôs, porém sem a utilização de multidões e outras que aplicam multidão, porém sem robôs. Neste trabalho, um simulador de multidão sem robôs foi escolhido para a realização dos testes de validação da teoria. Entretanto, em cenários reais de aplicação, o robô deve possuir um simulador de multidão embutido para que ele próprio realize a simulação, cujos dados são importantes para a classificação das células nos mapas a serem gerados para o momento que ocorrer uma situação de emergência.

Estes conceitos são fundamentais para este trabalho, pois justifica as escolhas e definições quanto aos robôs e técnicas utilizadas.

3 REVISÃO DA LITERATURA

Atualmente diversos estudos estão sendo realizados na área de robótica, em especial no que diz respeito às possibilidades de interações entre seres humanos e robôs. Em muitos aspectos estas máquinas já fazem parte da vida das pessoas e a tendência é que esta relação de convívio cresça cada vez mais.

Neste capítulo serão apresentados alguns referenciais teóricos e pesquisas relevantes para dar embasamento a esta tese.

3.1 ROBÔS MÓVEIS AUTÔNOMOS

A navegação autônoma trata da capacidade do robô móvel se locomover no ambiente, sem a necessidade de intervenção humana, de forma eficiente e segura (SOUTO; MAIA; SOUZA, 2014). A partir da representação dos obstáculos estáticos do ambiente em um modelo (mapa), o robô irá se locomover da sua configuração (posição) inicial a uma configuração final, desviando-se de obstáculos e buscando, idealmente, percorrer o melhor caminho (SOUTO; MAIA; SOUZA, 2014).

Robôs autônomos podem auxiliar, por exemplo, em trabalhos industriais (STEINBAUER; LOIGGE; MÜHLBACHER, 2016; ZHANG; WANG; XIONG, 2016); em resgate de pessoas quando ocorrem desabamentos (CHEN et al., 2016); podem ainda ser úteis para desvendarem casos de situações em que algum veículo foi submerso ou ainda para exploração subaquática (YOO et al., 2015); além disso, espera-se que possam ajudar pessoas em hospitais, com serviços do cotidiano (GRAF et al., 2016); no ambiente urbano como companheiro de pessoas que precisam se locomover de um lugar à outro (FERRER; GARRELL; SANFELIU, 2013; GARRELL; SANFELIU, 2012), e até mesmo no ambiente doméstico (MEHMOOD; HASHMI; AJMAL, 2015), onde muitas tarefas podem ser executadas para manter-se a limpeza e organização da residência.

Robôs autônomos devem ser capazes de reconhecer pessoas, objetos e outros robôs de modo que, a partir disso, consigam se relacionar e se comunicar. É fundamental também, em muitos aspectos, que consigam aprender com o resultado da interação e aperfeiçoem seus comportamentos por meio de novos conhecimentos. Por exemplo, pessoas não treinadas podem instruir robôs através de diálogo e eles aprendem interativamente executando tarefas, ao invés de terem acesso a um grande repositório de conhecimento ambiental para entender os coman-

dos recebidos (KOLLAR et al., 2013). Neste caso, quando o robô está em um novo ambiente e alguém diz: *Vá para a cozinha*, o robô é capaz de aprender que *ir para* se refere à tarefa *GoTo* e que *a cozinha* se refere a um conjunto de locais no ambiente (KOLLAR et al., 2013). Os robôs devem, ainda, possuir capacidades para perceber, compreender e responder de uma forma socialmente apropriada. Habilidades sociais como aproximação e distanciamento são requisitos fundamentais para a questão da interação (CHARALAMPOUS; EMMANOUILIDISY; GASTERATOS, 2014).

As tecnologias empregadas em robôs têm sido estudadas e, além disso, utilizadas para a realização de tarefas que em tempos passados eram realizadas apenas por seres humanos. Há situações em que estes sistemas devem percorrer uma trajetória em determinada área e, para isso, devem possuir desempenho autônomo de navegação (FILLIAT; MEYER, 2003; NGUYEN; STARZYK; GOH, 2013; SOUTO; MAIA; SOUZA, 2014). Entretanto, um dos grandes desafios de um robô autônomo é conseguir identificar sua posição em um ambiente e, a partir disso, estabelecer sua trajetória, uma vez que as características do local podem variar, por exemplo, em relação à iluminação e posição de objetos no contexto.

Laflaquière et al. (2015) entendem que os sistemas robóticos autônomos não devem se basear na forma como os humanos enxergam o mundo, mas sim devem desenvolver sua própria maneira, a fim de obter percepção das coisas, tais como padrões que podem ser repetitivos ou invariantes, principalmente no que diz respeito a espaço. Para eles, fazer robôs mais autônomos e capazes de operar, a priori, em ambientes desconhecidos é o principal objetivo da investigação robótica moderna. Um robô autônomo será capaz de capturar a existência destas restrições e construir uma representação interna de configuração no que diz respeito ao espaço externo no qual ele está imerso.

Diversos são os desafios que ainda permeiam este ramo de pesquisa. O comportamento dos robôs nos ambientes, o reconhecimento de pessoas e objetos, o seu movimento adequado e até mesmo a detecção de expressões faciais e corporais de seres humanos são alguns exemplos.

As simulações deste trabalho possuem um viés de estudo com foco em robôs móveis autônomos terrestres em ambientes internos com multidão que são capazes de reconhecer pessoas, objetos e outros robôs, considerando-se os aspectos de seu comportamento em situações de emergência.

3.1.1 Robôs móveis autônomos na multidão em ambientes fechados

Robôs móveis autônomos na multidão é um assunto que tem recebido atenção por parte dos pesquisadores envolvidos no tema robótica. Os estudos são basicamente voltados à interação que os robôs podem ter com os seres humanos. Tais sistemas devem se movimentar em ambientes onde há grande concentração de pessoas, sejam estes ambientes fechados ou abertos. Podem eles ser agentes que irão cuidar da limpeza de ambientes (LIU, S. et al., 2016) ou até mesmo guiar pessoas para outros espaços (TRIEBEL et al., 2015a) ou para saídas de emergência (BOUKAS et al., 2015).

O movimento dos robôs em áreas de grande concentração de pessoas representa grandes desafios, pois faz-se necessário garantir que estes sistemas robóticos consigam desviar de obstáculos e ainda cuidar para que sejam evitados quaisquer tipos de contatos com as pessoas, de modo geral. O controle de sua velocidade em situações variadas também é um fator que requer atenção (BÖLÖNI; KHAN; ARIF, 2013).

Bölöni, Khan e Arif (2013) propõem a movimentação de um robô autônomo em um espaço estreito de um mercado cercado com lojas cujas entradas servem como marcos e um número de compradores realizam movimentos propositais. O robô deve levar em consideração os custos sociais de seu movimento, expressos em termos de violação do espaço pessoal dos humanos, bloqueio de seu caminho ou mesmo o contato físico com eles. Levando em consideração o fato de que o objetivo do robô é equilibrar suas metas de missão com o intuito de minimizar os custos sociais, a meta-estratégia chamada *passar-pela-frente*, que define que o robô deve tomar a decisão de atravessar na frente da multidão, evitando contato com ela, foi bem sucedida em relação às outras, que definem que o robô deve respeitar o espaço da multidão a todo custo, ou decidir deixar a multidão passar, ou ainda ignorar os custos sociais quase que integralmente.

Já o trabalho de Triebel et al. (2015b) apresenta o projeto SPENCER, uma plataforma de tamanho humano, onde o corpo se assemelha à funcionalidade de um balcão de informações e a cabeça serve como um dispositivo para uma comunicação não verbal compreensível, porém simplificada, que guia passageiros estrangeiros no aeroporto de Schiphol, Holanda, desde o portão de chegada, até um ponto de controle de passaportes mais distantes, ou até conexões europeias internas. Para interação física com o usuário, a plataforma possui uma tela e um leitor de cartão de embarque. Os sensores consistem de scanners a laser 2D, câmeras RGB-D frontais e traseiras e sistema de câmera estéreo à altura dos ombros. Foram desenvolvidas ferramentas

para a estimação automática de posturas corporais, classificação de atributos humanos como gênero e idade, estimativa de posicionamento de cabeça, detecção de voz e classificação de objetos importantes no ambiente. Experiências com o sistema completo mostraram que o robô é capaz de se aproximar e se envolver com uma pessoa, receber uma posição de objetivo e orientar a pessoa ou um grupo para a meta, mantendo o controle desta pessoa. Se uma falha de cooperação é detectada quando a pessoa não o segue mais, ele para e espera o reengajamento.

O trabalho de Bohórquez et al. (2016) aborda a navegação segura de um robô bípede em uma multidão, mantendo o equilíbrio e evitando colisões. Mostram que a preservação do equilíbrio, bem como da segurança passiva podem ser analisadas como a capacidade do robô de parar com segurança em algum momento no futuro. Os autores projetaram controladores para a navegação segura de robôs bípedes em multidões que mesclam conceitos de viabilidade de equilíbrio e evitação de colisão, além de desenvolverem dois controladores de segurança passivos e uma nova estratégia que aborda as limitações da segurança passiva. Os modelos foram avaliados em simulações de multidões em caminhada, caracterizados por recomendações sobre os requisitos mínimos de percepção do robô para navegar com segurança, notificações que falhas podem ser antecipadas o mais cedo possível e sem falsos alarmes em ambientes perfeitamente conhecidos, e que a estratégia de segurança passiva descontraída produz menos colisões do que as de segurança passiva e, quando ocorrem, são menos severas.

A proposta apresenta por Grewal et al. (2018) é, segundo os autores, uma nova abordagem para a seleção de um destino para uma cadeira de rodas autônoma em um ambiente interno não mapeado usando uma câmera, abrangendo Light Detection and Ranging (LIDAR), e visão computacional. O sistema analisa o ambiente e identifica uma relação de objetivos para o usuário optar por uma delas. o ambiente de testes foi um shopping center e os objetivos incluíam diversas lojas. O sistema de visão computacional foi testado com imagens de fachadas em várias distâncias e ângulos. Foram feitos dez ensaios para testar o sistema de navegação com objetivos de curto, médio e longo alcances. Como resultado, o sistema navegou com êxito até o objetivo em 100% dos testes para objetivos próximos e em 90% dos testes para objetivos de médio e longo alcances. Os autores concluíram que o projeto proposto é um meio promissor de seleção de destino para cadeiras de rodas autônomas em ambientes internos não mapeados para pessoas com deficiências graves.

Os trabalhos de Bölöni, Khan e Arif (2013), Triebel et al. (2015b), Bohórquez et al. (2016) e Grewal et al. (2018) contribuem com a proposta deste trabalho, pois embora não lidam com situações emergenciais, apresentam casos em que o robô navega pelo ambiente fechado

estimulado a atentar para os custos sociais (violação do espaço pessoal dos humanos, bloqueio de seu caminho, contato físico) que envolvem a relação robôs-humanos em ambientes fechados.

3.1.2 Robôs móveis autônomos em situações de emergência

Giglio-Jacquemot (2005) conceituam emergência como o acontecimento de alguma coisa séria, cuja aparição súbita ameaça ou causa perigo.

Um dos grandes desafios para os pesquisadores, de modo geral, tem sido os problemas que envolvem situações de emergência, principalmente (mas não somente) em ambientes fechados. As ocorrências dizem respeito a, por exemplo, terremotos, incêndios, ameaças de desabamentos ou o próprio desabamento em si. Não são raras as vezes em que podem ser vistos em noticiários, acidentes de pequeno, médio e grande portes ocorridos em vários locais pelo mundo. Por mais que as pessoas que representam instituições se esforcem ao máximo para manterem um nível elevado de segurança, algumas situações emergenciais continuam acontecendo devido à falta de previsibilidade das ocorrências.

Sempre que estas situações ocorrem, a ação imediata e natural das pessoas tomadas pelo pânico é tentarem a evasão do local. O início do pânico em uma multidão de pessoas sob tais circunstâncias pode desencadear uma debandada com consequências sérias e até fatais como, por exemplo, pessoas esmagadas ou pisoteadas (JOHNSON, 1987; ELLIOTT; SMITH, 1993 apud TRIVEDI; RAO, 2018). Manter o controle emocional em momentos como estes não é uma tarefa fácil, pois suas vidas podem estar em risco e é bastante difícil aferir as consequências que poderão trazer as atitudes que demonstram pânico. Acontece que o desespero se manifesta, na maioria dos casos, de modo a refletir em tragédias que poderiam ser evitadas ou, na pior das hipóteses minimizadas, se houvessem estratégias de evacuação pré-definidas para casos assim. A incerteza ou até mesmo desconhecimento a respeito da existência de tais estratégias levam as pessoas a considerarem qualquer sugestão externa como informação útil para a tomada de decisão muitas vezes precipitada. Neste contexto, as informações que recebem as pessoas em casos de emergência devem ser exatas para que o fluxo de evasão seja controlado e eficaz na prevenção de acidentes (PACHECO; KARELOVIC; CIPRIANO, 2015).

Evacuações, particularmente aquelas que surgem durante situações de emergência, apresentam características comuns, sendo as mais presentes o aumento da velocidade das pessoas, as interações físicas, o efeito de gargalo e a obstrução de saída (HELBING; FARKAS; VICSEK,

2000 apud BOUKAS et al., 2015). O efeito de estrangulamento é definido como a tendência da multidão em apressar-se em direção a uma saída, o que resulta em entupimento (BOUKAS et al., 2015).

Tang et al. (2016) apresentam um robô na multidão para evacuação em situações de emergência, com base em um modelo de mobilidade humana proposto e na informação atualizada sobre o meio ambiente. Além disso, propõem um algoritmo para selecionar a saída com o tempo de escape mínimo para acelerar a evacuação. Para avaliar tal evacuação durante a emergência, os comportamentos de saída dos evacuados são simulados durante a evacuação de emergência, juntamente com a decisão de orientação do robô no sistema de evacuação assistida por ele. Descrevem o comportamento da multidão em uma emergência, estendendo o Social Force Model (SFM) - Modelo de Força Social - (HELBING; FARKAS; VICSEK, 2000 apud TANG et al., 2016), usando forças atrativas, autodirigidas e repulsivas. Consideram uma situação de emergência em um ambiente interno que tem um ou mais eventos emergentes, como incêndios ou explosões, e o alarme no ambiente está ativo quando ocorre a emergência. Como resultado da simulação, houve a eficácia dos modelos propostos. Mostrou a melhoria da evacuação devido à orientação assistida por robô. Embora o robô só ofereça as informações de evacuação para uma pequena porção de evacuados durante a emergência, a eficiência de evacuação de todo o sistema ainda pode ser melhorada, de acordo com os autores.

O estudo de Tang et al. (2016) se relaciona com a proposta deste trabalho, pois apresenta um robô inserido em um ambiente interno com multidão tomando decisões no momento em que há a ocorrência de situações de emergência. Entretanto, a proposta dos autores converge para que o robô escolha a saída ideal de um ambiente (menos congestionada), com base em informações atualizadas sobre o mesmo, para guiar a multidão até ela, enquanto que a proposta desta tese não é o robô servir de guia, mas sim procurar uma área segura do ambiente para posicionar-se no momento da emergência para não atrapalhar a evacuação da multidão.

3.1.3 Robôs móveis autônomos em ambientes fechados - uso de autômatos celulares

O conceito de ACs tem sido uma boa opção entre os métodos de simulação para simulação dinâmica de multidões (SAKOUR; HU, 2017). Nos últimos anos, a metodologia dos ACs foi aplicada à dinâmica de multidões durante as evacuações. O ambiente é dividido em um conjunto de células que podem estar vazias ou ocupadas (WINTER, 2012). Os modelos de

ACs são utilizados com sucesso. para simular o processo de evacuação. A abordagem dos ACs pode ser apropriada para descrever a dinâmica de pedestres em situações complexas devido à sua simplicidade, flexibilidade e eficiência (ALIZADEH, 2011).

Os estudos de Boukas et al. (2015) se voltam para a utilização de robô móvel para conduzir um grupo de pessoas para saídas menos congestionadas, em situações em que seres humanos não podem ser enviados para guiar as pessoas, ou em casos de emergência onde sinais e outros meios estão fora de ordem devido a perda de energia, ou ainda em qualquer cenário de evacuação que exija uma dinâmica de movimento de evacuação por uma multidão. O robô utilizado é uma plataforma móvel autônoma, equipada com sensores de visão múltipla, e é competente em aplicações de exploração em interiores. O trabalho compreende o uso de AC baseado em um campo de chão (YANAGISAWA et al., 2009 apud BOUKAS et al., 2015), visão computacional para detecção humana e técnicas de planejamento de trajetórias. Cada uma das células AC ocupa uma área de 0,4 x 0,4 m que, em bloco, formam uma grade bidimensional (2D) que representa o espaço de trabalho AC. Os experimentos foram realizados em uma área retangular de 256 m^2, incluindo quatro saídas. Obstáculos causam bloqueio e aprisionamento da multidão. Dado o fato de que o robô opera em um ambiente interno já conhecido, o mapa estático da cena explorada foi calculado previamente. No entanto, durante a evacuação, o robô faz uso do mapa estático já existente e lida com as mudanças dinâmicas do ambiente, enquanto executa rotinas de planejamento e localização do caminho. Foram verificados benefícios no modelo de evacuação implementado, pois supera os problemas típicos de algoritmos relativos, tais como a repetição infinita da rota ou o bloqueio completo devido a formações específicas, por exemplo, os obstáculos em forma de U.

O trabalho de Tsompanas et al. (2018) apresenta a aplicação de autônomos celulares ao problema do caminho mais curto no movimento coordenado da robotização de enxames (múltiplos robôs). Um algoritmo baseado em ACs é introduzido para extrair dinamicamente os caminhos livres de colisão necessários para cada membro de uma equipe de robôs. O planejador apresentado considera também o comportamento coletivo que os robôs devem exibir para manter sua formação. O algoritmo baseado em AC pode extrair as rotas ótimas de cada robô em direção ao seu destino, enquanto os caminhos mais curtos também são calculados para recuperar a formação desejada. Como uma abordagem de decomposição celular, a área de configuração onde as equipes operam é inicialmente dividida em uma rede retangular simples de células quadradas idênticas. Ambas as mensurações da rede necessária são expressas em células e, portanto, dependem do comprimento celular aplicado. O conjunto de regras de transição deve

considerar os procedimentos de prevenção de colisão e o comportamento coletivo das células-robôs do enxame. Evitar obstáculos de uma célula-robô depende da caracterização de uma célula adjacente em sua vizinhança de Moore (ZUBELDIA, 2014) como uma célula-obstáculo. As regras de transição para evitar obstáculos são aplicadas para contornar o obstáculo. Um robô central, que atua como mestre, comanda seu robô adjacente para trocar suas posições, a fim de recuperar a formação do time. Comportamentos coletivos são executados e seguindo os caminhos mais curtos extraídos, a formação é recuperada com o custo de tempo mínimo. Finalmente, a equipe continua com suas posições finais cobrindo a distância definida.

O trabalho de Boukas et al. (2015) se assemelha à proposta deste trabalho em muitos aspectos, pois utiliza um modelo de autômato celular que divide o campo de chão em células, aquele para auxiliar o robô no direcionamento da multidão para evadir o local e este para auxiliar o robô a escolher o melhor lugar para estacionar a fim de manter o fluxo de evasão o mais livre possível. Em outras palavras, a ideia dos autores é que o robô auxilie as pessoas a evadirem o ambiente quando ocorre uma situação de emergência, apresentando as saídas mais livres de pessoas, situação esta que não é abordada neste trabalho, uma vez que neste modelo, casas e ambiente terão robôs capazes de realizar serviços diversos e, no momento em que houver uma emergência, as pessoas terão a opção de escolher qual a saída utilizar, competindo aos robôs posicionarem-se em uma região do ambiente fechado de modo a não prejudicar a saída da multidão. Entre as principais diferenças de ambos os trabalhos, nota-se que em Boukas et al. (2015) as situações de emergência consideradas são aquelas que não possuem nível crítico elevado, ou seja, situações em que as pessoas têm condições de evacuarem o ambiente tendo uma velocidade de 9,6 Km/h (2,6667 m/s) considerada rápida em seu modelo, enquanto que neste trabalho as situações podem ser modeladas levando-se em consideração situações com baixo nível crítico de emergência até níveis de emergência que envolvem pânico. Além disso, observa-se que o primeiro classifica as células como livres ou ocupadas, enquanto que este trabalho utiliza ainda uma terceira classificação, chamadas de região de atenção (RA), que mostra ao robô que, mesmo em situações sem emergência, ele não deve estacionar em qualquer célula deste tipo, pois é uma região onde os acessos estão localizados e, durante a emergência, ele se tornaria um obstáculo. Este trabalho, ainda, descreve não apenas um tipo de mapa do ambiente, classificado com células livres ou ocupadas, mas também considera que os robôs inseridos em ambientes fechados possuem um simulador de evacuação de multidão embutido neles, que irá contribuir para que outros mapas sejam gerados, contendo classificações das regiões de baixa, moderada e alta intensidades, a fim de antecipar os cenários de emergência

dos ambientes fechados para que o robô os use para estacionar em áreas de segurança durante o processo de evacuação pela multidão.

Já o trabalho de Tsompanas et al. (2018) possui características similares a proposta deste trabalho no que diz respeito ao uso de ACs para encontrar um caminho mais curto para navegação do robô até um determinado objetivo. Neste caso não há o contexto de emergência, nem mesmo de um simulador de multidão ou mesmo mapas com classificações de regiões com intensidades que variam entre um nível baixo e alto.

3.2 SIMULAÇÃO DE MULTIDÃO E AUTÔMATOS CELULARES

Modelos de ACs têm sido desenvolvidos para estudar a evacuação de multidão em diferentes situações. Tais modelos podem ser baseados na interação ambiente-pedestre (ZHENG; TINGKUAN; MENGTING, 2009 apud CARNEIRO, 2012), que leva em consideração, entre outros fatores, investigar a influência do meio ambiente na evacuação, ou na interação pedestre-pedestre (ZHENG; TINGKUAN; MENGTING, 2009 apud CARNEIRO, 2012), que considera, entre outros fatores, a investigação sobre o comportamento, a dinâmica e a interação existente entre as pessoas (CARNEIRO, 2012).

O trabalho de Dadová (2012), apresenta os temas da simulação de multidão, do controle comportamental e do uso de ACs, bem como as soluções de problemas relevantes para eles. Além disso, o autor propõe a solução, usando ACs, para a animação de multidões em três aspectos, sendo o primeiro com regras semelhantes ao Game of Life de Conway para o controle do movimento dos participantes em uma multidão, situação semelhante à de performance de dança, onde os movimentos são mais aleatórios. O segundo para o controle do movimento e do comportamento dos indivíduos na situação de flash mob espalhado para os novos participantes de forma semelhante, como preferências sociais ou doença. Finalmente, o terceiro, no controle comportamental sobre a multidão no ambiente virtual, ou seja, usa o AC para o controle do comportamento e também para o movimento no ambiente. Os padrões comportamentais abordados são baseados em psicologia para essa última situação. Para o autor, os métodos resolvem parcialmente problemas diferentes com o uso de AC, pois as soluções não são ótimas e, portanto, há espaço para melhorias.

Li e Han (2015) propuseram um modelo de simulação de evacuação de pedestres baseado nos ACs estendidos com a consideração de tendências comportamentais heterogêneas

em humanos. As tendências investigadas foram a familiaridade e a agressividade. Os experimentos de simulação testaram os valores dos parâmetros comportamentais e a composição do movimento de pedestres. O modelo é calibrado e validado usando cálculo manual, diagramas fundamentais de tráfego e experimentos de evacuação. Os resultados mostraram que o tempo de evacuação diminui exponencialmente com um aumento na familiaridade dos evacuados e aumenta quando os pedestres são muito conservadores ou muito agressivos. Além disso, o modelo é aplicado para otimizar os edifícios para reduzir os tempos de evacuação. Para eles, as tendências comportamentais têm uma forte influência no tempo de evacuação.

O método proposto por Czerniak et al. (2018), chamado Avaliação de Autómatos Celulares (CAEva - Cellular Automata Evaluation), utiliza a teoria dos ACs e pode ser usado para verificar as condições dos edifícios em caso de acidente. Os testes realizados em acidentes reais mostraram que um programa apropriadamente configurado permite obter uma simulação realística da evacuação humana. Um dos casos analisados foi o episódio ocorrido na *boate Kiss*, no Brasil, em 2013 (CNN, 2013 apud CZERNIAK et al., 2018). Este cenário envolvia rotas, visibilidade, sinalização, obstáculos, rotas sem saída, entre outros. As pessoas não seguiram para as saídas corretas ou não conseguiram sair por causa da aglomeração e obstáculos (não robóticos) dispostos no caminho. O cenário da fuga de pessoas do edifício foi recriado por meio de ACs. Determinando a probabilidade de uma pessoa se dirigir para a saída, os parâmetros de incêndio e a definição do efeito de grupo permitiram que os autores concluíssem que o método CAEva aparece como uma solução muito promissora, especialmente nos casos de obras de renovação ou indisponibilidade temporária de rotas de fuga. Neste trabalho, considera-se que as rotas de fuga estão configuradas de maneira que estejam bem definidas.

3.3 CONCLUSÃO DO CAPÍTULO

Robôs móveis autônomos estão interagindo cada vez mais com os seres humanos. Eles são inseridos em ambientes fechados com multidão para, entre outras tarefas, auxiliar as pessoas com limpeza, sendo guias, transmitindo informações e transportando objetos.

Muitos são os desafios que envolvem os estudos relacionados a este tema, pois os robôs devem navegar pelo ambiente de maneira planejada, sendo capaz de prever os custos sociais da interação com a multidão como, por exemplo, colidir com elas, impedir o sentido de locomoção, sinalizar algo de maneira equivocada, entre outros custos (BÖLÖNI; KHAN; ARIF, 2013).

Quando ocorre uma situação de emergência, diversas são as possíveis reações das pessoas. Os robôs podem auxiliá-las a evacuarem um ambiente fechado com um nível satisfatório de segurança. A teoria dos ACs pode auxiliar neste sentido, indicando até mesmo os caminhos mais curtos para o abandono do local, pois cada uma das células da estrutura da grade de ocupação terá uma classificação e isto irá contribuir para que o robô autônomo realize seus movimentos.

Para este trabalho, o robô não pode se tornar um obstáculo para a multidão que evacua o ambiente fechado. A classificação celular adequada faz com que ele crie mapas adequados para este propósito.

Alguns trabalhos aqui apresentados são exemplos que demonstram que a teoria dos ACs vem sendo estudada no contexto de simulações de multidão, não somente no movimento das pessoas pelo ambiente como também na evacuação em situações normais e de emergência. Embora os aspectos comportamentais das pessoas sejam o alvo de alguns destes estudos, tais aspectos são importantes para a compreensão de como os simuladores de multidão são estruturados.

Os conceitos destes trabalhos se relacionam à proposta deste, pois demonstram que a teoria dos ACs é utilizada para gerenciar a dinâmica da multidão, por meio de simulação, em ambientes fechados. Neste sentido, a simulação da dinâmica do comportamento de robôs, inseridos em ambientes fechados, pode também apresentar resultados significativos para a evacuação segura da multidão em situações de emergência, utilizando ACs.

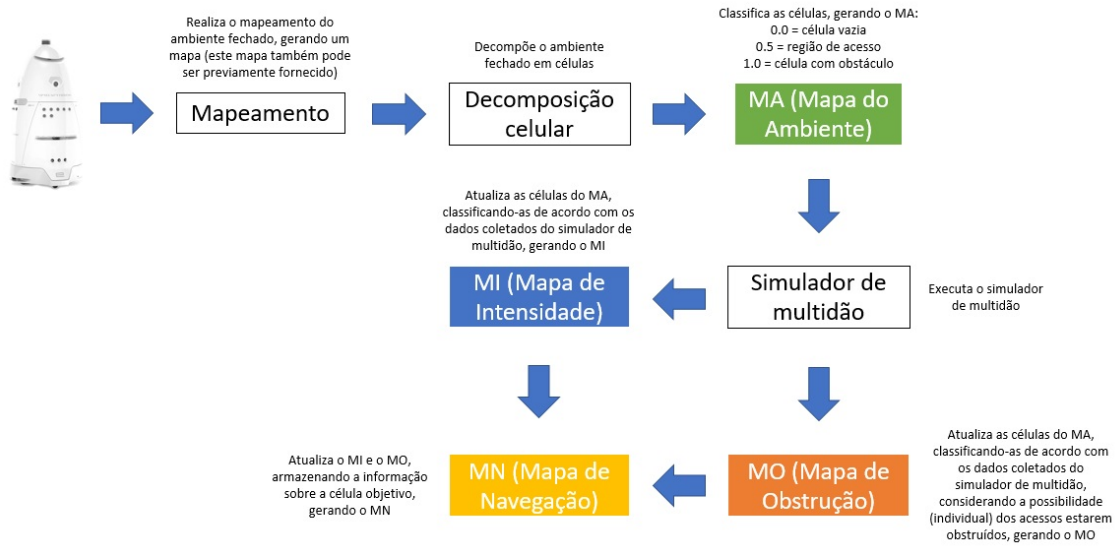
4 CLASSIFICAÇÃO CELULAR

Em ambientes fechados que abrigam multidão, é preciso considerar que quanto menos obstáculos neles localizados, melhores serão as condições de evacuação em situações de emergência. Neste contexto, sob a perspectiva do ser humano, o robô pode também ser um obstáculo, dependendo do local em que ele esteja estacionado. Se ele se posicionar, por exemplo, a qualquer momento, em frente a um acesso que não esteja obstruído, o sistema já se apresenta como vulnerável, pois o robô se tornará um empecilho, ao invés de auxiliar a multidão. Com respeito aos acessos, é possível caracterizá-los conforme a definição 4.1.

Definição 4.1. *(Acesso) Acesso é uma passagem específica que permite que a multidão entre ou saia do ambiente ou passe de um espaço para outro como, por exemplo, portas, janelas e corredores.*

Um robô autônomo móvel navegando no contexto de multidão em ambientes fechados necessita possuir estratégias que irão guiar seu comportamento tanto em situações normais quanto em situações de emergência. Deste modo, é preciso que ele possua, basicamente, quatro tipos de mapas que serão utilizados em casos específicos. A figura 16 apresenta as etapas de construção dos mapas utilizados nesta teoria até a definição da célula objetivo.

Figura 16 – Etapas de construção dos mapas utilizados pelo robô.



Fonte: Autor.

O robô autônomo, inserido em um ambiente fechado contendo multidão, realiza o mapeamento utilizando um conjunto de sensores (sobre sensores, consultar o apêndice A), embora este mapa possa também ser fornecido ao robô juntamente com sua localização neste cenário. Após este mapeamento, ele utiliza a técnica de decomposição celular por grades de ocupação, conforme mencionado no capítulo 2, seção 2.4, para fracionar o ambiente fechado em células de mesmo tamanho e formato. Na sequência, o robô gera o Mapa do Ambiente (MA), ou seja, um mapa de obstáculos estáticos (grade de ocupação estática) contendo as classificações numéricas de cada célula, sendo 0.0 para células livres e 1.0 para células ocupadas por paredes, objetos ou pessoas. Em seguida, executa um simulador de multidão embutido nele, sendo capaz de armazenar um arquivo que contém as posições de cada agente virtual (pessoa) durante o processo de evacuação do ambiente. Estes dados são utilizados pelo robô para gerar o Mapa de Intensidade (MI), onde cada célula recebe uma contagem acerca da quantidade de acessos a ela pela multidão. Células que foram mais acessadas são classificadas como de alta intensidade enquanto que as menos acessadas são classificadas como de baixa intensidade. O MI é o mapa de percepção sensorial (grade de ocupação dinâmica) que possui a classificação atualizada do MA, com base nos dados extraídos do simulador de multidão. Tais dados são utilizados também para que o robô gere Mapas de Obstrução (MO), onde considera que os acessos podem estar, por qualquer razão, obstruídos no momento em que ocorre a emergência. Na etapa seguinte, é construído o Mapa de Navegação (MN) contendo as classificações atualizadas das células e também a marcação da célula escolhida como objetivo para orientar o robô a respeito de sua trajetória no momento da evacuação do ambiente pela multidão. Estes mapas serão definidos em seções adiante.

Neste trabalho, o Mapa do Ambiente (MA) é utilizado em situações normais, enquanto que o Mapa de Intensidade (MI), o Mapa de Obstrução (MO) e o Mapa de Navegação (MN) são utilizados em situações emergenciais. Entretanto, todos eles são gerados, um após outro, em situações normais de navegação.

Para a descrição detalhada destes mapas é preciso, antes, considerar que o robô, quando inserido no ambiente fechado, poderá receber um mapa pronto ou, ainda, poderá construí-lo realizando o mapeamento do local, este último conforme técnicas encontradas em Bresson et al. (2017) e Gupta et al. (2017), por exemplo.

Definição 4.2. *(Mapa recebido) O mapa recebido é um mapa da estrutura do ambiente que é modelado de acordo com suas dimensões reais e armazenado no robô.*

Definição 4.3. *(Mapa construído) O mapa construído é um mapa que o robô gera a partir de sua navegação pelo ambiente utilizando seus sensores.*

Tendo como base este mapa recebido ou construído, o próximo passo é a construção do Mapa do Ambiente, que é constituído por meio de uma estrutura baseada em células. Esta estrutura celular pode ocorrer de duas maneiras, conforme a seguir.

4.1 DECOMPOSIÇÃO DO MAPA DO AMBIENTE EM CÉLULAS

De acordo com Ottoni (2000), o método de decomposição em células consiste em dividir o espaço livre do robô em regiões simples (células), de forma que um caminho entre duas configurações em células diferentes possa ser facilmente obtido.

Segundo Russell, Norvig e Davis (2010), o método de decomposição em células decompõe o espaço livre em um número finito de regiões contíguas, chamadas células.

O método de decomposição celular é importante para este trabalho, pois permite ao robô visualizar o ambiente fechado como uma malha de células de tamanhos fixos e de formatos regulares e, então, classificar cada uma delas, gerando mapas contendo informações sobre regiões livres, de atenção ou de risco para a passagem e permanência do robô, em situações normais e com emergência. Além disso, a cada instante de tempo, as células recebem ou não novas classificações, uma vez que é necessário considerar o dinamismo existente nos ambientes fechados como, por exemplo, obstáculos que são movidos de um lugar para outro ou ainda pessoas que se deslocam de uma posição para outra no ambiente.

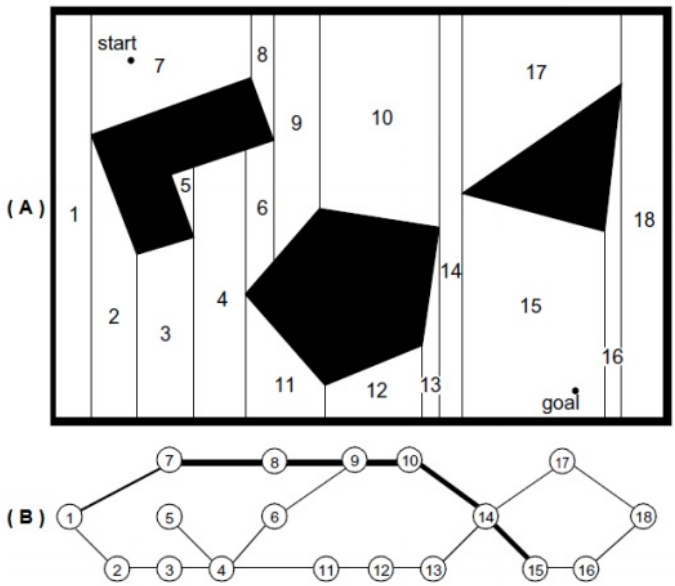
A estratégia de decomposição celular pode ser implementada por meio de dois métodos (SIEGWART; NOURBAKHSH, 2004; OTTONI, 2000 apud REINALDO, 2015), sendo a decomposição exata e a decomposição aproximada.

4.1.1 Método da decomposição exata

Este método decompõe o espaço livre em células cuja união é exatamente o espaço livre. Este conjunto de células é dito completo, pois sempre que possível este conjunto permite que um caminho entre duas configurações qualquer seja obtido, desde que seja utilizado um algoritmo de busca apropriado (OTTONI, 2000). Tal método precisa permitir que as células tenham forma

irregular nos lugares em que são adjacentes aos limites do espaço livre, porém as formas devem permanecer simples, facilitando os cálculos de percursos que passem por qualquer célula livre (RUSSELL; NORVIG; DAVIS, 2010). A figura 17a apresenta um exemplo da aplicação do método de decomposição exata.

Figura 17 – Exemplo de aplicação do método de decomposição exata e do grafo de conectividade.



Fonte: (SIEGWART; NOURBAKHSH, 2004 apud REINALDO, 2015).

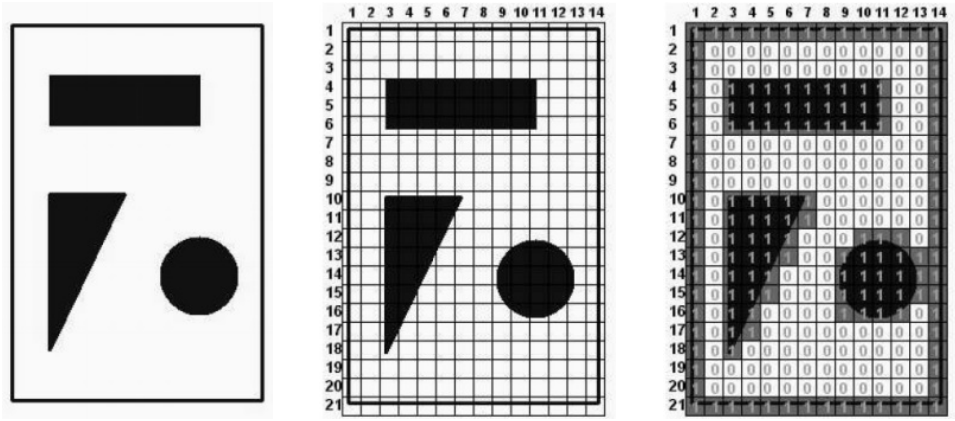
O grafo de conectividade pode ser construído (figura 17b), especificando através de uma linha mais cheia, o caminho entre a configuração inicial e final do robô (SIEGWART; NOURBAKHSH, 2004 apud REINALDO, 2015).

4.1.2 Método da decomposição aproximada

O método da decomposição por células aproximadas, também conhecida como mapeamento de grade de ocupação (MOJTAHEDZADEH, 2011), consiste em dividir o espaço livre do robô em células, considerando a mesma forma geométrica (quadrado, por exemplo). A figura 18 apresenta a decomposição celular aproximada, por etapas (OTTONI, 2000 apud REINALDO, 2015).

Figura 18 – Exemplo de aplicação do método de decomposição aproximada.

(a) Configuração do ambiente (b) Decomposição do ambiente em células (c) Aproximações realizadas pela discretização



Fonte: (OTTONI, 2000 apud REINALDO, 2015).

Na etapa 1, figura 18a, o mapa do ambiente está representado com três obstáculos. Na segunda etapa, esse mapa está discretizado em uma matriz de células de mesmo tamanho e é apresentado na figura 18b. Na etapa 3, figura 18c, observa-se o mapa anterior contendo, agora, valores indexados, sendo o valor zero para células sem obstáculos e valor 1 para células com obstáculos. Quando uma célula fica localizada numa região pertencente a um obstáculo, ela é considerada como ocupada, mesmo que não esteja totalmente preenchida. Ainda na figura 18c, as células onde não haviam obstáculos foram coloridas em branco, as células que estavam totalmente preenchidas com obstáculos foram coloridas de preto e as células que estavam parcialmente preenchidas pelos obstáculos foram coloridas de cinza. Pela figura 18c percebe-se que as células de cor cinza representam a imprecisão causada pela definição do tamanho das células. Uma forma de amenizar esse problema seria definir valores variáveis para o tamanho das células (OTTONI, 2000 apud REINALDO, 2015).

A teoria deste trabalho contempla o método da decomposição aproximada, mais precisamente do tipo grade de ocupação determinística, pois o formato geométrico deve ser único e de mesmo tamanho, uma vez que o ambiente é dinâmico, ou seja, os estados das células sofrem alterações com o tempo. Aplicações da técnica de decomposição por grade de ocupação podem ser encontradas nos trabalhos de Elfes (1989), Thrun (2001), Souza (2008), Mojtahedzadeh (2011), Dubois, Dib e Charpillet (2011), Carrillo et al. (2015), Shi, Kodagoda e Dissanayake (2010 apud REINALDO, 2015), Santana, Aires e Medeiros (2016), Wirges et al. (2018).

A seção seguinte irá apresentar a teoria sobre o mapa do ambiente que é gerado pelo robô utilizando decomposição celular e classificação de células com base no cenário (obstáculos, acessos, por exemplo) do ambiente fechado.

4.2 MAPA DO AMBIENTE - MA

Nas primeiras etapas de navegação do robô pelo ambiente fechado, um mapa que descreve este ambiente é gerado e, em seguida, ocorre a decomposição celular, seguindo o método da decomposição aproximada discutido na seção anterior (4.1.2). A partir deste resultado, o robô irá classificar cada uma das células, gerando assim o MA.

Definição 4.4. *(Mapa do Ambiente) O MA é o mapa decomposto em células regulares, de tamanhos e formas iguais, contendo a classificação de cada célula com base nas dimensões do ambiente e obstáculos nele localizados.*

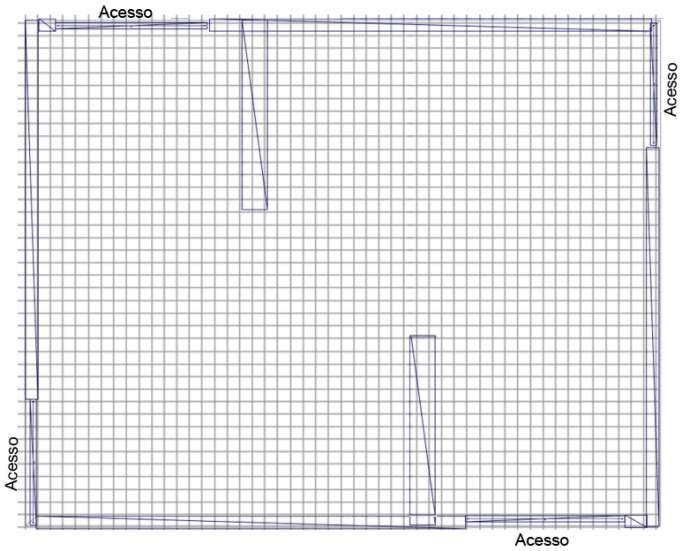
Definição 4.5. *(Classificação celular) Classificar uma célula significa atribuir a ela um valor numérico (0.0, 0.5 ou 1.0) que representa o estado desta célula no mapa.*

A classificação celular é importante para auxiliar o robô em sua navegação pelo ambiente. O MA é utilizado pelo robô durante toda a sua navegação em situações normais (sem emergência), usando um algoritmo de planejamento de trajetórias qualquer. Entretanto, o robô precisa construir este mapa, a partir do mapa recebido ou gerado. Esta teoria faz uso de um algoritmo de decomposição aproximada que divide o ambiente em células.

Definição 4.6. *(Célula) Uma célula possui um formato quadrado com área igual a 1 m^2.*

Uma célula é definida, neste trabalho, com formato quadrado e dimensões 1 x 1 metro, pois os experimentos analisados adiante consideram robôs que se enquadram neste tipo de célula sem ultrapassar os limites dela. A figura 19 apresenta um exemplo de mapa que pode ser recebido ou ainda gerado pelo robô, com decomposição celular, contendo quatro acessos (um em cada lado) e dois obstáculos (neste caso, paredes).

Figura 19 – Exemplo de mapa com decomposição celular.



Fonte: Autor.

Uma vez que o robô mapeou o ambiente e realizou a decomposição do mapa em células, a próxima etapa é classificar cada célula como livre ou ocupada, conforme definições a seguir. Há ainda a possibilidade da atribuição de valores intermediários, caso seja necessário definir outros status para as células.

Definição 4.7. *(Célula livre) Uma célula é considerada livre, no MA, com valor igual a 0, quando não houver nenhum obstáculo sobre ela, desde que não esteja na região de acesso.*

Definição 4.8. *(Célula ocupada) Uma célula é considerada ocupada, no MA, com valor igual a 1, quando houver algum obstáculo que a ocupe parcial ou totalmente.*

A classificação das células, bem como suas atualizações, pode ocorrer, por exemplo, fazendo-se uso de sensores do tipo laser ou sonar. Estes sensores também possibilitam que o robô se localize no mapa do ambiente (LIU, H. et al., 2017; MARINHO et al., 2017; LANG; WANG; SILVA, 2008 apud ALMEIDA et al., 2018). Entretanto, há outras maneiras de tratamento da localização do robô no ambiente como, por exemplo, por odometria e por utilização de sensor de profundidade tridimensional (GANGANATH; LEUNG, 2012 apud ALMEIDA et al., 2018), por utilização de encoders e visão computacional (LANG; WANG; SILVA, 2008 apud ALMEIDA et al., 2018) e, ainda, por utilização de landmarks, que são marcas de material

reflexivo posicionados no ambiente, de forma que refletem a luz emitida por sensores infravermelhos acoplados ao robô (LEE; SONG, 2007 apud ALMEIDA et al., 2018). A célula que está ocupada pelo robô no ambiente é classificada por ele com valor 1.

Ambientes fechados possuem acessos de entrada e saída da multidão e estes normalmente são estruturados com portas ou corredores. O robô precisa ser capaz de detectar tais acessos e classificar as células destas regiões, considerando que poderá se tornar um obstáculo no caso de estar navegando por ali no momento em que ocorre uma situação de emergência e a multidão necessite evacuar o ambiente. Algumas literaturas descrevem técnicas de detecção de portas baseadas no pressuposto de uma largura de porta fixa (MOZOS; STACHNISS; BURGARD, 2005 apud SHI; KODAGODA; DISSANAYAKE, 2010), estados dinâmicos de portas (ANGUELOV et al., 2004 apud SHI; KODAGODA; DISSANAYAKE, 2010) e regras temporais difusas (CARINENA et al., 2004 apud SHI; KODAGODA; DISSANAYAKE, 2010). Estas ideias citadas foram combinadas por Shi, Kodagoda e Dissanayake (2010) para construir um detector de porta usando heurísticas definidas por eles.

A identificação de células de área de acesso é fundamental para que ocorra a classificação adequada da região de próxima ao acesso, para que o robô evite estacionar em um local que o faça obstruir a evacuação da multidão no momento da emergência, principalmente.

Definição 4.9. *(Célula de área de acesso) Uma célula é considerada área de acesso, no MA, com valor igual a 0.5, quando não houver algum obstáculo que a ocupe parcial ou totalmente e quando estiver em uma área quadrada de algum acesso.*

Células de áreas que contém acessos são classificadas pelo robô com um valor variável, diferente de 0 (zero) e de 1, pois não deve considerar simplesmente uma área livre (0) a ponto de ocupar alguma destas células e permanecer parado ali, nem tampouco uma área ocupada (1) que o impeça de passar nesta região. Neste caso, o valor 0.5 deve ser atribuído às células dos acessos, representando, este valor, uma célula com status indeterminado. A quantidade de células a serem consideradas, próximas aos acessos, deve formar uma região quadrada a partir delas, tendo como base a largura (em linha) destes acessos.

Definição 4.10. *(Quantidade de células - CL) A quantidade de células CL representa a quantidade de células que forma a largura total do acesso (em linha).*

Definição 4.11. *(Quantidade de células - QC) A quantidade de células QC representa a quantidade total de células na região de acesso que tem formato quadrado, calculada a partir de CL, conforme equação QC = $(CL)^2$.*

A formação desta região quadrada é justificada pelo acúmulo de pessoas na emergência, além da necessidade da abertura do acesso e, neste caso, o robô posicionado nesta área poderá atrapalhar a evacuação rápida da multidão.

Além das classificações numéricas (0, 0.5 e 1), são utilizadas três cores, representando, cada uma delas, os valores numéricos de possíveis status das células, conforme a tabela 1.

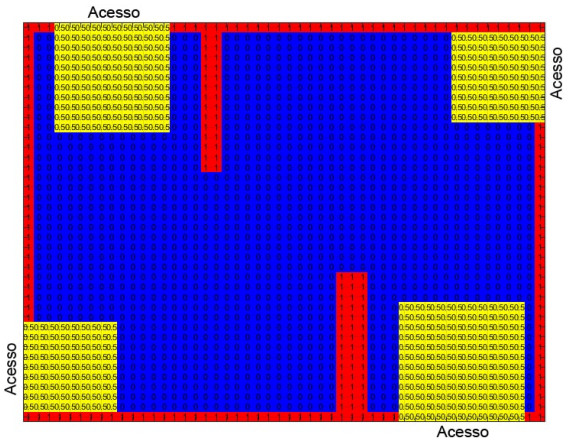
Tabela 1 – Possíveis estados e respectivas cores.

Estado	**Valor de status**	**Cor**
Livre (RL)	0	Azul
Região de atenção (RA)	0.5	Amarela
Região de risco (RR)	1	Vermelha

Fonte: Autor.

A figura 20 apresenta o MA com todas as células classificadas numericamente e com as respectivas cores.

Figura 20 – Exemplo de mapa de ambiente com decomposição celular e classificação das células.



Fonte: Autor.

Neste exemplo, é possível observar a existência de regiões livres, regiões de atenção e regiões de risco, utilizadas pelo robô para referenciar suas ações de navegação.

Definição 4.12. *(Região Livre - RL) Uma RL do MA é aquela formada por um conjunto de células com status igual a 0 (zero), representada pela cor azul, por onde o robô pode passar ou parar durante sua navegação pelo ambiente.*

Definição 4.13. *(Região de Atenção - RA) Uma Região de Atenção (RA) do MA é aquela formada por um conjunto de células com status igual a 0.5, representada pela cor amarela, por onde o robô pode passar, mas não pode parar durante sua navegação pelo ambiente.*

Definição 4.14. *(Região de Risco - RR) Uma RR do MA é aquela formada por um conjunto de células com status igual a 1, representada pela cor vermelha, por onde o robô não pode passar ou parar durante sua navegação pelo ambiente.*

A teoria descrita nesta seção envolve a geração do Mapa do Ambiente (MA) que é utilizado pelo robô em situações normais, ou seja, sem emergência. Entretanto, assim que uma situação emergencial ocorre, o robô precisa ter como base um mapa com configurações diferentes, com o intuito de procurar áreas no ambiente em que possa permitir que o deslocamento da multidão para os acessos não seja impedido. A seção seguinte apresenta a teoria que descreve este mapa, denominado de Mapa de Intensidade (MI).

4.3 MAPA DE INTENSIDADE - MI

Ambientes fechados não estão isentos da ocorrência de situações emergenciais em que as pessoas deverão evacuá-lo para que suas integridades física e psicológica sejam preservadas. Em ocasiões assim, um robô inserido neste contexto deve procurar estacionar em regiões do ambiente por onde as pessoas não passariam ou, na pior das hipóteses, passariam com pouca intensidade. Estas regiões são consideradas seguras e chamadas de regiões livres (RLs).

Definição 4.15. *(Mapa de Intensidade) O MI é o mapa decomposto em células, gerado pelo robô a partir do MA e dos dados referentes às posições de passagem dos agentes (pessoas) durante a evacuação do ambiente, extraídos de um simulador de multidão embutido no robô.*

O MI é utilizado pelo robô no momento em que receber a informação de uma situação de evacuação do ambiente fechado e indica quais são as regiões, na estrutura de decomposição celular, de alta, média e baixa intensidade de passagem de pessoas. Ele utiliza o MA como base, simula a evacuação na emergência e atualiza as células do ambiente com os dados do simulador, classificando-as novamente.

O robô armazena em sua memória ambos os mapas, cada um para ser utilizado em uma situação específica. A quantidade de agentes a ser configurada na simulação pode ser estabelecida de acordo com uma de três possíveis formas, conforme definições:

Definição 4.16. *(Quantidade fornecida) A Quantidade Fornecida (QF) de agentes é o número inteiro de agentes pré-definido e informado ao robô.*

Para os casos da definição 4.16, os administradores do ambiente realizam uma estimativa média do número de frequentadores e este dado é utilizado na simulação. Neste caso, o mapa de intensidade irá representar situações mais próximas da realidade de modo geral, porém não irá representar sempre o cenário mais crítico de uma situação de emergência.

Definição 4.17. *(Quantidade lida) A Quantidade Lida (QL) de agentes é o número inteiro de agentes que o robô utiliza após realizar uma contagem das pessoas no ambiente real.*

No caso da definição 4.17, o mapa de intensidade irá representar situações mais próximas da realidade em momentos pontuais, porém não irá representar sempre o cenário mais crítico de uma situação de emergência.

Definição 4.18. *(Quantidade por hipótese) A Quantidade por Hipótese (QH), de agentes, é o número inteiro truncado obtido pela razão entre a área total livre do MA, em m^2, e a quantidade de agentes por m^2 (ou por célula).*

Para os casos da definição 4.18 o mapa de intensidade não irá sempre representar situações mais próximas da realidade, porém irá representar sempre o cenário mais crítico de uma situação de emergência.

O modelo de ambiente criado no simulador deve levar em consideração os detalhes do ambiente real quanto às suas dimensões e posição de objetos, que normalmente ocupam

a mesma posição no cenário, como mesas, vasos de plantas, entre outros. Estes dados são representados no MA.

4.3.1 Parâmetros da simulação

Conforme mencionado anteriormente, o robô faz a simulação da evacuação dos agentes inseridos neste ambiente por meio de um simulador embutido nele. Para que seja possível a realização da simulação, alguns parâmetros devem ser estabelecidos, conforme apresentados nas seções a seguir:

4.3.1.1 Quantidade de agentes

A quantidade de agentes é fundamental para a simulação. Qualquer uma das definições 4.16, 4.17 e 4.18 pode ser utilizada para estabelecer o número de agentes a ser considerado para mostrar a intensidade de ocupação em cada célula do ambiente.

É possível, por meio deste parâmetro, simular os aspectos de emergência mais ou menos críticos ou ainda aqueles que mais se aproximam da realidade.

Na simulação, a quantidade de agentes calculada deve ser distribuída por toda área livre do MA.

4.3.1.2 Partida da multidão

Um ponto a ser considerado é a partida da multidão no momento em que ocorre a emergência, mais precisamente quando a multidão tem a informação da emergência por um alarme, por exemplo. A reação para a evacuação do ambiente pode ser imediata ou não.

Definição 4.19. *(Partida) A partida da multidão refere-se ao modo como os agentes reagem no exato momento em que ocorre a sinalização de uma situação de emergência.*

O parâmetro da definição pode ser de modo aleatório, quando os agentes saem de suas posições em direção aos acessos em espaços de tempos distintos, ou instantâneo, quando os agentes saem de suas posições em direção aos acessos no mesmo instante.

Este parâmetro é importante de ser observado, pois as intensidades no MI poderão ficar mais compactas ou mais espalhadas, dependendo da escolha que for feita, aumentando ou diminuindo as possibilidades de áreas livres no mapa.

4.3.1.3 Destino da multidão

O destino que a multidão irá escolher para a evacuação deve também ser um parâmetro para o robô.

Definição 4.20. *(Destino) O destino da multidão refere-se ao critério de como a multidão deverá escolher o acesso para a evacuação, podendo ser por pesos ou por nível de congestionamento.*

A escolha do destino por nível de congestionamento considera que cada agente busca o acesso mais próximo e, caso esteja congestionado, busca um novo acesso. Já a escolha por pesos considera que cada acesso possui um valor numérico que representa o grau de prioridade que possuem e, neste caso, os agentes procuram os acessos mais prioritários, porém se chegar em algum que esteja congestionado eles procuram outro acesso.

Este parâmetro é importante de ser observado, pois ele determina, entre outras coisas, o comportamento da multidão durante a evasão, na emergência.

Após realizar a simulação com estes parâmetros, o robô armazena os dados das posições de cada agente, a cada segundo, em uma matriz para utilizá-los na classificação celular do MI.

4.3.2 Classificação celular no MI

Conforme mencionado anteriormente, durante a navegação no ambiente fechado, realizada pelo robô, o MA é construído e servirá como base para a construção do MI, que ocorrerá após o robô utilizar seu simulador de multidão para obtenção dos dados necessários para a nova classificação das células.

Os dados extraídos do simulador de multidão contendo as coordenadas (x,y) de cada agente tornam-se parâmetro para classificar as células correspondentes no cenário, de acordo com a quantidade de ocorrências (passagem de agentes) em cada uma delas. As regiões mais

intensas são aquelas que possuem células com quantidades maiores de ocorrências e o oposto serve para as regiões menos intensas. Um exemplo é apresentado na figura 21.

Figura 21 – Exemplo de valores de coordenadas de cinco agentes extraídos de um simulador de multidão convertidos em quantidades de acessos em cada célula.

(a) Coordenadas (x,y) de cada agente em tempos distintos

Agente	x	y
A	2	3
A	3	2
A	3	1
B	4	4
B	4	3
B	4	2
B	3	1
C	2	2
C	3	1
D	3	4
D	2	3
D	2	2
D	3	1
E	5	3
E	4	2
E	3	1

(b) Grade de ocupação contendo valores atribuídos referentes à quantidade de acessos por célula

	1	2	3	4	5
1	0	0	5	0	0
2	0	2	1	2	0
3	0	2	0	1	1
4	0	0	1	1	0
5	0	0	0	0	0

(c) Mapa de intensidade (MI) - Grade de ocupação contendo valores atribuídos normalizados referentes à quantidade de acessos por célula

	1	2	3	4	5
1	0	0	1	0	0
2	0	0,4	0,2	0,4	0
3	0	0,4	0	0,2	0,2
4	0	0	0,2	0,2	0
5	0	0	0	0	0

Fonte: Autor.

Neste exemplo, observa-se na figura 21b que cinco agentes ocuparam, em algum momento, a célula de coordenadas (3,1). Pode-se deduzir que, por se tratar de uma emergência, esta célula faz parte de uma região de acesso por onde as pessoas evacuam o ambiente fechado. Já na célula de coordenadas (4,2) da figura 21b, por exemplo, passaram apenas 2 agentes na evacuação, sendo eles o agente B e o agente E, o que pode ser confirmado na figura 21a. Além disso, dezesseis células possuem classificação 0 (zero), significando que em nenhum momento foram visitadas por qualquer agente.

Um exemplo de algoritmo que calcula a quantidade de acessos dos agentes em cada célula é representado no algoritmo 1:

Algoritmo 1 – Algoritmo que calcula o número de ocorrências em cada coordenada (x,y) do mapa.

```
Entrada: Inteiro i,k,l,n; Matriz matriz,MAP
n = length(matriz); %armazena a quantidade de linhas da matriz.
para i=1:n faça
    k = matriz(i,1);
    l = matriz(i,2);
    MAP(l,k) = MAP(l,k) + 1;
fim
retorna MAP
```

Na primeira leitura, o algoritmo armazena na variável ‘k’ (tipo inteiro), o valor correspondente à posição (1,1) da matriz denominada ‘matriz’, enquanto que a variável ‘l’ (tipo inteiro) armazena o valor correspondente à posição (1,2) da matriz de mesmo nome, que foi criada anteriormente. A matriz de nome ‘MAP’ (matriz de coordenadas), deve atribuir o valor ‘1’ (um) para a coordenada (l,k) lida. A cada nova leitura, sempre que a mesma coordenada lida já tiver algum valor atribuído, o algoritmo soma um valor inteiro ao anterior e atribui o novo número nesta coordenada.

A figura 21c apresenta o Mapa de Intensidade (MI) gerado pelo simulador de multidão embutido no robô, ou seja, a grade de ocupação normalizada para valores que vão de 0 a 1, que representam, respectivamente, nenhuma e máxima intensidade de acesso. Há, ainda, valores intermediários que serão importantes para a decisão do robô em relação ao seu posicionamento.

O valor normalizado de cada célula é obtido por meio da simples equação:

$$C_{\text{i,j(norm)}} = C_{\text{i,j}}/q \tag{4}$$

onde $C_{\text{i,j(norm)}}$ é o valor normalizado, $C_{\text{i,j}}$ é o valor anterior da célula e q é o maior valor encontrado na grade.

Deste modo, é possível definir as áreas do mapa de três formas:

Definição 4.21. *(Área de alta intensidade) A área de alta intensidade corresponde às regiões que foram muito visitadas pelos agentes, na simulação. Isto quer dizer que* $C_{i,j(norm)}$ *é 1 ou está próximo a ele.*

Definição 4.22. *(Área de moderada intensidade) A área de moderada intensidade corresponde às regiões que foram moderadamente visitadas pelos agentes (nível intermediário), na simulação.*

Definição 4.23. *(Área de baixa intensidade) A área de baixa intensidade corresponde às regiões que foram pouco visitadas pelos agentes, na simulação, ou seja, zero ou próximo a zero para* $C_{i,j(norm)}$.

A tabela 2 apresenta um exemplo de níveis de emergência e as respectivas classificações de suas áreas. A configuração dos tipos de intensidade permite que sejam construídos modelos diferentes para diversas situações de emergência, pois admite-se que algumas delas são menos críticas e a multidão evacua o ambiente sem causar tumultos (conforme apresentado em trabalhos como Boukas et al. (2015), por exemplo), ao contrário de outras que são mais críticas e podem causar pânico na multidão (um incêndio no prédio, por exemplo). Para a primeira situação, o nível de emergência NE1 é mais indicado, pois contribui para que o mapa do ambiente possua um número maior de células livres (RL), conforme mostra a figura 22. Para a segunda situação, o nível de emergência NE2 é mais indicado, pois contribui para que o mapa do ambiente possua um número maior de regiões de baixa, moderada e alta intensidade, conforme mostra a figura 22. Assim, o robô irá procurar regiões que não atrapalhem o sentido de evacuação no momento da emergência.

Tabela 2 – Exemplo de níveis de emergência com base na classificação das áreas para geração do MI.

Estado	**Cor**	**Nível de emergência NE1 (menos crítica)**	**Nível de emergência NE2 (mais crítica)**
Região Livre (RL)	Azul	0 a 0.4	0 a 0.19
Região de atenção (RA)	Amarela	0.41 a 0.8	0.2 a 0.39
Região de risco (RR)	Vermelha	0.81 a 1	0.4 a 1

Fonte: Autor.

É possível observar que as situações mais críticas (NE2, por exemplo) exigem que os parâmetros de configuração para a região de risco (RR) sejam mais espaçados.

A figura 22 ilustra o MI que foi gerado na figura 21c, implementando os dois níveis de emergência apresentados na tabela 2, usando cores, sendo a azul uma região de intensidade baixa para o robô, enquanto que a amarela é de intensidade moderada e a vermelha é de alta intensidade.

Figura 22 – MIs com as classificações dos exemplos de níveis de emergência da tabela 2.

(a) MI para situações de NE1, ou seja, movimento sem pânico

	1	2	3	4	5
1	0	0	1	0	0
2	0	0,4	0,2	0,4	0
3	0	0,4	0	0,2	0,2
4	0	0	0,2	0,2	0
5	0	0	0	0	0

(b) MI para situações de NE2, ou seja, movimento com possível pânico

	1	2	3	4	5
1	0	0	1	0	0
2	0	0,4	0,2	0,4	0
3	0	0,4	0	0,2	0,2
4	0	0	0,2	0,2	0
5	0	0	0	0	0

Fonte: Autor.

Pode-se observar na figura 22 que, dependendo do nível de emergência, o MI se apresenta de forma diferente, contribuindo para que o robô decida seus movimentos de maneira adequada, em cada situação.

A figura 22a mostra um exemplo de MI que poderia ser utilizado em situações com baixo nível de emergência (caso da falta de emergia elétrica), pois como pode ser observado, o robô poderia estacionar em quase todas as células do ambiente, exceto a de cor vermelha que é o acesso das pessoas. Se for definido um objetivo pelo menor valor, ele estacionaria em uma célula qualquer de valor 0 (zero), a que estivesse mais próxima a ele, por exemplo.

A figura 22b mostra um exemplo de MI que poderia ser utilizado em situações críticas de emergência (um incêndio, por exemplo). Observa-se, neste caso, que somente as células com valores 0 (zero) podem ser o objetivo do robô, porém não necessariamente a mais próxima a ele, já que em certos casos, para ele chegar ao objetivo, deveria passar por uma região vermelha, ou seja, região de risco (RR), e isto não é recomendado, uma vez que esta região representa movimento intenso e ele poderia, então, causar acidentes.

Considera-se a normalização das células do MI para o mesmo padrão estabelecido no MA, na tabela 1 da seção 4.2, ou seja, valor 0 (zero) para uma região RL, 0.5 para uma região RA e 1.0 para uma região RR. Assim, um segundo nível de normalização deve ser realizado para que os valores intermediários a 0 e 1 (conforme figura 21c, os valores 0.2 e 0.4) sejam relacionados ao padrão do MA.

Esta nova normalização utiliza como base o exemplo anteriormente apresentado da tabela 2 e está descrito na tabela 3. Neste trabalho, os parâmetros desta tabela foram utilizados, pois expressam os casos mais críticos de emergência.

Tabela 3 – Valores de referência para classificação das células do MI.

Estado	**Cor**	**Intervalo de classificação**	**Valor normalizado**
Região Livre (RL)	Azul	0	0.0
Região de atenção (RA)	Amarela	> 0 e ≤ 0.1	0.5
Região de risco (RR)	Vermelha	> 0.1	1.0

Fonte: Autor.

As definições de estado das células no MI são estabelecidas conforme a seguir:

Definição 4.24. *(Célula livre) Uma célula é considerada livre quando seu valor, após a normalização, é igual a 0.0 (zero).*

Definição 4.25. *(Célula de RA) Uma célula é considerada de atenção quando seu valor, após a normalização, é igual a 0.5.*

Definição 4.26. *(Célula de RR) Uma célula é considerada de risco quando seu valor, após a normalização, é igual a 1.0. No caso específico deste tipo de célula, diferentemente do que trata a definição 4.14, o robô pode passar por esta região, mantendo-se a determinação de que não pode parar.*

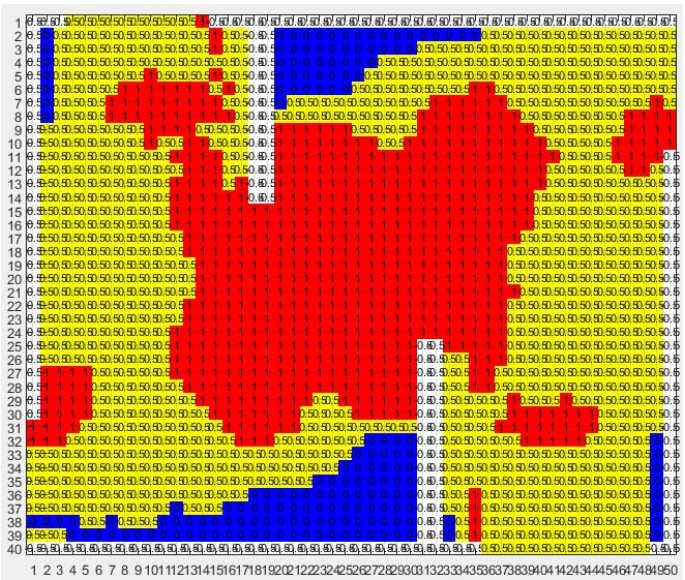
O algoritmo que realiza esta normalização é dado conforme a seguir:

Algoritmo 2 – Algoritmo que normaliza os valores das células, conforme padrão do MA.

```
Entrada: Inteiro nx, ny, maior; Matriz MAP1,MAP2
para nx=1:40 faça
    para ny=1:50 faça
        MAP1(nx,ny) = MAP(nx,ny) / maior; % maior é o valor mais alto da matriz
        se (MAP1(nx,ny) > 0.1) então
            MAP2(nx,ny) = 1;
        senão
            se (MAP1(nx,ny) > 0) então
                MAP2(nx,ny) = 0.5;
        senão
            MAP1(nx,ny) = 0;
    fim
fim
retorna MAP2
```

Este algoritmo utiliza a variável *maior* para realizar a primeira normalização que compõe a matriz *MAP1* com valores intermediários a 0 (zero) e 1 e, para a segunda normalização, os valores desta matriz são condicionados aos parâmetros da tabela 3 para gerar a matriz normalizada *MAP2*, ou seja, o MI. A figura 23 apresenta um exemplo de MI construído com a ferramenta MATLAB®, usando os parâmetros de configuração citados anteriormente.

Figura 23 – Exemplo de MI gerado pelo robô, com o status de cada célula representado por cores.



Fonte: Autor.

Nesta figura, é possível verificar uma região central de alta intensidade de passagem da multidão acompanhada de uma grande região de moderada intensidade, regiões estas que serão evitadas pelo robô durante uma emergência. O valor numérico -0.5, na cor branca, serviu apenas para identificação das paredes do ambiente neste mapa, não representando uma nova classificação para as células. A seguir será discutida brevemente a situação particular da obstrução de acessos, na emergência.

4.4 MAPA DE OBSTRUÇÃO - MO

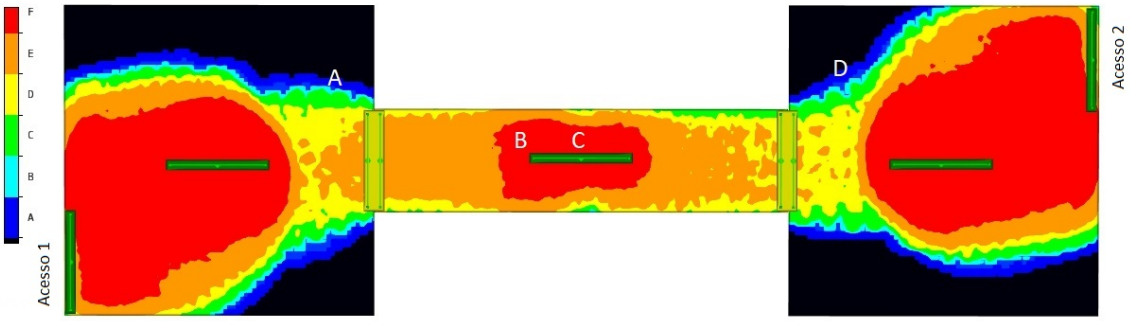
Uma situação de emergência pode ocorrer por razões diversas como, por exemplo, incêndio, desabamento ou risco de desabamento, ameaça de bomba, entre outras. Como o ambiente é fechado, tais situações podem ocorrer em qualquer localização, inclusive em regiões próximas aos acessos, impedindo, assim, que a multidão abandone o local por tais acessos obstruídos. Se este for o caso, a multidão que passaria por determinado acesso deverá procurar outro, fazendo com que o sentido da multidão se comporte de maneira diferente, alterando com isso o MI.

Definição 4.27. *(Mapa de Obstrução) O MO é o mapa decomposto em células, gerado pelo robô a partir do MA e dos dados referentes às posições de passagem dos agentes virtuais (pessoas) durante a evacuação do ambiente, extraídos de um simulador de multidão embutido neste robô, para uso quando houver obstrução de algum acesso.*

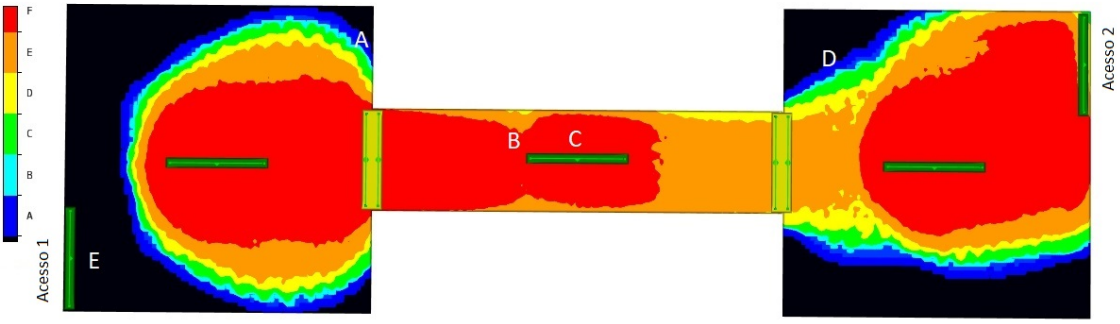
A figura 24 apresenta dois comportamentos da multidão em um cenário que contém um corredor que une dois ambientes, cada um com um acesso em suas extremidades. A figura 24a mostra o cenário onde a situação de emergência ocorre em qualquer local que não prejudica o sentido de movimento da multidão para nenhum dos acessos (qualquer região na cor preta, por exemplo). Se o robô estiver posicionado na região C do mapa e considerando que ele deverá procurar uma região de pouca intensidade para estacionar, nesta situação ele poderia percorrer a trajetória C $\rightarrow$ A ou C $\rightarrow$ D. Entretanto, se estiver posicionado na região B, a trajetória mais adequada é B $\rightarrow$ A, uma vez que o tempo para esta navegação é menor do que o tempo de navegação na trajetória B $\rightarrow$ D. Para estas duas situações a navegação do robô não atrapalha o sentido da evacuação, pois em qualquer uma delas ele estará acompanhando a multidão.

Figura 24 – Exemplo de simulações, obtido do simulador de multidão, considerando alteração no mapa de intensidade com base na obstrução de um dos acessos do ambiente.

(a) Situação que não prejudica o sentido de movimento da multidão.



(b) Situação onde o sentido de movimento da multidão é alterado em razão do local da obstrução, em *E*.



Fonte: Autor.

Já na figura 24b, considera-se que uma ocorrência como, por exemplo, um incêndio, surge em uma região próxima ao acesso 1, mais precisamente na área representada pela letra E. Deste modo, ele ficará obstruído. Observa-se neste mapa que todos os agentes do cenário procuram evacuar o ambiente pelo acesso 2, tornando único o sentido da multidão. Por esta razão, a trajetória B $\rightarrow$ A, embora seja a mais econômica na questão do tempo, que era a mais adequada na situação da figura 24a, passa a ser inapropriada, pois o robô estaria se movimentando contra o sentido da multidão. O mesmo ocorre para a trajetória C $\rightarrow$ A. Esta situação poderia ocasionar problemas graves quanto à integridade física das pessoas, pois o robô estaria sendo um obstáculo à evasão delas. Sendo assim, o robô deverá decidir acompanhar o sentido da multidão até sua meta, mesmo que o tempo para encontrar a posição mais segura (RL) seja maior. A trajetória mais indicada seria B $\rightarrow$ D, neste caso (C $\rightarrow$ D, se ele estiver localizado em C).

No ambiente fechado da figura 24, o robô precisará dos dados da simulação para cada situação em que o sentido de movimento da multidão possa ser alterado. Assim, ele construirá pelo menos três tipos diferentes de mapas:

a) o primeiro (MI), para os casos onde o sentido de movimento da multidão não é comprometido;

b) o segundo (MO), para os casos onde o acesso 1 está obstruído;

c) o terceiro (MO), para os casos onde o acesso 2 está obstruído.

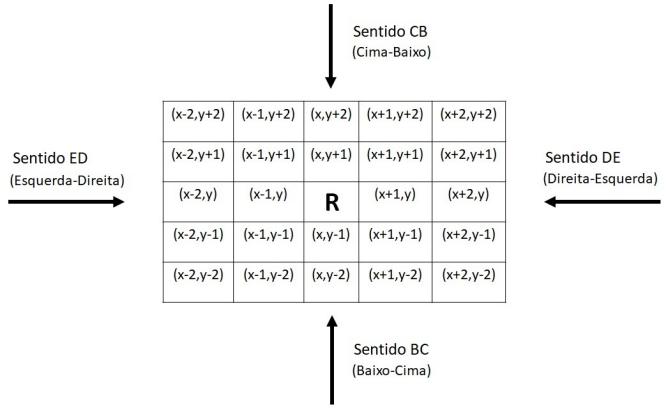
Em qualquer tipo de ambiente, faz-se necessário que o robô realize a simulação e gere os MOs adequados para cada situação onde os acessos poderão estar obstruídos.

Além destes, outros casos deverão ser considerados como, por exemplo, situações em que a emergência ocorre no corredor. Entretanto este estudo envolve variáveis que não fazem parte do escopo deste trabalho como, por exemplo, o conhecimento do fluxo em cada célula do ambiente.

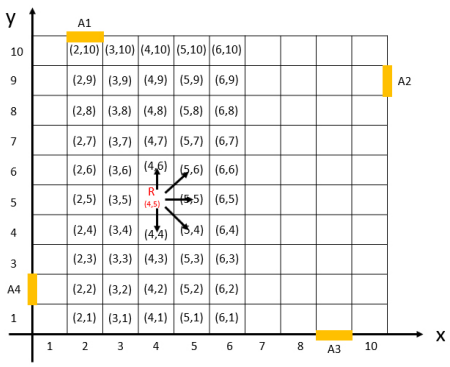
Para que o robô possua conhecimento do sentido de movimento da multidão, fundamental para que não navegue em sentido contrário a ela no momento da emergência, ele recebe a informação de obstrução de determinado acesso, por meio um sistema de segurança qualquer. Sendo assim, o robô utiliza as referências das coordenadas de suas células vizinhas, baseando-se em suas próprias coordenadas (x,y) para desqualificar as células que fariam com que ele definisse um objetivo contra o deslocamento da multidão. A figura 25 apresenta possíveis sentidos de movimento das pessoas, além das referências das coordenadas da vizinhança em relação à coordenada (x,y) do robô.

Figura 25 – Exemplos de sentidos de movimento da multidão e coordenadas da vizinhança do robô.

(a) Exemplo de possíveis sentidos de movimento da multidão.



(b) Exemplo de posição das células e posicionamento do robô na malha.



Fonte: Autor.

Na figura 25a é possível observar que o robô está localizado ao centro da malha de células, representado pela letra R, e sua posição é (x,y). A vizinhança do robô está representada, nesta figura, de acordo com suas respectivas coordenadas, a partir de R. Deste modo, assim que o robô recebe a informação de que há uma emergência e de que a ocorrência obstruiu o acesso à esquerda, por exemplo, ele utiliza seu conhecimento acerca de sua orientação para excluir a possibilidade de navegar para qualquer célula que possua coordenada x com valor inferior a sua (ou seja, qualquer célula que tenha x-1).

Na figura 25b é possível observar que o robô está localizado na célula de coordenadas (4,5). No caso de receber a informação de que o acesso A4 (em laranja, à esquerda do mapa), por exemplo, está obstruído, o algoritmo de navegação do robô impede que ele se desloque para qualquer célula de coordenada x = 3. Deste modo, o sentido de seu movimento será para cima, para baixo, para a direita ou para uma das diagonais à direita, mas não para a esquerda ou para uma das diagonais à esquerda. Para os demais casos, onde outros acessos do exemplo da figura 25b possam estar obstruídos, o tratamento do robô deve ocorrer conforme a seguir:

a) Se A1 estiver congestionado, o robô não irá se deslocar para células com coordenadas y = 6;
b) Se A2 estiver congestionado, o robô não irá se deslocar para células com coordenadas x = 5;
c) Se A3 estiver congestionado, o robô não irá se deslocar para células com coordenadas y = 4;

A próxima seção apresentará a teoria do Mapa de Navegação, necessário para que o robô procure a melhor célula para estacionar enquanto há uma emergência e as pessoas evacuam o ambiente.

4.5 MAPA DE NAVEGAÇÃO - MN

O Mapa de Navegação é destinado a contribuir para que o robô, em uma situação de emergência, procure se deslocar para as regiões livres (RL) do mapa, utilizando um algoritmo de navegação qualquer como, por exemplo, o algoritmo A* (HART; NILSSON; RAPHAEL, 1968 apud RUSSELL; NORVIG; DAVIS, 2010).

Definição 4.28. *(Mapa de navegação) O Mapa de Navegação (MN) é o mapa decomposto em células, gerado pelo robô a partir do MI e do MO, que armazena a informação a respeito de qual é a célula objetivo a partir da posição do robô no ambiente.*

Este mapa requer que o robô conheça sua posição no ambiente, ou seja, a célula na qual está localizado, e também defina a sua célula objetivo, que deverá ser alguma célula onde a região seja livre (RL), conforme tabela 3 apresentada anteriormente.

Ao chegar na célula objetivo o robô deve ficar parado até que receba a informação de que não há mais uma emergência, fazendo com que ele reconstrua o MA para sua navegação, ou até que seja manualmente retirado daquela posição.

4.5.1 Definição da célula objetivo

Em uma situação de emergência, o robô precisa estacionar em algum local do ambiente fechado em que, preferencialmente, não haja passagem de pessoas durante a evasão. Para que isso seja possível é necessário que a célula objetivo seja definida e, assim, ele poderá utilizar alguma técnica de planejamento de trajetória como, por exemplo, A* que o conduzirá até sua meta.

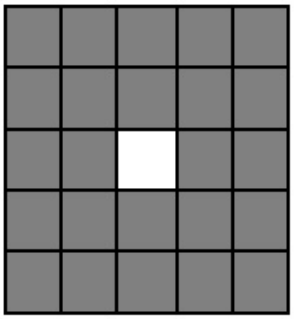
Neste trabalho, considerou-se que a definição da célula objetivo e a definição da trajetória do robô ocorrem em momentos distintos. Isso porque existem critérios da teoria para a escolha do objetivo que contribuem para que o robô evite passar por regiões de alta intensidade de ocupação da multidão, durante a emergência. A definição da trajetória do robô é um processo que sucede à definição da meta, apresentando o caminho que o robô deverá seguir no momento em que uma emergência ocorrer.

Para a definição da célula objetivo deve-se considerar as situações em que não haja impedimento de evacuação por qualquer acesso do ambiente e também as situações em que um ou mais acessos estejam obstruídos. Para o primeiro caso, o robô utiliza o MI, enquanto que para o segundo, ele utilizará o MO mais apropriado. A justificativa para isso encontra-se nas próximas subseções, 4.5.1.1 e 4.5.1.2, respectivamente.

Com base na teoria dos ACs, o conhecimento dos estados das células vizinhanças ao robô é fundamental para as ações de definição de objetivos e também de planejamento de trajetórias. O modelo de vizinhança estendida de Moore (ZUBELDIA, 2014) considera como vizinhas as oito células que estão ao redor da origem, além de todas as demais células nos ní-

veis seguintes. A figura 26 apresenta um exemplo do modelo de vizinhança estendida de Moore, com nível 2.

Figura 26 – Exemplo do modelo de vizinhança estendida de Moore, com nível 2.



Fonte: Autor.

A célula central (cor branca) é a origem, ou seja, a posição em que o robô está localizado. Como ele possui o MI e um conjunto de MOs, todos eles contendo o estado de cada célula, é possível a definição da célula objetivo para a qual ele deverá se mover no momento em que ocorre uma emergência.

O custo para se determinar a célula objetivo é fundamental. O menor custo é aquele cuja soma das células até chegar ao objetivo é menor. O oposto disso se aplica ao maior custo.

Definição 4.29. *(Custo) O custo é a medida calculada até uma célula de RL, levando em consideração os valores de cada célula no intervalo entre a posição do robô e a célula objetivo.*

O movimento do robô, para encontrar o custo, pode ser em qualquer sentido, inclusive na diagonal.

Definição 4.30. *(Regra da contagem) Para encontrar os valores de custos para atingir uma célula objetivo, os valores das células devem ser adicionados um ao outro sempre do maior para o menor e nunca ao contrário.*

O que a definição 4.30 sugere é que, se o robô está em uma célula de valor 1,0 (RR), a próxima célula a ser contada para efeito de custo pode ter qualquer valor, inclusive o valor 1,0.

Entretanto, se o robô estiver em uma célula 0.5, a próxima célula a ser contada pode apenas ter valor 0.5 ou 0 (zero), mas nunca o valor 1,0. Esta regra é importante para garantir que o robô não saia de uma região RA e passe por uma região RR, o que poderia causar acidentes.

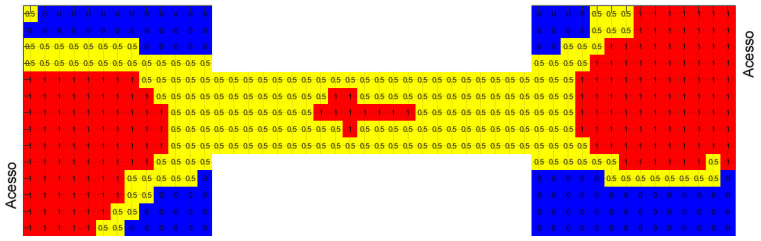
4.5.1.1 Célula objetivo onde não há impedimento de acesso

Nos casos onde os acessos não estão comprometidos pelo motivo da emergência, a multidão tem a opção de escolher qualquer um deles para evacuar do ambiente fechado. De posse do MI, a definição da célula objetivo, que é realizada pelo próprio robô, se baseia na busca da célula com estado zero (status RL) mais próxima a ele. Entretanto, o custo para chegar a esta célula também deve ser considerado, pois poderá haver casos em que chegar à célula mais próxima fará com que ele tenha que atravessar uma região RR desnecessariamente.

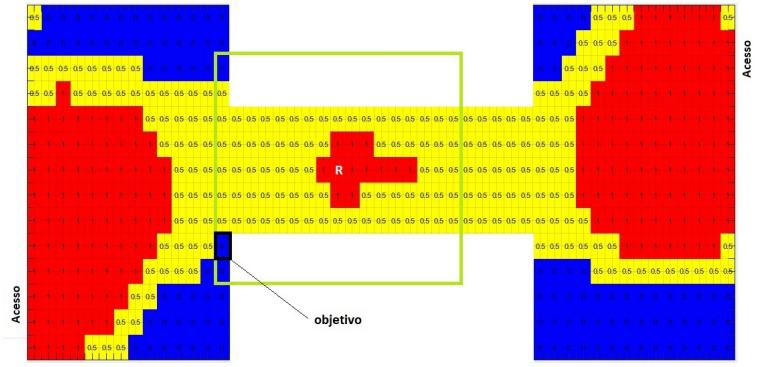
Observando o exemplo da figura 27, é possível observar que o robô, representado pela letra R na cor branca na figura 27b, está localizado em uma região RR. Entretanto, ele possui conhecimento dos valores de cada célula do ambiente e isto contribui para a escolha da célula objetivo.

Figura 27 – Definição da célula objetivo.

(a) Exemplo de MI e os valores dos estados das células.



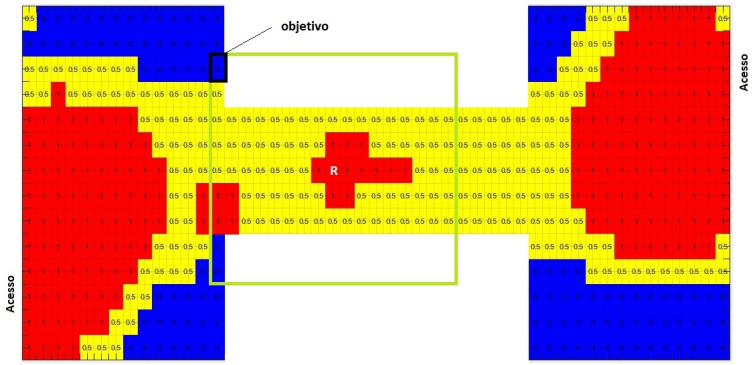
(b) Apresentação da célula objetivo definida pelo robô, sem a influência de regiões RRs próximas à célula de RL mais próxima.



Fonte: Autor.

A célula objetivo (em destaque na cor preta na figura 27b) é aquela que possui o valor zero mais próxima a ele, conforme mencionado acima, e está localizada em uma região próxima a parede. O caminho para o objetivo não possui regiões de risco. O valor do menor custo para atingir este objetivo é de 3,5.

Figura 28 – Apresentação da célula objetivo definida pelo robô, com a influência de regiões RRs próximas à célula de RL mais próxima.



Fonte: Autor.

No caso da figura 28 foi adicionada, intencionalmente, uma região RR próxima à célula objetivo da figura anterior (27b). É possível observar que o robô teria que passar por esta região, caso a célula objetivo fosse aquela. Assim, a célula objetivo escolhida pelo robô deve ser outra, indicada na figura 28, pois, deste modo, a região de risco é evitada, levando-se em consideração a regra da contagem (definição 4.30). O valor do menor custo para a nova célula objetivo é de 4,0, enquanto que para o objetivo anterior, com a região de risco adicionada, é de 6,0.

Há, porém, situações em que um acesso pode estar obstruído para a evacuação da multidão. Casos como estes serão discutidos a seguir.

4.5.1.2 Célula objetivo onde há impedimento de acesso ou há sentido único de evacuação

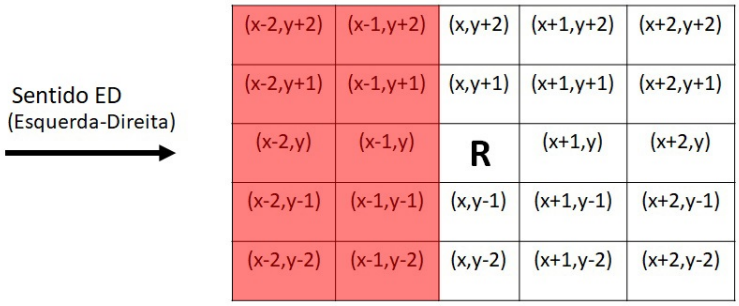
Uma situação de emergência pode ocorrer em qualquer região de um ambiente fechado e por motivos diversos. É possível, por exemplo, o surgimento de um incêndio próximo à um dos acessos, fazendo com que ele fique obstruído para a passagem da multidão.

Em casos como estes, conforme exposto na seção 4.4, o robô necessita considerar que o sentido de movimento da multidão se altera e, deste modo, irá utilizar um dos MOs que possui para definir qual deverá ser a célula objetivo, uma vez que a célula com estado zero mais

próxima e de menor custo pode ser aquela que faria com que o robô se movimentasse contra o sentido da multidão na emergência, potencializando a ocorrência de acidentes. Para que isso não aconteça, o sentido do movimento da multidão é um parâmetro a ser considerado nesta teoria.

Em uma situação de emergência, para a definição da célula objetivo, o robô irá desconsiderar, para efeito de escolha desta célula objetivo, um conjunto de células que poderiam fazer com que ele navegasse contra o sentido de movimento da multidão ou oferecesse grande resistência à saída dela. Para isso, todas as células que forem desqualificadas para serem a célula objetivo terão seus valores atualizados para 1 (em vermelho na figura 29 para o sentido ED), ou seja, passarão a pertencer a uma região RR.

Figura 29 – Células a serem desqualificadas no sentido ED.



Fonte: Autor.

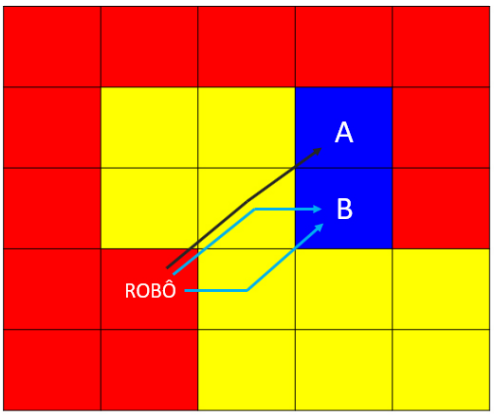
Observa-se que o robô não irá contra o sentido de movimento da multidão, pois todas as células à esquerda dele foram desqualificadas para serem a célula objetivo, passando a possuírem valor 1. Considerando-se as coordenadas (x,y) do robô, nota-se que todas as células que foram atualizadas (e somente elas) possuem a coordenada x menor que a coordenada x do robô.

Após as células do ambiente serem desqualificadas, poderá haver situações em que custos iguais sejam encontrados até duas ou mais candidatas diferentes à célula objetivo. Assim, o robô irá sempre fazer uso da definição 4.31.

Definição 4.31. *(Custos iguais) Sempre que houver custos iguais para alcançar objetivos diferentes, a célula objetivo será aquela cuja quantidade de possíveis caminhos, com mesmo custo, para chegar até ela seja maior.*

A figura 30 apresenta um exemplo onde a definição 4.31 é aplicada. Observa-se que, dada a posição do robô no cenário proposto, as células livres A e B são candidatas a serem células objetivo, pois os custos são os mesmos.

Figura 30 – Escolha da célula objetivo quando os custos são iguais para mais de uma célula livre.



Fonte: Autor.

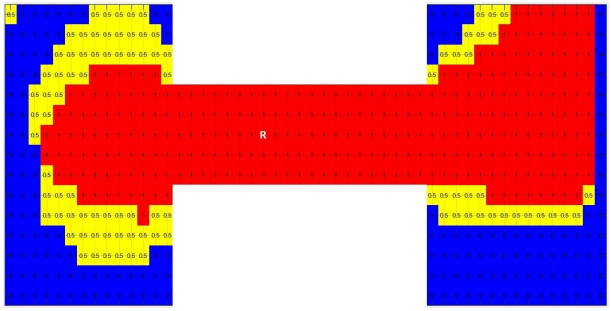
As setas em preto e ciano (figura 30) representam os possíveis caminhos do robô para chegar às células livres A ou B. Deste modo, nota-se que existem diferentes formas do robô chegar em A, porém com custo superior a B e também diferentes formas de chegar a B, porém as de menor custo estão em ciano. Sendo assim, a célula B será escolhida como objetivo.

A figura 31 apresenta um exemplo de construção da primeira etapa do MN, com base na teoria de desqualificação de células que não poderão ser a célula objetivo. Na figura 31a, é possível observar que o sentido de movimento da multidão será da esquerda para a direita do mapa. O robô está posicionado em uma região RR, no corredor.

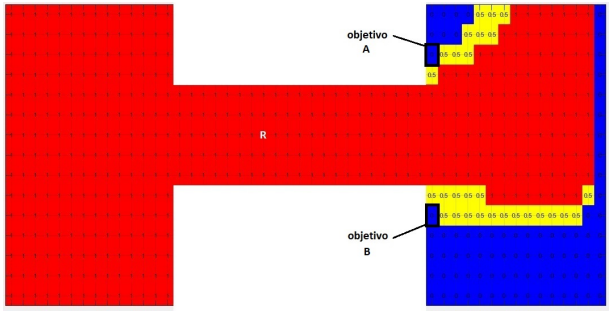
Já na figura 31b, observa-se que todas as células à esquerda do robô foram desqualificadas como célula objetivo para que o robô não se movimento no sentido contrário ao movimento de evasão da multidão. Assim que o robô atualiza as células que são desqualificadas, o procedimento para determinar qual deve ser a célula objetivo é o mesmo apresentado na subseção anterior (4.5.1.1).

Figura 31 – Desqualificação de células que não podem ser a célula objetivo.

(a) MO com sentido ED.



(b) MN contendo as células desqualificadas.



Fonte: Autor.

No caso da figura 31, a célula objetivo estará ao final do corredor, à direita, no sentido em que a multidão caminha. É possível observar que o custo é o mesmo para qualquer uma das duas células de RL, apresentadas como *objetivo A* e *objetivo B* na figura 31b. Neste caso, a célula objetivo escolhida pelo robô deve ser a *objetivo A*, pois o sentido de evasão favorece o posicionamento do robô nesta célula.

Um exemplo de algoritmo escrito para o sentido ED, pode ser representado conforme a seguir.

Algoritmo 3 – Algoritmo que desqualifica algumas células para serem objetivo.

```
1 Entrada: Inteiro x_robot, y_robot, x, y, n, m; Matriz viz
2 para y=1:n faça
3     para x=1:m faça
4         se (x < x_robot) então
5             viz(x,y) = 1;
6         fim
7     fim
8 fim
9 retorna viz(x,y)
```

Os casos de sentidos que não foram abordados nesta subseção como, por exemplo, a evasão em um sentido diagonal ao sentido de orientação do robô, serão tratados em trabalhos futuros, uma vez que para a realização dos testes foram consideradas apenas situações em que não há acessos obstruídos nos ambientes. O próximo capítulo apresentará os resultados dos testes realizados para a análise da teoria desenvolvida.

5 RESULTADOS E DISCUSSÕES

Para a realização dos testes que validam a teoria descrita no capítulo anterior, foram criados nove ambientes fechados, para efeito de simulação, de dimensões e quantidade de acessos (definição 4.1) distintos, assemelhando-se com ambientes reais. As situações mais críticas, baseadas na definição 4.18 onde a quantidade de agentes a ser considerada é por hipótese, foram consideradas nas simulações, ou seja, aquelas onde o ambiente está repleto de pessoas e a ação de evacuação delas (definição 4.19) é instantânea ao recebimento da informação de emergência. Deste modo, a revisão da literatura pouco contribui para efeito de comparações, uma vez que esta abordagem se difere das demais, como na situação apresentada no trabalho de Boukas et al. (2015). A tabela 4 apresenta uma descrição de cada um destes ambientes fechados, bem como alguns aspectos de configuração necessários para realizar a simulação de multidão.

Tabela 4 – Exemplos de ambientes fechados e seus respectivos mapas.

Ambiente	**Acessos**	**Corredores**	**Área total (m^2)**	**Área livre**	**Pessoas/m^2**	**Agentes**
Centro de exposições	4	0	2000	1876	1	1876
Salão de festas	1	1	600	300	1	300
Universidade	2	1	550	550	1	550
Biblioteca	2	3	924	577	1	577
Apartamento	1	1	267	230	1 para cada 4 m^2	57
Centro comercial - shopping center	4	6	4508	1850	1	1850
Supermercado	3	10	800	664	1	664
Posto de saúde	3	4	270	158	1	158
Hotel	2	2	540	454	1	454

Fonte: Autor.

Para cada um dos ambientes, serão apresentados, nas seções a seguir, quatro mapas, sendo o mapa desenhado do ambiente, o Mapa do Ambiente - MA - (definição 4.4), o mapa de intensidade de um simulador e o Mapa de Intensidade - MI - (definição 4.15) gerado pelo robô.

O primeiro mapa foi construído por meio da ferramenta de simulação de multidão chamada Oasys MassMotion®, versão 8.0.9.0, e, então, foi aplicada a matriz celular, pois o método escolhido para decomposição é o mapeamento por grade de ocupação (decomposição aproxi-

mada), descrito na subseção 4.1.2. Todas as células possuem o mesmo formato e tamanho, conforme definição 4.6.

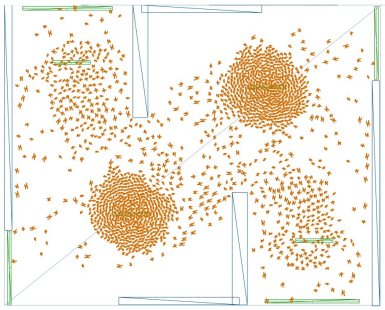
O segundo mapa é a mesma representação do primeiro, porém foram definidos os valores de cada célula, bem como as cores que representam cada valor, indicando as regiões livres (RL), as regiões de atenção (RAs) e as regiões de risco (RRs), regiões estas compostas por células cujas definições são, respectivamente, 4.24, 4.25 e 4.26.

Para o terceiro mapa, utilizou-se a mesma ferramenta e mapa construído na etapa 1 para a inserção dos obstáculos, marcos dos agentes para guiá-los da origem ao alvo (origem aos acessos) e parâmetros de configuração. Em seguida, a simulação foi executada e, então, gerado o mapa de intensidade do simulador.

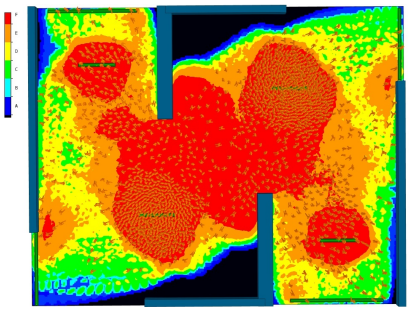
A figura 32 apresenta um exemplo de simulação, de mapa de intensidade e de dados, obtidos por meio do simulador.

Figura 32 – Exemplo de simulação, de mapa de intensidade e de dados obtidos por meio do simulador.

(a) Simulação



(b) Mapa de intensidade gerado no simulador



(c) Amostra dos dados obtidos por meio do simulador

	A	B	C	D	E
1	#Frame	AgentID	X	Y	Z
2	0	1001	19	0	27
3	0	1002	36	0	10
4	0	1003	34	0	10
5	0	1004	15	0	27
6	0	1005	36	0	10
7	0	1006	18	0	27
8	0	1007	35	0	11
9	0	1008	15	0	27
10	0	1009	17	0	27
11	0	1010	18	0	27
12	0	1011	18	0	27
13	0	1012	33	0	10
14	0	1013	14	0	27
15	0	1014	35	0	10
16	0	1015	16	0	27
17	0	1016	35	0	10
18	0	1017	17	0	27
19	0	1018	15	0	27
20	0	1019	16	0	27
21	0	1020	17	0	27
22	0	1021	14	0	27

Fonte: Autor.

Na figura 32a, observa-se uma simulação de pessoas evacuando um ambiente cujas dimensões são 40 x 50m. Para este ambiente, a fim de se obter os resultados para o caso mais crítico de emergência que é a máxima concentração de pessoas, considerou-se utilizar a forma QH, onde cada metro quadrado (uma célula) é capaz de ser ocupado por uma única pessoa. Assim, descontando ainda as áreas de obstáculos (paredes), a quantidade de agentes usados na simulação foi de 1876.

Na figura 32b, é possível obter um mapa de intensidade do simulador, que considera as regiões quentes (vermelho) como as mais intensas (grande concentração de agentes durante a evasão), as regiões frias (azul) como as menos intensas (pouca concentração de agentes durante

a evasão) e as regiões em preto representando aquelas onde não há passagem de agentes. As outras regiões são intermediárias entre o quente e o frio.

A figura 32c apresenta uma amostra dos dados obtidos por meio do simulador, que são organizados, no modelo mais simples de configuração, da seguinte maneira:

a) Frame: nomenclatura que distingue uma posição das outras no mapa. Para cada agente (pessoa), o software marca seus movimentos e armazena-os em uma ordem numérica crescente, iniciando do zero;
b) AgentID: é a identificação numérica de cada agente, iniciando pelo número 1001;
c) X: é a coluna onde são registradas as posições x dos agentes em cada frame;
d) Y: é a coluna onde são armazenadas as posições y (tridimensionalidade) dos agentes em cada frame, ou seja, é o componente 3D de altura. Neste caso, todos os valores são zero, pois assume-se que os agentes se deslocam em um cenário bidimensional;
e) Z: é a coluna onde são registradas as posições z dos agentes em cada frame.

Para o robô, apenas os valores contidos nas colunas X e Z serão importantes. Os dados de Z irão compor a coordenada y neste trabalho, por uma questão conceitual habitual.

Para a construção do mapa de intensidade do simulador de multidão, foram utilizadas as seguintes configurações, em cada ambiente:

a) Reação à informação de emergência: por meio da definição 4.19, ao iniciar a simulação, todos os agentes, no mesmo instante, saem de suas posições e caminham para o acesso mais próximo ou, em alguns ambientes, como no caso do shopping center, para os acessos que possuem peso (prioridade) maior de evacuação por se tratarem de acessos principais.
b) Custo: por meio da definição 4.29, a evacuação considera o menor custo para a multidão, em alguns ambientes. Neste caso, os agentes procuram os acessos mais próximos de si (referência à definição 4.20), tentando seguir a Lei do Menor Esforço (DADOVÁ, 2012). Segundo Still (2000 apud DADOVÁ, 2012), o menor esforço (ou menor consumo de energia) é o caminho ou rota mais fácil de A para B que os indivíduos da multidão tomam à medida que progridem através de um ambiente. Além desta opção, é possível configurar a evacuação atribuindo pesos

para os acessos, fazendo com que os agentes priorizem aqueles de pesos maiores (referência à definição 4.20). Esta segunda alternativa de configuração também foi utilizada, neste trabalho, em testes de alguns ambientes que possuem mais que um acesso.

O quarto mapa (MI) foi gerado com o auxílio da ferramenta Mathworks® Matlab® R2018a, 64 bits, utilizando-se os dados coletados do simulador. Neste caso, um arquivo com extensão .csv foi criado contendo as coordenadas de cada um dos agentes, a cada 1 segundo de simulação, durante todo o período que cada agente leva para evacuar o ambiente fechado.

É possível observar, em cada ambiente, as semelhanças de intensidades entre o mapa que foi gerado no simulador de multidão e aquele que foi gerado a partir dos dados dele, no Matlab, apesar do fato do simulador considerar sete níveis de intensidade representados pelas cores que variam da preta (menor intensidade de ocupação) até a vermelha (maior intensidade de ocupação) e este trabalho considerar, para o MI, apenas três níveis de intensidade, cujas cores são: azul (menor intensidade de ocupação), amarela (intensidade média de ocupação) e vermelha (maior intensidade de ocupação). Isto mostra que o sistema é capaz de classificar adequadamente as células do MI com base nos dados coletados por meio do simulador.

Para cada um dos nove ambientes de testes são apresentados resultados, com base na teoria do capítulo 4, de simulações que envolvem a navegação de três robôs autônomos terrestres diferentes, individualmente, cada um deles realizando a navegação, definindo a célula objetivo e usando a técnica de planejamento de trajetória A* para chegar a esta célula. Para que as simulações e os resultados fiquem mais próximos da realidade, as características escolhidas são de robôs reais. Estes robôs foram escolhidos porque já são usados em ambientes internos com multidão. Tais robôs são apresentados na figura 33.

O robô da figura 33a é o Knightscope, modelo K5. Ele é utilizado para promover segurança, fazendo patrulhamento em ambientes abertos e fechados como, por exemplo, shoppings, armazéns, hospitais, aeroportos, casinos, estádios, estruturas de estacionamento, entre outros (KNIGHTSCOPE, 2019). Possui altura de 62,5 cm, largura de 33,5 cm, comprimento de 36 cm e pesa cerca de 180 kg. A velocidade deste robô é semelhante à velocidade média que um ser humano consegue desenvolver, caminhando, conforme estudos de Weidmann (1993 apud LINO; MAIONE; MAIONE, 2009).

Figura 33 – Robôs cujas características são consideradas para os resultados.

(a) Knightscope k5.

(b) Signage HOSPI.

(c) Relay

Fonte: (a) (KNIGHTSCOPE, 2019), (b) (PANASONIC, 2018), (c) (IEEE.ORG, 2019).

O robô da figura 33b é o Panasonic Signage HOSPI. É um robô que faz entregas em nome de pessoas. Ele é equipado com 3 displays em seu corpo para comunicar informações sobre itens como meio de publicidade e navega por áreas designadas (PANASONIC, 2018). Possui altura de 138,6 cm, largura de 72,5 cm, comprimento de 63 cm e pesa cerca de 211 kg.

O robô da figura 33c é o Savioke Relay. É um robô de entrega de objetos que funciona em hotéis, hospitais e outros espaços públicos. Ele pode navegar por multidões e operar elevadores para levar o serviço de quarto (IEEE.ORG, 2019). Possui altura de 92 cm, largura de 51 cm, comprimento não especificado e pesa cerca de 40 kg.

A tabela 5 apresenta a característica de velocidade de cada um dos robôs (KNIGHTSCOPE, 2019; PANASONIC, 2018; IEEE.ORG, 2019) escolhidos para este estudo.

Tabela 5 – Velocidade de cada robô.

Robô	**Velocidade em m/s**
Knightscope K5	1,34
Panasonic Signage HOSPI	1,0
Savioke Relay	0,7

Fonte: Autor.

A característica de velocidade para os robôs reais de serviço foram apresentadas para analisar o tempo de deslocamento de cada um deles até a célula objetivo, com base na trajetória escolhida. Isso é importante, pois alcançar o objetivo significa que o robô não estará atrapalhando a evacuação da multidão na emergência e, quanto mais rápido fizer isso, mais seguro serão as ações de saída das pessoas.

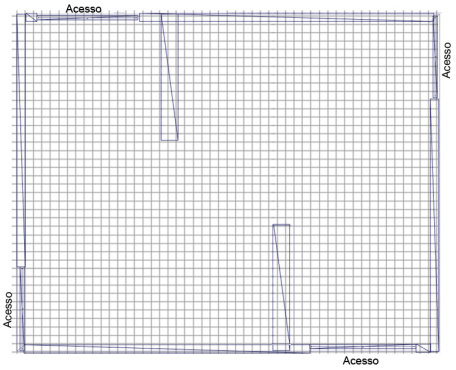
A seguir serão descritos os ambientes fechados escolhidos para os testes e também serão apresentados os mapas de cada um deles. Os casos indicados como *aceitável* para as definições da célula objetivo e da trajetória do robô são aqueles em que ele não navega contra o sentido de movimento da multidão na emergência e também não avança em uma região RR sem ter a necessidade disso como, por exemplo, nesta última situação, porque está em uma célula 0.5 próxima à uma parede (se em um corredor que seja amplo). Os casos indicados como *não aceitável*, ao contrário, consideram movimento do robô contra o sentido de movimento da multidão, na emergência, ou o acesso a uma região RR desnecessariamente.

5.1 CENTRO DE EXPOSIÇÕES

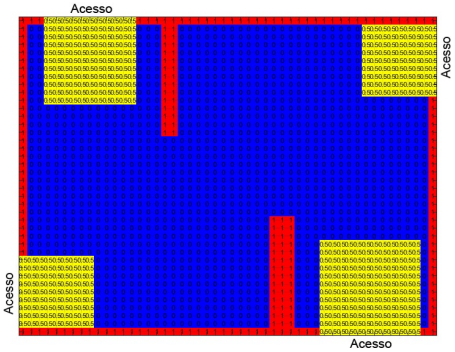
Este é um ambiente que simula um centro de exposições, onde os obstáculos são, basicamente, as paredes e stands de eventos. Possui 40 x 50 m e uma área livre de 1876 m^2 para movimentação do robô e das pessoas. A figura 34 apresenta os mapas deste ambiente.

Figura 34 – Exemplo de mapa de um centro de exposições.

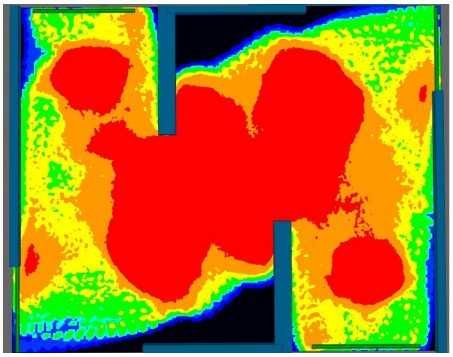
(a) Mapa decomposto em células.



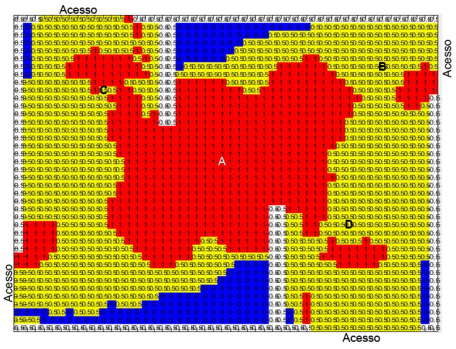
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.



Fonte: Autor.

Foi considerado que a multidão, de 1876 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local e, no momento de uma emergência, ela tem a opção de abandonar o ambiente por qualquer um dos quatro acessos, porém irá procurar o mais próximo, com base nos parâmetros de configuração do simulador. Caso o acesso mais próximo esteja congestionado, os agentes podem procurar evacuar o ambiente fechado por outro acesso.

A simulação da figura 34c gerou um arquivo com 149272 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 42 segundos.

O MI deste ambiente, representado na figura 34d, mostra quatro letras (A, B, C e D) que representam posições hipotéticas distintas onde o robô pode estar no momento em que recebe a informação de emergência. Para cada situação e para efeito de testes, foram definidas as células objetivo com base na teoria descrita na seção 4.5.1 e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

5.1.1 Definição da célula objetivo no centro de exposições

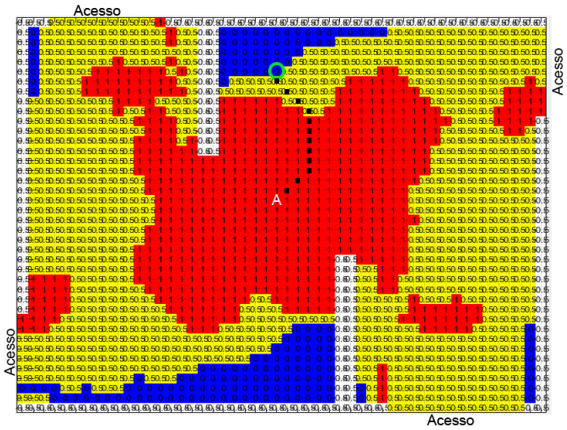
Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 34d. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

Primeiramente serão apresentados os casos em que nenhum acesso está obstruído. Sendo assim, o robô irá fazer uso do MI (definição 4.15) para encontrar a célula objetivo. A figura 35 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva de cada posição (A, B, C e D) em que ele se encontra no ambiente, no momento da emergência.

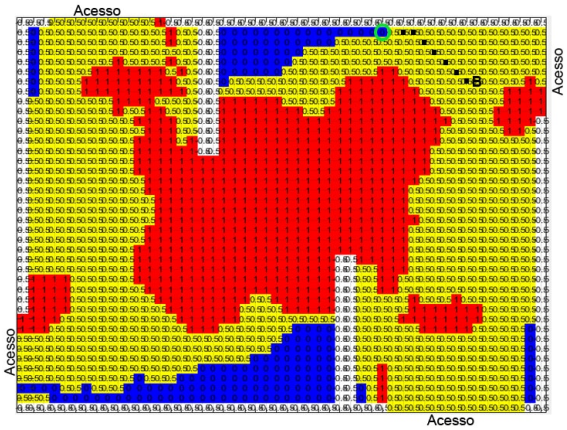
Os símbolos quadrados na cor preta representam o menor custo (um deles) para se alcançar o objetivo. Vale lembrar que, conforme a definição 4.28, assim que a célula objetivo é definida, o MN é gerado, contendo esta indicação.

Figura 35 – Células objetivo em cada posição do robô no centro de exposições.

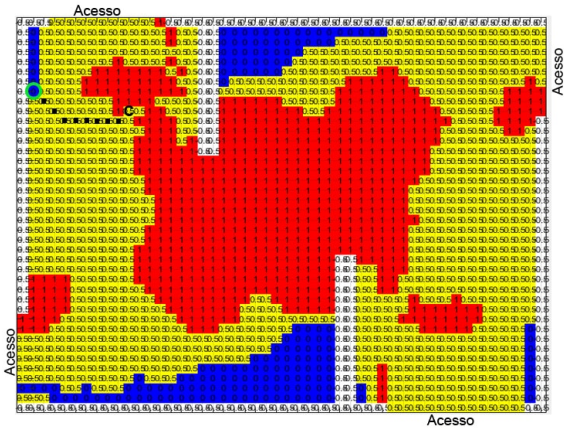
(a) Célula objetivo do robô posicionado em A.



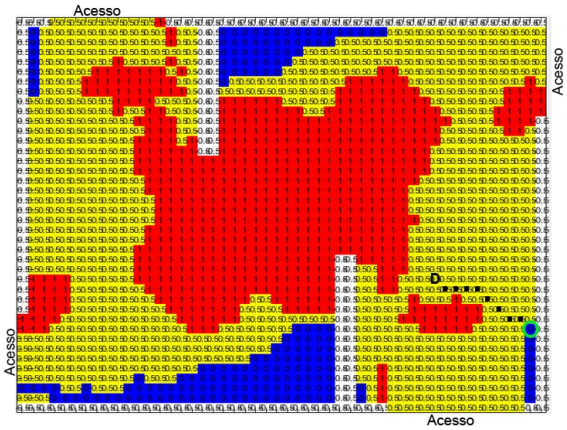
(b) Célula objetivo do robô posicionado em B.



(c) Célula objetivo do robô posicionado em C.



(d) Célula objetivo do robô posicionado em D.



Fonte: Autor.

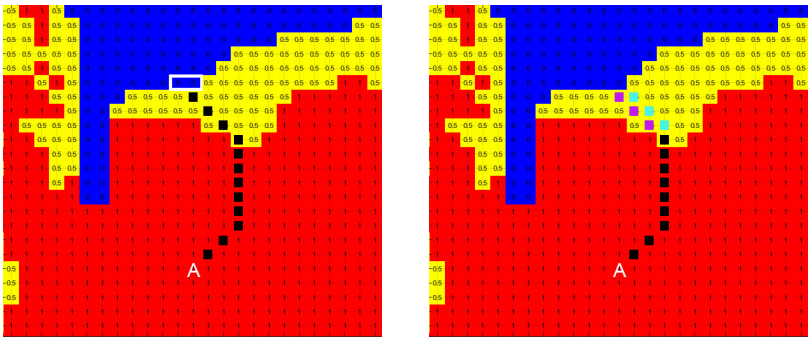
Para o caso em que o robô se encontra na posição A, que fica ao centro do ambiente, conforme figura 35a, é possível verificar que o valor do menor custo é 10,0. A célula com um círculo verde representa o objetivo para o caso em que o robô encontra-se em A. A região escolhida para a célula objetivo é como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência. Teoricamente, um custo linear seria mais propício, para que o robô mantenha um único sentido de movimento (sempre reto), porém o valor do custo, neste caso, seria 11, valor este que é superior ao encontrado anteriormente. A teoria do menor custo se faz necessária para que o robô permaneça o menor tempo possível em uma região RR (definição 4.26).

Um detalhe importante precisa ser abordado nesta situação: quando o robô adiciona o último valor (0,5) ao custo, antes de definir a célula objetivo, ele tem duas opções de escolha desta célula (indicadas no interior do retângulo branco), conforme mostrado na figura 36a.

Figura 36 – Situação onde duas células podem ser definidas como a célula objetivo.

(a) Possibilidades de escolha da célula objetivo.

(b) Possibilidades de menor custo para uma candidata a célula objetivo.



Fonte: Autor.

Neste caso, define-se um critério para esta escolha:

Definição 5.1. *(Escolha da célula objetivo) Se houver duas ou mais células capazes de serem a célula objetivo, o robô irá escolher aquela que possui maior número de possibilidades de ser escolhida pelo menor custo.*

Na figura 36a, observa-se que há duas células no interior do retângulo branco, sendo uma à esquerda e outra à direita. A figura 36b mostra que o menor custo, ou seja, o valor 10,0, é o mesmo para a célula à direita do retângulo branco (onde houve a determinação da célula objetivo na figura) quando o caminho é seguido pela cor lilás ou pela cor ciano. Além disso, outras duas possíveis combinações seriam: 9 pretos - 1 lilás - 2 cianos; 9 pretos - 2 lilases - 1 ciano. Neste caso, são quatro possibilidades de caminhos para a célula à direita, enquanto que há somente uma possibilidade de caminho para a célula à esquerda, quando se mantém o custo (menor custo).

Para o caso em que o robô se encontra na posição B, conforme figura 35b, é possível verificar que o valor do menor custo é 2,5. A região que possui a célula objetivo também é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Já para o caso em que o robô se encontra na posição C, conforme figura 35c, é possível verificar que o valor do menor custo é 4,0. A regra da contagem foi aplicada, por meio da definição 4.30, ou seja, observa-se que o robô poderia considerar a primeira célula a sua esquerda

para contagem, porém o valor dela é maior do que o valor da célula do robô. Outra definição importante aplicada foi a definição 5.1, referente à escolha da célula objetivo, uma vez que duas delas são candidatas. A região que possui a célula objetivo também é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Para o caso em que o robô se encontra na posição D, conforme figura 35d, é possível verificar que o valor do menor custo é 4,0. A regra da contagem foi aplicada, por meio da definição 4.30, durante a contagem das células. A definição 5.1, referente à escolha da célula objetivo, também teve que ser aplicada. A região que possui a célula objetivo também é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Neste tipo de ambiente fechado são previstos, pelo menos, quatro MOs, considerando que o sentido de movimento da multidão será alterado quando qualquer acesso for obstruído pelo motivo da emergência.

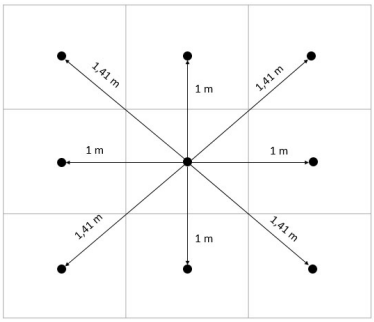
5.1.2 Definição da trajetória do robô no centro de exposições

Nesta etapa, será apresentado o processo de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A* descrita no capítulo 2, seção 2.3. O processo demonstrado com detalhes aqui será utilizado em todos os demais ambientes de testes.

O custo, quando necessário encontrar a célula objetivo, de acordo com a definição 4.29, leva em consideração os valores de cada célula. Entretanto, para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. Deste modo, uma distância em linha reta de centro-a-centro é diferente de uma distância na diagonal de centro-a-centro, conforme apresentado na figura 37.

Sempre que o robô necessitar se mover para uma célula vizinha à frente, traz ou lados, ele irá considerar um percurso de 1 metro de distância, com base na definição 4.6. Entretanto, se o robô necessitar se mover para uma célula vizinha na diagonal, ele irá considerar um percurso de 1,41 metro de distância.

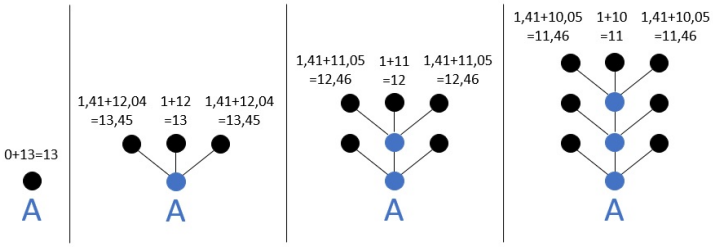
Figura 37 – Medidas entre centros de duas células vizinhas.



Fonte: Autor.

Com base na figura 35a, que mostra a posição (A) do robô e também a célula objetivo definida por ele, é possível usar o algoritmo A* para encontrar a trajetória a ser seguida. A figura 38 apresenta as primeiras etapas do uso deste algoritmo no caso em que o robô está na posição A (figura 35a) do MI.

Figura 38 – Esquema de trajetória criada pelo algoritmo A* para a situação em que o robô está na posição A, conforme figura 35a.

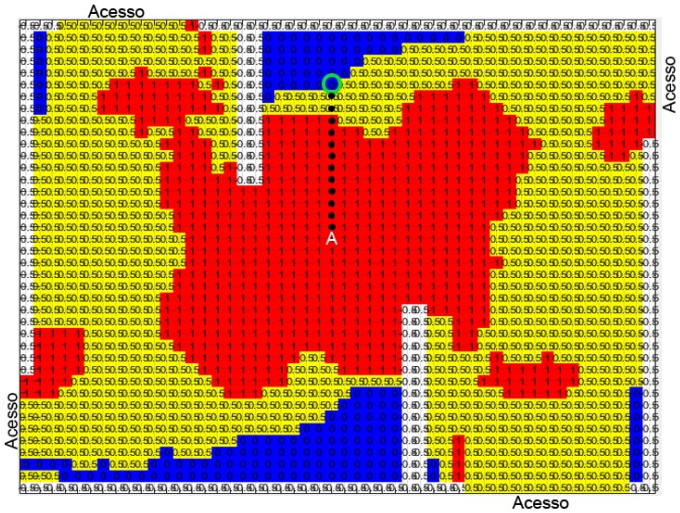


Fonte: Autor.

É possível perceber que, como a posição do robô está alinhada com a posição da célula objetivo, as células que o robô irá escolher para prosseguir com sua navegação será sempre, para este caso, a célula à sua frente. A trajetória do robô até a célula objetivo é apresentada pelos pontos na cor preta, na figura 39.

Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência em que ele esteja no centro de exposições, na célula A, é *aceitável*, embora ele tenha que passar por uma grande região RR, já que ele se encontra nesta área. Entretanto, nesta situação o robô poderá atrapalhar o sentido de movimento da multidão, já que não existe informação deste em cada célula do ambiente.

Figura 39 – Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo, no centro de exposições.

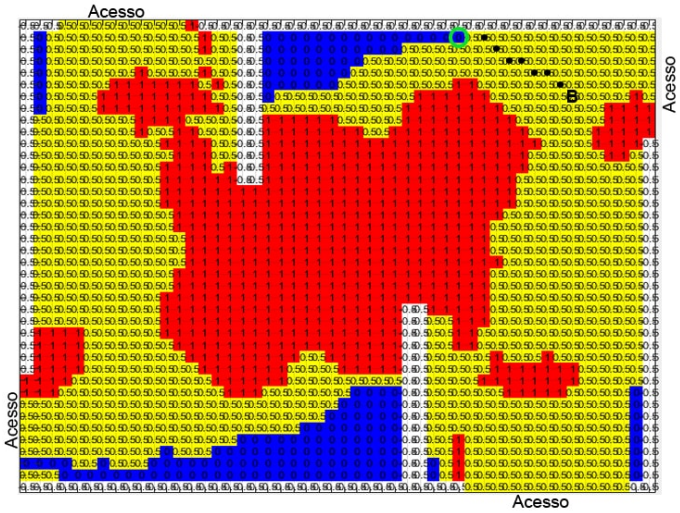


Fonte: Autor.

Definida a trajetória do robô até a célula objetivo quando ele estiver na posição A (figura 35a), o mesmo processo é realizado para sua localização nas posições B, C e D.

É possível observar (figura 40) que o robô irá realizar uma trajetória predominantemente contínua na diagonal, ou seja, sempre irá escolher a célula vizinha na diagonal no sentido do objetivo.

Figura 40 – Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



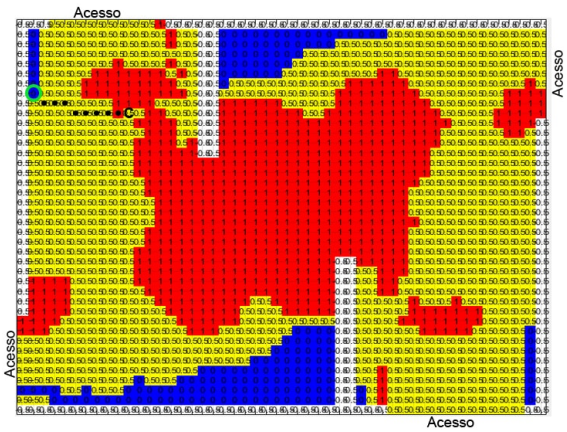
Fonte: Autor.

Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência em que ele esteja no centro de exposições, na célula B, é *aceitável*. Entretanto, embora o robô não realize uma navegação direta contra o sentido de movimento da multidão, a posição em que se encontra (B) fica próxima a um acesso, o que significa que a multidão virá em direção a ele. Como o robô respeita o espaço social das pessoas, é bem provável que, nesta condição, ele fique parado em grande parte do tempo em que houver a evacuação, aguardando as pessoas saírem para, assim, desenvolver sua trajetória de navegação. Entretanto, este cenário deve ser observado em trabalhos futuros quando a teoria for aplicada em um robô real.

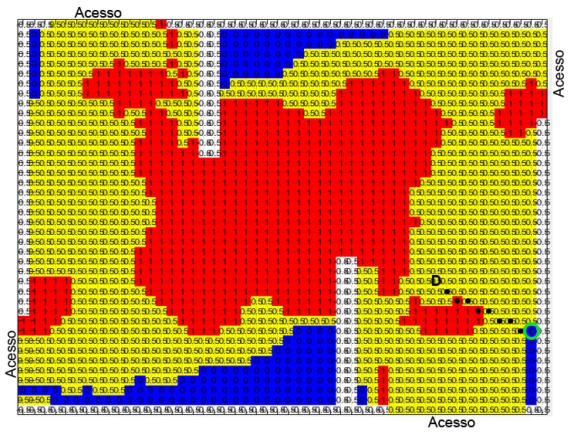
Para os mapas a seguir, que representam as posições C e D do robô no mapa, serão apresentadas as trajetórias geradas pelo algoritmo A* já definidas, bem como as discussões sobre elas. A figura 41 apresenta as trajetórias encontradas pelo robô para os casos em que ele esteja nas posições C (figura 35c) e D (figura 35d).

Figura 41 – Trajetória seguida pelo robô nos outros casos do centro de exposições.

(a) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



(b) Trajetória seguida pelo robô desde a sua posição (D) até a célula objetivo.



Fonte: Autor.

As posições C e D são semelhantes, embora estejam em situações invertidas no mapa. Observa-se na figura 41a que a trajetória do robô é quase perpendicular ao sentido de movimento da multidão. Entretanto, ele se movimenta em uma região de intensidade moderada de passagem da multidão e, portanto, esta trajetória é *aceitável*. Na figura 41b, o robô acompanharia, em certa medida, o sentido da multidão na evacuação, tornando a trajetória *aceitável*. A exemplo do caso anterior (posição B do robô), possivelmente o robô permaneça algum tempo parado para, então, seguir a trajetória até a célula objetivo.

5.1.3 Simulação com os robôs no centro de exposições

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo, a saber, o Knightscope K5, o Panasonic Signage HOSPI e o Savioke Relay.

Com base na figura 39, anteriormente apresentada, é possível perceber que o robô precisou percorrer treze células para chegar ao objetivo, a partir de sua posição A no MI. Para que isso fosse possível, ele navegou em uma linha reta, passando por dez células de risco (RR), duas células de atenção (RA) e, finalmente, parando na célula livre (RL).

Sendo assim, a distância total percorrida pelo robô é de treze metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

Neste sentido, o robô Knightscope K5, para a situação em que ele esteja na posição A do mapa e necessita navegar treze metros até a célula objetivo, levaria aproximadamente 9,7 segundos para o êxito de sua operação, conforme velocidade definida na tabela 5.

Neste mesmo cenário, o Panasonic Signage HOSPI levaria 13 segundos, enquanto que o Savioke Relay levaria cerca de 18,6 segundos para o êxito da operação.

Para o caso da figura 40, em que o robô esteja na posição B do mapa e realiza um trajeto na diagonal até a célula objetivo, o robô percorre um trajeto de 8,46 metros, passando por cinco células de RA do mapa até chegar na célula de RL.

O robô Knightscope K5, para esta situação, levaria aproximadamente 6,31 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 8,46 segundos e o Savioke Relay levaria cerca de 12 segundos.

Já para o caso da figura 41a, em que o robô esteja na posição C do mapa e realiza um trajeto retilíneo e também diagonal até a célula objetivo, o robô percorre um trajeto de 9,82 metros, passando por uma célula de RR, sete células de RA do mapa até chegar na célula de RL.

O robô Knightscope K5, para esta situação, levaria aproximadamente 7,32 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 9,82 segundos e o Savioke Relay levaria cerca de 14 segundos.

Para o caso da figura 41b, em que o robô esteja na posição D do mapa e realiza um trajeto retilíneo e também diagonal até a célula objetivo, o robô percorre um trajeto de 11,05 metros, passando por duas células de RR, seis células de RA do mapa até chegar na célula de RL.

O robô Knightscope K5, para esta situação, levaria aproximadamente 8,25 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 11,05 segundos e o Savioke Relay levaria cerca de 15,78 segundos.

Evidentemente, em todos os casos analisados até este momento e também em todos aqueles que ainda serão analisados, não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

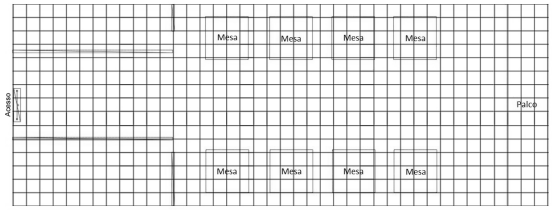
A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para outro ambiente, um salão de festas.

5.2 SALÃO DE FESTAS

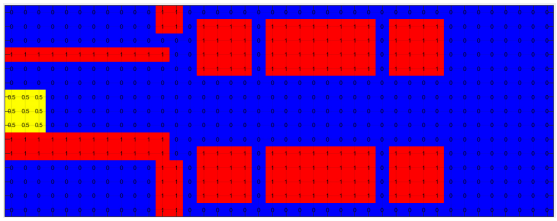
Este é um ambiente que simula um salão de festas, onde os obstáculos são, basicamente, as mesas e o único acesso fica ao final de um corredor. A figura 42 apresenta os mapas deste ambiente.

Figura 42 – Exemplo de mapa de um salão de festas.

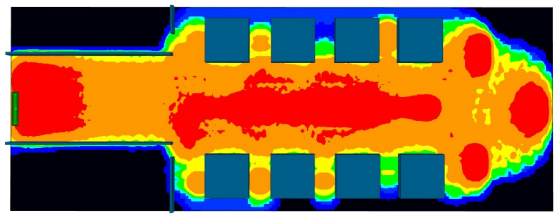
(a) Mapa decomposto em células.



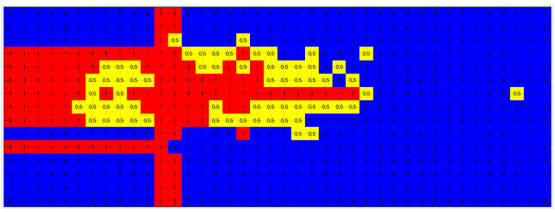
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.



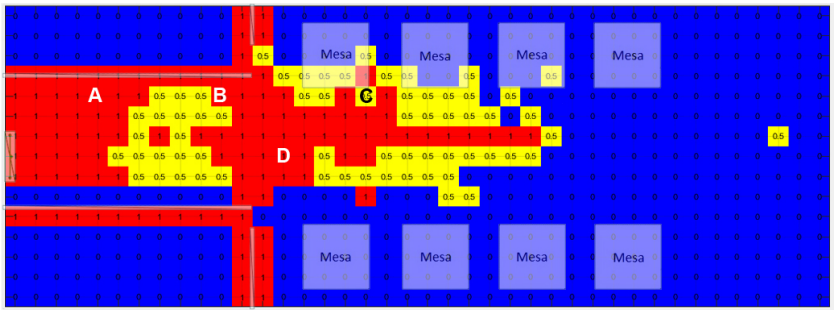
Fonte: Autor.

Possui 15 x 40 m e uma área livre de 300 m^2 para movimentação do robô e das pessoas. Foi considerado que a multidão, de 300 agentes (conforme apresentado na tabela 4), fica distribuída em áreas formadas por um palco e, também, ao redor das oito mesas. A simulação da figura 42c gerou um arquivo com 13007 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 15 segundos.

É possível observar que este ambiente possui um sentido único de evacuação, no corredor, pelo fato de haver apenas um acesso para este fim. Quando o robô estiver posicionado neste corredor, ele irá utilizar o MO para buscar seu posicionamento.

O MI deste ambiente, que também pode ser o MO, representado na figura 43, mostra quatro letras (A, B, C e D) que representam posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência. Para melhor visualização do cenário, foi realizada a sobreposição da estrutura que representa o mapa, no MI.

Figura 43 – MI ou MO do salão de festas com sobreposição da representação do mapa.



Fonte: Autor.

Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

As etapas a seguir mostrarão as definições das células objetivo para cada posição do robô, bem como a trajetória que o robô seguirá utilizando o algoritmo A*. Na sequência, serão apresentados os resultados de uma simulação com características de robôs reais relacionados às definições anteriores.

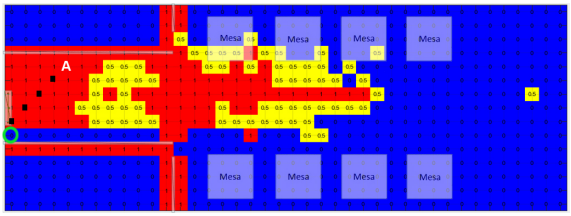
5.2.1 Definição da célula objetivo no salão de festas

Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 43. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

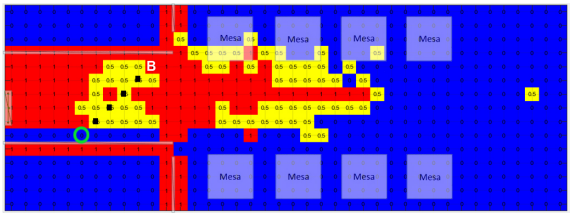
A figura 44 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A e B, no momento da emergência, considerando o custo e a regra da contagem. Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

Figura 44 – Células objetivo para as posições A e B do robô, no salão de festas.

(a) Célula objetivo do robô posicionado em A.



(b) Célula objetivo do robô posicionado em B.



Fonte: Autor.

Para o caso em que o robô se encontra na posição A, que fica no corredor, em uma região RR perto de uma parede, conforme figura 44a, é possível verificar que o valor do menor custo é 4,0. A célula com um círculo verde representa o objetivo para o caso em que o robô se encontra em A. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência. Entretanto, o robô quando posicionado em A está junto a uma parede e, em casos como estes, a melhor opção seria o robô permanecer parado, pois as chances de ocorrência de acidentes seriam menores. Estas decisões não fazem parte do escopo deste trabalho e, por isso, serão abordadas em trabalhos futuros.

Para o caso em que o robô se encontra na posição B, semelhantemente ao caso anterior, que fica no corredor, em uma região RR perto de uma parede, conforme figura 44b, é possível verificar que o valor do menor custo é 2,0. A célula com um círculo verde representa o obje-

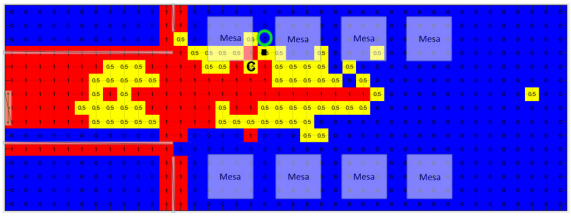
tivo para o caso em que o robô se encontra em B. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, porém o robô quando posicionado em B está junto a uma parede e, em casos como estes, a exemplo do que ocorre quando posicionado em A, a melhor opção seria o robô permanecer parado, pois as chances de ocorrência de acidentes seriam menores.

A figura 45 apresenta as células objetivo definidas pelo robô sob a perspectiva das posições C e D, no momento da emergência. O MI é utilizado para encontrar as células objetivo.

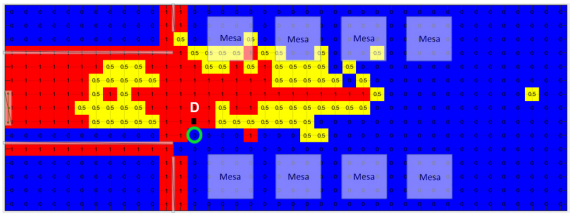
Para o caso em que o robô se encontra na posição C, que fica próxima a uma das mesas do ambiente, em uma região RA, conforme figura 45a, é possível verificar que o valor do menor custo é 0,5. A célula com um círculo verde representa o objetivo para o caso em que o robô se encontra em C. Esta região é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Figura 45 – Células objetivo para as posições C e D do robô, no salão de festas.

(a) Célula objetivo do robô posicionado em C.



(b) Célula objetivo do robô posicionado em D.



Fonte: Autor.

Já para o caso em que o robô se encontra na posição D, em uma região RR próxima ao corredor, conforme figura 45b, é possível verificar que o valor do menor custo é 1,0. Esta região da célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

5.2.2 Definição da trajetória do robô no salão de festas

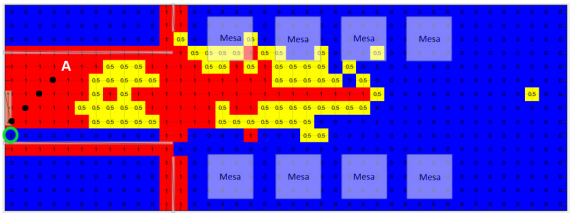
Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A* descrita no capítulo 2, seção 2.3.

Para a definição do melhor caminho o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. A figura 46 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*.

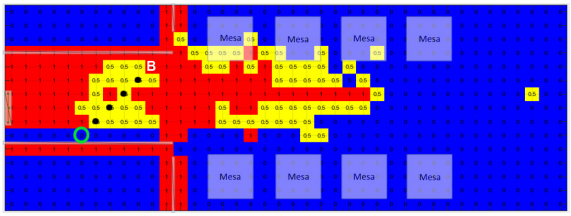
Na figura 46a, é possível perceber que o robô cria uma trajetória na diagonal, representada por pontos na cor preta, a favor do sentido de deslocamento da multidão até a célula objetivo. Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência em que ele esteja no salão de festas, na posição A, é *aceitável*, embora ele tenha que passar por uma grande região RR, já que ele se encontra nesta área. O sentido de movimento da multidão não é comprometido, pois o robô não realizará seu movimento contra ele. Como o robô respeita o espaço social das pessoas, é bem provável que, nesta condição, ele fique parado em grande parte do tempo em que houver a evacuação, aguardando as pessoas saírem para, assim, desenvolver sua trajetória de navegação. Entretanto, este cenário deve ser observado em trabalhos futuros quando a teoria for aplicada em um robô real.

Figura 46 – Trajetória seguida pelo robô utilizando a técnica A*, no salão de festas.

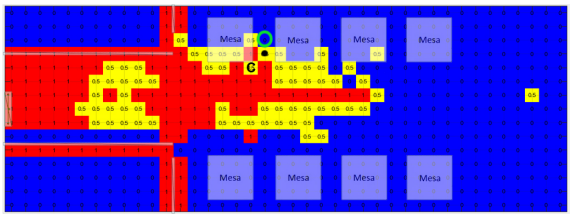
(a) Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo.



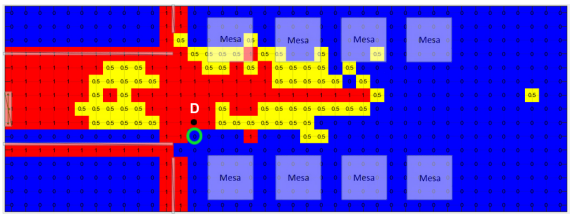
(b) Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



(c) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



(d) Trajetória seguida pelo robô desde a sua posição (D) até a célula objetivo.



Fonte: Autor.

Já na figura 46b, a situação é semelhante à situação anterior. O robô cria uma trajetória na diagonal a favor do sentido de movimento das pessoas, até a célula objetivo. Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência em que ele esteja no salão de festas, na posição B, é *aceitável*, não tendo que passar por regiões RRs, embora se localize em uma delas no momento da emergência. O sentido de movimento da multidão não é comprometido, pois o robô não realizará seu movimento contra ele. A exemplo do que acontece com a situação na posição A, é bem provável que o robô fique parado em grande parte do tempo em que houver a evacuação, aguardando as pessoas saírem para, assim, desenvolver

sua trajetória de navegação. Entretanto, este cenário deve ser observado em trabalhos futuros quando a teoria for aplicada em um robô real.

Nos casos apresentados para o robô nas posições C ou em D, como estão localizadas próximas à paredes, a melhor situação seria permanecer nesta posição para evitar acidentes. Situações como estas serão abordadas em trabalhos futuros.

As posições C e D são mais simples, pois o robô não realiza grande navegação para chegar à célula objetivo, por estar próximo à regiões RLs.

5.2.3 Simulação com os robôs no salão de festas

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo.

Com base na figura 46a, anteriormente apresentada, é possível perceber que o robô precisou percorrer cinco células para chegar ao objetivo, a partir de sua posição A. Para que isso fosse possível, ele navegou na diagonal, passando por quatro células de risco (RR), parando na célula livre (RL). Sendo assim, a distância total percorrida pelo robô é de 6,64 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa, levaria aproximadamente 5 segundos para o êxito de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1 m/s, levaria 6,64 segundos, enquanto que o Savioke Relay, cuja velocidade é de cerca de 0,7 m/s, levaria cerca de 9,5 segundos para o êxito da operação.

Para a situação em que o robô esteja na posição B do mapa (figura 46b), anteriormente apresentada, o robô também precisou percorrer cinco células para chegar ao objetivo. Para que isso fosse possível, ele navegou na diagonal, passando por quatro células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 7,05 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 5,26 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 7,05 segundos e o Savioke Relay levaria cerca de 10 segundos para alcançar o objetivo.

Considerando a situação em que o robô esteja na posição C do mapa (figura 46c), anteriormente apresentada, o robô precisou percorrer duas células para chegar ao objetivo. Ele passou por uma célula de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo

robô é de 2,41 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 1,79 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 2,41 segundos e o Savioke Relay levaria cerca de 3,44 segundos para alcançar o objetivo.

Por fim, para a situação em que o robô esteja na posição D do mapa (figura 46d), anteriormente apresentada, o robô precisou percorrer duas células para chegar ao objetivo. Ele passou por uma célula de risco (RR), parando na célula livre (RL). A distância total percorrida pelo robô é de 2 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 1,5 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 2 segundos e o Savioke Relay levaria cerca de 2,85 segundos para alcançar o objetivo.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para o ambiente de uma Universidade.

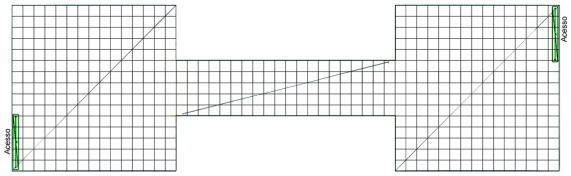
5.3 UNIVERSIDADE

Este é um ambiente que simula um espaço de uma Universidade em que não há obstáculos, porém há um corredor que une dois prédios deste ambiente fechado. Possui 15 x 50 m e uma área livre de 550 m^2 para movimentação do robô e das pessoas. Foi considerado que a multidão, de 550 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local e, no momento de uma emergência, ela tem a opção de abandonar o ambiente por qualquer um dos dois acessos, porém irá procurar o mais próximo, com base nos parâmetros de configuração do simulador. Caso o acesso mais próximo esteja congestionado, os agentes podem procurar evacuar o ambiente fechado pelo outro acesso. A figura 47 apresenta os mapas deste ambiente.

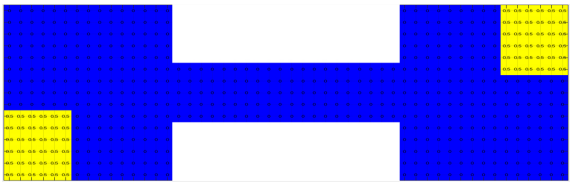
A simulação da figura 47c gerou um arquivo com 15959 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 11 segundos.

Figura 47 – Exemplo de mapa de uma Universidade.

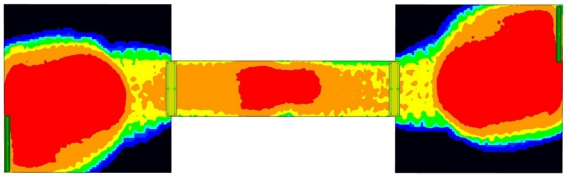
(a) Mapa decomposto em células.



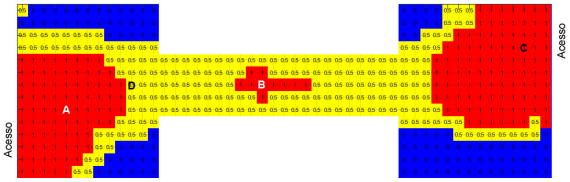
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.



Fonte: Autor.

O MI deste ambiente, representado na figura 47d, mostra quatro letras (A, B, C e D), representando posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência. Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

5.3.1 Definição da célula objetivo na Universidade

Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 47. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

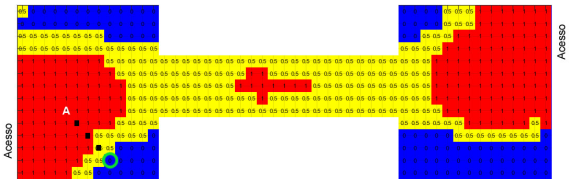
A figura 48 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A, B, C e D, no momento da emergência. Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

Para o caso em que o robô se encontra na posição A, que fica em uma área onde há movimento intenso da multidão, em uma região RR, conforme figura 48a, é possível verificar que o valor do menor custo é 2,5. A célula com um círculo verde representa o objetivo para o

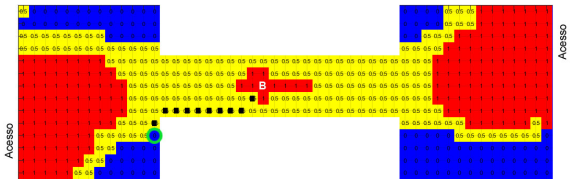
caso em que o robô se encontra em A. Esta região é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Figura 48 – Células objetivo em cada posição do robô na Universidade.

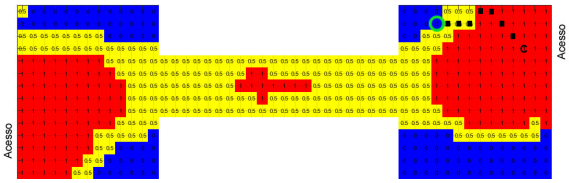
(a) Célula objetivo do robô posicionado em A.



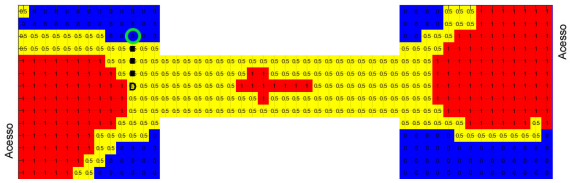
(b) Célula objetivo do robô posicionado em B.



(c) Célula objetivo do robô posicionado em C.



(d) Célula objetivo do robô posicionado em D.



Fonte: Autor.

Para o caso em que o robô se encontra na posição B, que fica no corredor, em uma pequena região RR, conforme figura 48b, é possível verificar que o valor do menor custo é 5,0. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Para o caso em que o robô se encontra na posição C, que fica próxima a um acesso, em uma região RA, conforme figura 48c, observa-se que o valor do menor custo é 5,5. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Já para o caso em que o robô se encontra na posição D, em uma região RR próxima ao corredor, conforme figura 48d, verifica-se que o valor do menor custo é 1,5. A região escolhida para a célula objetivo também é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

5.3.2 Definição da trajetória do robô na Universidade

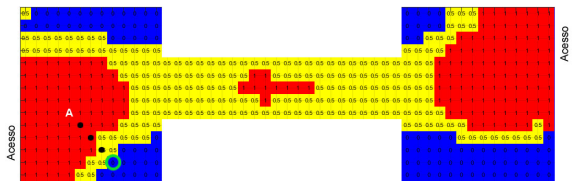
Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A* descrita no capítulo 2, seção 2.3.

Para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. A figura 49 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*.

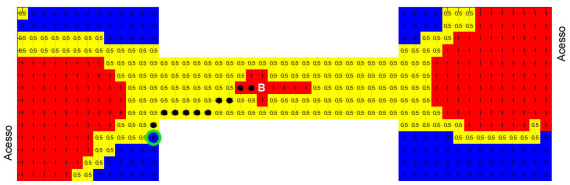
Na figura 49a, é possível perceber que o robô cria uma trajetória na diagonal, representada por pontos na cor preta, até a célula objetivo, porém não é a favor do sentido de movimento da multidão, embora não seja também contra ele. Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência em que ele esteja na Universidade, na posição A, é *aceitável*, embora ele tenha que passar por uma grande região RR, já que ele se encontra nesta área. Como o robô respeita o espaço social das pessoas, é bem provável que, nesta condição, ele fique parado em grande parte do tempo em que houver a evacuação, aguardando as pessoas saírem para, assim, desenvolver sua trajetória de navegação. Entretanto, este cenário deve ser observado em trabalhos futuros quando a teoria for aplicada em um robô real.

Figura 49 – Trajetória seguida pelo robô utilizando a técnica A*, na Universidade.

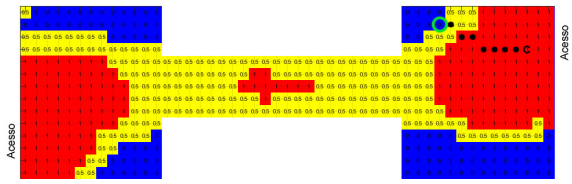
(a) Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo.



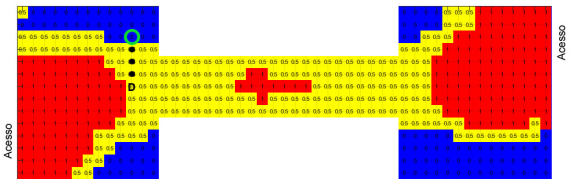
(b) Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



(c) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



(d) Trajetória seguida pelo robô desde a sua posição (D) até a célula objetivo.



Fonte: Autor.

Já na figura 49b, o robô cria uma trajetória buscando ficar próximo às paredes até a célula objetivo, no sentido do movimento da multidão. Esta trajetória seguida pelo robô até a célula objetivo em uma situação em que ele esteja na posição B, é *aceitável*, não tendo que passar por grandes regiões RRs, embora se localize em uma delas no momento da emergência. O sentido de movimento da multidão não é comprometido, pois o robô não realizará seu movimento contra ele. Entretanto, ele percorre um caminho longo até chegar ao objetivo. Em casos como este,

procurar estabelecer sua posição próximo a alguma parede iria reduzir o percurso e diminuir as chances de acidentes.

Na figura 49c, observa-se que o robô, muito próximo a um acesso e em uma região RR do mapa, cria uma trajetória até a célula objetivo que, inicialmente, não é a mais adequada, pois ela vai contra o sentido de evacuação. Deste modo, esta trajetória seguida pelo robô até a célula objetivo em uma situação em que ele esteja na posição C, é considerada *não aceitável*. Como o robô respeita o espaço social das pessoas, é bem provável que, nesta condição, ele fique parado em grande parte do tempo em que houver a evacuação, aguardando as pessoas saírem para, assim, desenvolver sua trajetória de navegação. Entretanto, este cenário deve ser observado em trabalhos futuros quando a teoria for aplicada em um robô real. Para evitar ao máximo que acidentes ocorram, talvez a melhor estratégia para o robô seria procurar estacionar próximo a uma parede no sentido da evacuação (direita do mapa) ou, ainda, seguir a multidão e sair do ambiente juntamente com a multidão.

No caso da posição do robô em D (figura 49d), observa-se que é criada uma trajetória perpendicular ao corredor. Entretanto o caminho é curto e, além disso, ele não navega por regiões de risco (RR), o que faz com que este trajeto seja *aceitável*. A evacuação da multidão não é comprometida, pois o robô não realizará seu movimento contra ela.

Casos como aqueles em que o robô esteja posicionado em B ou C serão tratados em trabalhos futuros.

5.3.3 Simulação com os robôs na Universidade

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo.

Com base na figura 49a, anteriormente apresentada, é possível perceber que o robô precisou percorrer quatro células para chegar ao objetivo, a partir de sua posição A. Para que isso fosse possível, ele navegou na diagonal, passando por duas células de risco (RR), por uma de atenção (RA) e parando na célula livre (RL). Sendo assim, a distância total percorrida pelo robô é de 5,64 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa, levaria aproximadamente 4,2 segundos para o êxito

de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1 m/s, levaria 5,64 segundos, enquanto que o Savioke Relay, cuja velocidade é de cerca de 0,7 m/s, levaria cerca de 8 segundos para o êxito da operação.

Para a situação em que o robô esteja na posição B do mapa (figura 49b), anteriormente apresentada, o robô precisou percorrer onze células para chegar ao objetivo. Para que isso fosse possível, ele navegou em linha reta e também na diagonal, passando por duas células de risco (RR), oito células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 12,23 metros. Assim, o robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 9,13 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 12,23 segundos e o Savioke Relay levaria cerca de 17,47 segundos para alcançar o objetivo.

Considerando a situação em que o robô esteja na posição C do mapa (figura 49c), anteriormente apresentada, o robô precisou percorrer oito células para chegar ao objetivo. Ele passou por seis células de risco (RR), uma célula de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 8,82 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 6,58 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 8,82 segundos e o Savioke Relay levaria cerca de 12,6 segundos para alcançar o objetivo. Conforme discutido anteriormente, a mais adequada estratégia do robô poderia ser evadir o ambiente juntamente com a multidão, já que inicialmente ele vai contra o sentido de evacuação. Nesta posição C, ele teria que navegar em linha reta para a direita, passando por 3 células de risco, totalizando três metros. O primeiro robô levaria 2,24 segundo, aproximadamente, para evacuar o ambiente junto com a multidão, enquanto que o segundo robô levaria 3 segundos e o terceiro levaria 4,3 segundos, aproximadamente. Embora os tempos sejam curtos e aparentemente seria rápida a saída do robô, aspectos relevantes como o respeito social, velocidade em que a multidão corre para abandonar o ambiente, entre outros, deve ser considerado. Assim, esta discussão será feita em trabalhos futuros, conforme já mencionado.

Por fim, para a situação em que o robô esteja na posição D do mapa (figura 49d), anteriormente apresentada, o robô precisou percorrer quatro células para chegar ao objetivo. Ele passou por três células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 4 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 3 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 4 segundos e o Savioke Relay levaria cerca de 5,7 segundos para alcançar o objetivo.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

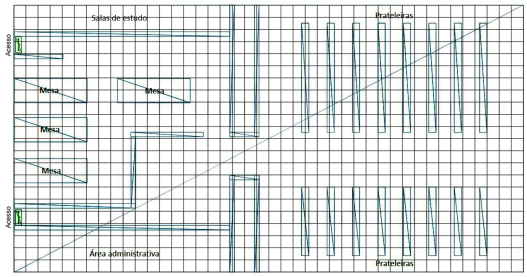
A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para uma biblioteca.

5.4 BIBLIOTECA

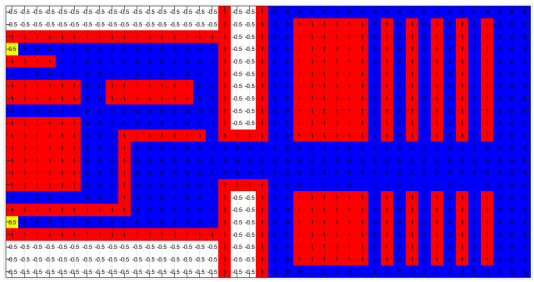
Este é um ambiente que simula uma biblioteca onde há diversos obstáculos como, por exemplo, paredes, mesas de estudo e estantes de livros. Além disso há muitos corredores e os espaços de passagem são estreitos. Possui 22 x 42 m e uma área livre de 557 m^2 para movimentação do robô e das pessoas. A figura 50 apresenta os mapas deste ambiente.

Figura 50 – Exemplo de mapa de uma biblioteca.

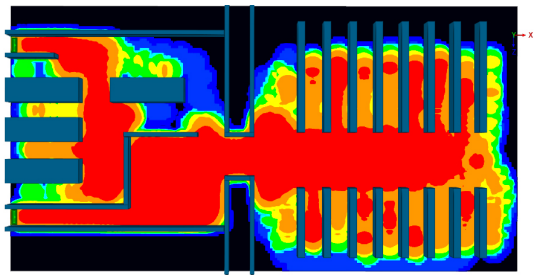
(a) Mapa decomposto em células.



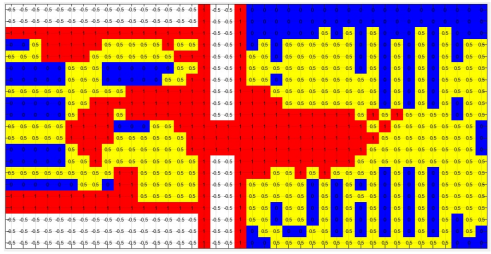
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.



Fonte: Autor.

Foi considerado que a multidão, de 557 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local e, no momento de uma emergência, ela tem a opção de abandonar o ambiente por qualquer um dos dois acessos, porém irá procurar o mais próximo, com base nos

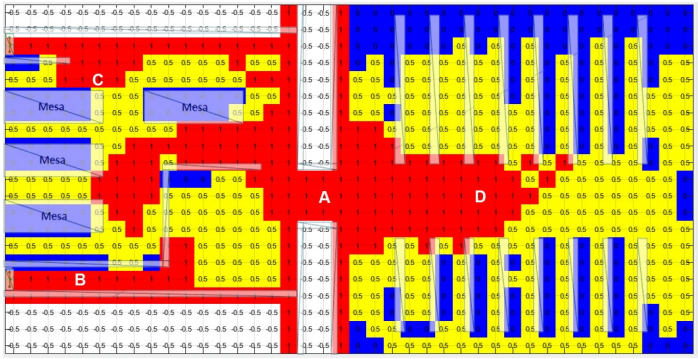
parâmetros de configuração do simulador. Caso o acesso mais próxima esteja congestionada, os agentes podem procurar evacuar o ambiente fechado pelo outro acesso.

A simulação da figura 50c gerou um arquivo com 72315 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 29 segundos.

Nas figuras 50b e 50c, a região em branco são as áreas inacessíveis pelo robô. O valor -0.5 é apenas ilustrativo, não configurando uma nova classificação feita por ele.

O MI deste ambiente, representado na figura 51, mostra quatro letras (A, B, C e D) que representam posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência. Para melhor visualização do cenário, foi realizada a sobreposição da estrutura que representa o mapa, no MI. Neste caso, todas as posições foram escolhidas em regiões RRs do mapa, para que as situações mais críticas sejam analisadas. Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

Figura 51 – MI da biblioteca com sobreposição da representação do mapa.



Fonte: Autor.

As etapas a seguir mostrarão as definições das células objetivo para cada posição do robô, bem como a trajetória que o robô seguirá utilizando o algoritmo A*. Na sequência, serão apresentados os resultados de uma simulação com características de robôs reais relacionados às definições anteriores.

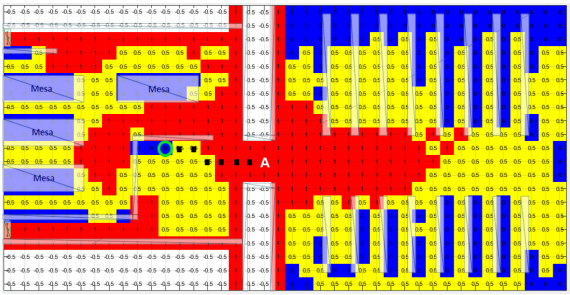
5.4.1 Definição da célula objetivo na biblioteca

Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 51. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

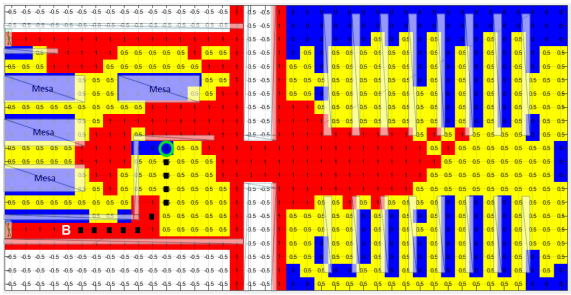
A figura 52 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A, B, C e D, no momento da emergência. Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

Figura 52 – Células objetivo em cada posição do robô na biblioteca.

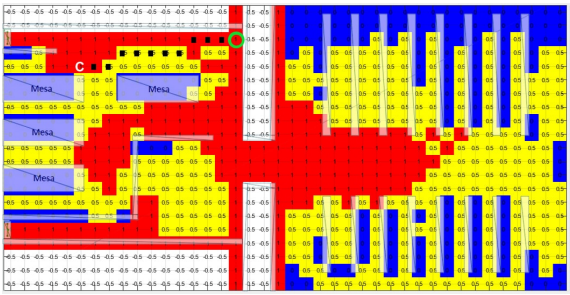
(a) Célula objetivo do robô posicionado em A.



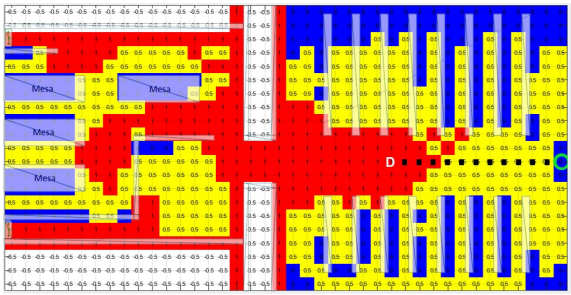
(b) Célula objetivo do robô posicionado em B.



(c) Célula objetivo do robô posicionado em C.



(d) Célula objetivo do robô posicionado em D.



Fonte: Autor.

Para o caso em que o robô se encontra na posição A, que fica em um acesso que separa dois ambientes da biblioteca, onde há movimento intenso da multidão, em uma região RR, conforme figura 52a, é possível verificar que o valor do menor custo é 4,5. A célula com um círculo verde representa o objetivo para o caso em que o robô se encontra em A. Esta região é

aceitável como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, uma vez que está junto à uma parede.

O exemplo em que o robô se encontra na posição B, que fica em um corredor estreito, em uma região RR, conforme figura 52b, mostra que o valor do menor custo é 8,0. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência.

Para o caso em que o robô se encontra na posição C, que fica próxima a um corredor de acesso, em uma região RA, conforme figura 52c, observa-se que o valor do menor custo é 5,5. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, uma vez que está junto à uma parede.

Já para o caso em que o robô se encontra na posição D, em uma região RR no corredor mais largo, conforme figura 52d, verifica-se que o valor do menor custo é 7,0. Esta região da célula objetivo também é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois fica ao final da biblioteca, junto à parede.

5.4.2 Definição da trajetória do robô na biblioteca

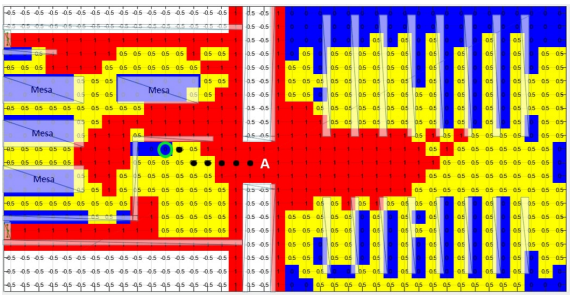
Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A*, descrita no capítulo 2, seção 2.3.

Para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer.

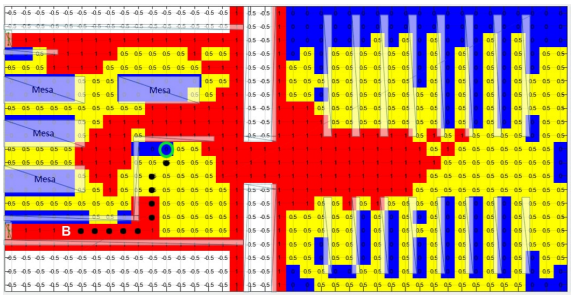
A figura 53 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*. Na figura 53a, é possível perceber que o robô cria uma trajetória no mesmo sentido da multidão que passa por este acesso estreito, até a célula objetivo. Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência em que ele esteja na biblioteca, na posição A, é *aceitável*. O sentido de movimento da multidão não é comprometido, pois o robô não realizará seu movimento contra ele. É importante analisar que, para este exemplo, o robô inicialmente pode causar uma considerável obstrução no acesso, aumentando esta expectativa quanto menor for a sua velocidade, pois nesta posição, há um acúmulo de pessoas tentando evacuar o ambiente.

Figura 53 – Trajetória seguida pelo robô utilizando a técnica A*, na biblioteca.

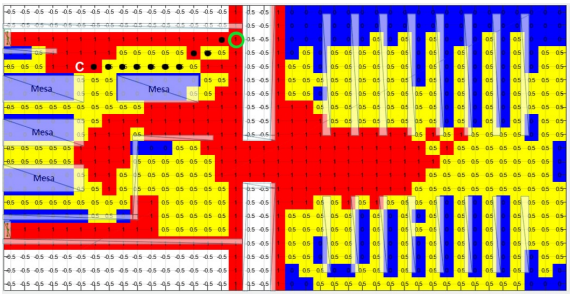
(a) Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo.



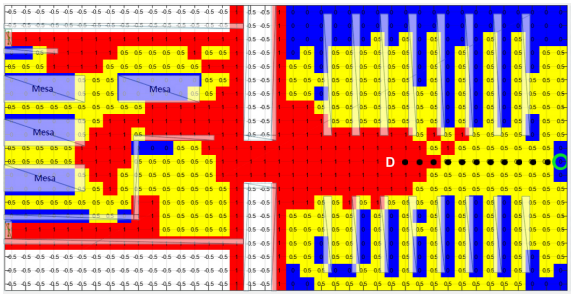
(b) Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



(c) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



(d) Trajetória seguida pelo robô desde a sua posição (D) até a célula objetivo.



Fonte: Autor.

Já na figura 53b, o robô cria uma trajetória contra o sentido de evacuação, até a célula objetivo, pois não há células livres à frente, no mesmo sentido em que a multidão corre para evacuar o ambiente. Esta trajetória seguida pelo robô até a célula objetivo é considerada *não aceitável*, pois neste corredor estreito certamente ele congestionaria o acesso por completo, não permitindo que as pessoas saiam do local. Possivelmente a melhor solução para este caso seria o robô movimentar-se no mesmo sentido da multidão e sair com ela. Entretanto, este tipo de situação será abordado em trabalhos futuros, uma vez que há outras variáveis a serem estudadas, como o comportamento do robô em áreas abertas, por exemplo.

Na figura 53c, é possível perceber que o robô cria uma trajetória junto à uma mesa do ambiente, até a célula objetivo. É evidente que as mesas podem ser deslocadas de suas posições, mas isto não está sendo considerado aqui. A trajetória seguida pelo robô até a célula objetivo, é *aceitável*, pois seu movimento ocorre principalmente em uma região RA e o sentido de movimento da multidão não é comprometido, uma vez que o robô não realiza seu movimento contra ele.

Na figura 53d, observa-se que o robô cria uma trajetória de sentido contrário ao sentido de evacuação da multidão, até a célula objetivo, porém em uma região RA de corredor largo,

onde a intensidade de passagem dos agentes é moderada. A trajetória seguida pelo robô até a célula objetivo é, portanto, *aceitável*.

5.4.3 Simulação com os robôs na biblioteca

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo, a saber, o Knightscope K5, o Panasonic Signage HOSPI e o Savioke Relay.

Com base na figura 53a, anteriormente apresentada, é possível perceber que o robô precisou percorrer sete células para chegar ao objetivo, a partir de sua posição A no MI. Para que isso fosse possível, ele realizou uma navegação predominantemente em linha reta, com um deslocamento na diagonal, passando por três células de risco (RR), três células de atenção (RA) e, finalmente, parando na célula livre (RL).

Sendo assim, a distância total percorrida pelo robô é de 7,41 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa e necessita navegar 7,41 metros até a célula objetivo, levaria aproximadamente 5,5 segundos para o êxito de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1,0 m/s, levaria 7,41 segundos, enquanto que o Savioke Relay, cuja velocidade é de 0,7 m/s, levaria cerca de 10,6 segundos para cumprir seu propósito.

Para a situação em que o robô esteja na posição B do mapa (figura 53b), anteriormente apresentada, o robô precisou percorrer onze células para chegar ao objetivo. Para que isso fosse possível, ele realizou uma navegação predominantemente em linha reta, com dois deslocamentos na diagonal, passando por sete células de risco (RR), três células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 11,82 metros, sendo que necessita se movimentar 6,41 metros somente para sair do corredor, contra o sentido de evacuação. O robô Knightscope K5, para percorrer o trajeto completo, levaria aproximadamente 8,82 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 11,82 segundos e o Savioke Relay levaria cerca de 16,88 segundos para alcançar o objetivo.

Considerando a situação em que o robô esteja na posição C do mapa (figura 53c), anteriormente apresentada, o robô precisou percorrer onze células para chegar ao objetivo. Ele passou por duas células de risco (RR), oito células de atenção (RA), parando na célula livre

(RL). A distância total percorrida pelo robô é de 11,82 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 8,82 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 11,82 segundos e o Savioke Relay levaria cerca de 16,88 segundos para alcançar o objetivo.

Por fim, para a situação em que o robô esteja na posição D do mapa (figura 53d), anteriormente apresentada, o robô precisou percorrer doze células para chegar ao objetivo. Ele passou por três células de risco (RR), oito células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 12 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 8,96 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 12 segundos e o Savioke Relay levaria cerca de 17,14 segundos para alcançar o objetivo.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para o ambiente de um apartamento.

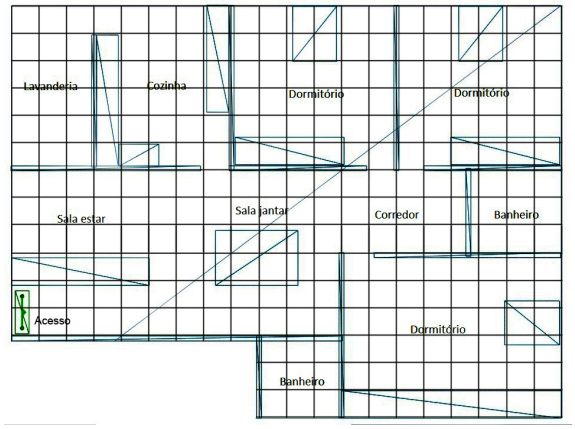
5.5 APARTAMENTO

Este é um ambiente que simula um apartamento, onde há obstáculos como, por exemplo, paredes e móveis em geral. Possui 15 x 20 m e uma área livre de 230 m^2 para movimentação do robô e das pessoas. Foi considerado que a multidão, de 57 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local, porém neste caso específico, a quantidade de pessoas por m^2 é menor, sendo uma pessoa a cada 4 m^2. No momento de uma emergência, a multidão tem apenas uma opção de acesso para abandonar o ambiente. A figura 54 apresenta os mapas deste ambiente.

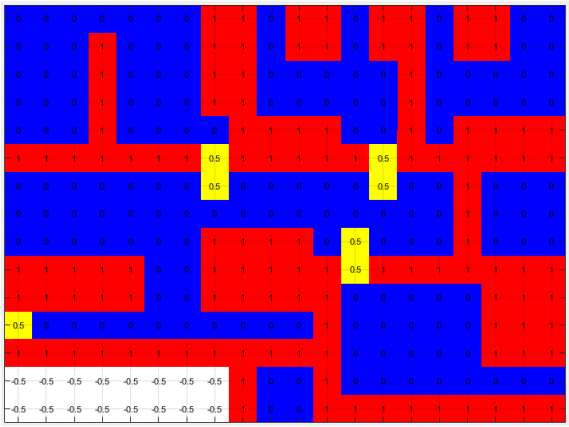
A simulação da figura 50c gerou um arquivo com 1496 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 9 segundos.

Figura 54 – Exemplo de mapa de um apartamento.

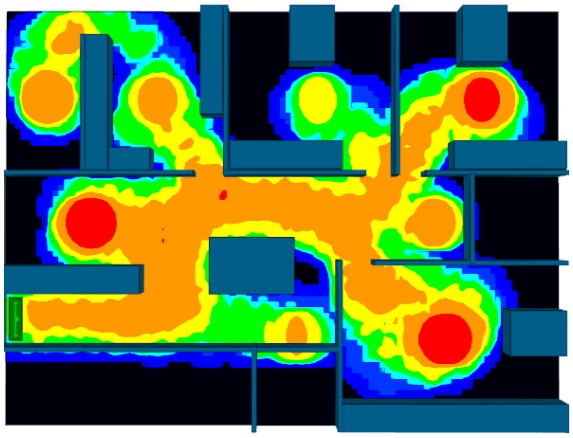
(a) Mapa decomposto em células.



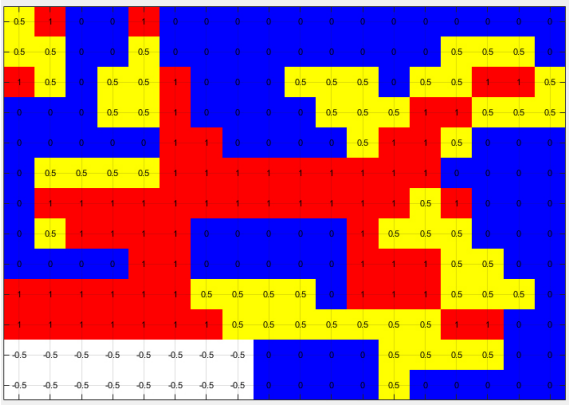
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.



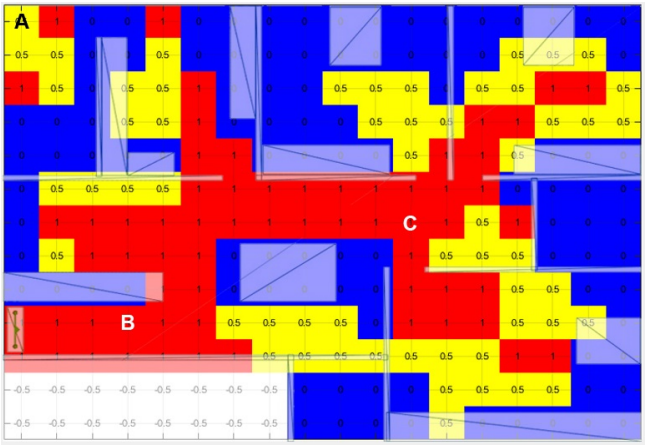
Fonte: Autor.

Na figura 54b, a região em branco são as áreas inacessíveis pelo robô. O valor -0.5 é apenas ilustrativo, não configurando uma nova classificação feita por ele.

O MI deste ambiente, representado na figura 55, mostra três letras (A, B e C) que representam posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência.

Para melhor visualização do cenário, foi realizada a sobreposição da estrutura que representa o mapa, no MI. Neste caso, duas das posições estão em regiões RRs do mapa e uma delas está em uma região RA. Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

Figura 55 – MI do apartamento com sobreposição da representação do mapa do apartamento.



Fonte: Autor.

As etapas a seguir mostrarão as definições das células objetivo para cada posição do robô, bem como a trajetória que o robô seguirá utilizando o algoritmo A*. Na sequência, serão apresentados os resultados de uma simulação com características de robôs reais relacionados às definições anteriores.

5.5.1 Definição da célula objetivo no apartamento

Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 55. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

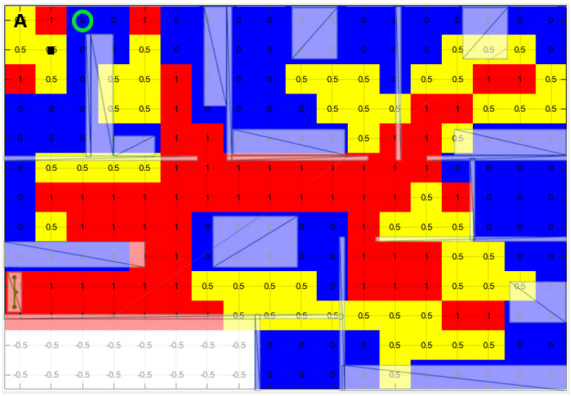
A figura 56 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A, B e C, no momento da emergência. Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

Para o caso em que o robô se encontra na posição A, que fica no encontro de duas paredes em um cômodo do apartamento, em uma região RA, conforme figura 56a, é possível verificar que o valor do menor custo é 0,5. A célula com um círculo verde representa o objetivo para o caso em que o robô se encontra em A. Esta região é considerada *não aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois

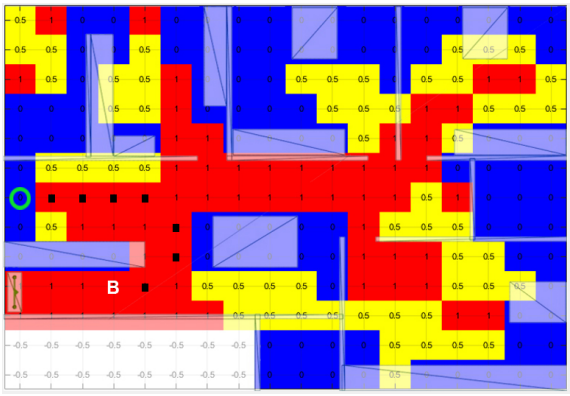
ele estacionaria em frente a um acesso, impedindo a saída das pessoas. Este acesso, inclusive, é o único que permite evacuar tal cômodo do apartamento. É importante salientar que os dados do robô são arredondados para que seja possível a obtenção de valores inteiros para as coordenadas x e y de cada agente no espaço de tempo da simulação. Ao fazer isso, o algoritmo interpretou que a célula com maior intensidade de passagem de agentes foi a célula vizinha à do acesso.

Figura 56 – Células objetivo em cada posição do robô no apartamento.

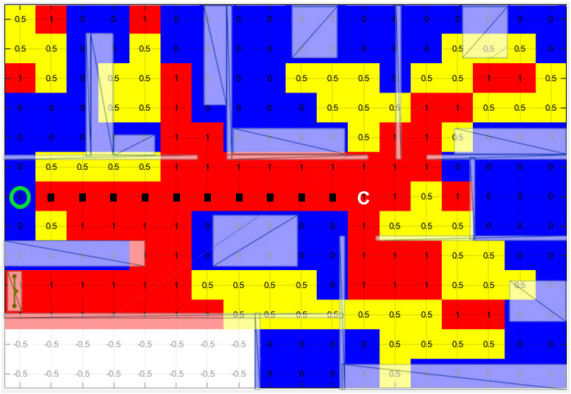
(a) Célula objetivo do robô posicionado em A.



(b) Célula objetivo do robô posicionado em B.



(c) Célula objetivo do robô posicionado em C.



Fonte: Autor.

O exemplo em que o robô se encontra na posição B, que fica no corredor que permite a evacuação da multidão pelo ambiente, em uma região RR, conforme figura 56b, mostra que o valor do menor custo até a célula objetivo é 7,0. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, uma vez que está junto à uma parede.

Para o caso em que o robô se encontra na posição C, que fica em uma região RR do ambiente, conforme figura 56c, observa-se que o valor do menor custo até a célula objetivo é 10. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o

posicionamento do robô quando ocorre uma situação de emergência, uma vez que está junto à uma parede.

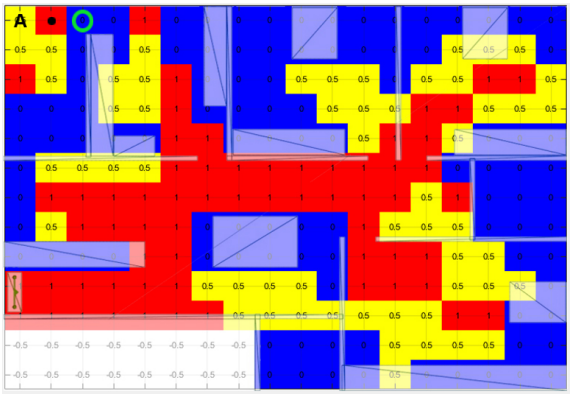
5.5.2 Definição da trajetória do robô no apartamento

Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A*, descrita no capítulo 2, seção 2.3.

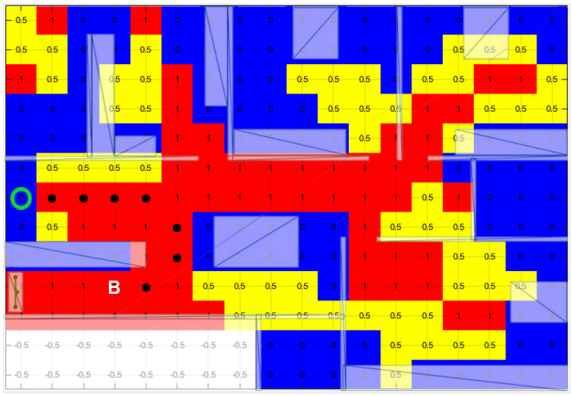
Para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. A figura 57 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*.

Figura 57 – Trajetória seguida pelo robô utilizando a técnica A*, no apartamento.

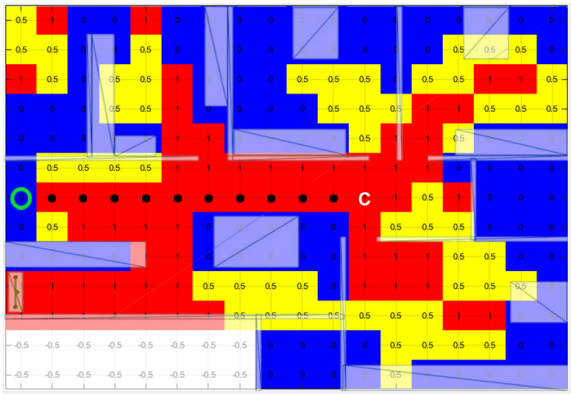
(a) Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo.



(b) Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



(c) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



Fonte: Autor.

Na figura 57a, é possível perceber que o robô cria uma trajetória em linha reta, até a célula objetivo. Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência em que ele esteja no apartamento, na posição A, é considerada *não aceitável*, pois ele se posiciona exatamente em frente ao acesso estreito (cerca de 1 metro) que permite que as pessoas evacuem este cômodo antes de evacuarem o ambiente. A melhor solução para esta situação seria o robô permanecer parado na posição inicial (A), uma vez que se encontra em um canto do cômodo no ambiente, junto às paredes. Deste modo, não haveria impedimento da evacuação da multidão neste acesso. Este tipo de situação será abordado em trabalhos futuros.

Já na figura 57b, o robô cria uma trajetória contra o sentido de movimento da multidão, até a célula objetivo, pois ele se encontra em uma região RR no corredor que possui o acesso principal por onde a multidão evacua o ambiente e não há células livres à frente. Esta trajetória seguida pelo robô até a célula objetivo é considerada *não aceitável*, pois neste corredor estreito certamente ele congestionará o acesso por completo, não permitindo que as pessoas saiam do local. Possivelmente a melhor solução para este caso seria o robô movimentar-se no mesmo sentido da multidão e sair com ela. Entretanto, este tipo de situação será abordado em trabalhos futuros, uma vez que há outras variáveis a serem estudadas, como o comportamento do robô em áreas abertas, por exemplo.

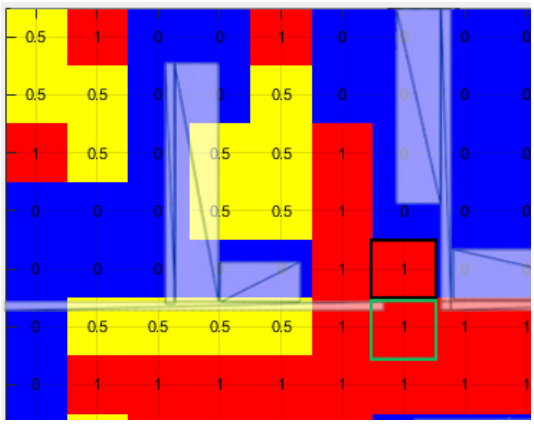
Na figura 57c, é possível perceber que o robô cria uma trajetória em linha reta, até a célula objetivo, passando por uma considerável região RR. A trajetória seguida pelo robô até a célula objetivo, é *aceitável*, pois o sentido de movimento da multidão não é comprometido, uma vez que o robô não realiza seu movimento contra ele. Como o robô respeita o espaço social das pessoas, é bem provável que, nesta condição, ele fique parado em grande parte do tempo em que houver a evacuação, aguardando as pessoas saírem para, assim, desenvolver sua trajetória de navegação. Entretanto, este cenário deve ser observado em trabalhos futuros quando a teoria for aplicada em um robô real.

5.5.3 Particularidade do ambiente apartamento

Este tipo de ambiente possui algumas características particulares que necessitam de análise. Como exemplo, alguns acessos possuem largura de porta de 82 cm, medida esta que é inferior ao tamanho de uma célula, conforme definição 4.6.

Sendo assim, todas as células que estejam exatamente nos acessos que possuem dimensões como esta representarão uma célula com obstáculo, pois pelo menos parte dela contém uma parede. Isto é possível observar na figura 58, onde há um quadrado na cor verde.

Figura 58 – Representação de célula parcialmente ocupada por um obstáculo (parede).



Fonte: Autor.

Sabe-se que a célula (indicada em destaque no quadrado preto), na figura, possui o valor correspondente mostrado no centro dela. A célula do quadrado verde sempre terá o valor 1,0, desde a geração do MA, pois uma parte da parede ocupa esta célula. A implicação disso é que o robô localizado na parte superior da figura 58, no momento de uma situação de emergência, não poderá sair do ambiente, já que a única célula do acesso apresenta uma ocupação. Se um cômodo qualquer do ambiente (não é este o caso) não possuir uma célula livre, deverá haver um tratamento para isso, situação esta que será abordada em trabalhos futuros.

Em ambientes que possuem muitos objetos com dimensões inferiores a 1 metro, como por exemplo as mobílias de um apartamento, muitas são as células que terão o status 1,0 por estarem parcialmente ocupadas. Isto pode fazer com que haja pouco aproveitamento de espaços que poderiam estar livres para o posicionamento do robô em situações de emergência. Uma possível solução para este problema seria considerar dimensões menores para as células, como é o exemplo mostrado na figura 59.

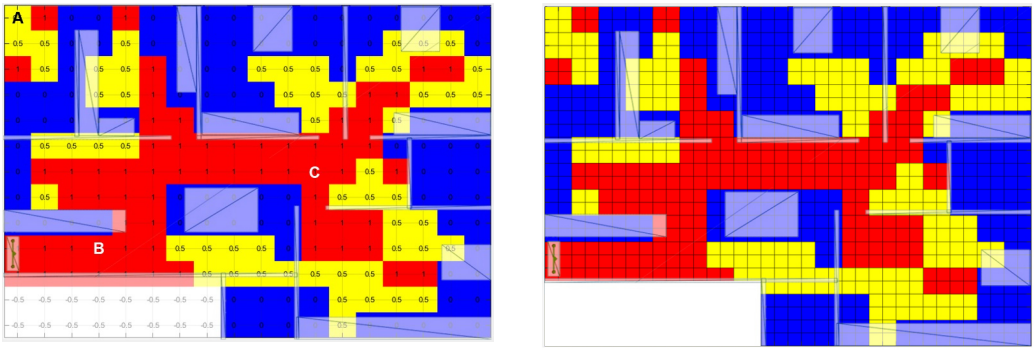
É possível, por meio da figura 59b, observar que os acessos entre cômodos possuem células que permitiriam a passagem do robô de um cômodo para outro em casos de emergência, desde que as dimensões dele sejam compatíveis com as dimensões das células. Entretanto, di-

mensões menores para as células não estão no escopo deste estudo e será abordado em trabalhos futuros.

Figura 59 – Decomposição em células de tamanhos variados.

(a) Decomposição celular do apartamento com células de tamanho 1,0 x 1,0 metro, conforme definição 4.6.

(b) Decomposição celular do apartamento com células de tamanho 0,5 x 0,5 metro.



Fonte: Autor.

Outro ponto importante a ser considerado é que foram considerados os casos mais críticos para todos os ambientes deste trabalho. Isso quer dizer que os testes utilizam a quantidade de agentes (para a simulação) que superlota os ambientes. O resultado disso é que há muitas áreas de risco (RR) sendo apresentadas e no ambiente do apartamento, mesmo sendo previsto 1 pessoa para cada 4 m^2, conforme tabela 4, poucas são as células livres em que o robô poderá escolher para estacionar em uma emergência (a mobília contribui também este problema). Uma possível solução em casos como este, considerando que, geralmente, ambientes residenciais não ficam superpopulados, seria utilizar critérios moderados de utilização da quantidade de agentes na simulação ao invés de utilizar critérios para os casos mais críticos. Deste modo, novas células livres surgiriam no ambiente, tornando as possibilidades de escolha da célula objetivo mais ampla. Entretanto, casos como este serão tratados em trabalhos futuros.

5.5.4 Simulação com os robôs no apartamento

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo, a saber, o Knightscope K5, o Panasonic Signage HOSPI e o Savioke Relay.

Com base na figura 57a, anteriormente apresentada, é possível perceber que o robô precisou percorrer apenas duas células para chegar ao objetivo, a partir de sua posição A no MI.

Para que isso fosse possível, ele realizou uma navegação em linha reta saindo de uma célula de atenção (RA), passando por uma célula de risco (RR) e, finalmente, parando na célula livre (RL). Sendo assim, a distância total percorrida pelo robô é de 2,0 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa e necessita navegar 2,0 metros até a célula objetivo, levaria aproximadamente 1,5 segundo para o êxito de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1,0 m/s, levaria 2,0 segundos, enquanto que o Savioke Relay, cuja velocidade é de 0,7 m/s, levaria cerca de 2,85 segundos para cumprir seu propósito.

Para a situação em que o robô esteja na posição B do mapa (figura 57b), anteriormente apresentada, o robô precisou percorrer oito células para chegar ao objetivo. Ele realizou uma navegação predominantemente em linha reta, com dois deslocamentos na diagonal, passando apenas por células de risco (RR) antes de parar na célula livre (RL). A distância total percorrida pelo robô é de 8,82 metros, sendo que necessita se movimentar 2,41 metros somente para sair do corredor, contra o sentido de evacuação. O robô Knightscope K5, para percorrer o trajeto completo, levaria aproximadamente 6,58 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 8,82 segundos e o Savioke Relay levaria cerca de 12,6 segundos para alcançar o objetivo.

Considerando a situação em que o robô esteja na posição C do mapa (figura 57c), anteriormente apresentada, o robô precisou percorrer onze células para chegar ao objetivo, em linha reta, passando apenas por células de risco (RR) até chegar na célula livre (RL). A distância total percorrida pelo robô é de 11 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 8,2 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 11 segundos e o Savioke Relay levaria cerca de 15,7 segundos para alcançar o objetivo.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô. Além disso, é evidente que alguns itens de mobília podem ser deslocados de suas posições com a reação das pessoas na emergência, fazendo com que haja um ajuste de valores das células, mas isto não está sendo considerado aqui.

Outra questão a considerar é que o robô Panasonic Signage HOSPI, por exemplo, possui dimensões muito próximas às medidas de largura dos acessos entre cômodos, não sendo indicado, portanto, para navegar por um ambiente com estas características, pois poderia atrapalhar a evacuação da multidão até mesmo quando não houver emergência.

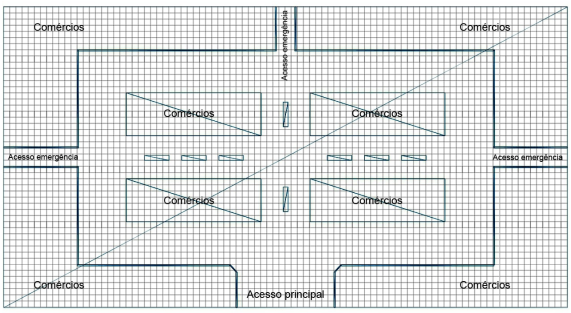
A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para o ambiente de um centro comercial.

5.6 CENTRO COMERCIAL - SHOPPING CENTER

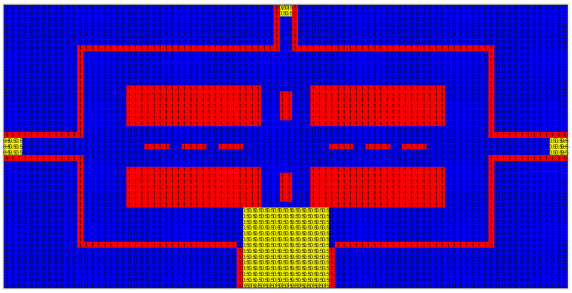
Este é um ambiente que simula um centro comercial, onde há obstáculos como, por exemplo, as lojas (comércios) e também alguns assentos (bancos) de esperadescanso (mobília). Possui 49 x 92 m e uma área livre de 1850 m^2 para movimentação do robô e das pessoas. Considera-se que a multidão, de 1850 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local, sendo uma pessoa a cada m^2. A figura 60 apresenta os mapas deste ambiente.

Figura 60 – Exemplo de mapa de um shopping center.

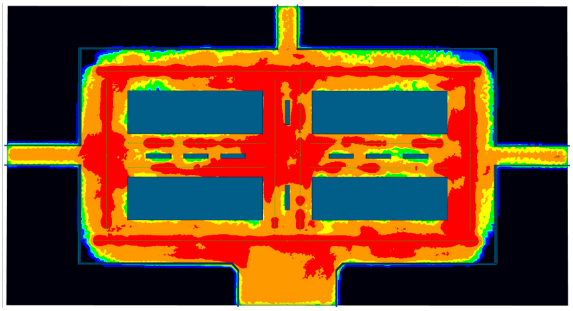
(a) Mapa decomposto em células.



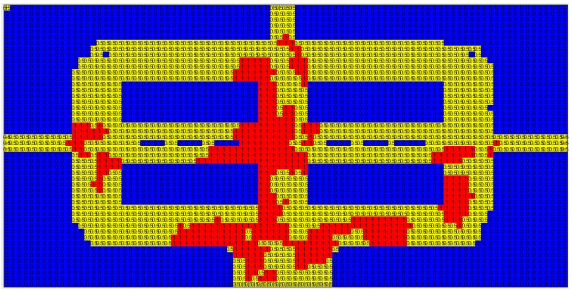
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.



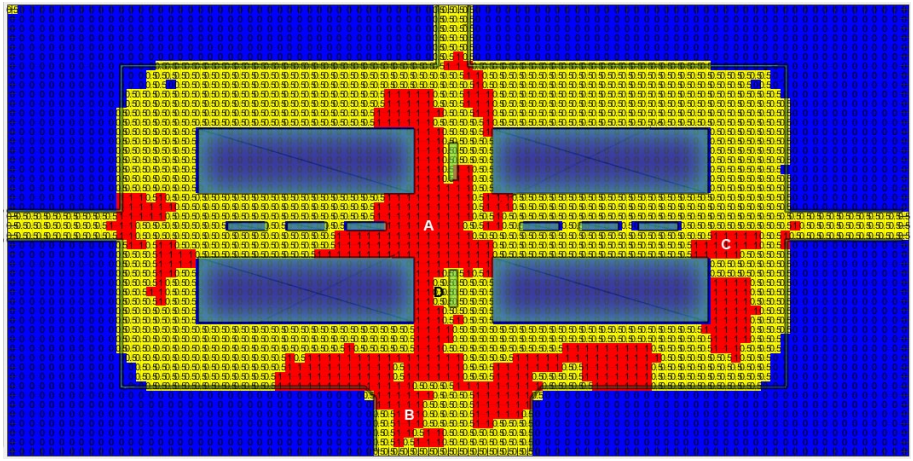
Fonte: Autor.

No momento de uma emergência, a multidão tem um acesso principal e três acessos de emergência para abandonar o ambiente. Para o acesso principal, foi atribuído peso de 70% para evacuação, enquanto que para as de emergência o peso é de 10% cada. Em caso de acesso congestionado, os agentes podem procurar evacuar o ambiente fechado por outro acesso.

A simulação da figura 60c gerou um arquivo com 134658 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 40 segundos.

O MI deste ambiente, representado na figura 61, mostra quatro letras (A, B, C e D) que representam posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência.

Figura 61 – MI do shopping center com sobreposição da representação do mapa.



Fonte: Autor.

Para melhor visualização do cenário, foi realizada a sobreposição da estrutura que representa o mapa, no MI. Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

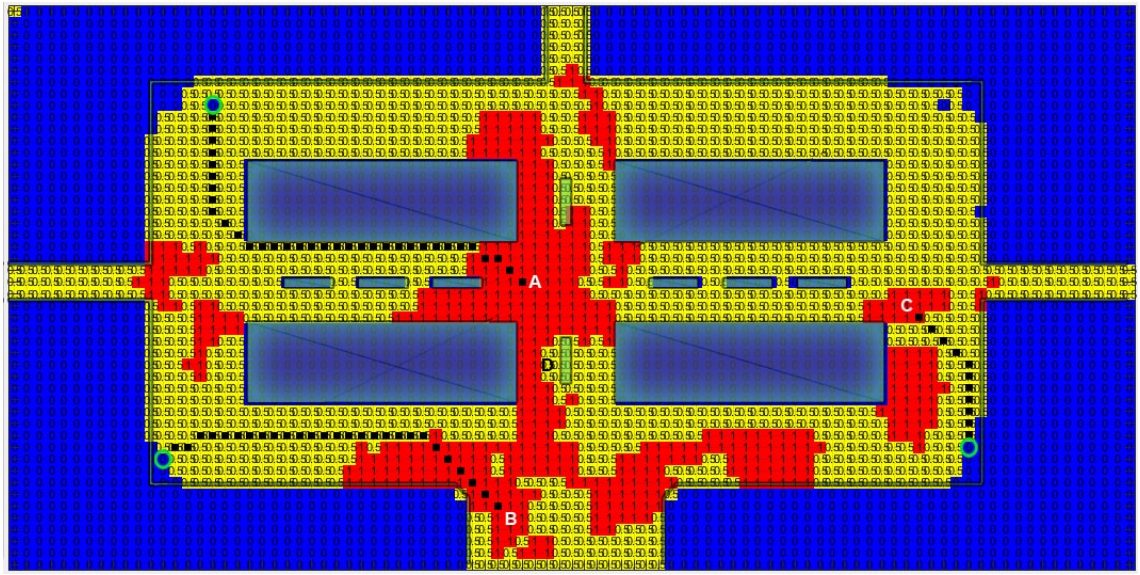
As etapas a seguir mostrarão as definições das células objetivo para cada posição do robô, bem como a trajetória que o robô seguirá utilizando o algoritmo A*. Na sequência, serão apresentados os resultados de uma simulação com características de robôs reais relacionados às definições anteriores.

5.6.1 Definição da célula objetivo no shopping center

Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 61. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

A figura 62 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A, B, C e D, no momento da emergência.

Figura 62 – Células objetivo em cada posição do robô no shopping center.



Fonte: Autor.

Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

A célula com um círculo verde representa o objetivo definido pelo robô para cada situação. Para o caso em que o robô se encontra na posição A, que fica em uma área onde há movimento intenso da multidão, em uma região RR, é possível verificar que o valor do menor custo é 19. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois está localizada próxima à paredes do ambiente.

Para o caso em que o robô se encontra na posição B, que fica próximo ao acesso principal do shopping center, em uma região RR, é possível verificar que o valor do menor custo é 16,5. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois está localizada próxima à paredes do ambiente.

Para o caso em que o robô se encontra na posição C, que fica em um corredor lateral do ambiente, em uma região RR, observa-se que o valor do menor custo é 6. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois está localizada próxima à paredes do ambiente.

Já para o caso em que o robô se encontra na posição D, em uma região RA no corredor central do ambiente, verifica-se que não é possível definir uma célula objetivo, pois o robô está em uma região cercada por células de região RR e por um obstáculo. A definição 4.30 estabelece que não é permitido ao robô adicionar a contagem de uma célula ao custo, que tenha valor superior ao da célula anterior.

5.6.2 Definição da trajetória do robô no shopping center

Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A* descrita no capítulo 2, seção 2.3.

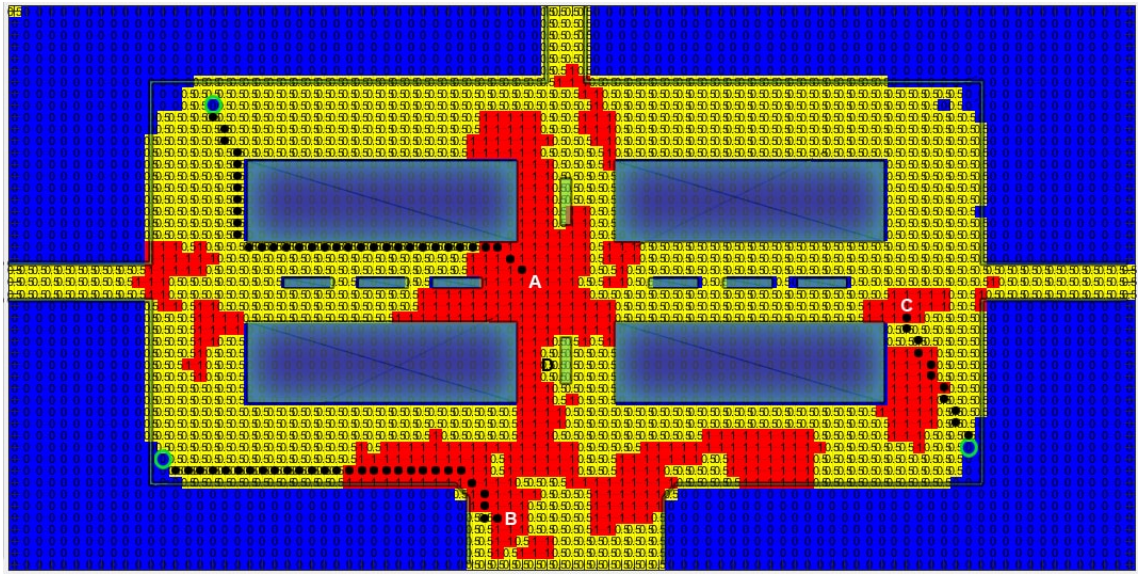
Para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. A figura 63 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*.

Na figura apresentada, para a posição A do robô, é possível perceber que o robô cria uma trajetória predominantemente em linha reta e próxima às paredes do espaço dos comércios do shopping center, representada por pontos na cor preta, até a célula objetivo. Esta trajetória seguida pelo robô até a célula objetivo, em uma situação de emergência, é *aceitável*, pois ele passa inicialmente por uma pequena região RR e, após isso, seu percurso ocorre em uma região RA. O sentido de movimento da multidão não é comprometido com esta ação do robô.

O robô, quando na posição B, cria uma trajetória buscando ficar próximo às paredes até a célula objetivo, porém contra o sentido de evacuação. Entretanto, é possível perceber que desde o início ele busca ficar próximo às paredes, o que é considerado como operação *aceitável* de navegação até a célula objetivo, uma vez que o corredor por onde a multidão se movimenta, até

o acesso principal é amplo. De qualquer maneira, situações como esta poderia trazer melhores resultados se o robô acompanhasse o sentido de movimento da multidão, evacuando o ambiente com ela, porém o tratamento para casos assim será realizado em trabalhos futuros.

Figura 63 – Trajetória seguida pelo robô utilizando a técnica A*, no shopping center.



Fonte: Autor.

O robô posicionado em C está localizado no corredor lateral do shopping center, em uma pequena região RR do ambiente. Ele cria uma trajetória até a célula objetivo que atravessa este corredor amplo, ação que poderia atrapalhar o movimento da multidão neste local. Entretanto, o sentido de navegação do robô até a célula objetivo é o mesmo do movimento de evacuação e, deste modo, a operação é *aceitável*. Em alguns momentos do trajeto, possivelmente o robô ficará parado respeitando os custos sociais envolvidos na interação com as pessoas.

No caso do robô na posição D do mapa, conforme mencionado na seção anterior (5.6.1), não há a definição da célula objetivo e, consequentemente, não há a definição da trajetória a ser seguida pelo robô. Neste caso, ele ficará parado nesta mesma posição do ambiente até que a situação se normalize ou até que seja removido manualmente ou por comando.

5.6.3 Simulação com os robôs no shopping center

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo. A figura 63 é a referência para esta análise.

O robô localizado na posição A precisou percorrer trinta e cinco células para chegar ao objetivo. Para que isso fosse possível, ele navegou predominantemente em linha reta, passando por quatro células de risco (RR), por trinta células de atenção (RA) e parando na célula livre (RL). Sendo assim, a distância total percorrida pelo robô até a célula objetivo é de 37,46 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa, levaria aproximadamente 27,9 segundos para o êxito de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1 m/s, levaria 37,5 segundos, enquanto que o Savioke Relay, cuja velocidade é de cerca de 0,7 m/s, levaria cerca de 53,5 segundos.

Para a situação em que o robô esteja na posição B do mapa, foi preciso percorrer trinta células para chegar ao objetivo. Para isso, ele navegou predominantemente em linha reta, passando por quatorze células de risco (RR), quinze células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 30,23 metros. Assim, o robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 22,6 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 30,2 segundos e o Savioke Relay levaria cerca de 43,18 segundos para alcançar o objetivo.

Considerando a situação em que o robô esteja na posição C do mapa, foi preciso percorrer doze células para chegar ao objetivo. Ele passou por cinco células de risco (RR), seis células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 13,05 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 9,73 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 13 segundos e o Savioke Relay levaria cerca de 18,6 segundos para alcançar o objetivo.

Por fim, para a situação em que o robô esteja na posição D do mapa, não há trajetória a ser percorrida pelo robô pela impossibilidade de definição da célula objetivo.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

Este tipo de ambiente possui poucas regiões RLs para onde o robô possa estacionar e aquelas que aparecem ficam nos cantos do ambiente, distantes das regiões centrais do mapa. Deste modo, dependendo da posição do robô, ele necessitará percorrer um longo caminho até alcançar o objetivo, sendo que poderia estacionar em regiões seguras mais próximas a ele.

A falta de outras células de RL ocorre porque o escopo deste trabalho prevê a classificação das células com base em dados do simulador para os cenários críticos, onde as áreas sem obstáculos possuem agentes a cada m^2.

Considerar cenários menos críticos para este tipo de ambiente faria com que novas células de RL surgissem no mapa, porém, em consequência disso, muitas células de RRs (status 1,0) poderiam deixar de existir, podendo tornar as possibilidades de acidentes mais reais. Sendo assim, este tipo de situação será tratado em trabalhos futuros.

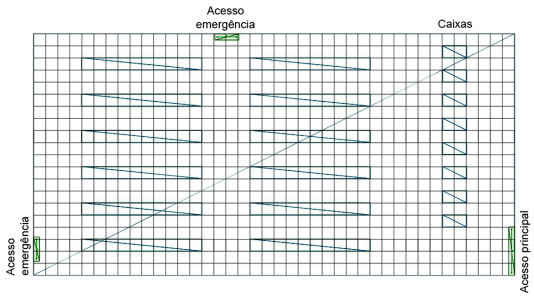
A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para um supermercado.

5.7 SUPERMERCADO

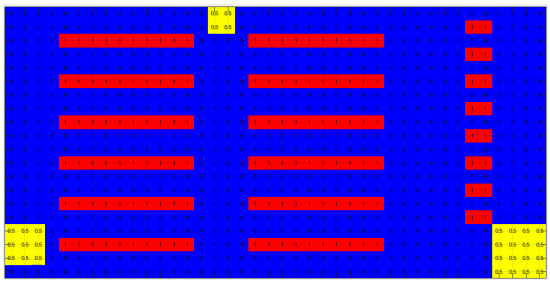
Este é um ambiente que simula um supermercado, onde há obstáculos como, por exemplo, as estantes dos produtos e também os caixas para pagamento de compras. Possui 20 x 40 m e uma área livre de 664 m^2 para movimentação do robô e das pessoas. Considera-se que a multidão, de 664 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local, sendo uma pessoa a cada m^2. A figura 64 apresenta os mapas deste ambiente.

Figura 64 – Exemplo de mapa de um supermercado.

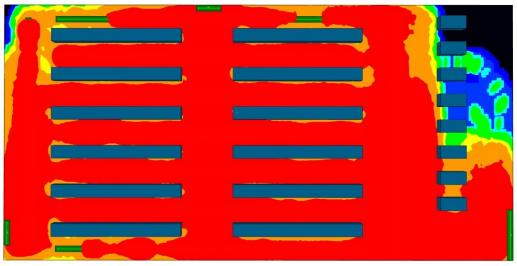
(a) Mapa decomposto em células.



(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.

Fonte: Autor.

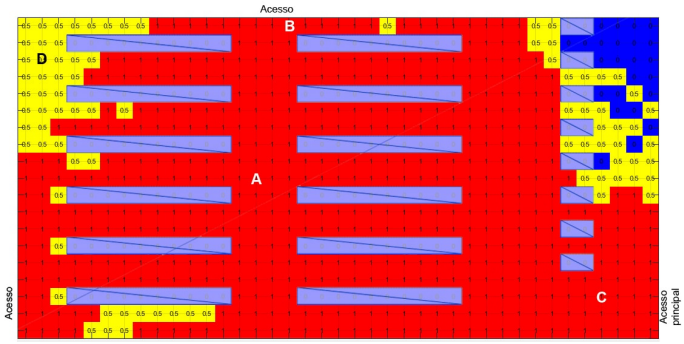
No momento de uma emergência, a multidão tem um acesso principal e dois acessos de emergência para abandonar o ambiente. Para o acesso principal, foi atribuído peso de 80% para evacuação, enquanto que para os de emergência o peso é de 10% cada. Em caso de acesso congestionado, os agentes podem procurar evacuar o ambiente fechado por outro acesso.

A simulação da figura 64c gerou um arquivo com 36637 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 21 segundos.

O MI deste ambiente, representado na figura 65, mostra quatro letras (A, B, C e D) que representam posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência.

Para melhor visualização do cenário, foi realizada a sobreposição da estrutura que representa o mapa, no MI. Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

Figura 65 – MI do supermercado com sobreposição da representação do mapa.



Fonte: Autor.

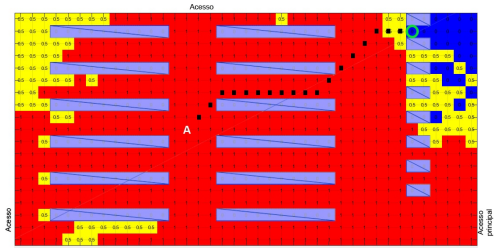
As etapas a seguir mostrarão as definições das células objetivo para cada posição do robô, bem como a trajetória que o robô seguirá utilizando o algoritmo A*. Na sequência, serão apresentados os resultados de uma simulação com características de robôs reais relacionados às definições anteriores.

5.7.1 Definição da célula objetivo no supermercado

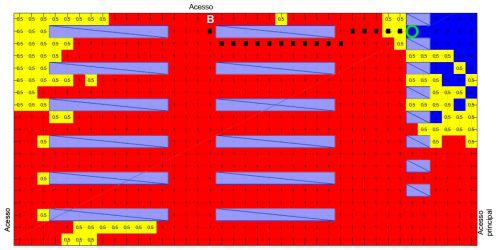
Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 65. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular. A figura 66 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A, B e C, no momento da emergência.

Figura 66 – Células objetivo em cada posição do robô no supermercado.

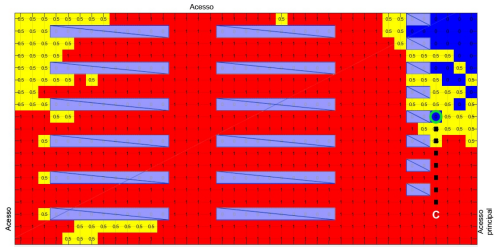
(a) Célula objetivo do robô posicionado em A.



(b) Célula objetivo do robô posicionado em B.



(c) Célula objetivo do robô posicionado em C.



Fonte: Autor.

Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

Para o caso em que o robô se encontra na posição A, que fica entre as estantes do supermercado, ao centro, em uma região RR ampla, conforme figura 66a, é possível verificar que o valor do menor custo é 17,0. A célula com um círculo verde representa o objetivo. Esta região é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois fica afastada de qualquer acesso, entre dois caixas de pagamento de compras.

O exemplo em que o robô se encontra na posição B, que fica próximo a um acesso, em uma região RR, conforme figura 66b, mostra que o valor do menor custo até a célula objetivo é 16,0. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, a exemplo do que ocorre quando posicionado em A. É interessante notar que o robô poderia tentar um custo menor para o mesmo objetivo se acompanhasse à direita, em linha reta, a parede em que está localizado. Entretanto, a sexta célula é de RA e a seguinte volta a ser de RR. A definição 4.30 estabelece que não é permitido ao robô adicionar a contagem de uma célula ao custo, que tenha valor superior ao da célula anterior e, deste modo, esta célula fica cercada por células vizinhas de valor 1,0. Por este motivo o custo teve que ser ajustado, possuindo novo valor.

Para o caso em que o robô se encontra na posição C, que fica em uma região RR do ambiente, próxima ao acesso principal, conforme figura 66c, observa-se que o valor do menor custo até a célula objetivo é 6,0. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois a intensidade de passagem de agentes, segundo dados do simulador, é moderado.

A posição D no mapa está localizada em uma região RA, cercada por células de valor 1,0, não possuindo entre elas nenhuma célula de RL. Deste modo, o robô não faz a definição da célula objetivo (ou define a célula objetivo como sendo a mesma posição do robô), permanecendo parado em sua posição, pois a definição 4.30 estabelece que não é permitido ao robô adicionar a contagem de uma célula ao custo, que tenha valor superior ao da célula anterior. O tratamento para casos assim será realizado em trabalhos futuros.

5.7.2 Definição da trajetória do robô no supermercado

Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A*, descrita no capítulo 2, seção 2.3.

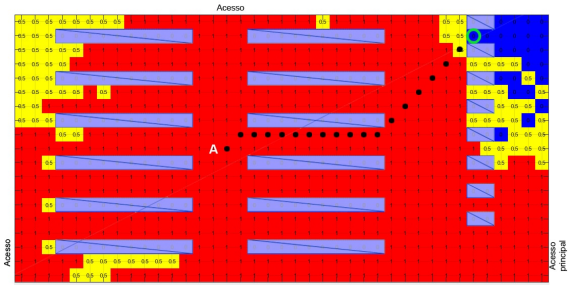
Para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. A figura 67 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*.

Na figura 67a, é possível perceber que o robô cria uma trajetória em linha reta e também na diagonal, até a célula objetivo. Esta trajetória seguida pelo robô, em uma situação de emergência em que ele esteja no apartamento, na posição A, é parcialmente *aceitável*, pois no início

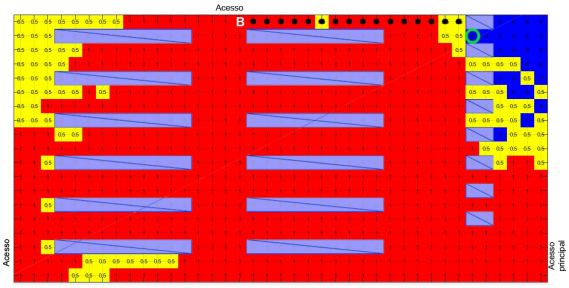
de seu movimento (linha reta) ele se desloca no sentido do movimento da multidão, próximo a um obstáculo (estante de produtos), por cerca de dez metros, o que daria cerca de 7,5 segundos até iniciar seu percurso na diagonal (com o robô mais veloz dos testes aqui simulados), este sim em sentido não favorável ao sentido de evacuação. Observa-se que a célula objetivo fica distante do da posição do robô, pelo fato de não haver célula de RL próximas a ele. Este problema poderia ser melhor solucionado se uma situação menos crítica fosse considerada, pois novas regiões RL apareceriam, diminuindo por consequência, as regiões RAs e regiões RRs. Entretanto, casos como este não fazem parte do escopo deste trabalho e serão abordados em trabalhos futuros.

Figura 67 – Trajetória seguida pelo robô utilizando a técnica A*, no supermercado.

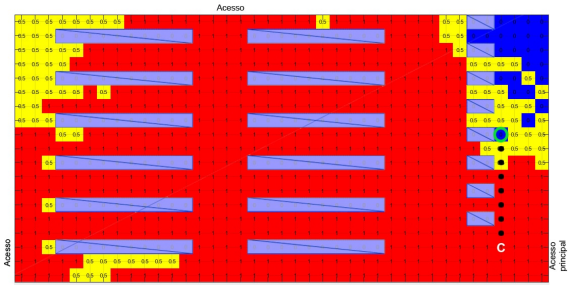
(a) Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo.



(b) Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



(c) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



Fonte: Autor.

Já na figura 67b, o robô cria uma trajetória predominantemente em linha reta, favorável ao sentido de evacuação, até a célula objetivo. Ele se encontra em uma região RR em um dos corredores do mapa e permanece quase todo o tempo em uma região deste tipo. Esta trajetória seguida pelo robô até a célula objetivo é *aceitável*, pois seu percurso desobstrui um acesso, além de permanecer próximo a um obstáculo por cerca de dez metros.

Na figura 67c, é possível perceber que o robô cria uma trajetória em linha reta, até a célula objetivo, passando por uma considerável região RR. A trajetória seguida pelo robô até a célula objetivo, é *aceitável*, pois desobstrui o principal acesso do ambiente. Entretanto, como o robô respeita o espaço social das pessoas, corre-se o risco dele ficar parado nesta posição em grande parte do tempo em que houver a evacuação, aguardando as pessoas saírem para, assim, desenvolver sua trajetória de navegação, tornando um obstáculo extra à multidão. Uma melhor alternativa seria o robô evacuar o ambiente juntamente com a multidão, porém casos assim serão tratados em trabalhos futuros.

No caso do robô na posição D do mapa, conforme mencionado na seção anterior (5.7.1), não há a definição da célula objetivo e, consequentemente, não há a definição da trajetória a ser seguida pelo robô. Neste caso, ele ficará parado nesta mesma posição do ambiente até que a situação se normalize ou até que seja removido manualmente ou por comando.

5.7.3 Simulação com os robôs no supermercado

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo. A figura 67 é a referência para esta análise.

O robô localizado na posição A precisou percorrer dezenove células para chegar ao objetivo. Navegou em linha reta e também na diagonal, passando por dezessete células de risco (RR), por uma célula de atenção (RA) e parando na célula livre (RL). Sendo assim, a distância total percorrida pelo robô até a célula objetivo é de 22,28 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa, levaria aproximadamente 16,6 segundos para o êxito de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1 m/s, levaria 22,28 segundos, enquanto que o Savioke Relay, cuja velocidade é de cerca de 0,7 m/s, levaria cerca de 31,8 segundos.

Para a situação em que o robô esteja na posição B do mapa, foi preciso percorrer dezessete células para chegar ao objetivo. Para isso, ele navegou predominantemente em linha reta, passando por treze células de risco (RR), três células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 17,41 metros. Assim, o robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 13 segundos para o êxito de sua

operação, enquanto que o Panasonic Signage HOSPI levaria 17,41 segundos e o Savioke Relay levaria cerca de 24,9 segundos para alcançar o objetivo.

Considerando a situação em que o robô esteja na posição C do mapa, foi preciso percorrer oito células, em linha reta, para chegar ao objetivo. Ele passou por cinco células de risco (RR), duas células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 8,0 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 6,0 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 8,0 segundos e o Savioke Relay levaria cerca de 11,4 segundos para alcançar o objetivo.

Por fim, para a situação em que o robô esteja na posição D do mapa, não há trajetória a ser percorrida pelo robô pela impossibilidade de definição da célula objetivo. Entretanto, a célula objetivo pode, nesta situação, ser a mesma que a célula original do robô.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

Este tipo de ambiente possui poucas regiões RLs para onde o robô possa estacionar, além do fato delas se concentrarem em apenas uma área (canto superior direito do mapa). Deste modo, dependendo da posição do robô, ele necessitará percorrer um longo caminho até alcançar o objetivo, sendo que poderia estacionar em regiões seguras mais próximas a ele.

A falta de outras células de RL ocorre porque o escopo deste trabalho prevê a classificação das células com base em dados do simulador para os cenários críticos, onde as áreas sem obstáculos possuem agentes a cada m^2.

Considerar cenários menos críticos para este tipo de ambiente faria com que novas células de RL surgissem no mapa, porém, em consequência disso, muitas células de regiões RRs (status 1,0) poderiam deixar de existir, podendo tornar as possibilidades de acidentes mais reais. Sendo assim, este tipo de situação será tratado em trabalhos futuros.

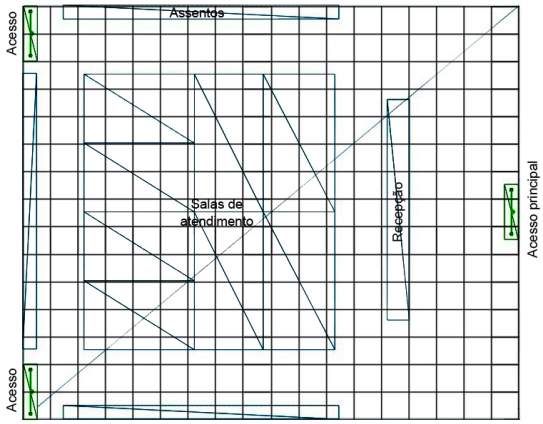
A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para o ambiente de um posto de saúde.

5.8 POSTO DE SAÚDE

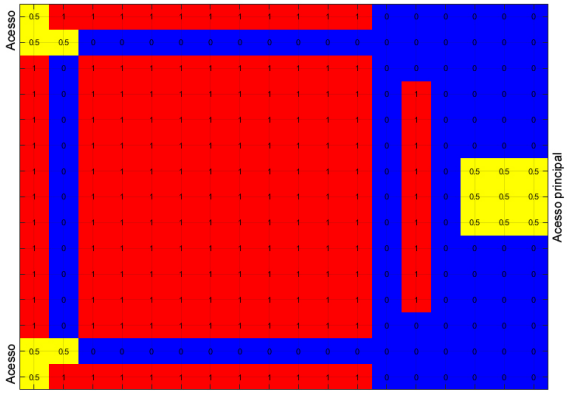
Este é um ambiente que simula um posto de saúde, onde os obstáculos são, basicamente, as paredes das salas e os assentos (bancos) que ficam na área de espera. Possui 15 x 18 m e uma área livre de 158 m^2 para movimentação do robô e das pessoas. Foi considerado que a multidão, de 158 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local e, no momento de uma emergência, ela tem somente uma opção de acesso do ambiente fechado. A figura 68 apresenta os mapas deste ambiente.

Figura 68 – Exemplos de mapas de um posto de saúde.

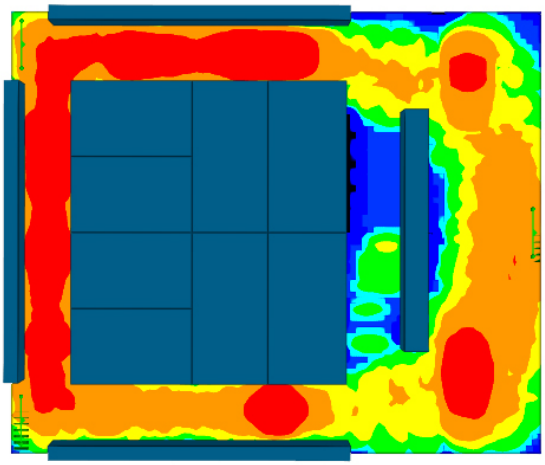
(a) Mapa decomposto em células.



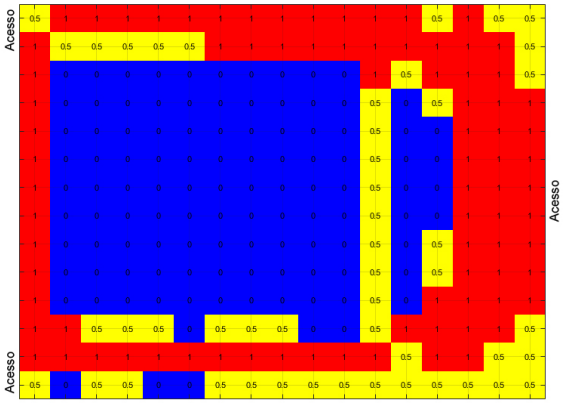
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.



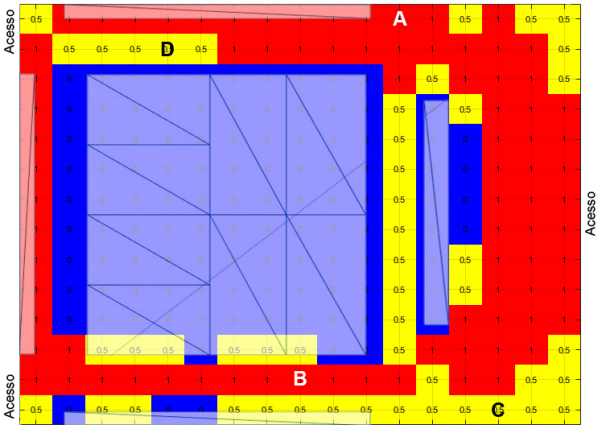
Fonte: Autor.

A simulação da figura 68c gerou um arquivo com 3793 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da

simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 9 segundos.

O MI deste ambiente, representado na figura 69, mostra quatro letras (A, B, C e D) que representam posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência. Para melhor visualização do cenário, foi realizada a sobreposição da estrutura que representa o mapa, no MI. Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

Figura 69 – MI do posto de saúde com sobreposição da representação do mapa.



Fonte: Autor.

As etapas a seguir mostrarão as definições das células objetivo para cada posição do robô, bem como a trajetória que o robô seguirá utilizando o algoritmo A*. Na sequência, serão apresentados os resultados de uma simulação com características de robôs reais relacionados às definições anteriores.

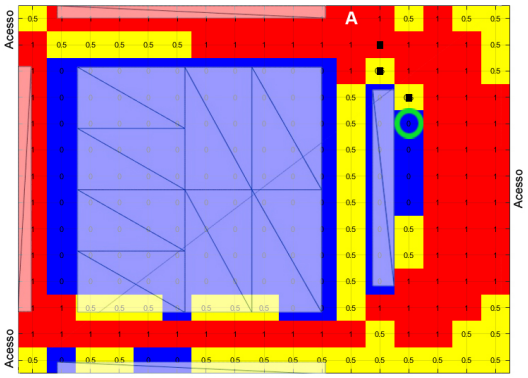
5.8.1 Definição da célula objetivo no posto de saúde

Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 69. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

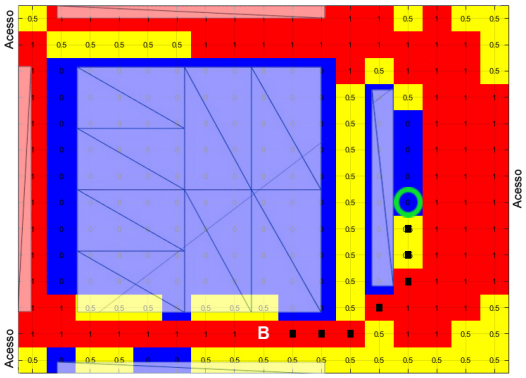
A figura 70 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A, B, C e D, no momento da emergência.

Figura 70 – Células objetivo em cada posição do robô no posto de saúde.

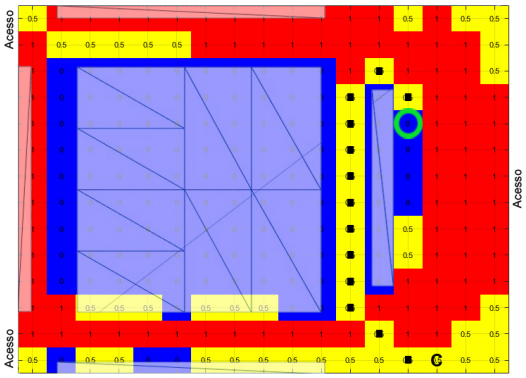
(a) Célula objetivo do robô posicionado em A.



(b) Célula objetivo do robô posicionado em B.



(c) Célula objetivo do robô posicionado em C.



Fonte: Autor.

Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

Para o caso em que o robô se encontra na posição A, que fica junto a uma parede do ambiente, em uma região RR, conforme figura 70a, é possível verificar que o valor do menor custo é 2,0. A célula com um círculo verde representa o objetivo. Esta região é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois próxima a um obstáculo geralmente fixo (balcão de atendimento).

O exemplo em que o robô se encontra na posição B, que fica em um corredor estreito do ambiente, em uma região RR, conforme figura 70b, mostra que o valor do menor custo até a célula objetivo é 6,0. A região escolhida para a célula objetivo é *aceitável* como área

de segurança para o posicionamento do robô quando ocorre uma situação de emergência, a exemplo do que ocorre quando posicionado em A.

Para o caso em que o robô se encontra na posição C, que fica em uma região RA do ambiente, junto a uma parede, conforme figura 70c, observa-se que o valor do menor custo até a célula objetivo é 6,5. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, a exemplo do que acontece quando o robô está posicionado em A. Próximo a posição C, há uma célula de RL mais próxima que a definida como objetivo, porém não é possível defini-la como tal, uma vez que para isso deveria passar de uma célula de RA para uma célula de RR em algum momento.

A posição D no mapa está localizada em uma região RA, cercada por células de valor 1,0, não possuindo entre elas nenhuma célula de RL. Deste modo, o robô não faz a definição da célula objetivo, permanecendo parado em sua posição, pois a definição 4.30 estabelece que não é permitido ao robô adicionar a contagem de uma célula ao custo, que tenha valor superior ao da célula anterior. O tratamento para casos assim será realizado em trabalhos futuros.

5.8.2 Definição da trajetória do robô no posto de saúde

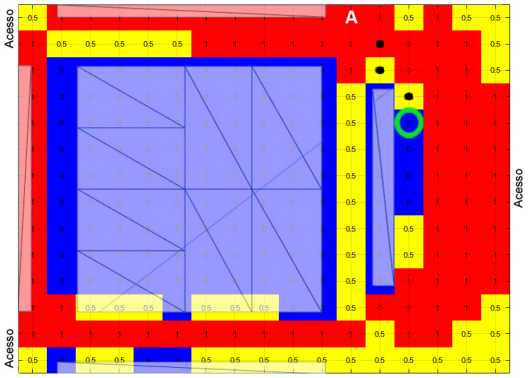
Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A*, descrita no capítulo 2, seção 2.3.

Para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. A figura 71 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*.

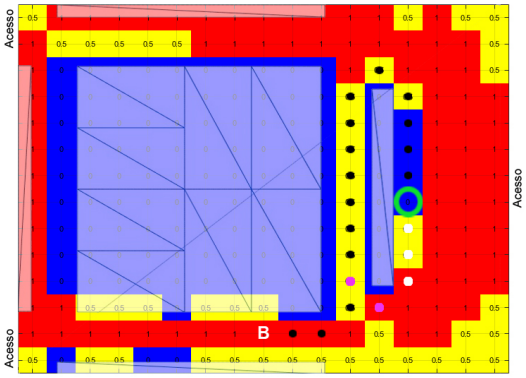
Na figura 71a, é possível perceber que o robô cria uma trajetória curta até a célula objetivo. Esta trajetória seguida pelo robô, em uma situação de emergência em que ele esteja no posto de saúde, na posição A, é considerada *não aceitável*, pois ele está próximo a uma parede e seu deslocamento até a célula objetivo atravessa uma região por onde há um sentido de movimento de evacuação. A melhor solução para este tipo de situação seria o robô permanecer parado na sua posição (A). Entretanto, casos como este não fazem parte do escopo deste trabalho e serão abordados em trabalhos futuros.

Figura 71 – Trajetória seguida pelo robô utilizando a técnica A*, no posto de saúde.

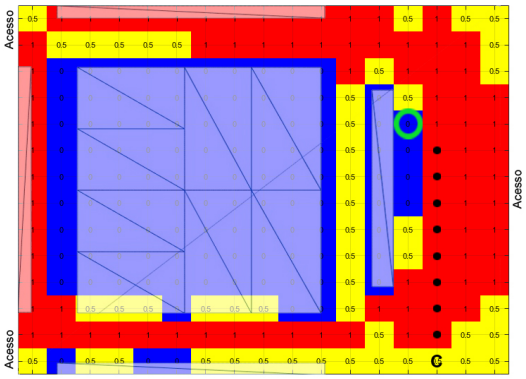
(a) Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo.



(b) Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



(c) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



Fonte: Autor.

Já na figura 71b, o robô cria uma trajetória predominantemente em linha reta, favorável ao sentido de movimento da multidão, até a célula objetivo (marcações na cor preta). Ele se encontra em uma região RR em um dos corredores do mapa e seu trajeto é realizado entre paredes e o balcão de atendimento do ambiente, não apresentando riscos de acidentes. Esta trajetória seguida pelo robô até a célula objetivo é *aceitável*. Neste exemplo específico, tomando as três primeiras marcações na cor preta, percebemos que a quarta marcação poderia ser em qualquer uma das duas direções indicadas pela marcação na cor lilás. Entretanto, aquela que dá sentido às marcações na cor branca (que faria com que o robô desenvolvesse um caminho completo mais curto até o objetivo), possui maior uma distância em linha reta (heurística admissível (RUSSELL; NORVIG, 2010)), fazendo com que o robô decida pelo caminho indicado pelas marcações em preto, segundo o algoritmo A*.

Na figura 71c, é possível perceber que o robô cria uma trajetória em linha reta, até a célula objetivo, passando por uma considerável região RA. A trajetória seguida pelo robô

até a célula objetivo, é considerada *não aceitável*, pois ele está próximo a uma parede e seu deslocamento até a célula objetivo atravessa uma região por onde há um sentido de evacuação, a exemplo do que acontece quando está na posição A do mapa. A melhor solução para este tipo de situação seria o robô permanecer parado na sua posição (C), pois está localizado em uma área onde não há grande intensidade de passagem da multidão na emergência, segundo dados do simulador. Entretanto, casos como este não fazem parte do escopo deste trabalho, uma vez que o robô deveria conhecer o fluxo em cada célula do ambiente fechado. Deste modo, este assunto será abordado em trabalhos futuros.

No caso do robô na posição D do mapa, conforme mencionado na seção anterior (5.8.1), não há a definição da célula objetivo e, consequentemente, não há a definição da trajetória a ser seguida pelo robô. Neste caso, ele ficará parado nesta mesma posição do ambiente até que a situação se normalize ou até que seja removido manualmente ou por comando.

5.8.3 Simulação com os robôs no posto de saúde

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo. A figura 71 é a referência para esta análise.

O robô localizado na posição A precisou percorrer quatro células para chegar ao objetivo. Navegou por uma célula de risco (RR), por duas células de atenção (RA) e parando na célula livre (RL). Sendo assim, a distância total percorrida pelo robô até a célula objetivo é de 4,82 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa, levaria aproximadamente 3,6 segundos para o êxito de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1 m/s, levaria 4,82 segundos, enquanto que o Savioke Relay, cuja velocidade é de cerca de 0,7 m/s, levaria cerca de 6,9 segundos.

Para a situação em que o robô esteja na posição B do mapa, foi preciso percorrer dezessete células para chegar ao objetivo. Para isso, ele navegou predominantemente em linha reta, passando por duas células de risco (RR), onze células de atenção (RA) e por quatro células livres (RL). A distância total percorrida pelo robô é de 18,23 metros. Assim, o robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 13,6 segundos para o êxito de

sua operação, enquanto que o Panasonic Signage HOSPI levaria 18,23 segundos e o Savioke Relay levaria cerca de 26 segundos para alcançar o objetivo. Para efeito de comparação, se hipoteticamente outro planejador de trajetórias fizesse com que o robô seguisse o trajeto indicado na cor branca, ele navegaria 8,82 metros até a célula objetivo e, deste modo, o robô Knightscope K5 levaria cerca de 6,6 segundos, o Panasonic Signage HOSPI levaria cerca 8,82, enquanto que o Savioke Relay levaria aproximadamente de 12,6 segundos para alcançarem o objetivo. Isto significa que cada robô economizaria mais da metade do tempo para chegar ao seu destino (sem considerar, evidentemente, desvios de obstáculos, tempo que poderia permanecer parado respeitando custos sociais, entre outros). Entretanto, a utilização de outros planejadores de trajetória não faz parte do escopo deste trabalho.

Considerando a situação em que o robô esteja na posição C do mapa, foi preciso percorrer nove células, em linha reta, para chegar ao objetivo. Ele passou por oito células de risco (RR), parando na célula livre (RL). A distância total percorrida pelo robô é de 9,41 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 7,0 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 9,41 segundos e o Savioke Relay levaria cerca de 13,44 segundos para alcançar o objetivo.

Por fim, para a situação em que o robô esteja na posição D do mapa, não há trajetória a ser percorrida pelo robô pela impossibilidade de definição da célula objetivo.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação, desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

A próxima seção irá apresentar as discussões referentes à definição da célula objetivo, definição da trajetória e simulação com características de robôs reais para o ambiente de um hotel.

5.9 HOTEL

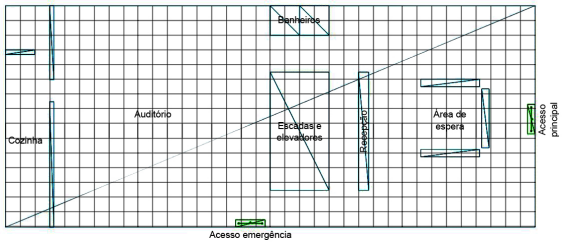
Este é um ambiente que simula o piso térreo de um hotel, onde os obstáculos são, basicamente, as paredes e assentos de espera. Possui 15 x 36 m e uma área livre de 454 m^2 para movimentação do robô e das pessoas. A figura 72 apresenta os mapas deste ambiente.

Foi considerado que a multidão, de 454 agentes (conforme apresentado na tabela 4), ocupa toda a área deste local e, no momento de uma emergência, ela tem a opção de abandonar o ambiente por qualquer um dos dois acessos, porém irá procurar o mais próximo, com base nos parâmetros de configuração do simulador. Caso o acesso mais próximo esteja congestionado, os agentes podem procurar evacuar o ambiente fechado por outro acesso.

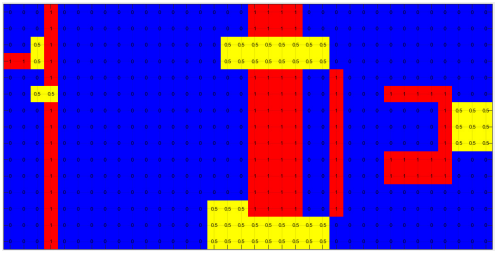
A simulação da figura 72c gerou um arquivo com 16958 linhas que estão relacionadas às marcações de todos os agentes nas coordenadas (x,y) do ambiente, contadas as cada segundo da simulação. O tempo simulado para que toda a multidão abandone este ambiente fechado, para uma situação real, foi de aproximadamente 15 segundos.

Figura 72 – Exemplo de mapa do piso térreo de um hotel.

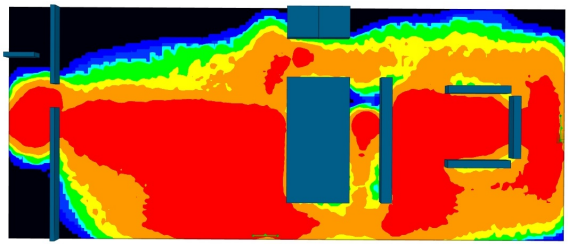
(a) Mapa decomposto em células.



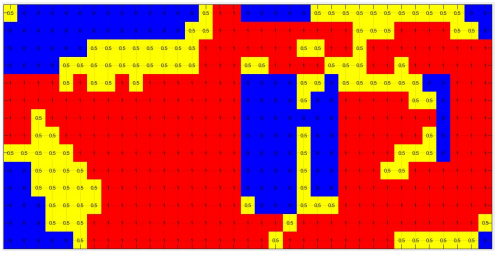
(b) Mapa do Ambiente (MA).



(c) Mapa de Intensidade do simulador.



(d) Mapa de Intensidade (MI) gerado pelo robô.

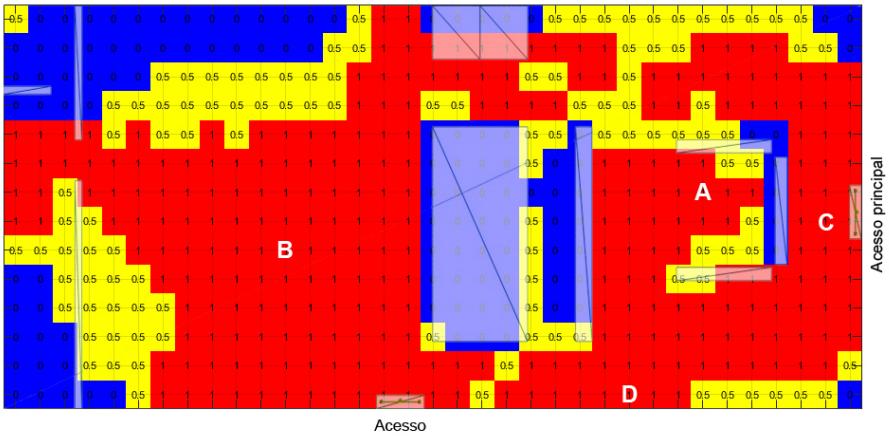


Fonte: Autor.

O MI deste ambiente, representado na figura 73, mostra quatro letras (A, B, C e D) que representam posições hipotéticas distintas onde o robô pode estar posicionado no momento em que recebe a informação de emergência.

Para melhor visualização do cenário, foi realizada a sobreposição da estrutura que representa o mapa, no MI. Para cada situação e para efeito de testes, foram definidas as células objetivo e, a partir disso, definida também a trajetória que o robô seguiria usando o algoritmo de planejamento A*.

Figura 73 – MI do hotel com sobreposição da representação do mapa.



Fonte: Autor.

As etapas a seguir mostrarão as definições das células objetivo para cada posição do robô, bem como a trajetória que o robô seguirá utilizando o algoritmo A*. Na sequência, serão apresentados os resultados de uma simulação com características de robôs reais relacionados às definições anteriores.

5.9.1 Definição da célula objetivo no hotel

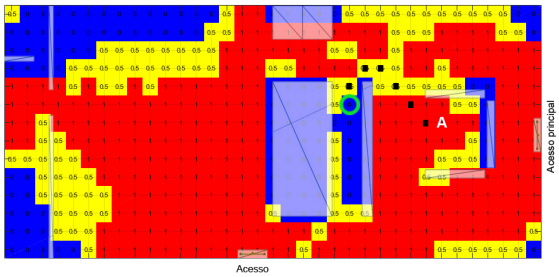
Nesta etapa, será mostrado o processo para a definição da célula objetivo para cada um dos casos de posicionamento do robô apresentados na figura 73. Os critérios para a escolha seguem as definições 4.29 e 4.30 apresentadas anteriormente na subseção 4.5.1, sobre custos e regra de contagem dos valores das células. Isto é possível por meio da definição 4.5 que diz respeito à classificação celular.

A figura 74 apresenta as células objetivo, indicadas por um círculo verde, definidas pelo robô sob a perspectiva das posições A, B, C e D, no momento da emergência. Os símbolos quadrados na cor preta representam o menor custo para se alcançar o objetivo. A definição 4.31, que estabelece o sentido prioritário de caminho quando há custos iguais, foi observado.

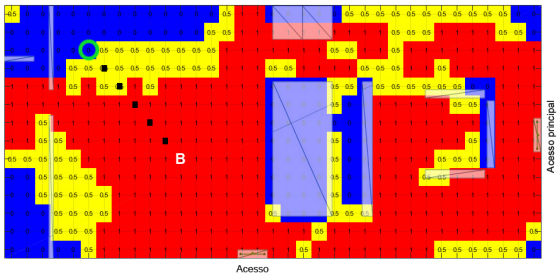
Para o caso em que o robô se encontra na posição A, que fica entre obstáculos (mobília) do ambiente, em uma região RR, conforme figura 74a, é possível verificar que o valor do menor custo é 4,0. A célula com um círculo verde representa o objetivo. Esta região é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois próxima a um obstáculo geralmente fixo (balcão de atendimento).

Figura 74 – Células objetivo em cada posição do robô no hotel.

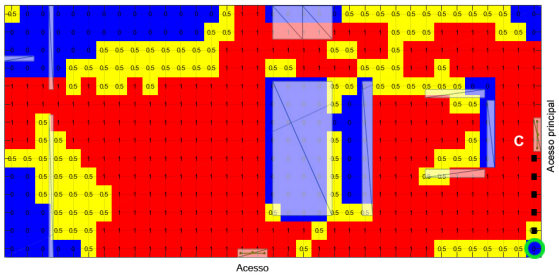
(a) Célula objetivo do robô posicionado em A.



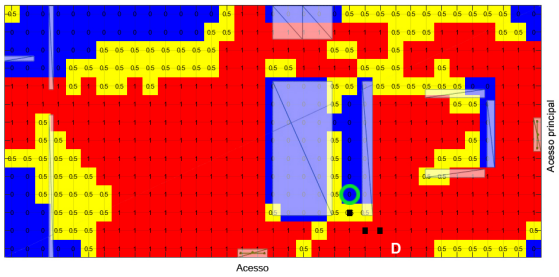
(b) Célula objetivo do robô posicionado em B.



(c) Célula objetivo do robô posicionado em C.



(d) Célula objetivo do robô posicionado em C.



Fonte: Autor.

O exemplo em que o robô se encontra na posição B, que fica em uma região central do espaço do auditório do ambiente, em uma região RR, conforme figura 74b, mostra que o valor do menor custo até a célula objetivo é 4,0. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois há várias células vizinhas livres, o que significa que a intensidade de passagem da multidão é baixo.

Para o caso em que o robô se encontra na posição C, que fica em uma região RR do ambiente, próxima ao acesso principal, conforme figura 74c, observa-se que o valor do menor custo até a célula objetivo é 4,5. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, pois ela está localizada no canto do ambiente.

Na posição D no mapa está localizada em uma região RR, junto a uma parede, conforme figura 74d, observa-se que o valor do menor custo até a célula objetivo é 2,5. A região escolhida para a célula objetivo é *aceitável* como área de segurança para o posicionamento do robô quando ocorre uma situação de emergência, a exemplo do que ocorre quando ele está posicionado em A.

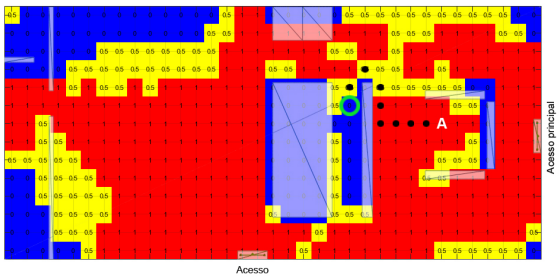
5.9.2 Definição da trajetória do robô no hotel

Nesta etapa, serão apresentados os resultados de definição do melhor trajeto para o robô seguir, com base na técnica de planejamento de trajetória A*, descrita no capítulo 2, seção 2.3.

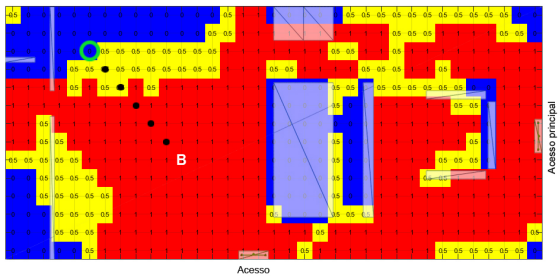
Para a definição do melhor caminho utilizando-se o algoritmo A*, o robô considera a medida que terá que percorrer do centro de uma célula ao centro de outra vizinha, a cada transição que fizer. A figura 75 apresenta as trajetórias para alcançar a célula objetivo por meio da técnica A*.

Figura 75 – Trajetória seguida pelo robô utilizando a técnica A*, no hotel.

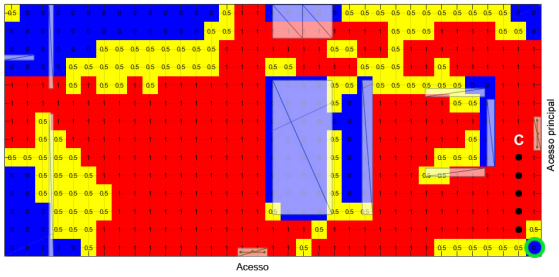
(a) Trajetória seguida pelo robô desde a sua posição (A) até a célula objetivo.



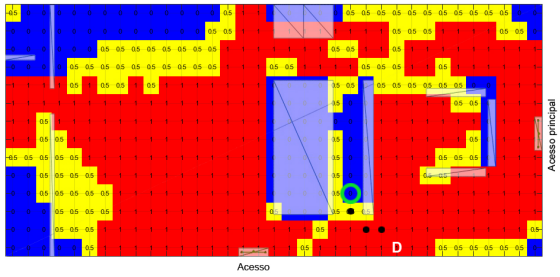
(b) Trajetória seguida pelo robô desde a sua posição (B) até a célula objetivo.



(c) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



(d) Trajetória seguida pelo robô desde a sua posição (C) até a célula objetivo.



Fonte: Autor.

Na figura 75a, observa-se que o robô cria uma trajetória, utilizando A*, saindo de uma região cercada por obstáculos (mobília do ambiente) até a célula objetivo que está localizada atrás de outro obstáculo (balcão de atendimento). Esta trajetória seguida pelo robô, em uma situação de emergência em que ele esteja no hotel, na posição A, é *aceitável*. Entretanto, uma trajetória inicialmente na diagonal, passando por apenas uma célula de região RR até chegar em uma região RA, poderia ser mais eficiente, já que evitaria células de grande intensidade de passagem da multidão no percurso. Para isso, o robô deveria utilizar uma estratégia diferente de

planejar sua trajetória, ficando este tratamento para trabalhos futuros, pois não faz parte deste escopo.

Já na figura 75b, o robô cria uma trajetória na diagonal até a célula objetivo. Ele se encontra em uma região RR em um auditório amplo e seu trajeto é realizado contra o sentido de movimento da multidão. Esta trajetória seguida pelo robô até a célula objetivo é considerada *não aceitável*, pois haveria grandes possibilidades de ocorrência de acidentes. Abaixo, no sentido da evacuação, à esquerda, há uma célula livre, próxima a uma parede, que poderia ser utilizada como célula objetivo. Entretanto, seu deslocamento poderia fazê-lo chocar-se com a multidão que evacua o ambiente pelo acesso do auditório, vindo da esquerda deste ambiente. De qualquer maneira, como o robô está localizado em uma região de alta intensidade de passagem da multidão, é possível que em uma situação de emergência, e por respeitar o espaço social das pessoas, ele permaneça parado nesta posição até que grande parte das pessoas tenham evacuado o ambiente, podendo ele desenvolver sua trajetória sem causar acidentes. Casos assim serão abordados em trabalhos futuros, uma vez que os testes com robôs reais, em ambientes reais, deverão auxiliar na análise para a melhor solução deste problema.

Na figura 75c, é possível perceber que o robô, localizado próximo ao acesso principal, cria uma trajetória em linha reta, até a célula objetivo, passando por uma região RR. A trajetória seguida pelo robô até a célula objetivo, é inadequada, pois ele tenta passar por um local onde há grande parte da multidão tentando evacuar o ambiente. A melhor solução para este tipo de situação seria o robô deslocar-se apenas uma célula para a sua esquerda, ficando bem próximo a um obstáculo. Entretanto, esta célula possui valor 1,0, movimento este que não aconteceria, de acordo com a teoria proposta. Esta é mais uma situação que será abordada em trabalhos futuros.

Na figura 75d, o robô, localizado próximo a uma parede do ambiente, em uma região RR, cria uma trajetória inadequada até a célula objetivo, pois ele tenta atravessar uma zona por onde há grande parte da multidão tentando evacuar o ambiente. A melhor solução para este tipo de situação seria o robô movimentar-se para a sua direita, até chegar na célula de RL que fica em um dos cantos do mapa.

5.9.3 Simulação com os robôs no hotel

Nesta subseção serão analisados os resultados das simulações em relação à características de robôs reais. Os robôs escolhidos para as discussões são aqueles apresentados no início deste capítulo. A figura 75 é a referência para esta análise.

O robô localizado na posição A precisou percorrer nove células para chegar ao objetivo. Navegou por cinco células de risco (RR), por três células de atenção (RA) e parou na célula livre (RL). Sendo assim, a distância total percorrida pelo robô até a célula objetivo é de 9,82 metros. Dependendo da característica de velocidade de cada robô, o tempo de percurso será diferente.

O robô Knightscope K5, que possui velocidade aproximada de 1,34 m/s, para a situação em que ele esteja na posição A do mapa, levaria aproximadamente 7,3 segundos para o êxito de sua operação. Neste mesmo cenário, o Panasonic Signage HOSPI, com velocidade de 1 m/s, levaria 9,82 segundos, enquanto que o Savioke Relay, cuja velocidade é de cerca de 0,7 m/s, levaria cerca de 14 segundos.

Para a situação em que o robô esteja na posição B do mapa, foi preciso percorrer seis células para chegar ao objetivo. Para isso, ele navegou na diagonal do espaço, passando por três células de risco (RR), duas células de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 8,46 metros. Assim, o robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 6,3 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 8,46 segundos e o Savioke Relay levaria cerca de 12 segundos para alcançar o objetivo.

Considerando a situação em que o robô esteja na posição C do mapa, foi preciso percorrer seis células, predominantemente em linha reta, para chegar ao objetivo. Ele passou por cinco células de risco (RR), parando na célula livre (RL). A distância total percorrida pelo robô é de 6,41 metros. O robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 4,78 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 6,41 segundos e o Savioke Relay levaria cerca de 9,16 segundos para alcançar o objetivo.

Por fim, para a situação em que o robô esteja na posição D do mapa, foi preciso percorrer quatro células para chegar ao objetivo. Para isso, ele navegou por duas células de risco (RR), uma célula de atenção (RA), parando na célula livre (RL). A distância total percorrida pelo robô é de 4,82 metros. Assim, o robô Knightscope K5 para esta situação no mapa, levaria aproximadamente 3,6 segundos para o êxito de sua operação, enquanto que o Panasonic Signage HOSPI levaria 4,82 segundos e o Savioke Relay levaria cerca de 7 segundos para alcançar o objetivo.

É importante lembrar que não foram considerados os tempos que o robô aguarda a multidão passar (célula vizinha ocupada) ou ainda tempos relacionados à sua necessidade de rotação,

desvio de objetos, entre outros. Considerou-se aqui o tempo ininterrupto de percurso com base na capacidade de velocidade de cada robô.

5.10 ANÁLISE GERAL DOS RESULTADOS

Para a realização dos testes nos nove ambientes escolhidos, em oito deles foram definidas quatro posições aleatórias em que o robô pode estar no momento de emergência e em um deles, três posições aleatórias. Deste modo, trinta e cinco situações iniciais foram estudadas, tanto para a definição da célula objetivo quanto para a definição da trajetória seguida pelo robô, a partir da classificação celular obtida por meio do simulador de multidão.

No que diz respeito à definição da célula objetivo, das 35 situações, 31 (88,57%) são *aceitáveis* como célula escolhida como objetivo, pois são estabelecidas em regiões RL do mapa, onde a passagem das pessoas não é comprometida quando o robô estaciona nela. Além disso, 1 situação (2,86%) foi considerada *não aceitável* como definição da célula objetivo, pois foi definida exatamente no acesso, tornando o robô um obstáculo que impede a passagem das pessoas para a evacuação. Foram identificadas, ainda, 3 situações (8,57%) onde a escolha da célula objetivo é impossível, baseadas na definição 4.30 que estabelece que não é permitido ao robô adicionar a contagem de uma célula ao custo que tenha valor superior ao da célula anterior. Neste caso, o robô permanece estacionado na posição em que está localizado, o que não representa uma desvantagem, já que o robô, nas 3 situações, está em região RA e não atrapalha o sentido de movimento da multidão.

Nos resultados referentes à definição das trajetórias para cada uma das 32 situações (já que 3 delas não possuem célula objetivo definidas) usando o planejador de trajetória A*, com base nas definições das células objetivo, 18 delas (56,25%) são *aceitáveis* para que o robô siga no momento da emergência. Quanto às situações consideradas *não aceitáveis* foram registradas 13 delas (40,63%). As razões se baseiam no fato do robô seguir contra o sentido de movimento da multidão (6 casos), estar posicionado junto à uma parede (5 casos), obstruir acesso (1 caso) ou passar por um local onde há concentração de pessoas (1 caso). Uma situação é parcialmente *aceitável* (3,13%), pois grande parte do percurso inicial no sentido da célula objetivo é a favor do sentido de evacuação e o restante contra ele.

5.10.1 Casos de exceção

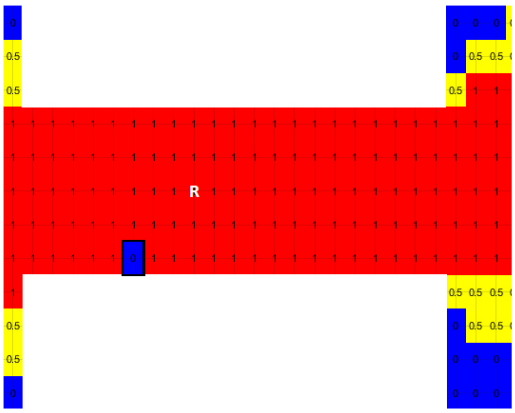
A teoria proposta neste trabalho, bem como os resultados dos experimentos realizados, permitiram identificar alguns casos em que a solução do problema exige considerar novas variáveis de estudo, por se tratarem de situações específicas.

Exemplos disso envolvem casos em que um determinado acesso do ambiente está comprometido em decorrência da emergência, situação apresentada anteriormente nas figuras 25b e 31, ou em casos que o robô se movimenta no sentido contrário à multidão (39, 49c, 53b, 57b, 67a, 75b), ou ainda em situações em que ele está junto à uma parede (71a, 71c) e, mesmo sem necessidade de qualquer deslocamento ele o faz, passando por uma região de alta intensidade. Um robô inserido neste ambiente deve evitar se deslocar no sentido contrário ao da multidão ou ainda atravessar sem necessidade por qualquer região RR. Entretanto, para que ele seja capaz de identificar o fluxo, cada célula do ambiente necessita possuir uma orientação deste fluxo proveniente dos dados coletados no simulador de multidão, porém isso nem sempre é provido por ele ou os dados do simulador nem sempre permitem a extração dessa informação. Deste modo, seria possível ao robô interpretar o sentido do fluxo contido nas células e definir seu objetivo em alguma célula que permitisse seu deslocamento em sentido favorável ao movimento das pessoas.

A questão do fluxo em cada célula seria importante também para solucionar problemas onde situações de emergência ocorrem em corredores estreitos. Neste caso, o robô não deve permanecer parado, mesmo que não haja células em regiões RL, pois impediria o deslocamento da multidão para o acesso.

Outro caso a ser considerado é o da figura 31, da seção 4.5.1, que trata da possibilidade de desqualificar um conjunto de células, dada uma situação de emergência que obstrua um dos acessos do ambiente fechado, para que o robô não se movimente contra o sentido da multidão. Entretanto, ao desqualificar células por este motivo, existe a possibilidade de algumas delas serem células de região RL que passarão a possuir classificação de região RR e motivarão o robô a se deslocar no mesmo sentido de movimento da multidão. A figura 76 mostra o exemplo de um corredor (o mesmo da figura anteriormente apresentada) que possui um robô (indicado pela letra R, em branco). Caso o acesso obstruído esteja à esquerda deste mapa, todas as células à esquerda do robô serão desqualificadas como célula objetivo, inclusive uma célula de RL próxima a ele (destacada com contorno na cor preta).

Figura 76 – Caso de exceção once um acesso está obstruído.



Fonte: Autor.

É possível observar que o caminho para a célula de RL destacada é mais curto que o caminho até qualquer uma das células RL à direita do robô, o que poderia indicar que em algumas situações é mais vantajoso o robô movimentar-se em um curto caminho contra o sentido de movimento da multidão, ao invés de movimentar-se um longo caminho no mesmo sentido dela. Entretanto, variáveis como largura do corredor, velocidade da multidão e também do robô, quantidade de pessoas, entre outras, tornam este um problema particular a ser explorado em trabalhos futuros.

Outro exemplo de situação de exceção a ser considerada ocorre quando há o entendimento de que uma boa alternativa para que o robô não atrapalhe a evacuação da multidão seria ele procurar estacionar próximo ao local do evento da emergência, especialmente se este local for um dos acessos, pois, deste modo, um sistema de segurança disponível no próprio robô seria capaz de alertar às pessoas de que tal acesso estaria obstruído para a saída. Entretanto, é preciso avaliar se o movimento do robô para chegar até o local do evento da emergência seria considerado de risco por atravessar por regiões RR ou ainda por ser contra o sentido de movimento da multidão.

O próximo capítulo irá apresentar as conclusões extraídas deste trabalho, além de descrever os principais assuntos de trabalhos futuros.

6 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho, apresentou-se uma teoria sobre a classificação celular em ambientes fechados com multidão, realizada por robô móvel autônomo terrestre, para definição de célula objetivo e planejamento de trajetória, a fim de contribuir para que o ele procure estacionar em regiões de pouca intensidade de passagem de pessoas no ambiente, em situações de emergência. O que determina se as regiões possuem baixa, moderada ou alta intensidade de passagem de pessoas é um simulador de multidão, incorporado no próprio robô, que armazena os dados referentes à quantidade de ocorrência de ocupação em cada célula durante todo o período de evacuação do ambiente. Células que estiveram muito ocupadas durante este período recebem classificação de alta intensidade, enquanto que aquelas que não tiveram passagem dos chamados agentes (pessoas, no simulador) recebem classificação de baixa intensidade.

As pesquisas realizadas durante a etapa de revisão bibliográfica mostraram que não existem contribuições neste sentido para o problema posicionamento do robô em local seguro para a multidão em situações de emergência, utilizando classificação celular com base em dados de simulador de multidão.

Os testes foram realizados em nove tipos de ambientes distintos (formato, tamanho, quantidade de obstáculos, entre outros), considerando situações críticas de emergência, isto é, a superlotação de cada um deles, onde há uma pessoa em cada metro quadrado, na maioria dos ambientes.

A construção dos mapas de intensidade (MIs) no simulador e também com os dados dele no MATLAB®, para todos os casos, apresentaram as similaridades desejadas, permitindo concluir que este processo de classificação celular retorna soluções eficientes para a estrutura base de determinação da célula objetivo realizada pelo robô, tornando o objetivo desta tese alcançado. O mapa de cores que representam estes mapas dão uma visão importante para identificar as regiões por nível de intensidade (baixa, moderada ou alta).

Nos mapas de intensidade de cada ambiente, foram escolhidas três ou quatro posições para serem feitas as análises do comportamento do robô no que diz respeito à definição da célula objetivo e à definição da trajetória do robô para o momento em que ocorre a emergência.

No que se refere à definição da célula objetivo, mais de 88% das situações são *aceitáveis*, já que o robô estacionado nesta células não atrapalha a evacuação da multidão do ambiente fechado. Este fato demonstra que a teoria proposta de classificação celular, por meio dos dados de um simulador de multidão, é eficaz.

Quanto à definição da trajetória para estes casos onde foi possível estabelecer uma célula objetivo, embora tenha retornado resultados favoráveis, superiores a 55%, ainda há a preocupação referente ao seu deslocamento. Entretanto, os problemas apresentados para resultados não eficientes de definição da trajetória revelam que grande parte deles ocorrem porque o robô navega contra o sentido de movimento da multidão na emergência, podendo provocar acidentes. Tais problemas podem ser solucionados se cada célula possuir orientação de fluxo, mas outros estudos devem ser realizados neste sentido, uma vez que variáveis como a variação do fluxo no tempo, a complexidade de se encontrar a célula objetivo, entre outras, estão além do que foi proposto no escopo deste trabalho. Outros problemas apresentados são referentes ao fato de que o robô já está posicionado junto à alguma parede do ambiente, não necessitando realizar movimentos que atrapalhariam a evacuação da multidão. Conclui-se que a teoria de classificação celular proposta é capaz de dar suporte aos resultados obtidos, tanto os favoráveis quanto aos não favoráveis, para os aspectos de definição de trajetória.

Os trabalhos futuros serão importantes para os ajustes necessários para tentar resolver os problemas apresentados acima.

Além disso, poderão auxiliar a resolver problemas referentes a situações como:

a) casos em que não foram encontradas células livres (RL) no mapa do ambiente após a coleta dos dados pelo simulador de multidão. A teoria deste trabalho prevê o posicionamento do robô somente em células de RL.
b) casos em que o robô está em uma célula de valor 0,5 ou 1,0 e na sua vizinhança há duas ou mais células de RL (0,0). Ele deverá decidir em qual delas irá estacionar, tendo como referência as consequências de sua escolha.
c) casos em que o robô define um objetivo, porém no percurso algum obstáculo ocupa aquela célula. Poderá haver situações em que ele deverá escolher nova célula objetivo e definir nova trajetória, porém fará isso durante a emergência.
d) casos em que a melhor escolha é uma célula de região RA (0.5) ao invés de uma célula de RL (0.0). Regiões próximas às paredes podem conter valores 0.5, enquanto que uma célula de RL faria com que o robô tivesse que navegar contra o sentido de evacuação ou ainda atravessar um corredor ou passagem da multidão.
e) casos em que o robô esteja em uma região RA, mas não há células de RL vizinhas a esta região. Nas ocorrências dos testes desta tese foi possível verificar que se o robô permanecer estacionado no local em que está, já que não é possível definir uma célula objetivo, ele não atrapalha a multidão. Entretanto, espera-se que uma

quantidade maior de testes focados nesta situação conduzam a conclusões mais precisas a este respeito.

f) casos em que o tamanho reduzido de cada célula do ambiente produz resultados mais significativos, como no exemplo do ambiente *apartamento* deste trabalho.

g) casos em que diminuir a quantidade de agentes por metro quadrado (redução da criticidade) produzirão resultados mais eficientes, apresentando maior número de células de RL, permitindo que o robô tenha maiores possibilidades de escolha da célula objetivo.

REFERÊNCIAS

ABDULKAREEM, Ademola et al. Development of a Programmable Mobile Robot. **International Journal of Applied Engineering Research**, v. 11, n. 4, p. 2314–2320, 2016. Disponível em <https://www.researchgate.net/publication/315706059_Development_of_a_Programmable_Mobile_Robot>. Acesso em 04 mar 2017.

ABU-MAHFOUZ, Adnan M; HANCKE, Gerhard P. Localised information fusion techniques for location discovery in wireless sensor networks. **International Journal of Sensor Networks**, Inderscience Publishers (IEL), v. 26, n. 1, p. 12–25, 2018.

AFZAAL, H.; ZAFAR, N. A. Robot-based forest fire detection and extinguishing model. In: 2016 2ND INTERNATIONAL CONFERENCE ON ROBOTICS e ARTIFICIAL INTELLIGENCE (ICRAI). PROCEEDINGS... Los Angeles, USA: [s.n.], nov. 2016. p. 112–117. Disponível em <https://ieeexplore.ieee.org/document/7791238>. Acesso em 04 mar 2017.

ALIZADEH, Rahim. A dynamic cellular automaton model for evacuation process with obstacles. **Safety Science**, Elsevier, v. 49, n. 2, p. 315–323, 2011.

ALMEIDA, Jefferson Silva et al. Localization System for Autonomous Mobile Robots Using Machine Learning Methods and Omnidirectional Sonar. **IEEE Latin America Transactions**, IEEE, v. 16, n. 2, p. 368–374, 2018.

ALSAAB, Ahmad. **Behavioural strategy for indoor mobile robot navigation in dynamic environments**. 2015. f. 204. thesis (PhD in Mechanical and Systems Engineering) – Newcastle University, United Kingdom, 2015.

ÁLVARES, Alberto J et al. A navigation and path planning system for the Nomad XR4000 mobile robot with remote web monitoring. In: ABCM SYMPOSIUM SERIES IN MECHATRONICS. PROCEEDINGS... Escola Politécnica da Universidade de São Paulo - São Paulo, SP: ABCM, nov. 2004. v. 1, p. 18–24. Disponível em <https//abcm.org.br/pages/abcm_symposium_series_in_mechatronics_vol_6>. Acesso em 13 ago 2018.

ALVES, Raulcezar; LINHARES, Brian Admas; SOUZA, Jefferson R. Autonomous Shopping Cart: A New Concept of Service Robot for Assisting Customers. In: 2018 LATIN AMERICAN ROBOTIC SYMPOSIUM, 2018 BRAZILIAN SYMPOSIUM ON ROBOTICS (SBR) e 2018 WORKSHOP ON ROBOTICS IN EDUCATION (WRE). PROCEEDINGS... João Pessoa, Brazil: LARS/SBR, nov. 2018. p. 451–456. Disponível em <https://www.researchgate.net/publication/329952049_Autonomous_Shopping_Cart_A_New_Concept_of_Service_Robot_for_Assisting_Customers>. Acesso em 03 fev 2018.

AMBOT. **Wheeled Platforms**. [S.l.]: AMBOT, 2017. Disponível em <http://www.ambot.com/ip-wheel.shtml>. Acesso em 28 abr 2017.

ANGELO, A. **A reference guide to new technology**. Boston, MA: Greenwood Press, 2007.

ANGUELOV, Dragomir et al. Detecting and modeling doors with mobile robots. In: ROBOTICS e AUTOMATION, 2004 IEEE INTERNATIONAL CONFERENCE ON. PROCEEDINGS... New Orleans, USA: [s.n.], abr. 2004. v. 4, 3777–3784 vol.4. Disponível em <https://ieeexplore.ieee.org/abstract/document/1308857>. Acesso em 03 fev. 2018.

ARGOS-SIM. **ARGoS**. [S.l.]: ARGoS, 2019. Disponível em <https://www.argos-sim.info>. Acesso em 15 jan 2019.

AROOR, Anoop; EPSTEIN, Susan L; KORPAN, Raj. MengeROS: a Crowd Simulation Tool for Autonomous Robot Navigation. **arXiv preprint arXiv:1801.08823**, 2018.

ASIMOV, Isaac. **The Three Laws of Robotics - in I, Robot**. New York, USA: Signet Books, 1956.

AZZOLINI, Alisson Gusatti. **Aprendizado por reforço em modelos probabilísticos de redes imunológicas para robótica autônoma**. 2011. f. 170. dissertation (Mestrado em Engenharia Elétrica) – Universidade Estadual de Campinas, Campinas, 2011.

BAILEY, Tim. **Mobile robot localisation and mapping in extensive outdoor environments**. 2002. f. 202. thesis (Doctoral program in philosophy) – University of Sydney, Sydney, 2002.

BENICASA, Alcides. Navegação autônoma de robôs baseada em técnicas de mapeamento e aprendizagem de máquina. **Revista Brasileira de Computação Aplicada**, v. 4, n. 1, p. 102–111, abr 2012.

BERGBREITER, S.; PISTER, K. S. J. CotsBots: an off-the-shelf platform for distributed robotics. In: 2003 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS e SYSTEMS (IROS 2003). PROCEEDINGS... Las Vegas, USA: [s.n.], out. 2003. p. 1632–1637. Disponível em <https://ieeexplore.ieee.org/document/1248878>. Acesso em 03 fev 2018.

BERLEKAMP, E.; CONWAY, J.; GUY, R. **Winning Ways for your Mathematical Plays**. Wellesley, MA: Academic, 1982. v. 2, p. 212.

BIGHETI, Jeferson André. **Mestrado em Engenharia Elétrica**. 2011. f. 110. dissertation () – Universidade Federal Paulista, Bauru, 2011.

BILANCIA, Louis. F. Safe product design, forensic engineering, and Asimov's Laws of Robotics. In: 2014 IEEE SYMPOSIUM ON PRODUCT COMPLIANCE ENGINEERING (ISPCE). PROCEEDINGS... San Jose, USA: [s.n.], mai. 2014. p. 17–24. Disponível em <https://ieeexplore.ieee.org/document/6841995/>. Acesso em 03 fev 2018.

BIZONY, P. Focus: THE 3 Laws of Robotics. **Engineering Technology**, v. 10, n. 6, p. 44–45, jul. 2015. Disponível em <https://ieeexplore.ieee.org/document/7636746/>. Acesso em 26 jan. 2018.

BOAS, Elias Ramos Vilas. **Mapeamento e Localização Simultânea de Ambientes Dinâmicos Aplicados na Navegação de Veículo Autônomo Inteligente**. 2011. f. 80.

dissertation (Programa de Pós-Graduação em Engenharia Elétrica) – Universidade Federal de Itajubá, Itajubá, 2011.

BOHÓRQUEZ, Néstor et al. Safe navigation strategies for a biped robot walking in a crowd. In: 2016 IEEE-RAS 16TH INTERNATIONAL CONFERENCE ON HUMANOID ROBOTS (HUMANOIDS). PROCEEDINGS... Cancun, Mexico: [s.n.], nov. 2016. p. 379–386. Disponível em <https://ieeexplore.ieee.org/document/7803304/>. Acesso em 11 dez 2018.

BÖLÖNI, Ladislau; KHAN, Saad Ahmad; ARIF, Saad. Robots in Crowds — Being Useful While Staying Out of Trouble. **Intelligent Robotic Systems: Papers from the AAAI 2013 Workshop**, v. 2, n. 3, p. 2–7, jan. 2013. Disponível em <https://www.researchgate.net/publication/287852884_Robots_in_crowds_being_useful_while_staying_out_of_trouble>. Acesso em 23 fev 2018.

BOUKAS, Evangelos et al. Robot Guided Crowd Evacuation. **IEEE Transactions on Automation Science and Engineering**, v. 12, n. 2, p. 739–751, Abr. 2015. Disponível em <https://ieeexplore.ieee.org/document/6824875>. Acesso em 07 jun. 2017.

BRAÜNL, Thomas. **Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems**. 3. ed. Crowley, Perth, Australia: Crawley: Springer, 2008. p. 560.

BRESSON, Guillaume et al. Simultaneous localization and mapping: A survey of current trends in autonomous driving. **IEEE Transactions on Intelligent Vehicles**, v. 20, p. 1–1, 2017.

CARINENA, Purificacion et al. Landmark detection in mobile robotics using fuzzy temporal rules. **IEEE Transactions on Fuzzy Systems**, IEEE, v. 12, n. 4, p. 423–435, 2004.

CARNEIRO, Lilian De Oliveira et al. Crowd Evacuation using Cellular Automata: Simulation in a Soccer Stadium. In: IEEE VIRTUAL e AUGMENTED REALITY (SVR), 2013 XV SYMPOSIUM ON. PROCEEDINGS... Cuiabá, Brazil: [s.n.], mai. 2013. p. 240–243. Disponível em <https://ieeexplore.ieee.org/abstract/document/6655789>. Acesso em 03 fev 2018.

CARNEIRO, Lilian Oliveira. **Simulação de evacuação de multidão por autômato celular - Estudo de caso em um estádio de futebol**. 2012. f. 73. dissertation (Mestrado em Ciência da Computação) – Universidade Federal do Ceará, Fortaleza, 2012.

CARRERA, Tânia Ferreira. **Movimentos reativos no robô turtlebot utilizando o kinect**. 2013. f. 93. dissertation (Mestrado emEngenharia Industrial) – Instituto Politécnico de Bragança, Bragança, 2013.

CARRILLO, Henry et al. Autonomous robotic exploration using occupancy grid maps and graph slam based on shannon and rényi entropy. In: ROBOTICS e AUTOMATION (ICRA), 2015 IEEE INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Seattle, USA: [s.n.], mai. 2015. p. 487–494. Disponível em <https://ieeexplore.ieee.org/abstract/document/7139224>. Acesso em 12 fev 2019.

CASSOL, Vinícius Jurinic et al. Crowdsim: Uma ferramenta desenvolvida para simulação de multidões. **I Workshop de Simulação Militar-SBGames**, p. 1–4, 2012.

CHARALAMPOUS, Konstantinos; EMMANOUILIDISY, Christos; GASTERATOS, Antonios. Social Mapping on RGB-D Scenes. In: 2014 IEEE INTERNATIONAL CONFERENCE ON IMAGING SYSTEMS e TECHNIQUES (IST). PROCEEDINGS... Bejing, China: [s.n.], out. 2014. p. 398–403. Disponível em <https://www.researchgate.net/publication/288280189_Social_mapping_on_RGB-D_scenes>. Acesso em 03 fev 2018.

CHATTERJEE, Ishani. **A Low-Cost, Human-Inspired Perception Approach for Dense Moving Crowd Navigation**. 2016. f. 43. dissertation (Master of Science in Robotics) – Carnegie Mellon University, Pittsburgh, 2016.

CHEN, Kui et al. Fundamental Development of a Virtual Reality Simulator for Four-Arm Disaster Rescue Robot OCTOPUS. In: 2016 IEEE INTERNATIONAL CONFERENCE ON ADVANCED INTELLIGENT MECHATRONICS (AIM). PROCEEDINGS... Alberta, Canada: [s.n.], jul. 2016. p. 721–726. Disponível em <https://ieeexplore.ieee.org/abstract/document/7576853>. Acesso em 03 fev 2018.

CHENGA, Jiantong et al. Robust linear pose graph-based SLAM. **Robotics and Autonomous Systems**, v. 72, n. 5, p. 71–82, out. 2015. Disponível em <https://www.researchgate.net/publication/275897659_Robust_linear_pose_graph-based_SLAM>. Acesso em 24 abr. 2018.

CHONG, T. J. et al. Sensor Technologies and Simultaneous Localization and Mapping (SLAM). **IEEE International Symposium on Robotics and Intelligent Sensors (IRIS 2015)**, v. 76, p. 174–179, dez. 2015. Disponível em <https://core.ac.uk/download/pdf/82329715.pdf>. Acesso em 04 abr. 2018.

CHOTIKARN, Ponlachart et al. Low cost submarine robot. **Songklanakarin Journal of Science and Technology**, v. 32, n. 5, p. 513–518, 2010.

CHUNG, Michael Jae-Yoon; CAKMAK, Maya. Exploring the use of robots for gathering customer feedback in the hospitality industry. In: HRI WORKSHOP ON SOCIAL ROBOTS IN THE WILD. PROCEEDINGS... Chicago, EUA: [s.n.], mar. 2018. p. 1–5. Disponível em <http://socialrobotsinthewild.org/wp-content/uploads/2018/02/HRI-SRW_2018_paper_7.pdf>. Acesso em 03 abr 2019.

CNN. **Brazil nightclub fire: 'Like a war zone' with bodies piled**. [S.l.]: CNN, 2013. Disponível em <https://edition.cnn.com/2013/01/28/world/americas/brazil-nightclub-fire/>. Acesso em 15 jan 2019.

CONDE, Miguel Heredia. **Compressive Sensing for the Photonic Mixer Device: Fundamentals, Methods and Results**. Heidelberg, Germany: Springer, 2017.

COOK, Albert M.; HUSSEY, S. M. **Assistive technologies: principles and practice**. 3. ed. St. Louis: Mosby, 2002.

COPPELIAROBOTICS. **V-Rep**. [S.l.]: Coppelia Robotics, 2019. Disponível em <http://www.coppeliarobotics.com>. Acesso em 15 jan 2019.

CÓRDOVA, Fabricio Alexander Lucero. **Detección de robo/abandono de objetos en interiores utilizando cámaras de profundidad**. 2012. f. 131. thesis (PhD en Tecnología Electrónica y de las Comunicaciones) – Universidad Autónoma de Madrid, Madrid, 2012.

CORKE, Peter. **Robotics, vision and control: fundamental algorithms in MATLAB**. Heidelberg, Germany: Springer, 2011. v. 73.

CORTÉS, Alberto Sanfeliu. **Millora de braç mecànic per a robots humanoides en àrees urbanes**. 2014. f. 74. dissertation (Maîtrise en Enginyeria Industrial) – Universitat Politècnica de Catalunya, Barcelona, 2014.

CURTIS, Sean; BEST, Andrew; MANOCHA, Dinesh. Menge: A modular framework for simulating crowd movement. **Collective Dynamics**, v. 1, p. 1–40, 2016.

CYBERBOTICS. **Webots User Guide 8.5.3: Using the Pioneer 3-AT and Pioneer 3-DX robots Pioneer 3-AT**. [S.l.]: Cyberbotics, 2017. Disponível em <https://www.cyberbotics.com/doc/guide/using-the-pioneer-3-at-and-pioneer-3-dx-robots>. Acesso em 23 jan 2017.

_____. **Webots, Open Source Robot Simulator**. [S.l.]: Cyberbotics, 2019. Disponível em <https://cyberbotics.com/>. Acesso em 15 jan 2019.

CZERNIAK, Jacek M et al. A Cellular Automata-Based Simulation Tool for Real Fire Accident Prevention. **Mathematical Problems in Engineering**, Hindawi, v. 2018, 2018.

DADOVÁ, Jana. **Cellular Automata Approach for Crowd Simulation**. 2012. f. 78. thesis (PhD in Computer Graphics and Geometry) – Comenius University, Bratislava, 2012.

DANG, Chien Van et al. Implementation of a Refusable Human-Robot Interaction Task with Humanoid Robot by Connecting Soar and ROS. **Journal of Korea Robotics Society**, Korea Robotics Society, v. 12, p. 55–64, 1 2017. Disponível em: <http://www.jkros.org/journalarticle.php?code=50702>.

DIRECTINDUSTRY. **Sensor de temperatura**. [S.l.]: Direct industry, 2017. Disponível em <http://www.directindustry.com/prod/calex-electronics-limited/product-9294-31101.html>. Acesso em 15 mar 2017.

_____. **Sensor de visão Lector®65x series**. [S.l.]: Direct Industry, 2017. Disponível em <http://www.directindustry.com/pt/prod/sick/product-894-1536081.html>. Acesso em 01 maio 2017.

_____. **Sensor laser**. [S.l.]: Direct industry, 2017. Disponível em <http://www.directindustry.com/prod/balluff/product-7415-1681838.html?utm_source=ProductDetail&utm_medium=Web&utm_content=SimilarProduct&utm_campaign=CA>. Acesso em 15 mar 2017.

DOAN, N. T.; HAYASHI, T.; YAMAKITA, M. High speed running of flat foot biped robot with inerter using SLIP model. In: 2015 IEEE INTERNATIONAL CONFERENCE ON ADVANCED INTELLIGENT MECHATRONICS (AIM). PROCEEDINGS... Busan, Korea:

[s.n.], jul. 2015. p. 110–115. Disponível em <https://ieeexplore.ieee.org/document/7222517>. Acesso em 13 maio 2017.

DOBRA, A. General classification of robots. Size criteria. In: 2014 23RD INTERNATIONAL CONFERENCE ON ROBOTICS IN ALPE-ADRIA-DANUBE REGION (RAAD). PROCEEDINGS... Castle, Slovakia: [s.n.], set. 2014. p. 1–6. Disponível em <https://ieeexplore.ieee.org/document/7002249>. Acesso em 18 dez 2018.

DUBOIS, Amandine; DIB, Abdallah; CHARPILLET, François. Using hmms for discriminating mobile from static objects in a 3d occupancy grid. In: TOOLS WITH ARTIFICIAL INTELLIGENCE (ICTAI), 2011 23RD IEEE INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Florida, USA: [s.n.], nov. 2011. p. 170–176. Disponível em <https://ieeexplore.ieee.org/document/6103323>. Acesso em 03 jan 2019.

DUCHON, Frantisek et al. Path planning with modified A star algorithm for a mobile robot. **Procedia Engineering**, Elsevier, v. 96, p. 59–69, 2014.

DUDEK, G.; JENKIN, M. **Computational Principles of Mobile Robotics**. Cambridge, UK: Cambridge University Press, 2000.

ELFES, Alberto. Using occupancy grids for mobile robot perception and navigation. **Computer**, IEEE, n. 6, p. 46–57, 1989.

ELLIOTT, Dominic; SMITH, Denis. Football stadia disasters in the United Kingdom: learning from tragedy? **Industrial & Environmental Crisis Quarterly**, v. 7, n. 3, p. 205–229, 1993. Disponível em <https://www.researchgate.net/publication/239997436_Football_Stadia_Disasters_in_the_United_Kingdom_Learning_from_Tragedy>. Acesso em 03 dez 2018.

ELPROCUS. **Infrared IR Sensor Circuit Diagram and Working Principle**. [S.l.]: Elprocus, 2017. Disponível em <https://www.elprocus.com/infrared-ir-sensor-circuit-and-working>. Acesso em 24 mar 2017.

FERNANDES JUNIOR, Francisco Erivaldo. **Estudo e implementação de redes neurais e algoritmos genéticos para resolução de cinemática inversa de um manipulador robótico com 5 graus de liberdade**. 2014. f. 113. dissertation (Mestrado em Engenharia Elétrica) – Universidade Estadual de Campinas - UNICAMP, Campinas, 2014.

FERRER, Gonzalo; GARRELL, Anaís; SANFELIU, Alberto. Robot Companion: A Social-Force based approach with Human Awareness-Navigation in Crowded Environments. In: 2013 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS e SYSTEMS. PROCEEDINGS... Tokyo, Japan: [s.n.], nov. 2013. p. 1688–1694. Disponível em <http://www.iri.upc.edu/files/scidoc/1458-Robot-Companion:-A-Social-Force-based-approach-with-Human-Awareness-Navigation-in-Crowded-Environments.pdf>. Acesso em 03 fev 2018.

FILLIAT, David; MEYER, Jean-Arcady. Map-based navigation in mobile robots: I. A review of localization strategies. **Cognitive Systems Research 4**, v. 4, n. 4, p. 243–282, dez. 2003. Disponível em <https://www.sciencedirect.com/science/article/pii/S1389041703000081>. Acesso em 23 mar. 2017.

FOX, Dieter et al. Monte carlo localization: Efficient position estimation for mobile robots. **AAAI/IAAI**, v. 1999, n. 343-349, p. 2–2, 1999.

GANGANATH, N; LEUNG, H. Mobile robot localization using odometry and kinect sensor. In: EMERGING SIGNAL PROCESSING APPLICATIONS (ESPA), 2012 IEEE INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Las Vegas, USA: [s.n.], jan. 2012. p. 91–94. Disponível em <https://ieeexplore.ieee.org/document/6152453/>. Acesso em 03 fev 2018.

GARRELL, Anais; SANFELIU, Alberto. Cooperative social robots to accompany groups of people. **The International Journal of Robotics Research**, v. 31, n. 13, p. 1675–1701, set. 2012. Disponível em <https://www.researchgate.net/publication/258140878_Cooperative_social_robots_to_accompany_groups_of_people>. Acesso em 05 set. 2018.

GEEKY-GADGETS. **Hexapod KMR-M6 Six Legged Robot Up Close (video)**. [S.l.]: Geeky Gadgets, 2017. Disponível em <http://www.geeky-gadgets.com/hexapod-kmr-m6-six-legged-robot-up-close-video-26-04-2011/>. Acesso em 06 mar 2017.

GENERATIONROBOTS. **5 Front and Back Pioneer DX and AT Bumpers**. [S.l.]: Generation Robots, 2017. Disponível em <https://www.generationrobots.com/en/402512-5-front-and-back-pioneer-dx-and-at-bumpers.html?search_query=bumper&results=11>. Acesso em 04 abr 2017.

GERKEY, Brian P.; VAUGHAN, Richard T.; HOWARD, Andrew. The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In: 11TH INTERNATIONAL CONFERENCE ON ADVANCED ROBOTICS. PROCEEDINGS... Coimbra, Portugal: [s.n.], 2003. p. 317–323. Disponível em <https://www.researchgate.net/publication/2876085_The_PlayerStage_Project_Tools_for_Multi-Robot_and_Distributed_Sensor_Systems>. Acesso em 09 jun 2018.

GIGLIO-JACQUEMOT, Armelle. **Definições de urgência e emergência: critérios e limitações**. Rio de Janeiro, Brasil: Editora FIOCRUZ, 2005. p. 144. Disponível em <http://books.scielo.org/id/zt4fg/02>. Acesso em 23 fev. 2017.

GOMES, Samuel Silva. **Modelagem e controle de atuadores robóticos e veículos subaquáticos não tripulados**. 2011. f. 151. dissertation (Mestrado em Engenharia Oceânica) – Universidade Federal do Rio Grande, Rio Grande, 2011.

GONÇALVES, Rodrigo Mikosz; CENTERO, Tânia Mezzadri; CANDEIAS, Ana Lúcia Bezerra. Autômatos celulares aplicados na modelagem de tendência em imagens multi-temporais. **Revista Brasileira de Cartografia-RBC**, n. 63-02, p. 233–241, 2011.

GORSEVSKI, Pece V. et al. Detecting grain boundaries in deformed rocks using a cellular automata approach. **Computers & Geosciences**, v. 42, p. 136–142, 2012. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0098300411003086>.

GOULART, Christiano. **Modelagem, simulação e controle de um veículo submarino de operação remota**. 2007. f. 138. dissertation (Mestrado em Engenharia Elétrica) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2007.

GRAF, B. et al. **Development of an intelligent care cart and new supplies concepts for care homes and hospitals**. Munich, Germany: [s.n.], jun. 2016. p. 1–6. Disponível em <https://ieeexplore.ieee.org/document/7559152>. Acesso em 03 fev 2018.

GREWAL, Harkishan Singh et al. Autonomous wheelchair navigation in unmapped indoor environments. In: 2018 IEEE INTERNATIONAL INSTRUMENTATION e MEASUREMENT TECHNOLOGY CONFERENCE (I2MTC). PROCEEDINGS... Texas, USA: [s.n.], mai. 2018. p. 1–6. Disponível em <https://ieeexplore.ieee.org/document/8409854>. Acesso em 03 abr 2019.

GUPTA, Saurabh et al. Cognitive mapping and planning for visual navigation. **arXiv preprint arXiv:1702.03920**, v. 3, 2017. Disponível em <http://openaccess.thecvf.com/content_cvpr_2017/papers/Gupta_Cognitive_Mapping_and_CVPR_2017_paper.pdf>. Acesso em 19 nov 2018.

GURAVAIAH, M.; DANIEL, K. Robotics - Review. **World Journal of Pharmaceutical Research**, v. 4, n. 10, p. 406–417, dez. 2015. Disponível em <www.wjpr.net/download/article/1443847780.pdf>. Acesso em 28 apr. 2018.

HALL-HOLT, Olaf; RUSINKIEWICZ, Szymon. Stripe boundary codes for real-time structured-light range scanning of moving objects. In: COMPUTER VISION, 2001. ICCV 2001. EIGHTH IEEE INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Vancouver, Canada: [s.n.], jul. 2001. v. 2, p. 359–366. Disponível em <https://ieeexplore.ieee.org/document/937648>. Acesso em 05 abr 2017.

HART, Peter E.; NILSSON, Nils J.; RAPHAEL, Bertram. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. **IEEE Transactions on Systems Science and Cybernetics**, v. 4, n. 2, p. 100–107, jul. 1968. Disponível em <https://ieeexplore.ieee.org/abstract/document/4082128>. Acesso em 16 ago 2019.

HEINEN, Milton Roberto; OSÓRIO, Fernando S. Controle inteligente do caminhar de robôs móveis utilizando algoritmos genéticos e redes neurais artificiais. **VI Encontro Nacional de Inteligência Artificial**, 2007.

HELBING, D.; FARKAS, I.; VICSEK, T. Simulating dynamical features of escape panic. **Nature**, v. 407, p. 487–490, set. 2000. Disponível em <https://www.nature.com/articles/35035023>. Acesso em 15 fev. 2017.

HOKUYO. **Sensor scanner laser**. [S.l.]: Hokuyo, 2017. Disponível em <https://www.hokuyo-aut.jp/02sensor/07scanner/urg_04lx_ug01.html>. Acesso em 15 mar 2017.

IEEE.ORG. **Relay**. [S.l.]: IEEE Robots, 2019. Disponível em <https://robots.ieee.org/robots/relay>. Acesso em 23 jan 2019.

INDEPENDENT. **La UPC presenta a Gràcia els robots urbans**. [S.l.]: UPC, 2017. Disponível em <http://independent.cat/nova/noticia/2010/05/07/la-upc-presenta-a-gracia-els-robots-urbans>. Acesso em 13 mar 2017.

INOKUCHI, Seiji; SATO, Kosuke; MATSUDA, Fumio. Range-imaging system for 3D object recognition. **International Conference on Pattern Recognition**, v. 2, jan. 1984.

JACAK, Witold. **Intelligent Robotic Systems: Design, Planning, and Control**. Hagenberg, Austria: Springer Publishing Company, Incorporated, 2002. v. 14.

JAVAPOINT. **Robot Locomotion**. [S.l.]: Java T point, 2017. Disponível em <https://www.javatpoint.com/robot-locomotion>. Acesso em 28 abr 2017.

JIANG, Chao et al. Robot-assisted pedestrian regulation in an exit corridor. In: INTELLIGENT ROBOTS e SYSTEMS (IROS), 2016 IEEE/RSJ INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Daejeon, Korea: [s.n.], out. 2016. p. 815–822. Disponível em <https://ieeexplore.ieee.org/document/7759145>. Acesso em 12 nov 2018.

JIANG, Yangyang et al. Gait design of biped robot based on USBSSC32. In: 2013 IEEE 8TH CONFERENCE ON INDUSTRIAL ELECTRONICS e APPLICATIONS (ICIEA). PROCEEDINGS... Melbourne, Australia: [s.n.], jun. 2013. p. 978–981. Disponível em <https://ieeexplore.ieee.org/document/6566509/>. Acesso em 23 fev 2017.

JOHNSON, Norms R. Panic at “The Who concert stampede”: an empirical assessment. **Social Problems**, Oxford University Press Oxford, UK, v. 34, n. 4, p. 362–373, 1987.

KAMARUDIN, K. et al. Improving Performance of 2D SLAM Methods by Complementing Kinect with Laser Scanner. In: 2015 IEEE INTERNATIONAL SYMPOSIUM ON ROBOTICS e INTELLIGENT SENSORS (IRIS). PROCEEDINGS... Langkawi, Malaysia: [s.n.], out. 2015. p. 278–283. Disponível em <https://ieeexplore.ieee.org/document/7451625>. Acesso em 03 fev 2018.

KASSIM, Anuar Mohamed et al. Conceptual design and implementation of electronic spectacle based obstacle detection for visually impaired persons. In: JOURNAL OF ADVANCED MECHANICAL DESIGN, SYSTEMS, e MANUFACTURING. THE JAPAN SOCIETY OF MECHANICAL ENGINEERS. PROCEEDINGS... Tokyo, Japan: [s.n.], 2016. v. 10, p. 1–12. Disponível em <https: //www.researchgate.net/publication/309586575_Conceptual_design_and_implementation_of_electronic_spectacle_based_obstacle_detection_for_visually_impaired_persons>. Acesso em 01 out 2019.

KATEVAS, N. **Mobile robotics in healthcare**. Amsterdam: IOS Press, 2001.

KELLEY, Scott T; GILBERT, Jack a. **Studying the microbiology of the indoor environment**. [S.l.]: Genome Biology, 2013. v. 14, p. 202. Disponível em <https://genomebiology.biomedcentral.com/articles/10.1186/gb-2013-14-2-202>. Acesso em 24 maio 2017.

KLAPTOCZ, A. **Design of Flying Robots for Collision Absorption and Self-recovery**. 2012. f. 114. thesis (Programme doctoral en systèmes de production et robotique) – École Polytechnique Fédérale de Lausanne, Lausanne, 2012.

KLEEMAN, Lindsay; KUC, Roman. **Sonar Sensing**. [S.l.]: Springer Handbook of Robotics, 2016. p. 753–782. Disponível em <https://link.springer.com/referenceworkentry/10.1007%2F978-3-540-30301-5_22>. Acesso em 23 abr. 2018.

KNIGHTSCOPE. **K5 Autonomous Data Machine**. [S.l.]: Knightscope, 2019. Disponível em <https://www.knightscope.com/knightscope-k5>. Acesso em 23 jan 2019.

KOENIG, N.; HOWARD, A. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In: 2004 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS e SYSTEMS (IROS) (IEEE CAT. NO.04CH37566). PROCEEDINGS... Sendai, Japan: [s.n.], set. 2004. v. 3, 2149–2154 vol.3. Disponível em <https://ieeexplore.ieee.org/abstract/document/1389727/>. Acesso em 03 fev 2018.

KOLLAR, T. et al. Learning environmental knowledge from task-based human-robot dialog. In: 2013 IEEE INTERNATIONAL CONFERENCE ON ROBOTICS e AUTOMATION. PROCEEDINGS... Karlsruhe, Germany: [s.n.], mai. 2013. p. 4304–4309. Disponível em <https://ieeexplore.ieee.org/document/6631186>. Acesso em 07 fev 2017.

KRAMER, Klaus. **Dinâmica de padrões em autômatos celulares com inércia**. 2014. f. 132. thesis (Doutorado em Física) – Universidade Federal do Paraná, Curitiba, 2014.

LADOU, Joseph; HARRISON, Robert J. **Poluição atmosférica**. [S.l.]: Current Medicina ocupacional e ambiental: diagnóstico e tratamento [recurso eletrônico], 2016. v. 5. Disponível em <https://books.google.com.br/books?isbn=8580555655>. Acesso em 24 maio 2017.

LAFLAQUIÈRE, Alban et al. Emerging Spatial Competences: From Machine Perception to Sensorimotor Intelligence. **Robotics and Autonomous Systems**, v. 71, p. 49–59, mar. 2015. Disponível em <https://www.researchgate.net/publication/273836860_Emerging_spatial_competences_From_machine_perception_to_sensorimotor_intelligence>. Acesso em 23 mar. 2017.

LANG, Haoxiang; WANG, Ying; SILVA, Clarence W de. Mobile robot localization and object pose estimation using optical encoder, vision and laser sensors. In: AUTOMATION e LOGISTICS, 2008. ICAL 2008. IEEE INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Chindao, China: [s.n.], set. 2008. p. 617–622. Disponível em <https://ieeexplore.ieee.org/document/4636224>. Acesso em 09 fev 2018.

LEE, Sooyong; SONG, Jae-Bok. Mobile robot localization using infrared light reflecting landmarks. In: IEEE CONTROL, AUTOMATION e SYSTEMS, 2007. ICCAS'07. INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Seoul, Korea: [s.n.], out. 2007. p. 674–677. Disponível em <https://ieeexplore.ieee.org/document/4406984/>. Acesso em 03 fev 2018.

LEITE, I. O. B.; LINS, J. C. S.; CERQUEIRA, M. G. C. **Especificações na Construção de um Sistema Autômato Celular**. [S.l.]: UFPE, 2007. Disponível em <http://www.di.ufpe.br/iobl/monografia/especificacoes.htm>. Acesso em 14 maio 2017.

LEUZE. **Sensor scanner laser**. [S.l.]: Leuze Electronic, 2017. Disponível em <http://www.leuze.com.br/pt/brasil/produtos/sensores_de_chaveamento/sensores_3d_2/scanner_a_laser/index.php>. Acesso em 15 mar 2017.

LI, Dewei; HAN, Baoming. Behavioral effect on pedestrian evacuation simulation using cellular automata. **Safety science**, Elsevier, v. 80, p. 41–55, 2015.

LINO, Paolo; MAIONE, Guido; MAIONE, Bruno. Modeling and simulation of crowd egress dynamics in a discrete event environment. In: CONTROL APPLICATIONS,(CCA) & INTELLIGENT CONTROL,(ISIC), 2009 IEEE. PROCEEDINGS... St. Petersburg, Russia: [s.n.], jul. 2009. p. 843–848. Disponível em <https://ieeexplore.ieee.org/document/5281015>. Acesso em 03 fev 2018.

LIU, Huaping et al. Robotic room-level localization using multiple sets of sonar measurements. **IEEE Transactions on Instrumentation and Measurement**, IEEE, v. 66, n. 1, p. 2–13, 2017.

LIU, J. et al. Study of methods for identifying obstacles using electronic guide device. In: 2014 IEEE CHINESE GUIDANCE, NAVIGATION e CONTROL CONFERENCE. PROCEEDINGS... Yantai, China: [s.n.], ago. 2014. p. 340–343. Disponível em <https://ieeexplore.ieee.org/document/7007254>. Acesso em 02 abr 2017.

LIU, S. et al. Cognitive abilities of indoor cleaning robots. In: 2016 12TH WORLD CONGRESS ON INTELLIGENT CONTROL e AUTOMATION (WCICA). PROCEEDINGS... Guilin, China: [s.n.], jun. 2016. p. 1508–1513. Disponível em <https://ieeexplore.ieee.org/document/7578317>. Acesso em 03 abr 2018.

LOZANO, Kátia Kelvis Cassiano. **Autômatos celulares probabilísticos com aplicações a sistemas biológicos**. 2017. f. 143. thesis (Doutorado em Engenharia de Sistemas e Computação) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2017.

LUO, Yi et al. A biped humanoid robot's gait planning method based on Artificial Immune Network. In: 2014 INTERNATIONAL CONFERENCE ON MACHINE LEARNING e CYBERNETICS. PROCEEDINGS... Lanzhou, China: [s.n.], jul. 2014. v. 1, p. 292–297. Disponível em <https://ieeexplore.ieee.org/document/7009131/>. Acesso em 23 jun 2018.

MAITY, Kousik et al. Autonomous One Wheel Flying Robot. **International Journal of Advances in Engineering & Technology**, v. 6, n. 2, p. 961–972, 2013.

MAKARENKO, Alexei A et al. An experiment in integrated exploration. In: IEEE. PROCEEDINGS... Lausanne, Switzerland: [s.n.], set. 2002. v. 1, p. 534–539. Disponível em <Anexperimentinintegratedexploration>. Acesso em 11 fev 2017.

MARINHO, Leandro B. et al. A novel mobile robot localization approach based on topological maps using classification with reject option in omnidirectional images. **Expert Systems with Applications**, Elsevier, v. 72, p. 1–17, 2017.

MCBRIDE, N.; HOFFMAN, R. R. Bridging the Ethical Gap: From Human Principles to Robot Instructions. **IEEE Intelligent Systems**, v. 31, n. 5, p. 76–82, set. 2016. Disponível em <https://ieeexplore.ieee.org/document/7579396>. Acesso em 23 ago. 2018.

MCKERROW, P. K. **Introduction to robotics**. Reading, MA: Addison-Wesley, 1999. p. 811.

MEHMOOD, Tahir; HASHMI, Usman; AJMAL, Arsalan Akhter; Ammar. Techniques and approaches in Robocup@Home - A review. In: 2015 INTERNATIONAL CONFERENCE ON INFORMATION e COMMUNICATION TECHNOLOGIES (ICICT). PROCEEDINGS... Dubai, UAE: [s.n.], nov. 2015. p. 1–6. Disponível em <https://ieeexplore.ieee.org/document/7469481/>. Acesso em 03 fev 2018.

MELOTTI, Gledson. **Aplicação de Autômatos Celulares em Sistemas Complexos: Um estudo de caso em espalhamento de epidemias**. 2009. f. 116. dissertation (Mestrado em Engenharia Elétrica) – Universidade Federal de Minas Gerais, Belo Horizonte, 2009.

MEYERDELIUS, Daniel; BEINHOFER, Maximilian; BURGARD, Wolfram. Occupancy Grid Models for Robot Mapping in Changing Environments. In: AAAI. PROCEEDINGS... Toronto, Canadá: [s.n.], jul. 2012. p. 2024–2030. Disponível em <https://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/viewFile/4983/5367>. Acesso em 03 dez 2018.

MICROSOFT. **Kinect Sensor**. [S.l.]: Microsoft, 2017. Disponível em <https://msdn.microsoft.com/en-us/library/hh438998.aspx>. Acesso em 01 maio 2017.

MOBILEROBOTS. **Camera Systems**. [S.l.]: Mobile Robots, 2017. Disponível em <http://www.mobilerobots.com/Accessories/CameraSystems.aspx>. Acesso em 04 abr 2017.

_____. **Pioneer 3-DX**. [S.l.]: Mobile Robots, 2017. Disponível em <http://www.mobilerobots.com/Libraries/Downloads/Pioneer3DX-P3DX-RevA.sflb.ashx>. Acesso em 10 abr 2017.

MOJTAHEDZADEH, Rasoul. **Robot Obstacle Avoidance using the Kinect**. Ago. 2011. f. 62. dissertation (Systems, Control and Robotics Master's Program) – Stockholm.

MONDOL, Sudipto Shekhor. **Advanced Robotics Training Report**. 2016. f. 41. dissertation (Bachelor of Technology in Mechanical Engineering) – Institute of Technology, Kolkata, 2016.

MORAVEC, H.; ELFES, A. High resolution maps from wide angle sonar. In: 1985 IEEE INTERNATIONAL CONFERENCE ON ROBOTICS e AUTOMATION. PROCEEDINGS... St. Louis, Missouri, USA: [s.n.], mar. 1985. v. 2, p. 116–121. Disponível em <https://ieeexplore.ieee.org/document/1087316>. Acesso em 24 nov 2018.

MOZOS, O Martınez; STACHNISS, Cyrill; BURGARD, Wolfram. Supervised learning of places from range data using adaboost. In: ROBOTICS e AUTOMATION (ICRA 2005). 2005

IEEE INTERNATIONAL CONFERENCE ON. PROCEEDINGS... Barcelona, Spain: [s.n.], abr. 2005. p. 1730–1735. Disponível em <https://www.researchgate.net/publication/261834529_Supervised_Learning_of_Places_from_Range_Data_using_Adaboost>. Acesso em 03 fev 2018.

MULLER, Ana Paula Oliveira. **Dinâmica de domínios e difusão em autômatos celulares determinísticos**. 2008. f. 73. dissertation (Mestrado em Física) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2008.

NASA, Jpl. **ATHLETE Rover Busts a Move: A Dancing Robot**. [S.l.]: NASA, 2017. Disponível em <https://www.jpl.nasa.gov/video/details.php?id=922>. Acesso em 06 mar 2017.

NASCIMENTO, Tiago Pereira do. **Controle de trajetória de robôs móveis omni-direcionais: Uma abordagem multivariável**. 2009. f. 85. dissertation (Mestrado em Engenharia Elétrica) – Universidade Federal da Bahia, Salvador, 2009.

NEUMANN, John Von. **Theory of Self-Reproducing Automata**. Edição: Arthur W. Burks. Champaign, IL, USA: University of Illinois Press, 1966.

NGUYEN, Vu Anh; STARZYK, Janusz A.; GOH, Wooi-Boon. A spatio-temporal Long-term Memory approach for visual place recognition in mobile robotic navigation. **Robotics and Autonomous Systems**, v. 61, n. 12, p. 1744–1758, dez. 2013. Disponível em <https://www.sciencedirect.com/science/article/pii/S092188901300002X>. Acesso em 23 mar. 2017.

NI, National Instruments. **Using LabVIEW to Rapidly Build an Autonomous Robot and Compete in the Intelligent Ground Vehicle Competition**. [S.l.]: National Instruments, 2017. Disponível em <http://sine.ni.com/cs/app/doc/p/id/cs-13114>. Acesso em 06 mar 2017.

NOF, Shimon Y. **Handbook of Industrial Robotics**. 2. ed. Danvers, MA: Wiley, 1999. p. 1378.

NORTEK. **DXS-54: Supervised Passive Infrared Motion Detector Transmitter (PERS)**. [S.l.]: Nortek, 2017. Disponível em <https://www.nortekcontrol.com/product_detail.php?productId=1179>. Acesso em 24 mar 2017.

NUNES, Flavio et al. Uma Representação Visual para Gerenciamento de Emergências em Ambientes Fechados. In: XII BRAZILIAN SYMPOSIUM ON INFORMATION SYSTEMS. PROCEEDINGS... Florianópolis, Brasil: [s.n.], 2016. Disponível em <http://www.lbd.dcc.ufmg.br/bdbcomp/servlet/Trabalho?id=23924>. Acesso em 03 fev. 2018.

NUNES, Ricardo F. Mapeamento da cinemática inversa de um manipulador robótico utilizando redes neurais artificiais configuradas em paralelo. Universidade Estadual Paulista (UNESP), 2016.

NYMAN, Moa; D'ORTO, Manolo. **Validation of a vehicle motion planner in an open-source simulator**. 2018. f. 39. Project in technology (Bachelor in Computer Science) – KTH Royal Institute of Technology, Stockholm, 2018.

OASYS, Limited. **MassMotion**. [S.l.]: Oasys, 2018. Disponível em <https://www.oasys-software.com>. Acesso em 25 set 2018.

OKOLI, Franklin et al. Cable-Driven Parallel Robot Simulation Using Gazebo and ROS. In: ROMANSY 22–Robot Design, Dynamics and Control. [S.l.]: Springer, 2019. p. 288–295.

OLIVEIRA, Luiz Fernando Pinto de. **Modelação, Simulação e Implementação de Padrões de Locomoção para Robôs Hexápodes**. 2016. f. 131. dissertation (Mestrado em Engenharia Electrotécnica e de Computadores) – Instituto Superior de Engenharia do Porto, Porto, 2016.

OSÓRIO, Fernando S. et al. Increasing reality in virtual reality applications through physical and behavioural simulation. In: VIRTUAL CONCEPT CONFERENCE. PROCEEDINGS... Cancun, México: [s.n.], dez. 2006. v. 1, p. 1–45. Disponível em <http://osorio.wait4.org/publications/Osorio-et-al-Virtual-Concept2006.pdf>. Acesso em 03 fev 2018.

OTTONI, Guilherme Lima. **Planejamento De Trajetórias Para Robôs Móveis**. 2000. f. 89. Trabalho de Conclusão de Curso (Graduação em Engenharia da Computação) – Universidade Federal do Rio Grande, Rio Grande, 2000.

PACHECO, Carolina; KARELOVIC, Pablo; CIPRIANO, Aldo. Comparing dynamic evacuation control strategies for emergency situations. In: 2015 EUROPEAN CONTROL CONFERENCE (ECC). PROCEEDINGS... Linz, Austria: [s.n.], jul. 2015. p. 3053–3058. Disponível em <https://ieeexplore.ieee.org/document/7331002>. Acesso em 03 fev. 2018.

PANASONIC. **Panasonic Conducts Demonstration Experiment of Autonomous Signage Robot, "HOSPI(R)"at Narita Airport**. [S.l.]: Panasonic, 2018. Disponível em <https://news.panasonic.com/global/topics/2018/54270.html>. Acesso em 23 jan 2019.

PASCO. **Sensor de temperatura**. [S.l.]: Pasco, 2017. Disponível em <https://www.pasco.com/prodCatalog/PS/PS-3201_wireless-temperature-sensor/index.cfm>. Acesso em 15 mar 2017.

PEDROSA, Diogo P F; MEDEIROS, Adelardo; ALSINA, Pablo. Extração de características métricas locais a partir de grade de ocupação usando método de votação ponderada, p. 6, 2019.

PEDSIM. **PedSim, Pedestrian Crowd Simulation**. [S.l.]: PEDSIM, 2018. Disponível em <http://pedsim.silmaril.org>. Acesso em 13 out 2018.

_____. **PEDSIM: A Microscopic PedestrianCrowd Simulation System**. [S.l.]: PEDSIM, 2002. Disponível em <http://pedsim.silmaril.org>. Acesso em 15 nov 2018.

PEGAS, L. G. **Rastreamento Visual para Robôs usando Microsoft Kinect**. 2014. f. 94. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) – Universidade Federal de São Carlos, São Carlos, 2014.

PFEIFER, R. et al. **Artificial Life**. [S.l.]: Institute for Informatics, Zurich University, Germany, 2001. Disponível em <http://www.ifi.unizh.ch/ailab/teaching/AL00/>. Acesso em 25 ago 2018.

PITARMA, Rui; MARQUES, Gonçalo; FERREIRA, Bárbara Roque. Monitoring Indoor Air Quality for Enhanced Occupational Health. **Journal of Medical Systems**, v. 41, n. 2, p. 23, dez. 2016. Disponível em <https://link.springer.com/article/10.1007/s10916-016-0667-2>. Acesso em 19 jul. 2018.

PRONOBIS, A. et al. A realistic benchmark for visual indoor place recognition. **Robotics and Autonomous Systems**, v. 58, n. 1, p. 81–96, jan. 2010. Disponível em <https://www.sciencedirect.com/science/article/pii/S0921889009001146>. Acesso em 23 mar. 2017.

QUADROS, Marina Eller. **Qualidade do ar em ambientes internos hospitalares: parâmetros físico-químicos e microbiológicos**. 2008. f. 134. dissertation (Mestrado em Engenharia Ambiental) – Universidade Federal de Santa Catarina, Florianópolis, 2008.

RAMOS, Fabio et al. A Bayesian approach for place recognition. **Robotics and Autonomous Systems**, v. 60, n. 4, p. 487–497, abr. 2012. Disponível em <https://www.sciencedirect.com/science/article/pii/S0921889011002053>. Acesso em 23 abr. 2017.

RBR. **Global Outlook for Mobile Security Robots**. [S.l.]: RBR, 2019. Disponível em <https://www.roboticsbusinessreview.com/wp-content/uploads/2018/03/RBR-UGVs-For-Security.pdf>. Acesso em 25 fev 2019.

_____. **Qihan Robot Provides Legal Services at Chinese Courthouse**. [S.l.]: RBR, 2019. Disponível em <https://www.roboticsbusinessreview.com/ai/qihan-robot-provides-legal-services-chinese-court/>. Acesso em 28 fev 2019.

REINALDO, Jurasildo Oliveira. **Reconhecimento de objetos no desenvolvimento de um sistema de navegação inteligente para robôs móveis**. 2015. f. 63. dissertation (Mestrado em Ciência da Computação) – Universidade do Estado do Rio Grande do Norte, Mossoró, 2015.

RENKAVIESKI, Christopher et al. Autômato Celular 3D Aplicado à Simulação Computacional da Evolução de Volumes Tumorais. **Anais do Computer on the Beach**, p. 860–869, 2018.

REYNOLDS, Craig W. Steering behaviors for autonomous characters. In: CITESEER, GAME DEVELOPERS CONFERENCE. PROCEEDINGS... San Francisco, USA: [s.n.], mar. 1999. v. 1999, p. 763–782. Disponível em <https://www.semanticscholar.org/paper/Steering-Behaviors-For-Autonomous-Characters-Reynolds/0b563ad5fdea6923d2ee91f9ee4ad3d6dcdf9cd0>. Acesso em 07 jul 2018.

RIBO, Miguel; PINZ, Axel. A comparison of three uncertainty calculi for building sonar-based occupancy grids. **Robotics and autonomous systems**, Elsevier, v. 35, n. 3-4, p. 201–209, 2001.

ROBOSAVVY. **Hitec – ROBONOVA-1 Kit**. [S.l.]: Robo Savvy, 2017. Disponível em <https://robosavvy.com/store/hitec-robonova-1-kit.html>. Acesso em 06 mar 2017.

ROBOTICS. **Omni-Directional Vehicles**. [S.l.]: University of Western Australia, 2017. Disponível em <http://robotics.ee.uwa.edu.au/eyebot/doc/robots/omni.html>. Acesso em 06 mar 2017.

ROBOTICSTODAY. **Kondo Series : Kondo KHR-1HV**. [S.l.]: Robotics Today, 2017. Disponível em <http://www.roboticstoday.com/robots/kondo-khr-1hv>. Acesso em 06 mar 2017.

ROBUGTIX. **Looks like a spider. Moves like a spider**. [S.l.]: Robugtix, 2017. Disponível em <http://www.robugtix.com/t8x>. Acesso em 03 abr 2017.

RUCKER, Rudy. The lifebox, the seashell, and the soul. **Thunder's Mouth Press, New York**, 2005.

RUSSELL, Stuart; NORVIG, Peter. **Artificial Intelligence: A Modern Approach**. 3. ed. Englewood Cliffs, New Jersey, USA: Pearson, 2010.

RUSSELL, Stuart; NORVIG, Peter; DAVIS, Ernest. **Artificial intelligence: a modern approach**. 3. ed. Upper Saddle River, NJ: Prentice Hall, 2010.

SAHDEV, Raghavender; CHEN, Bao Xin; TSOTSOS, John K. Indoor localization in dynamic human environments using visual odometry and global pose refinement. In: 15TH CONFERENCE ON COMPUTER e ROBOT VISION (CRV), IEEE. PROCEEDINGS... Toronto, Canada: [s.n.], mai. 2018. p. 1–8. Disponível em <https://www.raghavendersahdev.com/uploads/3/9/6/2/39623741/localization_among_humans_crv-2018.pdf>. Acesso em 03 fev 2018.

SAKOUR, Ibraheem; HU, Huosheng. Robot-assisted crowd evacuation under emergency situations: A survey. **Robotics**, Multidisciplinary Digital Publishing Institute, v. 6, n. 2, p. 8, 2017.

SANTANA, André M.; AIRES, Kelson; MEDEIROS, Adelardo. Uma Abordagem Para Construção De Grade De Ocupação Visual 2D Usando Visão Monocular. **10th Brazilian Congress on Computational Intelligence (CBIC-2011)**, ResearchGate, p. 1–8, 2016.

SANTANA, André Macedo. **Localização e mapeamento simultâneos de ambientes planos usando visão monocular e representação híbrida do ambiente**. 2011. f. 149. thesis (PhD em Engenharia Elétrica e de Computação) – Universidade Federal do Rio Grande do Norte, Natal, 2011.

SANTOS KLIPP, Telmo dos. **Proposta de uma arquitetura para alocação de tarefas em grupos de robôs móveis baseada em acordo bizantino**. 2013. f. 73. TCC (Mestrado em Física) – Universidade Federal de Santa Catarina, Florianópolis, 2013.

SAQUI, Diego. **Estudo de uma técnica para o tratamento de dead-times em operações de rastreamento de objetos por servovisão**. 2014. f. 130. dissertation (Mestrado em Ciência da Computação) – Universidade Federal de São Carlos, São Carlos, 2014.

SCIENSE. **How ASIMO Works**. [S.l.]: Howstuffworks, 2017. Disponível em <http://science.howstuffworks.com/asimo.htm>. Acesso em 06 mar 2017.

SHI, L; KODAGODA, S; DISSANAYAKE, G. Environment classification and semantic grid map building based on laser range finder data. In: ONLINE PROCEEDING OF IROS 2010 WORKSHOP ON SEMANTIC MAPPING e AUTONOMOUS KNOWLEDGE ACQUISITION. PROCEEDINGS... Taipei, Taiwan: [s.n.], out. 2010. p. 1–6. Disponível em <http://www.ais.uni-bonn.de/~holz/iros10wssm/proceedings/02_shi_final.pdf>. Acesso em 03 fev 2018.

SHIOMI, Masahiro et al. Towards a Socially Acceptable Collision Avoidance for a Mobile Robot Navigating Among Pedestrians Using a Pedestrian Model. **Int J of Soc Robotics, Springer Science+Business Media Dordrecht**, v. 6, n. 1, p. 443–455, Abr. 2014. Disponível em <https://link.springer.com/article/10.1007/s12369-014-0238-y>. Acesso em 03 fev 2018.

SIEGWART, R.; NOURBAKHSH, I. R. **Introduction to autonomous mobile robots)**. Cambridge, Massachusetts, MA: Massachusetts: The MIT Press, 2004.

SILVA, Anderson Paulo da et al. Modelagem Computacional Sobre a Expansão Epidemiológica da Esquistossomose. **UFRPE-Universidade Federal Rural de Pernambuco, DEINFO-Departamento de Estatística e Informática**, 2008.

SOLOMON, Chris; BRECKON, Toby. **Fundamentos de processamento digital de imagens: uma abordagem prática com exemplos em Matlab**. Rio de Janeiro, RJ: LTC, 2013. Disponível em: <https://books.google.com.br/books?id=vSRKnQEACAAJ>.

SONY. **Sony-Aibo**. [S.l.]: Sony, 2017. Disponível em <http://www.sony-aibo.com>. Acesso em 28 abr 2017.

SOUTO, José Renato A.; MAIA, Rosiery S.; SOUZA, Anderson de. Um Sistema de Navegação para o Robô Móvel X80. In: XII SIMP´oSIO BRASILEIRO DE AUTOMAç˜aO INTELIGENTE (SBAI). PROCEEDINGS... Natal , Brasil: [s.n.], Out. 2014. p. 1800–1805. Disponível em <swge.inf.br/SBAI2015/anais/511.pdf>. Acesso em 03 fev. 2018.

SOUZA, Anderson Abner Santana. **Mapeamento com sonar usando grade de ocupação baseado em modelagem probabilística**. 2008. f. 71. dissertation (Mestrado em Ciências) – Universidade Federal do Rio Grande do Norte, Natal, 2008.

SPECTRUM, Ieee. **Video: SQ1 Quadruped Robot from South Korea**. [S.l.]: IEEE Spectrum, 2017. Disponível em <spectrum.ieee.org/automaton/robotics/robotics-software/video-sq1-quadruped-robot-from-south-korea>. Acesso em 06 mar 2017.

SPEICH, J. E.; ROSEN, J. Medical robotics. **Encyclopedia of biomaterials and biomedical engineering**, p. 983–993, 2004.

STACHNISS, Cyrill. **Robotic mapping and exploration**. Heidelberg, Germany: Springer, 2009. v. 55.

STEINBAUER, Gerald; LOIGGE, Stefan; MÜHLBACHER, Clemens. Supervision of Hardware, Software and Behavior of Autonomous Industrial Transport Robots. In: 2016 IEEE INTERNATIONAL CONFERENCE ON SOFTWARE QUALITY, RELIABILITY e SECURITY COMPANION (QRS-C). PROCEEDINGS... Vienna, Austria: [s.n.], jun. 2016. p. 298–300. Disponível em <https://ieeexplore.ieee.org/document/7573759>. Acesso em 13 fev 2018.

STILL, G. Keith. **Crowd Dynamics**. 2000. f. 281. thesis (PhD in Philosophy in Mathematics) – University of Warwick, Warwick, 2000.

SZELISKI, Richard. **Computer vision: algorithms and applications**. Heidelberg, Germany: Springer Science & Business Media, 2010.

TANG, Bo et al. Human Mobility Modeling for Robot-Assisted Evacuation in Complex Indoor Environments. **IEEE Transactions on Human-Machine Systems**, v. 46, n. 5, p. 694–707, out. 2016.

THRUN, Sebastian. Learning occupancy grids with forward models. In: 2001 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS e SYSTEMS-IROS 2001. PROCEEDINGS... Pittsburgh, USA: [s.n.], out. 2001. p. 1676–1681. Disponível em <https://ieeexplore.ieee.org/document/977219>. Acesso em 03 fev 2018.

TORABI, Sina. **Collective transportation of objects by a swarm of robots**. 2015. f. 34. Masters thesis (Masters thesis in Complex Adaptive Systems) – Institutionen för tillämpad mekanik, Göteborg, 2015.

TREUILLE, Adrien; COOPER, Seth; POPOVIĆ, Zoran. Continuum Crowds. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 25, n. 3, p. 1160–1168, 2006.

TRIEBEL, Rudolph et al. SPENCER: A Socially Aware Service Robot for Passenger Guidance and Help in Busy Airports. In: 10TH CONFERENCE ON FIELD e SERVICE ROBOTICS, FSR 2015. PROCEEDINGS... Toronto, Canada: [s.n.], jun. 2015. p. 607–622. Disponível em <https://link.springer.com/chapter/10.1007%2F978-3-319-27702-8_40>. Acesso em 07 fev. 2018.

TRIEBEL, Rudolph et al. SPENCER: a socially aware service robot for passenger guidance and help in busy airports. In: 10TH CONFERENCE ON FIELD e SERVICE ROBOTICS, FSR 2015. PROCEEDINGS... Toronto, Canada: University of Toronto, jun. 2015. p. 21. Disponível em <http://doc.utwente.nl/98273/>. Acesso em 03 fev 2018.

TRIVEDI, Ashutosh; RAO, Shrisha. Agent-Based Modeling of Emergency Evacuations Considering Human Panic Behavior. **IEEE Transactions on Computational Social Systems**, v. 5, p. 277–, mar. 2018.

TSOMPANAS, Michail-Antisthenis I et al. Cellular Automata Applications in Shortest Path Problem. **Shortest Path Solvers. From Software to Wetware**, Springer, p. 199–237, 2018.

TSUI, K. M. et al. Exploring use cases for telepresence robots. In: 2011 6TH ACM/IEEE INTERNATIONAL CONFERENCE ON HUMAN-ROBOT INTERACTION (HRI). PROCEEDINGS... Lausanne, Switzerland: [s.n.], mar. 2011. p. 11–18. Disponível em <https://ieeexplore.ieee.org/document/6281254>. Acesso em 16 jan 2018.

TZAFESTAS, Spyros G. Autonomous mobile robots in health care services (special issue). **J Intell Robot Syst**, p. 177–374, 1998.

_____. **Intelligent Robotic Systems**. New York, NY, USA: Marcel Dekker, Inc., 1991.

_____. **Introduction to mobile robot control**. New York, USA: Elsevier, 2013. (Intelligent Systems, Control and Automation: Science and Engineering).

_____. **Roboethics: A Navigating Overview**. [S.l.]: Springer International Publishing, 2015. (Intelligent Systems, Control and Automation: Science and Engineering). Disponível em <https://books.google.com.br/books?id=EdpECgAAQBAJ>. Acesso em 25 set 2018.

UCROWDS. **Welcome to uCrowds**. [S.l.]: UCrowds, 2019. Disponível em <https://www.ucrowds.com>. Acesso em 15 jan 2019.

ULAM, Stanislaw. On some mathematical problems connected with patterns of growth of figures. In: AM. MATH. SOC. VOL. 14, PROVIDENCE. PROCEEDINGS... [S.l.: s.n.], 1962. v. 14, p. 215–224. Disponível em <https://oeis.org/A002858/a002858.pdf>. Acesso em 02 fev 2018.

USC, Robotics Research Lab. **ROS, Player, Stage and Gazebo**. [S.l.]: USC Robotics, 2019. Disponível em <http://robotics.usc.edu/?l=projects:playerstagegazebo>. Acesso em 15 jan 2019.

VAHID, Frank. **Sistemas Digitais**. Porto Alegra, Rio Grande do Sul: Bookman Editora, 2009.

VALIENTE, David et al. Information-based view initialization in visual SLAM with a single omnidirectional camera. **Robotics and Autonomous Systems**, v. 72, n. 8, p. 93–104, out. 2015. Disponível em <https://www.sciencedirect.com/science/article/pii/S0921889015000974>. Acesso em 09 jun. 2018.

VÁSQUEZ, Álvaro Andrés Saltos. Desarrollo de un sistema de SLAM para el robot AR. Drone en entorno ROS, 2016.

VEMULA, Anirudh; MUELLING, Katharina; OH, Jean. Modeling Cooperative Navigation in Dense Human Crowds. **arXiv preprint arXiv:1705.06201**, 2017.

VILELA, Fernanda. **ICMC desenvolve robô vigilante com sensores 3D e câmera térmica**. [S.l.]: USP, 2013. Disponível em <http://www5.usp.br/30663/robo-vigilante-utiliza-sensores-3d-e-camera-termica-para-monitoramento-de-ambientes-internos>. Acesso em 24 maio 2017.

VIMOS, Ximena Alexandra Vargas. **Diseño de un prototipo de control de acceso del personal mediante reconocimiento facial en 3D para empresas públicas o privadas**. 2016.

f. 102. Trabajo de titulación (Graduación en Ingeniería Electrónica en Control y Redes Industriales) – Escuela Superior Politécnica de Chimborazo, Riobamba, 2016.

WEIDMANN, Ulrich. Transporttechnik der Fußgänger: transporttechnische Eigenschaften des Fußgängerverkehrs, Literaturauswertung. **IVT Schriftenreihe**, ETH Zurich, v. 90, 1993.

WEIMAR, Jörg Richard. **Simulation with cellular automata**. Berlin, Germany: Logos-Verlag Berlin, 1997.

WHITE, S. D. **Military robots**. New York: Book Works, LLC, 2007.

WINTER, Hugo C. Modelling Crowd Dynamics During Evacuation Situations Using Simulation. In: STOR601: PROJECT 2. PROCEEDINGS... Lancashire, England: [s.n.], mai. 2012. p. 1–20. Disponível em <https://www.semanticscholar.org/paper/Modelling-Crowd-Dynamics-During-Evacuation-Using-Winter/6467d87cd6830e6711f05097009b9b0776d2d93c>. Acesso em 03 fev 2018.

WIRGES, Sascha et al. Object Detection and Classification in Occupancy Grid Maps using Deep Convolutional Networks. **arXiv preprint arXiv:1805.08689**, 2018.

WOLF, D. F. et al. Robótica inteligente: Da simulação às aplicações no mundo real., p. 279–330, 2009.

WOLFRAM, Stephen. **A new kind of science**. [S.l.]: Wolfram media Champaign, 2002. v. 5. Disponível em <https://www.wolframscience.com/>. Acesso em 15 ago 2018.

_____. Statistical mechanics of cellular automata. **Reviews of modern physics**, APS, v. 55, n. 3, p. 601, 1983.

_____. **Theory and Applications of Cellular Automata: (including Selected Papers 1983-1986)**. [S.l.]: World Scientific, 1986. (Advanced series on complex systems). Disponível em <https://books.google.com.br/books?id=llBjQgAACAAJ>. Acesso em 09 set 2018.

_____. Universality and complexity in cellular automata. **Physica D: Nonlinear Phenomena**, Elsevier, v. 10, n. 1-2, p. 1–35, 1984.

WOWWEE. **Robots**. [S.l.]: Wowwee, 2017. Disponível em <http://wowwee.com/products/robots>. Acesso em 13 mar 2017.

WUENSCHE, Andrew; LESSER, Mike. **The global dynamics of cellular automata: An atlas of basin of attraction fields of one-dimensional cellular automata**. Santa Fe, New Mexico, USA: Addison-Wesley, 1993. v. 1, p. 1–250.

YANAGISAWA, D. et al. Analysis on pedestrian outflow through an exit with an obstacle. In: 2009 ICCAS-SICE. PROCEEDINGS... Fukuoka, Japan: [s.n.], ago. 2009. p. 5040–5045. Disponível em <https://ieeexplore.ieee.org/document/5334615/>. Acesso em 21 ago 2017.

YOO, Seong-Yeol et al. Preliminary Water Tank Test of a Multi-legged Underwater Robot for Seabed Explorations. In: OCEANS 2015 - MTS/IEEE. PROCEEDINGS... Washington: [s.n.],

out. 2015. p. 1–6. Disponível em <https://ieeexplore.ieee.org/document/7404409/>. Acesso em 22 jun. 2015.

ZALOGA, S. **Unmanned aerial vehicles: robotic air warfare 1917–2007**. Oxford, UK: Osprey Publishing, 2008.

ZHANG, Jiafan; WANG, Yue; XIONG, Rong. Industrial robot programming by demonstration. In: 2016 INTERNATIONAL CONFERENCE ON ADVANCED ROBOTICS e MECHATRONICS (ICARM). PROCEEDINGS... Macau, China: [s.n.], ago. 2016. p. 300–305. Disponível em <https://ieeexplore.ieee.org/document/7606936>. Acesso em 03 fev 2018.

ZHENG, Xiaoping; TINGKUAN, Zhong; MENGTING, Liu. Modeling crowd evacuation of a building based on seven methodological approaches. **Building and Environment**, v. 44, n. 3, p. 437–445, 2009.

ZUBELDIA, Elizabeth Hernández. Uso dos autômatos celulares bidimensionais e imagens tomográficas na geração de meios porosos artificiais, 2014.

APÊNDICE A – SENSORES

Os sensores são a interface perceptual entre robô e ambiente. Sensores passivos, como câmeras, são verdadeiros *observadores*: capturam sinais que são gerados por outras fontes (radiação solar emitida refletida ou sinais emitidos pelos objetos da superfície) (RUSSELL; NORVIG, 2010). Sensores ativos, como o sonar, enviam energia para o ambiente. Eles contam com o fato de que essa energia é refletida de volta para o sensor. Os sensores ativos tendem a fornecer mais informações do que os sensores passivos (RUSSELL; NORVIG, 2010).

Os robôs são equipados com sensores, que lhes permitem perceber seu ambiente (RUSSELL; NORVIG, 2010). Em estruturas de robôs móveis, os sensores podem ser classificados conforme a tabela 6 (WOLF et al., 2009):

Tabela 6 – Tipos de sensores e exemplos.

Sensor	***Principais funções***	***Exemplos***
De posição e orientação	Determinar a posição absoluta ou direção de orientação do robô	GPS (Sistema de posicionamento global), bússola, inclinômetro, triangulação usando marcas (beacons)
De obstáculo	Determinar a distância até um objeto ou obstáculo	Sensor infravermelho, ultrassom (sonar), radar, sensor laser, sistema de visão estéreo
De contato	Determinar o contato com um objeto ou posição de contato com marcação	Sensores de contato, antenas e "bigodes", marcações (barreiras ópticas e magnéticas)
De deslocamento e velocidade	Medir o deslocamento do robô. Medidas relativas a posição e orientação do robô	Inercial, odômetro, potenciômetro, sensores baseados em visão
Outros	Detectar a abertura de portas ou janelas, captar a perturbação de um metal próximo, detectar alteração de temperatura, detectar pressão e força mecânicas	Sensores magnéticos, indutivos, capacitivos, reflexivos, de temperatura, de pressão e de força

Fonte: Autor (adaptado de Dudek e Jenkin (2000 apud WOLF et al., 2009)).

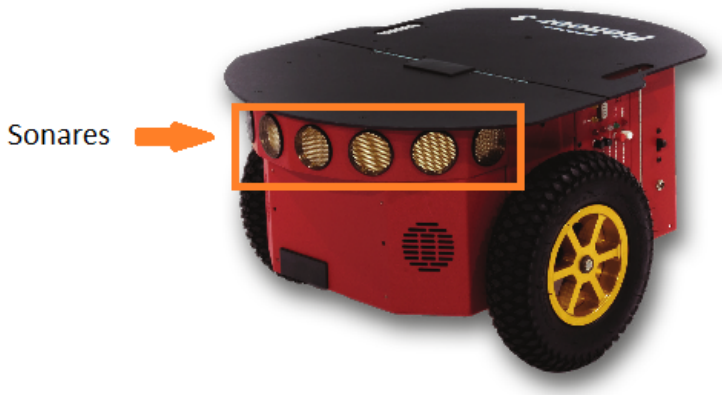
A seguir serão apresentados alguns exemplos de sensores com aplicação em sistemas robóticos autônomos terrestres.

SENSORES DE OBSTÁCULOS

Os sensores de obstáculos medem a distância até os objetos em sua área de operação e são usados para a navegação robótica com o propósito de evitar a colisão com obstáculos (MONDOL, 2016). São exemplos destes sensores:

a) Sonar (Ultrassom): abreviação de Sound Navigation And Ranging ou Navegação e Determinação de Distância pelo Som. São utilizados para detectar obstáculos por meio de energia acústica em frequências acima do que a audição humana. Permite captar a distância entre o robô e uma pessoa ou objeto. Isto pode ser medido por meio do tempo de viagem do eco, uma vez que a velocidade do som é conhecida (KLEEMAN; KUC, 2016). A figura 77 apresenta um exemplo deste tipo de sensor em uma base robótica Pioneer® 3DX. Em comparação com outros sensores, o sonar possui baixo custo, é leve, consome pouca energia e seu esforço computacional é baixo. Ele pode ser utilizado para reconhecimento de objetos de modo a evitar o contato com eles e também para mapeamento de ambientes (KLEEMAN; KUC, 2016);

Figura 77 – Sensores sonar em um Pioneer® 3DX.



Fonte: (MOBILEROBOTS, 2017b).

Entre as maiores limitações do sonar está a baixa resolução angular, ou seja, uma grande abertura do cone de onda impede que as medidas sejam interpretadas como pontos e, neste caso, é mais correto interpretá-las como arcos de circunferência. Isso significa também que, quanto mais longe o obstáculo detectado, menor a certeza quanto à sua localização angular (AZZOLINI, 2011). Outra

limitação importante do sonar é a interferência mútua, ou seja, em geral utilizam-se múltiplos sensores do tipo sonar, mesmo em um só robô, cada um responsável pela detecção de obstáculos em uma determinada faixa angular, porém o sinal emitido por um sonar pode interferir em outro (AZZOLINI, 2011). Para evitar tal fenômeno, em geral, faz-se a detecção em sequência, sendo um sonar por vez. Isso, em contrapartida, diminui bastante a velocidade de detecção (AZZOLINI, 2011).

b) Laser e scanner laser: são capazes de obter resultados robustos em ambientes internos e externos. Semelhante ao sonares, os sensores laser calculam sua distância em relação aos objetos pelo tempo que o feixe de laser é emitido até o momento em que é recebido pelo objeto ou pessoa, usando a velocidade da luz (VAHID, 2009). Alguns exemplos de sensores a laser são apresentados na figura 78: a) sensor a laser da Balluff® com saída analógica, capaz de medir distâncias entre 20 mm e 6 m; b) sensor scanner laser da Leuze Electronic®, capaz de mede distâncias até 65 m; c) sensor scanner laser da Hokuyo® modelo URG-04LX-UG01, que pesa cerca de 160 gramas, consome cerca de 2,5W de energia, mede 50mm x 50mm x 70mm e possui poder de alcance de 5,6 metros.

Figura 78 – Sensores a laser.

(a) Sensor de distância a laser Balluff®. (b) Sensor scanner laser Leuze Electronic®. (c) Sensor scanner laser Hokuyo®.

Fontes: (a) (DIRECTINDUSTRY, 2017c), (b) (LEUZE, 2017), (c) (HOKUYO, 2017)

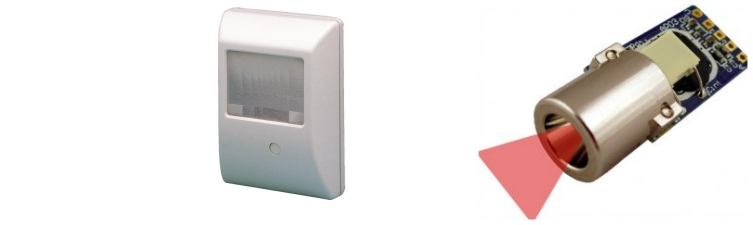
c) Infravermelho: Um sensor de infravermelho - Infrared (IR) - é um dispositivo eletrônico, usado para perceber alguns aspectos do ambiente. Pode medir o calor de um objeto, bem como detectar movimentos (ELPROCUS, 2017). O sensor IR é uma combinação de processador de sinal, indicador de localização sensível e

diodo emissor de laser. O laser é transmitido e é refletido quando atinge um obstáculo (KASSIM et al., 2016). O sensor de IR é usado para medir a distância entre o usuário e o obstáculo. Adotando o método de triangulação, a detecção de distância não será facilmente influenciada pela refletividade diversificada do objeto, pela temperatura do ambiente e pela duração do processamento. Este dispositivo produz uma tensão correspondente à distância de detecção. Assim, este sensor também pode ser usado como um sensor de proximidade (LIU, J. et al., 2014). A figura 79 apresenta um detector de movimento modelo DXS-54, da Nortek®, que utiliza sensor IR. Apresenta também um módulo sensor de temperatura IR que pode ser usado em projetos com arduino, capaz de medir temperaturas ente -33 a +220 graus Celsius.

Figura 79 – Sensores de infravermelho.

(a) Sensor de infravermelho da Nortek®.

(b) Sensor de infravermelho - módulo sensor de temperatura.



Fontes: (a) (NORTEK, 2017); b) (ELPROCUS, 2017)

SENSORES DE CONTATO

Sensores de contato são, normalmente, simples interruptores mecânicos que enviam um sinal quando o contato físico ocorre (MONDOL, 2016). A seguir, é descrito um exemplo deste tipo de sensor.

a) Bumpers (para-choques): Um sensor Bumper é usado para detectar um obstáculos a partir de uma colisão. A figura 80 apresenta uma estrutura de sensores do tipo Bumper e apresenta também esta mesma estrutura, montada em uma plataforma robótica Pioneer® 3DX.

Figura 80 – Estrutura de sensores Bumpers.

(a) Sensor de infravermelho da Nortek®.

(b) Estrutura de sensores Bumper montada em plataforma Pioneer® 3DX.

Fontes: (a) (GENERATIONROBOTS, 2017); b) Adaptada de (MOBILEROBOTS, 2017a)

Assim que o robô colide com algum objeto, ele repete uma série de ações de backup, girando e avançando até que seu caminho fique desobstruído (ALSAAB, 2015). Para Torabi (2015), assim que um sensor Bumper detecta contatos físicos com um obstáculo, um sinal 0 ou 1 é enviado ao robô, sendo 1 a indicação de que está havendo contato com algum objeto e 0 a indicação de que não há qualquer contato com objetos.

SENSORES DE DESLOCAMENTO E DE VELOCIDADE

Estes sensores são utilizados quando pretende-se capturar dados de pessoas, objetos e/ou equipamentos em movimento. A seguir um exemplo deste tipo de sensor.

a) Sensores baseados em visão: a visão do robô é um processo complexo de detecção que envolve extrair, caracterizar e interpretar informações de imagens para identificar ou descrever objetos no ambiente (MONDOL, 2016). Os sensores de visão são utilizados na forma de câmeras, possuem a vantagem de medição a distância e são capazes de obter um grande número de informações do ambiente (CORKE, 2011 apud SAQUI, 2014). A figura 81 apresenta um exemplo de sensor de visão da Sick®, com leitor de código Lector® 65x, resolução de câmera de 2 ou 4 megapixels, lente grande angular e foco dinâmico com altura e profundidade flexíveis. É o primeiro leitor de código baseado em imagens voltado para automação logística e rastreabilidade de ponta-a-ponta em automação industrial.

Figura 81 – Sensor de visão Sick®.

Fonte: (DIRECTINDUSTRY, 2017b).

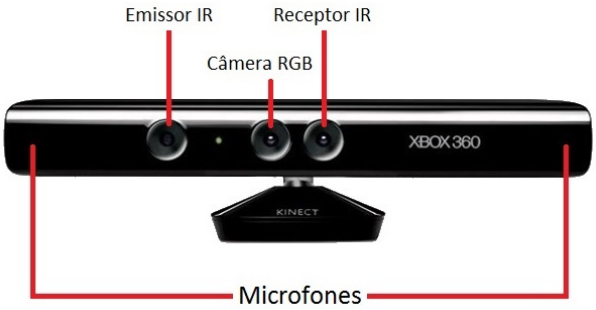
A imagem de uma cena captada por uma câmera é essencialmente uma projeção do mundo 3D (cena) em uma representação 2D (imagem). A maneira em que é feita esta projeção de 3D para 2D é central à capacidade de análise de imagens digitais. O modelo de projeção da câmera transforma coordenadas do mundo em 3D (X, Y, Z) em coordenadas de imagem 2D (x, y) no plano de imagem (SOLOMON; BRECKON, 2013).

SENSOR DE PROFUNDIDADE TRIDIMENSIONAL

Sensores de profundidade permitem a detecção de formas e tamanhos variados de objetos de modo a possibilitar o alcance mais confiável do mapeamento desejado (KAMARUDIN et al., 2015). Um exemplo de sensor deste tipo pode ser encontrado no Microsoft® Kinect®, um sensor de movimentos lançado para aplicações em consoles de jogos eletrônicos. Ele permite aos usuários controlar o jogo com gestos físicos e comandos de voz. A figura 82 apresenta o Microsoft® Kinect®, bem como os recursos que o compõem.

Este dispositivo de mapeamento 3D é composto por uma câmera de infravermelhos, uma câmera RGB, dois microfones e um motor que lhe possibilita variar a inclinação sobre o eixo horizontal. Sua faixa de operação é de 0.8 m - 3.5 m com resolução de 1 cm a uma distância de 2 m (a resolução varia de acordo com a distância) (GANGANATH; LEUNG, 2012; CARRERA, 2013; PEGAS, 2014).

Figura 82 – Microsoft® Kinect®.



Fonte: Autor (adaptada de Microsoft (2017) e Pegas (2014)).

Vilela (2013) e Vimos (2016) descrevem estes elementos da seguinte maneira:

a) Emissor e sensor infravermelhos: são usados para calcular a distância e a profundidade dos objetos e pessoas. Dessa forma, o robô visualiza o espaço onde está inserido e consegue se movimentar dentro daquele ambiente sem colidir com outros objetos. O Microsoft® Kinect® traça um mapa do local onde o robô está, fazendo com que o robô navegue de forma segura.
b) Câmera RGB (vermelho, verde, azul): ajuda a identificar e capturar imagens e vídeos. A câmara tem uma resolução de VGA de 8 bits (640 x 480 pixels) e opera a 30 quadros por segundo.
c) Microfones: localizados nas bordas frontais inferiores do sensor, são usados para o reconhecimento de voz e da fala.
d) Motor de inclinação: um impulso mecânico no sensor de base Kinect® inclina automaticamente o sensor para cima ou para baixo, conforme a necessidade.

Apesar da ideia inicial ser a utilização deste dispositivo na área de jogos, ele se tornou muito importante em outras aplicações. Conde (2017) apresenta uma relação de áreas de pesquisa que têm utilizado o Microsoft® Kinect®. São elas:

a) Reconhecimento e classificação de objetos: reconhecimento de objetos, classificação de objetos e cenas, reconhecimento de postura, reconhecimento de gestos, reconhecimento de ações, reconhecimento humano, reconhecimento facial, tradução de linguagem gestual, caracterização da vegetação.

b) Computação gráfica e Interface Homem-Máquina (IHM): aplicações de realidade aumentada, reconstrução e modelagem de cenas 3d, captura de movimento, animação 3d, acompanhamento de esqueleto, acompanhamento de articulações, rastreamento de mãos, impressões digitais e rastreamento de palma.

c) Robótica móvel: localização e mapeamento (ambos desacoplados e simultâneos), teleoperação.

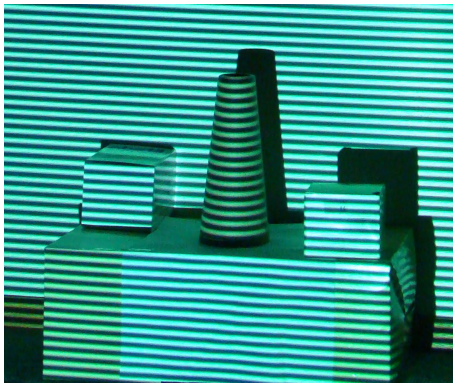
d) Medicina e cuidados de saúde: imagiologia médica, sistemas de reabilitação, assistência aos idosos, detecção de quedas, análise da marcha, assistência visual com deficiência, monitorização do movimento respiratório, rastreio de instrumentos cirúrgicos.

e) Segurança civil: detecção humana, estimativa de fluxo de pessoas.

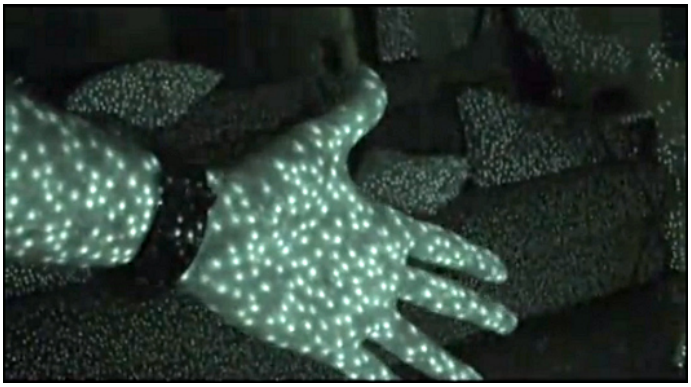
O sensor de profundidade tridimensional utiliza luz estruturada infravermelha para seu funcionamento. Este método consiste em projetar um padrão de luz sobre a cena e observar a deformação do padrão na superfície dos objetos (SZELISKI, 2010; HALL-HOLT; RUSINKIEWICZ, 2001; INOKUCHI; SATO; MATSUDA, 1984 apud CÓRDOVA, 2012). A figura 83a apresenta um exemplo de deformação deste padrão de luz por meio de linhas. No caso específico do Microsoft® Kinect®, o padrão de deformação deste padrão de luz é projetado por meio de pontos, conforme apresentado na figura 83b.

Figura 83 – Exemplo de deformação de luz estruturada infravermelha padrão em objetos de uma cena.

(a) Deformação por meio de linhas.



(b) Deformação por meio de pontos.



Fonte: Adaptada de Córdova (2012).

De acordo com Córdova (2012), a fonte de luz infravermelha do Microsoft® Kinect® projeta um padrão de pontos na cena, que é lido por um sensor IR monocromático CMOS. O sensor detecta os segmentos de pontos refletidos e estima a profundidade a partir da intensidade e distorção dos mesmos.

OUTROS EXEMPLOS DE SENSORES QUE PODEM SER UTILIZADOS PARA DETECÇÃO DE EMERGÊNCIA

Além dos sensores apresentados anteriormente, há também outros sensores que tratam de emergência como, por exemplo, sensores de temperatura e de fumaça, que podem ser utilizados em plataformas robóticas para finalidades específicas, conforme descritas a seguir:

a) Temperatura: dispositivos que transformam a temperatura e suas variações em sinais elétricos que podem ser interpretados por um sistema. Quando a temperatura aumenta, a resistência do dispositivo diminui, podendo ser identificado uma situação de emergência. Estes sensores podem detectar variação de temperatura havendo ou não contato com o objeto, dependendo das características técnicas de cada sensor.

Figura 84 – Sensores de temperatura.

(a) Sensor de temperatura infravermelho Calex Electronics®.

(b) Sensor de temperatura Pasco®.

Fontes: (a) (DIRECTINDUSTRY, 2017a), (b) (PASCO, 2017)

O uso de IR ocorre em sensores de temperatura quando há a necessidade de verificação de variação de temperatura sem contato. A figura 84 apresenta exemplos de sensores de temperatura: i) sensor de temperatura miniatura Calex

Electronics® com IR, que pode ser usado para medir temperaturas que variam de -20 a 500 graus Celsius de objetos e materiais estáticos ou dinâmicos (DIRECTINDUSTRY, 2017a); ii) sensor de temperatura sem fio Pasco® (sensor de contato), que pode ser submetido à temperaturas de -10^0 C a 40^0 C (PASCO, 2017). Robôs equipados com sensores de temperatura são usados para detecção e extinção de incêndios florestais (AFZAAL; ZAFAR, 2016), por exemplo;

b) Fumaça: sensores de fumaça usam um detector fotoelétrico em sua estrutura interna, tendo uma fonte de luz linear em uma extremidade e um sensor que recebe esta luz na outra. Uma vez que o feixe de luz sofre variações ou é interrompido, o sensor percebe a ausência de luz e ativa um sistema qualquer, que pode ser um alarme, por exemplo. Um detector de fumaça de 150 mm x 48 mm pode cobrir, por exemplo, uma área com cerca de 81 m^2 quando instalado em um ambiente livre e desobstruído, no teto;

Neste apêndice foram apresentados alguns sensores que podem ser utilizados no contexto de robôs móveis autônomos terrestres, de acordo as funcionalidades exigidas para as tarefas que precisam cumprir.

O apêndice B apresenta conceitos importantes sobre atuadores, exemplos e principais funções.

APÊNDICE B – ATUADORES

Os seres humanos têm olhos, ouvidos, entre outros órgãos, como sensores e mãos, pernas, trato vocal, entre outros, como atuadores. Um agente robótico pode ter câmeras e detectores de IR como sensores e vários motores como atuadores (RUSSELL; NORVIG, 2010). A tabela 7 apresenta alguns tipos de atuadores, bem como exemplos de aplicação em robôs, baseado em Dudek e Jenkin (2000 apud WOLF et al., 2009). Os atuadores dependem diretamente do ambiente em que o robô irá atuar, no qual deverão ser feitas escolhas primeiramente em relação ao meio: terrestre (indoor, outdoor estruturado ou não), aquático (superfície ou submarino) ou aéreo (avião, helicóptero ou dirigível). No caso de um robô terrestre, também é importante definir melhor o tipo de ambiente para que se possa configurar o método de locomoção adotado, usando rodas, esteiras ou pernas (WOLF et al., 2009).

Tabela 7 – Tipos de atuadores, funções e exemplos.

Atuador	***Principais tipos/funções***	***Exemplos***
Base fixa	Braço robótico com base fixa	Robôs industriais
Base móvel: rodas	2 rodas independentes (diferencial); 3 rodas (triciclo, omnidirecionais); 4 rodas (veículos robóticos-ackermann)	Robôs Khepera e Pioneer 3DX; Robô BrainStem PPRK; Stanley - Stanford
Base móvel: esteira	Esteira	Tanques e veículos militares
Base móvel: juntas e articulações	Bípedes; 4 patas (quadrapods); 6 patas (hexapods)	Robôs humanoides; robôs Sony Aibo, BigDog; Robôs inseto
Base móvel: propulsão, hélices ou turbinas	Veículos aéreos com hélices; veículos aquáticos com hélices; veículos subaquáticos	Aviões, helicópteros e dirigíveis; barcos autônomos; submarinos autônomos
Outros tipos	Braços manipuladores com base móvel; garras com ou sem feedback sensorial; mecanismos de disparo	Garras embarcadas; Mão robótica; Disparo do chute (futebol de robôs)

Fonte: Autor (adaptado de Dudek e Jenkin (2000 apud WOLF et al., 2009)).

.

Com relação aos robôs com rodas, são necessárias, pelo menos, duas delas para garantir a estabilidade de um robô. Entretanto, robôs diferenciais (que possuem duas rodas padrão fixas), têm sua estabilidade garantida somente se o seu centro de massa estiver abaixo do eixo das rodas. Já em robôs com três rodas, a garantia da estabilidade é satisfeita quando o centro de massa do robô está localizado dentro do triângulo formado por elas. Robôs com mais de três rodas necessitam de mecanismos de suspensão de modo a garantir que todas as rodas permaneçam em contato com o solo (NASCIMENTO, 2009). A figura 85a apresenta um robô com quatro rodas.

Os robôs com esteiras possuem direção diferencial que lhe permitem realizar boa manobra em terrenos irregulares. Durante a tração, possui maior contato e fricção com o solo. Os robôs que usam este atuador são mais robustos, dotados de sensores, geralmente na parte de cima, que lhes permitem navegar com certa segurança ao atuarem sobre terrenos irregulares, em atividades como busca e resgate, desarmamento de bombas, entre outras (BRAÜNL, 2008 apud SANTOS KLIPP, 2013). A figura 85b apresenta um exemplo de robô que utiliza esteira. Já na figura 85c, é apresentado um robô com juntas e articulações. Com o uso de juntas e motores angulares, é possível que sejam reproduzidas as diversas articulações presentes em um robô real em seres humanos e nos animais com alto nível de precisão (OSÓRIO et al., 2006 apud HEINEN; OSÓRIO, 2007).

Figura 85 – Exemplos de aplicações de atuadores.

(a) Robô com atuador 4 rodas - Ambot® GRP 4400.

(b) Robô com atuador esteira.

(c) Robô com atuador juntas e articulações - Aibo®.

Fontes: (a) (AMBOT, 2017), (b) (JAVAPOINT, 2017), (c) (SONY, 2017).

Neste apêndice foram apresentados alguns exemplos de atuadores e sua importância em plataformas robóticas móveis autônomas terrestres.