



**UNIVERSIDADE FEDERAL DE OURO PRETO
ESCOLA DE MINAS
COLEGIADO DO CURSO DE ENGENHARIA DE CONTROLE
E AUTOMAÇÃO - CEAU**



ANDRÉ LUIZ MACIEL CID

**AMBIENTE DE SIMULAÇÃO E GERAÇÃO DE MAPAS DE SINAL DE
RÁDIO COM UM ROBÔ MÓVEL TERRESTRE EM AMBIENTES
CONFINADOS**

**MONOGRAFIA DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E
AUTOMAÇÃO**

Ouro Preto, 2021

ANDRÉ LUIZ MACIEL CID

**AMBIENTE DE SIMULAÇÃO E GERAÇÃO DE MAPAS DE SINAL DE
RÁDIO COM UM ROBÔ MÓVEL TERRESTRE EM AMBIENTES
CONFINADOS**

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como parte dos requisitos para a obtenção do Grau de Engenheiro de Controle e Automação.

Orientador: Héctor Azpúrua, Msc.

Coorientador: Prof. Wolmar Araujo Neto

**Ouro Preto
Escola de Minas – UFOP
2021**

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

C568a Cid, Andre Luiz Maciel.

Ambiente de simulação e geração de mapas de sinal de rádio com um robô móvel terrestre em ambientes confinados. [manuscrito] / Andre Luiz Maciel Cid. - 2021.

60 f.: il.: color., gráf., mapa.

Orientador: Me. Hector Azpurua.

Coorientador: Prof. Dr. Wolmar Neto.

Monografia (Bacharelado). Universidade Federal de Ouro Preto. Escola de Minas. Graduação em Engenharia de Controle e Automação .

1. Simulação computacional. 2. Dispositivo automático - Robótica móvel. 3. Telecomunicação. 4. Radiação eletromagnética - Ondas de rádio. I. Azpurua, Hector. II. Neto, Wolmar. III. Universidade Federal de Ouro Preto. IV. Título.

CDU 681.5

Bibliotecário(a) Responsável: Maristela Sanches Lima Mesquita - CRB-1716



FOLHA DE APROVAÇÃO

Andre Luiz Maciel Cid

Ambiente de simulação e geração de mapas de sinal de rádio com um robô móvel terrestre em ambientes confinados

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Engenheiro de Controle e Automação

Aprovada em 16 de abril de 2021

Membros da banca

- [Ms.] - Héctor Ignacio Perez Azpúrua - Orientador (Instituto Tecnológico Vale)
[Dr.] - Wolmar Araújo Neto - Coorientador (Universidade Federal de Ouro Preto)
[Dr.] - Bruno Nazário Coelho - (Universidade Federal de Ouro Preto)
[Dr.] - Gustavo Medeiros Freitas - (Universidade Federal de Minas Gerais)

Bruno Nazário Coelho, membro da banca de avaliação, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 06/10/2023



Documento assinado eletronicamente por **Bruno Nazário Coelho, PROFESSOR DE MAGISTERIO SUPERIOR**, em 06/10/2023, às 17:15, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Luciana Gomes Castanheira, PROFESSOR DE MAGISTERIO SUPERIOR**, em 16/10/2023, às 15:05, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0601584** e o código CRC **5AFA5D64**.

Dedico este trabalho a todos aqueles que já me apoiaram com incentivo ou boas ideias.

AGRADECIMENTOS

Não tem como finalizar uma etapa sem antes agradecer a todas as pessoas que ajudaram de alguma maneira. Por isso agradeço a minha família, que sempre fez de tudo para que eu pudesse chegar aqui. Agradeço aos amigos de longa data e companheiros do curso de automação da Universidade Federal de Ouro Preto, pelo companheirismo, boas ideias e risadas. E por fim, aos funcionários e bolsistas do Instituto Tecnológico Vale, pela oportunidade única de aprendizado e de trabalhar com pessoas incríveis no laboratório de controle e robótica.

“Inteligência é a capacidade de se adaptar a mudanças.” (Stephen Hawking)

RESUMO

O presente trabalho apresenta uma proposta de mapeamento de sinal de rádio com um robô móvel terrestre em um ambiente conhecido, visando disponibilizar um mapa de conectividade entre a base central e o robô. Para tal, é proposto primeiro o desenvolvimento de um ambiente simulado que permita a utilização de um robô de inspeção para ambientes confinados, assim como a viabilidade de uso do robô em situações adversas. Testado a eficiência da simulação, será realizado o mapeamento e localização simultâneos do robô na cena simulada para gerar um mapa de ocupação 2D. Com o mapa pronto, será utilizado uma métrica para qualificar o sinal de rádio em diversos pontos e, com isso, gerar outro mapa contendo a qualidade deste sinal, que pode ser utilizado futuramente para planejar um caminhos considerando a qualidade do sinal de rádio como métrica principal.

Palavras-chaves: Simulação; Robótica móvel; Ambientes Confinados; SLAM; Telecomunicação; Sinal de rádio.

ABSTRACT

This paper presents a proposal for radio signal mapping with a land mobile robot in a known environment, with the aim of providing a connectivity map between the central base and the robot. To this end, it is first proposed to develop a simulated environment that allows the use of an inspection robot in confined environments, as well as the feasibility of using the robot in adverse situations. Once the efficiency of the simulation has been tested, simultaneous mapping and localization of the robot in the simulated scene will be carried out to generate a 2D occupancy map. Once the map is ready, a metric will be used to qualify the radio signal at various points and, with this, generate another map containing the quality of this signal, which can be used in the future to plan paths considering the quality of the radio signal as the main metric.

Key-words: Simulation; Mobile Robot; Confined Spaces; SLAM; Telecommunication; Radio signal.

LISTA DE ILUSTRAÇÕES

Figura 1 – Sensores embarcados no EspeleoRobô (CID et al., 2020).	19
Figura 2 – Tipos de comunicação presente no EspeleoRobô (FILHO, 2019).	20
Figura 3 – Representação de um robô diferencial.	21
Figura 4 – Interface do usuário do CoppeliaSim.	28
Figura 5 – EspeleoRobô real a esquerda e EspeleoRobô simulado a direita.	29
Figura 6 – Árvore simplificada de TFs do EspeleoRobô.	30
Figura 7 – Visão geral do posicionamento de sensores e componentes do robô simulado.	30
Figura 8 – Sistema de câmeras do EspeleoRobô.	31
Figura 9 – Imagem RGB e mapa de profundidade da câmera Intel Realsense simulada.	32
Figura 10 – Sistema de iluminação em ambiente simulado de mina subterrânea: (a) sem iluminação do robô, e (b) sistema de iluminação ativado.	32
Figura 11 – Simulação do Velodyne no CoppeliaSim (a) e visualização no RVIZ dos pontos lidos pelo Velodyne (b).	32
Figura 12 – IMU da Xsens simulada respeitando o sistema de coordenadas da Xsens real (XSENS, 2019). O bloco azul do sensor simulado representa a origem do sistema de coordenadas do sensor real.	33
Figura 13 – Junta de rotação para simular motores do EspeleoRobô.	34
Figura 14 – Ambientes projetados para a simulação: (a) pista de motocross, (b) mina de Ouro, (c) cavidade natural, (d) moinho de bolas, (e) sistema de tuneis da DARPA, (f) laboratório ITVRoC, (g) tubulação e (h) reconstrução interna de moinho com LiDAR.	36
Figura 15 – Análise de desempenho do modelo dinâmico do robô no CoppeliaSim em diferentes cenários. Linhas pretas em cada barra representam o desvio σ	37
Figura 16 – Análise de desempenho dos sensores embarcados no EspeleoRobô utilizados no CoppeliaSim. Linhas pretas em cada barra representam o desvio σ	38
Figura 17 – Interface gráfica do EspeleoRobô. A figura superior mostra a GUI numa inspeção de dutos com o robô real. Embaixo é mostrado a mesma GUI num cenário de mina simulado disponibilizado pela DARPA.	39
Figura 18 – Diagrama de teleoperação do EspeleoRobô através do ROS usando a plataforma real ou simulada.	39
Figura 19 – Plataforma robótica (a) com avaliação de estabilidade calculada com o polígono de sustentação (b).	40
Figura 20 – Pipeline de navegação em dois mapas representativos: pista de motocross (a-b) e sistema de túneis DARPA (c-d). As cores dos caminhos definem a métrica usada: Menor distância (vermelho), transversalidade do terreno (branco), menor consumo de energia (amarelo) e métricas combinadas (verde).	41

Figura 21 – Fluxograma para utilização das velocidades dos motores para localização do robô.	42
Figura 22 – Teste de odometria de rodas em comparação com localização verdadeira do robô (fornecida pelo simulador).	43
Figura 23 – Laboratório de controle e robótica do Instituto Tecnológico Vale, utilizado para teste de mapeamento.	44
Figura 24 – Representação 2D do EspeleoRobô com áreas de colisão.	44
Figura 25 – EspeleoRobô mapeando o laboratório simulado: (a) Processo de mapeamento, onde os pontos coloridos representam as leituras do LiDAR e as áreas escuras ao redor destes pontos são os obstáculos inflados. (b) Mapa de ocupação final do laboratório.	45
Figura 26 – Posição e orientação do robô dados pelo filtro de partículas. A elipse rosa representa a covariância da posição e o triângulo amarelo (abaixo da seta vermelha) a covariância da orientação	45
Figura 27 – Ângulo para manter o sensor de obstáculos apontado para o EspeleoRobô	47
Figura 28 – Distância entre o robô e a antena nos casos de visada direta (a) e a presença de um obstáculo (b).	47
Figura 29 – Ambiente simulado do laboratório: (a) Posição da antena de rádio no mapa (circulado em amarelo), e (b) sensor de obstáculos apontado para o EspeleoRobô.	48
Figura 30 – Mapa de rádio do laboratório simulado.	49
Figura 31 – Mapa do sinal de rádio pelo método da interpolação linear (a) e cúbica (b).	49
Figura 32 – Predição do mapa do sinal de rádio completo pelo método da interpolação linear (a) e cúbica (b).	50
Figura 33 – Resultado do mapeamento de uma seção do mapa, considerando diferentes pontos colhidos: (a) 4 pontos, (b) 15 pontos, (c) 25 pontos.	51
Figura 34 – Simulação de interpolação de sinal de rádio em ambiente representativo: (a) mina Edgar do DARPA challenge SubT, e (b) mapa de ocupação de uma seção do mapa e seu sinal de rádio.	52

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Motivação	14
1.2	Definição do Problema	15
1.3	Objetivos	16
1.3.1	<i>Objetivo Geral</i>	16
1.3.2	<i>Objetivos Específicos</i>	16
1.4	Contribuições	16
1.5	Estrutura do trabalho	17
2	REVISÃO BIBLIOGRÁFICA	19
2.1	EspeleoRobô	19
2.2	Modelo cinemático de um robô diferencial	20
2.3	Simuladores para robótica	22
2.4	Mapeamento e localização	23
2.4.1	<i>Algoritmos de mapeamento e localização simultâneos</i>	24
2.5	Sinal de rádio	25
2.5.1	<i>Interpolação do sinal de rádio</i>	26
3	SIMULAÇÃO DE ROBÔS EM AMBIENTES CONFINADOS	28
3.1	Escolha do simulador	28
3.2	Robô simulado	28
3.3	Sensores e Atuadores Simulados	30
3.3.1	<i>Câmeras</i>	30
3.3.2	<i>RealSense D435i</i>	31
3.3.3	<i>Iluminação</i>	31
3.3.4	<i>Velodyne VPL16</i>	32
3.3.5	<i>IMU</i>	33
3.3.6	<i>Motores</i>	33
3.4	Ambientes Modelados	34
3.5	Avaliação do ambiente simulado	35
3.6	Teleoperação e Treinamento	38
3.7	Análise de Mobilidade	38
3.7.1	<i>Planejamento de Caminhos, Mapeamento e Controle</i>	40
4	MAPA DE INTENSIDADE DO SINAL DE RÁDIO	42
4.1	Sistema de navegação	42

4.1.1	<i>Odometria</i>	42
4.1.2	<i>Mapeamento e localização simultâneos</i>	43
4.1.3	<i>Localização</i>	44
4.2	Simulação do sinal de rádio	46
4.3	Mapeamento do sinal de rádio	47
4.3.1	<i>Comparação entre métodos de interpolação</i>	48
5	CONCLUSÃO	53
	REFERÊNCIAS	55

1 INTRODUÇÃO

A robótica móvel tem-se mostrado cada vez mais uma alternativa viável para a realização automática de várias tarefas nas mais diversas áreas. Exemplos de tais utilizações podem ser observados nas áreas de transportes (HICHRI et al., 2014), organização de estoques (PAOLANTI et al., 2017) e exploração (LIDORIS et al., 2009).

Em processos de mineração, robôs móveis autônomos estão sendo usados cada vez mais devido a fatores como segurança e aumento da produtividade. Tais robôs desempenham papel importante em diversas áreas dos processos de extração mineral, estando presentes tanto em lavras a céu aberto quanto em minas subterrâneas. Nas lavras a céu aberto, além dos caminhões de carga, responsáveis por transportar toneladas de minério para as frentes de beneficiamento, há também trens, perfuradores e veículos para depositar os explosivos nas perfurações em processos de desmonte (TINTO, 2021). Além de aumentar a produtividade, veículos autônomos são responsáveis pela redução no custo de manutenção e aumento da vida útil do equipamento. Segundo a Vale (2018), é esperado um aumento de 15% da vida útil e redução de 10% do custo de manutenção em equipamentos e veículos que foram automatizados.

Já em ambientes subterrâneos, a aplicação de robôs autônomos é estudada tendo em vista os diversos papéis que robôs móveis podem desempenhar em todo o processo de mineração. Além do processo de carregamento de carga em locais perigosos, através de tratores e carregadeiras (ROBERTS et al., 2000), a utilização de robôs para a exploração também tem se mostrado relevante. Minas subterrâneas sofrem constantes alterações devido a exploração do minério e possuem um ambiente dinâmico, logo, manter um mapa local é de grande importância para evitar possíveis riscos e otimizar o planejamento da mina.

Algumas das tarefas mais demandantes em ambientes industriais são inspeções e mapeamento, particularmente em regiões que possam ser perigosas para operadores humanos. Áreas onde o time de inspeção precisa atuar, como tubulações, cavernas, galerias de barragens e tanques, também podem ser consideradas como ambientes confinados. Os espaços confinados são áreas que não foram projetadas para a ocupação contínua pelos humanos. Estes ambientes apresentam condições limitadas para entrada e saída e em muitas situações a ventilação é ruim ou inexistente, sendo comum a falta de oxigênio. Assim, os ambientes confinados são definidos como lugares de alta periculosidade onde existem riscos de desabamento, falta de ar, presença de gases tóxicos ou inflamáveis, fungos, animais peçonhentos, dentre outros riscos relacionados.

Se tratando de mapeamento e exploração, os robôs móveis são utilizados principalmente devido a possibilidade de se locomoverem e atuarem em espaços limitados e de difícil acesso para humanos. Para isso, é necessário que o robô seja devidamente preparado com os sensores e atuadores necessários para o ambiente em que vá trabalhar além de possuir um sistema de

locomoção apropriado, que possa ser utilizado através de um operador ou sistema autônomo. Para robôs não autônomos, é preciso considerar como ocorrerá a comunicação entre a base de operação e o dispositivo móvel. Alguns exemplos de comunicação incluem ondas eletromagnéticas como a utilizada na tecnologia Bluetooth (BUTNARU et al., 2005), infravermelho (ARVIN; SAMSUDIN; RAMLI, 2010) ou através de cabos.

Considerando os riscos de operação em um ambiente confinado, assim como a dificuldade de acesso nestes locais, é proposto o desenvolvimento de um ambiente simulado que permita a realização de testes de navegação, mapeamento, locomoção e treinamento de teleoperação sem a necessidade de por o robô ou o operador em risco. O ambiente simulado também facilitará o desenvolvimento e adaptação do robô, tanto na implementação de algoritmos quanto na análise de viabilidade de alterações mecânicas.

Dado um ambiente confinado, e considerando as dificuldades de atuar em tal ambiente, é proposto a utilização de um robô móvel equipado de sensores como LiDAR 2D ou 3D, Unidades de Medição Inercial e informações da localização do robô para gerar um mapa 2D do local. Simultaneamente ao mapeamento, dados da qualidade de sinal de rádio serão coletados em diversos pontos com a finalidade de gerar um outro mapa de qualidade de sinal de rádio.

1.1 Motivação

Muitas empresas ainda utilizam de operadores para realizar tarefas de inspeções em ambientes confinados. O maior problema destas inspeções é a exposição dos funcionários a perigos característicos destes ambientes. Além disto, é necessário que a empresa realize os devidos treinamentos, forneçam os materiais de proteção adequado para cada situação e disponha de uma equipe para auxílio da operação. Tais atitudes diminuem as chances de ocorrer um acidente porém não extinguem com todos os riscos, logo, é preciso de alternativas para a realização de tarefas de inspeção, como a utilização de plataformas robóticas móveis.

Ambientes confinados também são desafiadores para robôs pelas condições ruins de iluminação, visibilidade e posicionamento global, possibilidade de terrenos irregulares e escorregadiços, interferências nos sensores, entre outros. É necessário que os robôs sejam especialmente projetados para atuarem nos ambientes a serem inspecionados. Tal projeto é necessário pois é preciso garantir que o robô consiga colher os dados necessários, se locomover no ambiente sem intervenção direta de um operador e garantir a integridade da plataforma robótica.

Considerando todos os possíveis riscos de realizar inspeções em ambientes confinados, é ideal que haja uma plataforma adequada que permita realizar diversos testes com o robô sem submetê-lo a capotamento ou até mesmo perda dentro da mina ou cava. Tal plataforma deve permitir a simulação de forma mais próxima ao real para garantir que algoritmos desenvolvidos e utilizados na simulação possam ser implementados no robô real. Considerando ainda o investimento necessário para o desenvolvimento do robô, a simulação se torna essencial para realização

de testes de conceito e para garantir que ele vá cumprir as tarefas para o qual foi projetado.

Levando em conta as condições para que os diferentes sistemas de comunicação sejam usados e suas vantagens e desvantagens, a proposta de mapeamento da qualidade do sinal de rádio é implementada para resolver alguns problemas. O primeiro problema a ser citado é que a utilização de comunicação via cabo se torna inviável para cavernas com muitos obstáculos, havendo o risco deste ficar preso ou até mesmo ser danificado. Além disso, a utilização de um mapa de sinal facilitará a operação com múltiplos robôs, dado que este mapa pode ser utilizado como base para o planejamento de caminhos considerando a qualidade do sinal de rádio. Além disso, o uso de comunicação via cabo gera uma força de arrasto que aumenta conforme a distância entre o robô e a base de comunicação se estende, podendo afetar na carga máxima que o robô consegue carregar e até mesmo na sua mobilidade (MARTZ; AL-SABBAN; SMITH, 2020).

Caso a comunicação com o robô seja feita com cabo, o uso do sistema de mapeamento do sinal de rádio também é interessante devido justamente a essa redundância. Em situações que a comunicação com o cabo falhe por algum motivo, não será necessário realizar o resgate do robô, pois o mesmo já terá o mapa de sinal de rádio, e com isso, percorrerá apenas locais que garantam a comunicação com o operador. Tal resgate o expõe a possíveis danos físicos no seu içamento ou até mesmo na necessidade de algum operador ter que entrar no ambiente para buscá-lo.

Garantir a comunicação constante com o operador é importante para monitorar a plataforma e o ambiente em que o robô se encontra. Se tratando de ambientes confinados, como cavas terrestres e minas, sempre existe a possibilidade de alteração do ambiente devido a um deslizamento, desmoronamento ou outro fenômeno. Caso isto aconteça, mantendo o sinal constante com a base, o operador poderá tomar ciência do fato e conseguir agir antes de colocar o robô em risco.

Analisando o problema de navegação em ambientes internos de outro modo, a implementação de um mapa de rádio também abre precedente para a utilização de mais robôs para a exploração de ambientes confinados. Em sistemas de enxame de robôs, é fundamental que todos os robôs mantenham comunicação constante entre si. Outro possível uso é a criação de rotinas automáticas que auxiliem na recuperação dos robôs caso haja algum problema de comunicação com a base de controle.

1.2 Definição do Problema

Dado um robô móvel terrestre, projetado para realizar inspeções em ambientes confinados, é possível adaptá-lo para um ambiente simulado que seja capaz de fornecer condições semelhantes às que o robô real será submetido. Para tal, é necessário que seu funcionamento, estrutura física, arquitetura de software e sensores também sejam adaptados para a simulação.

Além disto, também é possível analisar quais métodos possibilitam a representação de ambientes de inspeção na simulação. Em seguida, é necessário validar a simulação, considerando o robô e os ambientes simulados, garantindo que estes estejam condizentes com as operações reais do robô.

Considerando o problema de mapeamento de rádio, dado um robô móvel equipado com sensores que permitam tanto o mapeamento 2D quanto sua localização, será primeiramente utilizado um algoritmo de mapeamento e localização simultânea (SLAM) para gerar um mapa de ocupação do ambiente. Simultâneo a tarefa de mapeamento, ocorrerá, em certos intervalos de tempos, a medição da intensidade de sinal de rádio captada pelo robô. Com tais valores, um algoritmo de interpolação será aplicado para gerar um segundo mapa de intensidade que considera a qualidade do sinal assim como o locais acessíveis para a locomoção do robô.

1.3 Objetivos

1.3.1 Objetivo Geral

Desenvolver metodologia de simulação para robôs de inspeção em ambientes confinados e estudo de viabilidade de um algoritmo de mapeamento de sinal de rádio.

1.3.2 Objetivos Específicos

- Adaptação do robô real para o simulador.
- Analisar métodos de inserção de diferentes ambientes modelados para o simulador.
- Analisar a responsividade de diferentes modelos e sensores e o quanto estes impactam na simulação.
- Realizar testes de teleoperação e navegação autônoma para averiguar usabilidade do simulador.
- Gerar um mapa de ocupação do ambiente usando LIDAR ou sensores de profundidade como câmeras estéreo e outros dispositivos.
- Estimar a qualidade do sinal de rádio em certos intervalos de tempo.
- Interpolar os dados do rádio para gerar um mapa de calor do sinal de rádio.
- Fusão do mapa de rádio com o mapa de ocupação.

1.4 Contribuições

Durante a realização deste trabalho foram realizados os seguintes relatórios técnicos e publicações em congressos:

Relatórios Técnicos:

- *Alexandre Magno; Nilton Costa; André Cid; Maurício Sathler; Levi Resende; Mateus Nazário; Mário Torres; Héctor Azpúrua; Gustavo Pessin. **Inspeção da Tubulação de Rejeitos das Usinas de Salobo.** (2019).*
- *Maurício Sathler; André Cid; Mateus Nazário; Jacó Domingues; Paulo Alves; Héctor Azpúrua; Gustavo Pessin. **Módulo de Sensores de Gás Embarcado no EspeleoRobô.** (2020).*
- *Gustavo Freitas; Adriano Rezende; Vitor Miranda; Gilmar Cruz; Rafale Silva; Alvaro Araujo; Carolina Fany; David Marques; Douglas Coutinho; Lucas Matos; Lucca Leão; Héctor Azpúrua; Jacó Domingues; Luís Guilherme Dias; André Cid; Mário Torres; Mateus Nazário; Maurício Sathler; Paulo Alves; Rodrigo Teotônio; Gustavo Pessin. **Quarto relatório técnico de acompanhamento do projeto Dispositivo Robótico de Inspeção de Ambientes Restritos e Confinados.** (2021).*
- *Paulo Alves; Rodrigo Teotônio; André Cid; Mário Torres; Jacó Domingues; Luís Guilherme Dias; Gustavo Freitas; Gustavo Tressia; Philip Pritzelwitz; Héctor Azpúrua; Gustavo Pessin. **Dispositivo Robótico de Inspeção de Trilhos.** (2020).*

Publicações em congressos nacionais:

- *Matheus Lima, et al. **Realimentação de Força para Teleoperação de Escavadeiras.** Anais do 14º simpósio brasileiro de automação inteligente (SBAI), 2019, Ouro Preto.*

Publicações em congressos internacionais:

- *Cid, André, et al. "A Simulated Environment for the Development and Validation of an Inspection Robot for Confined Spaces." 2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE).*

1.5 Estrutura do trabalho

O restante do trabalho está organizado da seguinte forma: O Capítulo 2 apresenta uma revisão da literatura sobre simuladores utilizados na robótica, alguns dos processos e algoritmos de mapeamento e métodos de interpolação do sinal de rádio. O robô e ambiente simulados são apresentados no Capítulo 3, juntamente com os sensores embarcados e técnicas utilizadas para a construção de ambos. Ainda no Capítulo 3 são apresentados os resultados de diversos experimentos realizados no ambiente simulado. Os testes englobam desde o tempo de resposta

da simulação, considerando diversos sensores assim como a utilização do robô para teleoperação e rotas autônomas. No Capítulo 4 é demonstrado a implementação dos algoritmos de SLAM, interpolação e mapeamento do sinal de rádio, assim como resultado de testes realizados em simulação. Finalmente, no Capítulo 5 encontra-se uma análise do método proposto, assim como as considerações finais e sugestões para trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

Dado a tarefa de desenvolvimento de um ambiente virtual que possa ser utilizado para realizar a simulação de um robô de inspeção, alguns pontos merecem destaque, como a estrutura do robô, seu modelo cinemático e o simulador utilizado. Já para a tarefa de averiguar a viabilidade de um algoritmo de mapeamento de sinal de rádio, o mapeamento e localização, assim como o estudo da propagação do sinal de rádio são necessários. Neste capítulo serão apresentados a base teórica e trabalhos relacionados necessários para o desenvolvimento dos métodos de simulação de robôs e mapeamento de sinal desenvolvidos neste trabalho.

2.1 EspeleoRobô

O EspeleoRobô é uma plataforma móvel desenvolvida pelo Instituto Tecnológico Vale (VALE, 2016; AZPURUA et al., 2019) que possui seis motores, com sistema de locomoção intercambiável que permite a adaptação do robô em diferentes tipos de terreno (ROCHA et al., 2017). Através de pinos de engate rápido é possível utilizar o robô com rodas, pernas, esteiras, rodas tipo estrela e configurações híbridas. O robô é equipado com 6 conjuntos de reduções planetárias, motores MCD EPOS da *Maxon Motors* que possuem *encoders* e drives já integrados e duas baterias de padrão militar da *Bren-Tronics*. Ele é projetado para se locomover em cavidades e colher informações por meio de câmeras de alta resolução, câmeras 360, sensores de profundidade (câmera estéreo e LIDAR), conforme Figura 1, além de poder ser equipado com outros sensores como de qualidade do ar e temperatura.

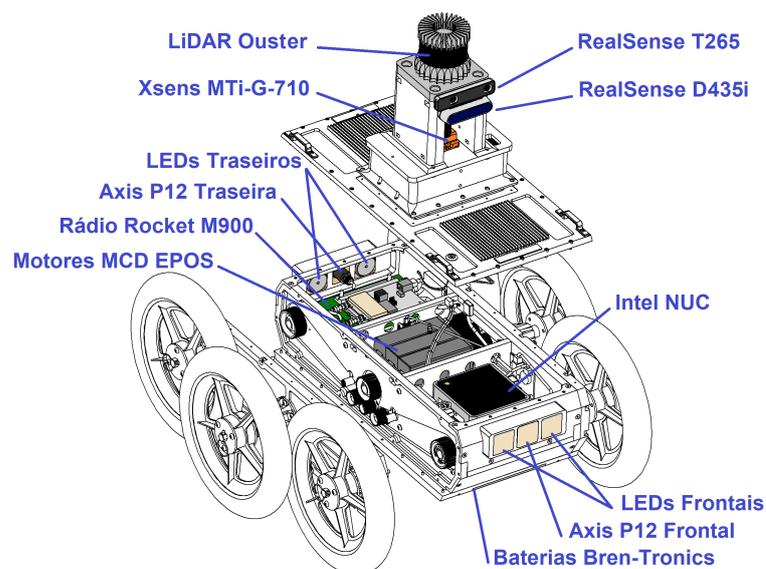


Figura 1 – Sensores embarcados no EspeleoRobô (CID et al., 2020).

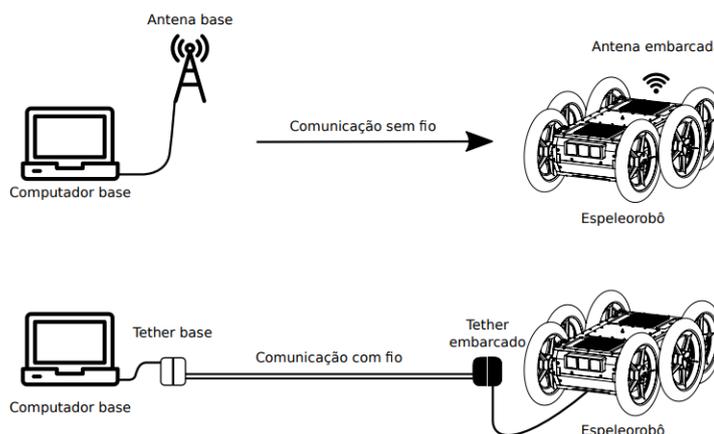


Figura 2 – Tipos de comunicação presente no EspeleoRobô (FILHO, 2019).

A comunicação entre o operador e o EspeloRobô pode ser realizada através de sinal de rádio, por meio do transmissor *Ubiquiti Rocket M900* e antenas receptoras, além de comunicação através de duas placas *Fathom-X board* conectadas por um cabo de 200 metros, conforme Figura 2. Seus dispositivos são controlados por um mini computador *Intel NUC 6i5SYH* através do ROS (Robot Operation System), um framework open-source para programação de robôs, que permite a modularização de todo o sistema, facilitando a utilização de dispositivos específicos e reduzindo o tempo de eventuais mudanças na sua programação (ROS, 2010).

2.2 Modelo cinemático de um robô diferencial

Para utilizar robôs móveis autônomos é preciso conhecer sua descrição matemática, conhecida como modelo cinemático. Por possuir configurações de quatro ou seis rodas, o EspeleoRobô pode ser modelado como um veículo Skid-steering (ROCHA et al., 2017), porém tal modelagem precisa ser adaptada para cada terreno que o robô for atuar, isso pois é necessário saber as forças de interação entre o terreno e os pneus. Considerando tal problema, e a fim de simplificar as operações de navegação, o EspeleoRobô pode ser considerado como um robô diferencial.

O modelo diferencial se caracteriza por possuir apenas duas rodas no mesmo eixo, tracionadas de forma independente, conforme Figura 3. Para encontrar as velocidades a serem aplicadas nos motores, deve-se utilizar primeiro a Equação 2.1 para encontrar a velocidade linear (v) e a velocidade angular (w).

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & 0 \\ \sin(\varphi) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (2.1)$$

Em seguida, a equação 2.2 é utilizada para encontrar as velocidades a serem aplicadas

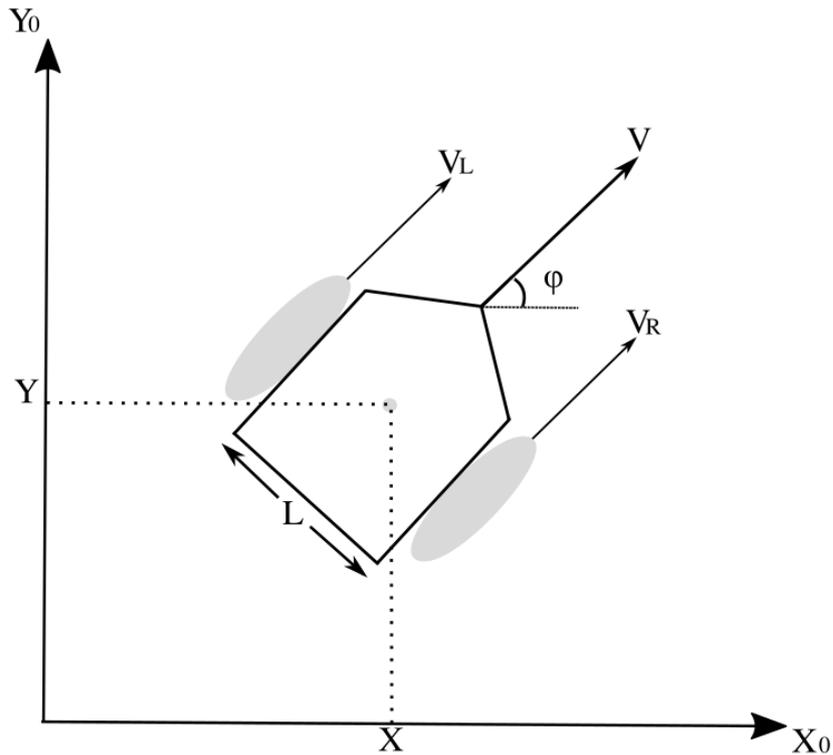


Figura 3 – Representação de um robô diferencial.

nas rodas da esquerda (v_L) e da direita (v_R), considerando o distanciamento entre rodas L :

$$\begin{bmatrix} v_R \\ v_L \end{bmatrix} = \begin{bmatrix} 1 & \frac{L}{2} \\ 1 & -\frac{L}{2} \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (2.2)$$

Além de permitir encontrar as velocidades aplicadas às rodas, a modelagem cinemática do robô também permite estimar a posição do robô no plano. Para isto, basta encontrar as velocidades no plano X,Y, conforme Equação 2.3 e então integrar em função do tempo, conforme Equação 2.4.

$$\begin{bmatrix} v_X \\ v_Y \end{bmatrix} = \begin{bmatrix} \cos(\varphi) \\ \sin(\varphi) \end{bmatrix} \begin{bmatrix} v \end{bmatrix} \quad (2.3)$$

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta \varphi \end{bmatrix} = \begin{bmatrix} v_X \\ v_Y \\ \varphi \end{bmatrix} \begin{bmatrix} \Delta t \end{bmatrix} \quad (2.4)$$

A utilização das fórmulas acima em um robô Skid-steer gerará erro na sua localização (COTA, 2019), devido principalmente ao escorregamento das rodas (ANOUSAKI; KYRIAKOPOULOS, 2004). Logo, se torna necessário a fusão de dados de diversos sensores para aumentar a precisão da localização.

2.3 Simuladores para robótica

A função básica de um simulador é reproduzir com fidelidade o ambiente e o robô com seus determinados atributos, sendo uma ferramenta indispensável para prototipagem de sistemas robóticos. Um simulador possibilita testar ideias em ambientes seguros e controlados, evitando danos à estrutura do robô. Escolher o simulador apropriado é importante, pois diferentes ambientes de simulação oferecem desempenhos, detalhamentos e características variadas. Além de evitar danos ao robô, as simulações tornam possível testar rapidamente algoritmos, executar testes de teleoperação e treinamentos para sistemas que utilizam inteligência artificial (IA).

Embora interagir fisicamente com o robô seja crítico para a operação do mesmo, a simulação também faz importante parte no ciclo de desenvolvimento de sistemas robóticos. Desta forma, existem vários softwares de simulação de robôs que podem ser usados.

Considerando a arquitetura do EspeleoRobô (FILHO, 2019), é importante que o simulador escolhido tenha compatibilidade com o ROS. O *Robot Operating System* (ROS) é um meta sistema operacional de código aberto, que contém os serviços esperados de um sistema operacional, incluindo abstração de hardware, controle de dispositivos baixo nível, implementação de funcionalidades comumente utilizadas, passagem de mensagens entre processos e manutenção de pacotes (QUIGLEY et al., 2009). O ROS também fornece ferramentas e bibliotecas para obter, construir, escrever e executar códigos através de múltiplos computadores. Uma das ferramentas mais usadas do ROS é o Rviz, um software de visualização 3D (ROS, 2018b) que auxilia o usuário a enxergar, interagir e entender como os dados estão sendo interpretados pelo robô. Devido a sua robustez, o ROS é amplamente utilizado para o desenvolvimento de diversos robôs, tanto na indústria quanto para pesquisas.

Simuladores de robótica, quando integrados com motores de física/dinâmica, representam melhor o ambiente real além de simular interações como peso, atrito, esforço, entre outras características de objetos estáticos e dinâmicos. Cada motor possui características e desempenhos diferentes, tanto na velocidade de processamento quanto na fidelidade da reprodução do modelo real. Alguns dos motores de física mais usados na robótica são: Open Dynamics Engine (ODE) (ODE, 2020), Bullet engine e a sua versão em python PyBullet (BULLET, 2020), e o Vortex (CM-LABS, 2020).

- **Gazebo:** Criado para ser o sucessor de simuladores como *Player* e *Stage*, que já se encontram em desuso, Gazebo (KOENIG; HOWARD, 2004) foi desenvolvido para permitir a simulação de robôs móveis em ambientes externos. Este é um simulador compatível com ROS, desenvolvido pela Open Robotics, com código aberto e livre. O simulador dinâmico padrão é ODE e motores opcionais podem ser adicionados por meio de plug-ins. Este é um dos simuladores mais populares dada a grande comunidade ao seu redor, controles simplificados e sua construção em torno do ecossistema ROS.

- **CoppeliaSim:** Sucessor do VREP, é um software comercial de código fechado que pode ser obtido gratuitamente em sua versão educacional (COPPELIAROBOTICS, 2019). O ambiente de desenvolvimento integrado no CoppeliaSim é baseado na arquitetura de controle distribuído, permitindo que os objetos e modelos sejam controlados de forma individual via scripts. Este simulador possui múltiplas opções de motores de física realística.
- **Webots:** É um simulador de robótica, lançado pela empresa suíça (CYBERBOTICS, 2020), que visa a facilidade no uso. Este é um simulador gratuito de código aberto que também pode ser controlado externamente usando sua API ou qualquer rede TCP/IP padrão, tem suporte ao ROS por padrão e possui o motor de física baseado no ODE.

Além dos simuladores anteriores, que possuem propósito geral, existem simuladores que tem recursos mais específicos para outras tarefas como carros autônomos ou enxames de robôs. Autonomous Robots Go Swarming (ARGoS) foi desenvolvido no projeto Swarmanoid especificamente para simulação de enxames e possui parte gráfica simples com desempenho aprimorado (PINCIROLI et al., 2012). Com arquitetura baseada em plugins, é possível realizar a troca do motor de física e adicionar sensores via estes módulos.

2.4 Mapeamento e localização

Para que um robô seja capaz de se locomover de forma autônoma, é necessário que ele possua sensores para entender, descrever e se localizar no ambiente em que se encontra. Além de sensores, é preciso utilizar algum algoritmo de navegação que será responsável pela tomada de decisão das rotas, realizando o desvio de obstáculos e respeitando outras possíveis métricas. Sensores como câmeras e LIDAR são os mais utilizados para detecção de possíveis obstáculos, além de poder serem usados para auxiliar na localização do robô.

Segundo Thrun (2003), o mapeamento é o processo de adquirir um modelo espacial do ambiente em que o robô se encontra, dessa forma, tal processo possui alguns problemas que devem ser levados em conta ao construir um mapa, sendo eles:

- **Dimensionalidade:** Está relacionada a quantidade de dados necessário para representar um mapa. Em mapas 3D este número tende a ser bem maior que em um mapa 2D, aumentando assim sua complexidade.
- **Problema de correspondência:** Ocorre quando é necessário saber se um obstáculo lido por um sensor já foi processado em outro momento. Este é um ponto importante pois o robô tende a acumular erro com o tempo, aumentando a chance de registrar pontos iguais em localizações diferentes do ambiente mapeado.

- **Mudança do ambiente:** Não há a garantia que os locais a serem percorridos permaneçam inalterados a cada inspeção. A exemplo, pode ocorrer mudanças nas disposições dos móveis do cômodo, caso seja um robô para ambientes internos.
- **Caminho a ser percorrido ao mapear:** Se tratando de um robô de exploração, é preciso levar em consideração o caminho a ser percorrido. Tal caminho é importante pois vários aspectos podem influenciar no mapeamento, como possível perda da localização do robô, energia consumida e áreas com prioridade de mapeamento.

Considerando os problemas citados acima, a plataforma a ser utilizada e o problema de mapeamento do sinal de rádio, o primeiro passo é encontrar um método viável de mapeamento do ambiente que permita que o robô percorra esse mapa registrando os sinais de rádio.

Atualmente o EspeleoRobô pode ser equipado com diversos sensores para detecção do ambiente, sendo eles o LiDAR Hokuyo UTM-30LX, o Ouster OS1-16 e as câmeras estéreo Intel Realsense D435i, ZED e ZED mini. A escolha de qual sensor usar depende da finalidade da inspeção que o robô irá fazer. As câmeras permitem mapeamento 3D do ambiente e ainda respeitam as texturas do local, porém possuem alcance limitado (até 15 metros, dependendo da câmera) e campo de visão pequeno. Já os sensores que utilizam da tecnologia do LiDAR permitem realizar o mapeamento com distâncias maiores e sem a necessidade de manobrar várias vezes o robô para tentar capturar o ambiente todo. Sua maior desvantagem comparado às câmeras é a falta de textura do ambiente.

Além dos sensores para leitura do ambiente, o EspeleoRobô possui dois métodos para gerar sua odometria ou localização. O robô é equipado com encoders que disponibilizam a velocidade de rotação de cada roda. Com tal velocidade, e levando em conta seu modelo dinâmico, é possível calcular a distância percorrida pelo EspeleoRobô assim como sua posição no plano em relação a sua posição inicial. O EspeleoRobô também pode ser equipado com uma câmera Intel Realsense T265, que gera dados de odometria através de dados visuais.

Outro fator importante é o método de mapeamento. Sensores e atuadores possuem certa incerteza, tornando ainda mais complicado a construção do mapa. A melhor alternativa para tratar tal problema é a utilização de um método probabilístico de mapeamento. Tais métodos levam em conta as incertezas dos sensores, tornando o mapa gerado mais preciso. Segundo [Milford \(2008\)](#), há três abordagens principais, onde cada método possui suas vantagens e desvantagens, que são: Filtro de Kalman, algoritmos de maximização de expectativa e filtro de partículas.

2.4.1 Algoritmos de mapeamento e localização simultâneos

No ROS há vários algoritmos de SLAM já implementados e prontos para serem usados em diferentes plataformas robóticas, sendo necessário apenas alguns ajustes para melhor adequação ao robô. Levando em conta o problema da dimensionalidade, é possível dividi-los em dois grupos: Os que constroem o mapa em duas ou três dimensões (2D ou 3D). Os mapas em 3D,

como Octmap (HORNUNG et al., 2013a) e Rtabmap (LABBÉ; MICHAUD, 2018), são os mais utilizados pela comunidade devido a fácil instalação e alta performance. Já os mais utilizados para o mapeamento 2D são Gmapping, Hector Slam (KOHLBRECHER et al., 2011) e Google Cartographer (HESS et al., 2016).

Tais mapas são utilizados em problemas de mapeamento e localização simultâneas quando há incerteza nos dados fornecidos pelos sensores e a posição do robô (THRUN, 2002). Cada célula do mapa pode ser definida de forma binária, onde cada uma pode ter um valor atribuído como ocupada (quando há um obstáculo) ou livre.

Para o trabalho proposto, a construção de um mapa de ocupação, que servirá como base para locomoção do robô gerar o mapa de sinal de rádio, é importante. Logo, para teste de conceito e aplicabilidade, um algoritmo que consiga gerar um mapa 2D basta.

2.5 Sinal de rádio

Quando utilizadas para sistemas de telecomunicação, é preciso considerar as várias possíveis perturbações que o sinal de rádio pode sofrer. Em ambientes confinados, como cavernas, as ondas de rádio se propagam por meio do fenômeno da difração e estão susceptíveis principalmente aos efeitos das ondas de multi percurso, que ocorre quando ondas secundárias, decorrentes de reflexões nas paredes e relevo das cavernas, chegam ao receptor de sinal com diferentes intensidades e defasadas em relação à onda principal, causando alteração no sinal resultante e conseqüentemente o desvanecimento do sinal recebido (SERRALHEIRO, 2017).

Considerando tais perturbações, é necessário encontrar um modo de qualificar o sinal de rádio. A qualificação da comunicação pode ser realizada considerando diferentes métricas como a latência do sinal, a perda de pacote ou potência do sinal que chega no receptor.

- **Latência:** Responsável por dizer o intervalo de tempo entre o envio da mensagem de um ponto da rede até sua chegada em outro ponto. Esta métrica é muito importante para ser analisada em casos de teleoperação do robô pois está diretamente ligada ao tempo de resposta, apesar de não dizer sobre a comunicação estar constante ou não. A exemplo, o rover construído pela NASA (NASA, 2021), que se encontra em Marte, possui latência variável de cinco a vinte minutos, dependendo da posição dos planetas.
- **Perda de pacotes:** A perda de pacotes diz respeito justamente a possível perda de informação quando dados são transmitidos de um ponto ao outro. Caso grande porcentagem de dados seja perdida, é possível que haja falha nas transmissões das imagens de câmera, arquivos corrompidos e até mesmo a perda de comunicação com o robô.
- **Potência de sinal:** A potência do sinal que chega no receptor está relacionada à intensidade do sinal lido. Há duas grandezas que representam esta intensidade: Decibéis por miliWatt (Dbm) ou Received Signal Strength Indicator (RSSI). Ambos se diferem pelo RSSI ser

uma medida relativa e Dbm ser um valor absoluto. Com baixa intensidade de sinal, o sistema começa a apresentar falhas como perda de pacotes e até mesmo aumento da latência da rede.

2.5.1 Interpolação do sinal de rádio

Considerando que a coleta de dados da potência do sinal de rádio é realizada de forma discreta, torna-se interessante aplicar métodos de interpolação nos pontos medidos para conseguir um mapa mais denso e diminuir consideravelmente o tempo de aquisição dos dados e criação do mapa.

Uma alternativa é a utilização de processos gaussianos (HÄHNEL; FOX, 2006). Além de possuir alta acurácia no mapa gerado, com este método, é possível realizar a predição do sinal em áreas não visitadas. Um fator importante na utilização de processos Gaussianos é seu alto custo computacional e necessidade de dados para o treinamento do algoritmo.

Outra forma mais simples de gerar um mapa a partir dos pontos de rádio coletado é através de métodos de interpolação. Arai e Tolle (2013) utilizaram de um algoritmo de interpolação linear para criar um mapa de rádio a partir alguns pontos de sinal lidos. Para encontrar a posição desconhecida (x, y) , utiliza-se a equação 2.5, onde (x_0, y_0) e (x_1, y_1) são pontos conhecidos e x um valor dentro do intervalo de x_0 a x_1 . Após a interpolação no plano, ocorre a interpolação da intensidade do sinal de rádio.

$$y = y_0 + (x - x_0) * \frac{y_1 - y_0}{x_1 - x_0} \quad (2.5)$$

A interpolação linear pode ser utilizada também através do método de Dalaunay, que fornece o sinal interpolado P_RSSI a partir dos triângulos (x_1, y_1, P_1) , (x_2, y_2, P_2) e (x_3, y_3, P_3) formado por três pontos conhecidos e seus respectivos sinais de potência de sinal (TALVITIE; RENFORS; LOHAN, 2015), conforme equação 2.6. Os valores λ_1 e λ_2 são definidos como as coordenadas do baricentro do triângulo, conforme equação 2.7 a 2.9.

$$P_RSSI = \lambda_1 P_1 + \lambda_2 P_2 + \lambda_3 P_3 \quad (2.6)$$

$$\lambda_1 = \frac{(y_2 - y_3)(x - x_3) + (x_3 - x_2)(y - y_3)}{(y_2 - y_3)(x_1 - x_3) + (x_3 - x_2)(y_1 - y_3)} \quad (2.7)$$

$$\lambda_2 = \frac{(y_3 - y_1)(x - x_3) + (x_1 - x_3)(y - y_3)}{(y_2 - y_3)(x_1 - x_3) + (x_3 - x_2)(y_1 - y_3)} \quad (2.8)$$

$$\lambda_3 = 1 - \lambda_1 - \lambda_2 \quad (2.9)$$

Para a geração dos mapas de potência do sinal de rádio, o mais usual é a utilização de funções que realizem a interpolação das coordenadas e do valor de intensidade do sinal de rádio juntos, pois estes permitem a visualização do mapa de forma contínua, gerando assim um mapa de calor com os sinais obtidos. Dentre estes métodos, os mais utilizados são: Interpolação bilinear, Interpolação cúbica e por vizinho mais próximo (DOURADO, 2014), sendo a interpolação bilinear e cúbica uma variação do método linear, tendo como diferença apenas o número de pontos utilizados para o cálculo do valor interpolado.

Outros métodos como Interpolação por função de base radial (RBF) (KRUMM; PLATT, 2003), inverso do quadrado da distância (IDW) (BI et al., 2018) e Krigagem (ZHAO; HUANG; JIA, 2016) também são utilizados para a interpolação do sinal de rádio. Estes se diferenciam por utilizar de uma função base de regressão para realizar a interpolação, sendo que a escolha desta função afetará diretamente nos resultados.

3 SIMULAÇÃO DE ROBÔS EM AMBIENTES CONFINADOS

Neste capítulo serão apresentados os procedimentos de simulação de ambientes confinados desenvolvido para o EspeleoRobô. O ambiente é baseado no CoppeliaSim, que é um simulador de alta precisão, e serão mostrados as configurações e adaptações realizadas para validação de tarefas de inspeção e mapeamento com robôs móveis terrestres em ambientes como cavernas, túneis e outros ambientes com restrição de localização global.

3.1 Escolha do simulador

Dados os simuladores citados na Seção 2.3, o CoppeliaSim e o Gazebo se adaptam melhor a aplicações com um único robô móvel em grandes ambientes detalhados, como mostrado por Melo et al. (2019), incluindo o uso para simulação de robôs de inspeção (CISZEWSKI et al., 2017). Um simulador para espaços confinados deve ter uma representação confiável do ambiente, fácil modelagem e edição de robôs móveis terrestres, suporte para diversos sensores, boa documentação, interface gráfica intuitiva, conforme Figura 4, e suporte para ROS. Desta forma, o CoppeliaSim foi selecionado como o simulador adequado para o robô de inspeção em ambientes confinados, dada a fidelidade nas interações do robô com outros objetos, a facilidade de manipulação de modelos e malhas diretamente no sistema virtual (PITONAKOVA et al., 2018) e facilidade para desenvolver algoritmos de planejamento (LI; GAO, 2019).

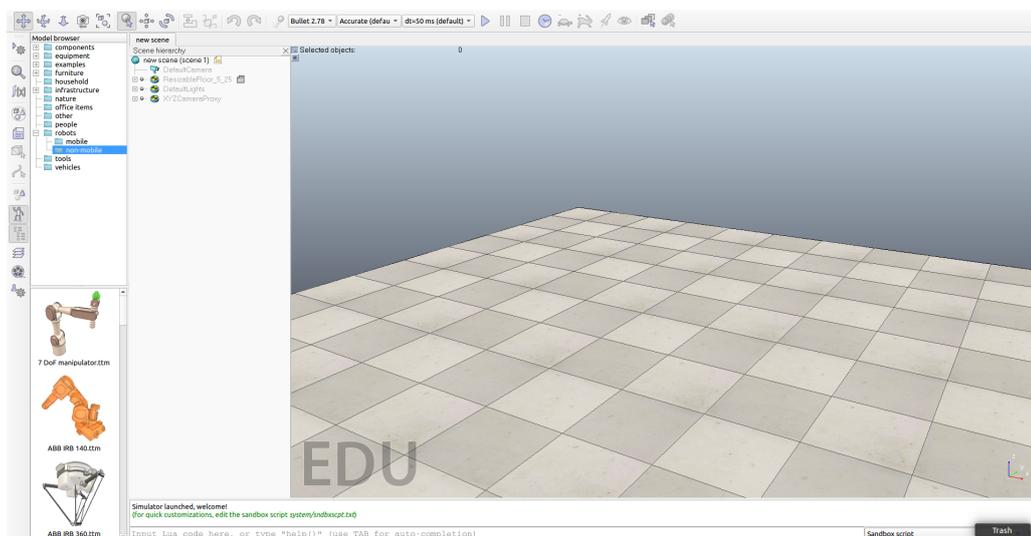


Figura 4 – Interface do usuário do CoppeliaSim.

3.2 Robô simulado

O EspeleoRobô foi desenvolvido utilizando processos computacionais como Computer Aided Engineering (CAE) e Computer Aided Design (CAD). Estes processos se tornaram

ferramentas essenciais à engenharia devido a redução do tempo de design, rápida produção de protótipos e alta qualidade do produto final (SHIH, 2020).

Para realização de desenhos e elementos de construção mecânica, optou-se pela utilização do SolidWorks devido a quantidade de recursos disponíveis em modelagem e simulação. O SolidWorks é um software de modelagem 3D para sistemas Windows que possui um banco de dados de materiais e suas propriedades e que permite a simulação de propriedades visuais e mecânicas dos elementos. Assim, podem ser realizadas simulações de movimento, tensões, deformação, escoamento, dentre outros recursos que foram utilizados para validar inicialmente o conceito do robô antes de realizar a sua fabricação.

A importação dos modelos do CAD no ambiente de simulação do CoppeliaSim requer o uso de algum dos formatos suportados, dos quais STL e Unified Robot Description Format (URDF) são os mais utilizados. A principal diferença entre estes dois formatos é que o STL salva apenas gráficos e todas as juntas de rotação e texturas devem ser posicionadas manualmente dentro do software, o que demanda um grande tempo. O formato URDF, amplamente usado na robótica, permite manter as propriedades de montagem, relações físicas, texturas e propriedade dos materiais utilizadas no CAD dentro do ambiente simulado, facilitando a construção da simulação. Dado que o SolidWorks não possui compatibilidade com URDF de forma nativa, é necessário a instalação de um plug-in para usar este formato. O robô simulado, ao lado do robô real, pode ser visto na Figura 5.

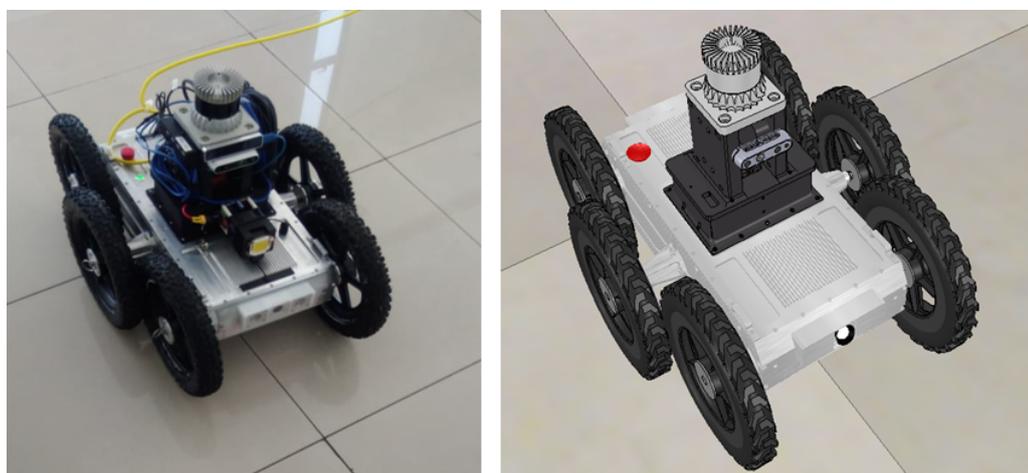


Figura 5 – EspeleoRobô real a esquerda e EspeleoRobô simulado a direita.

A representação do posicionamento das partes do robô, como seus sensores, atuadores e juntas passivas, está relacionada com seus frames. Os frames são sistemas de coordenadas globais que contém posição (x, y, z) e orientação, geralmente descrita usando quatérnions (i, j, k, ω) ou ângulos de Euler (roll, pitch, yaw).

No ROS é usada a ferramenta TF para resolver este problema. Essa ferramenta adota um padrão de representação do robô com coordenadas e unidades de medida, mantém o relacionamento entre os frames de coordenadas contido em estrutura de árvore e possibilita o usuário a

transformar pontos, vetores e planos entre quaisquer dois frames de coordenadas (ROS, 2017). A árvore de TF simplificada do robô pode ser observada na Figura 6.

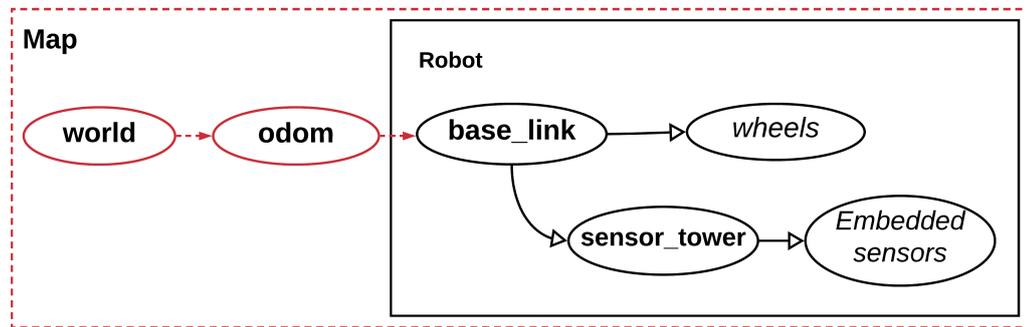


Figura 6 – Árvore simplificada de TFs do EspeleoRobô.

Com o objetivo de referenciar o robô em determinado ambiente, é necessário criar frames que representem as relações do robô com o mapa do local. A localização e orientação do robô no ambiente são representadas pelo frame da odometria. Este por sua vez, está conectado ao frame principal do robô, que possui ligação com sensores e motores.

3.3 Sensores e Atuadores Simulados

O CoppeliaSim já oferece de forma nativa diversos sensores e equipamentos utilizados na robótica. Porém, para uma simulação mais aproximada do robô real, alguns dos sensores foram adaptados a partir destas estruturas já disponibilizadas. A representação do robô virtual com todos os sensores é ilustrada na Figura 7

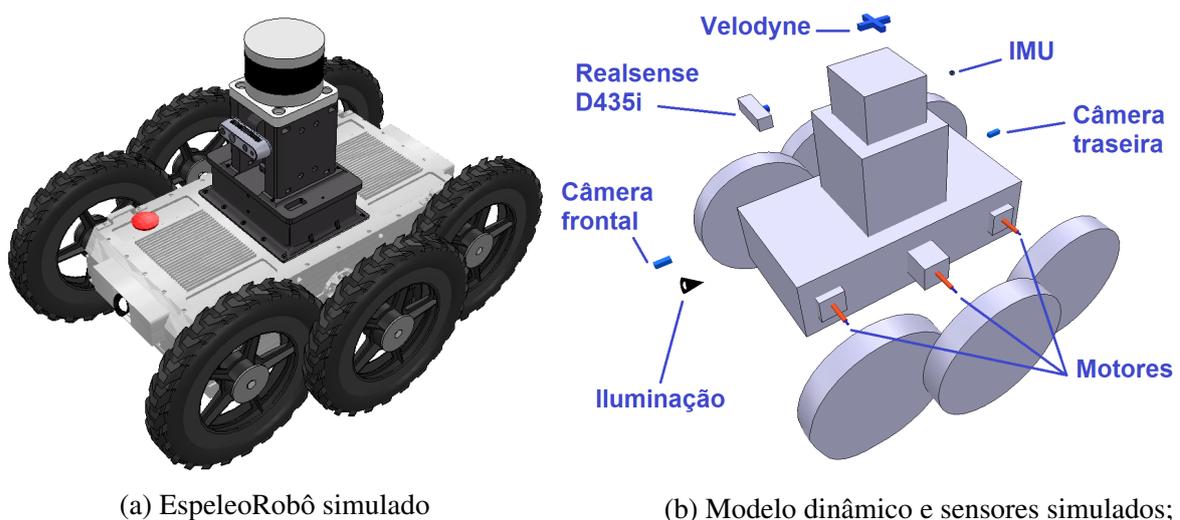


Figura 7 – Visão geral do posicionamento de sensores e componentes do robô simulado.

3.3.1 Câmeras

Para a simulação das câmeras, foi utilizada a estrutura *Vision Sensor* do CoppeliaSim, conforme Figura 8, que fornece uma renderização de imagem por frame, além de transmitir

dados de profundidade. A partir disso, as imagens e dados de cada câmera são publicadas em tópicos ROS. Para visualizar essas imagens na interface gráfica desenvolvida, é necessária a compatibilidade com protocolos de streaming de vídeo como o RTSP (Real Time Streaming Protocol). Desta forma, foi usado o pacote Web-video-server (WILLS, 2019), desenvolvido para streaming em protocolo HTTP e RTSP a partir de tópicos de imagem do ROS.



(a) Simulação do EspeleoRobô em um ambiente subterrâneo.



(b) Câmera frontal.

(c) Câmera traseira.

Figura 8 – Sistema de câmeras do EspeleoRobô.

3.3.2 RealSense D435i

Esse sensor foi desenvolvido a partir de adaptações no modelo do Kinect, câmera RGBD já disponível no simulador, conforme Figura 9. Um tratamento dos dados de profundidade garante a compatibilidade com os padrões de mensagem do ROS.

3.3.3 Iluminação

A partir da estrutura *Lights* do simulador, é possível iluminar uma parte específica do cenário, o cenário inteiro ou uma iluminação com foco direcional. Assim, uma iluminação direcional é atrelada a cada câmera, a fim de simular o posicionamento do sistema de LEDs real do EspeleoRobô. A iluminação artificial permite simular de forma mais realística o comportamento do robô em ambientes com pouca iluminação, como cavernas e minas subterrâneas (Figura 10).

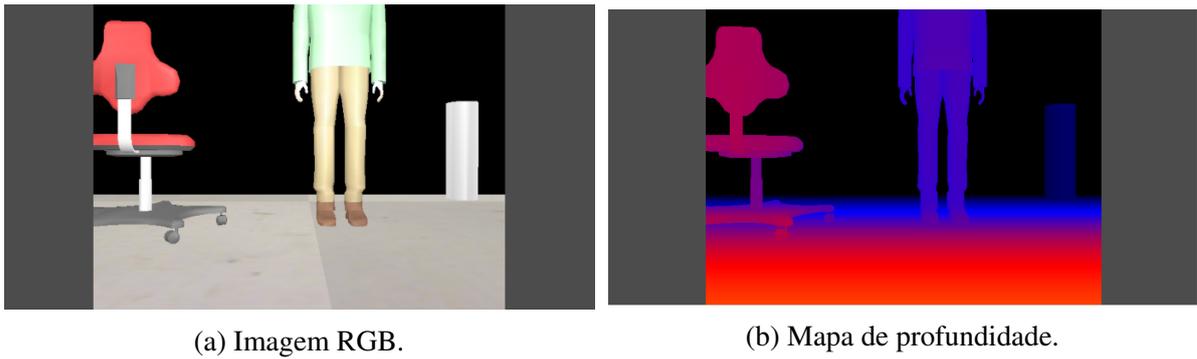


Figura 9 – Imagem RGB e mapa de profundidade da câmera Intel Realsense simulada.

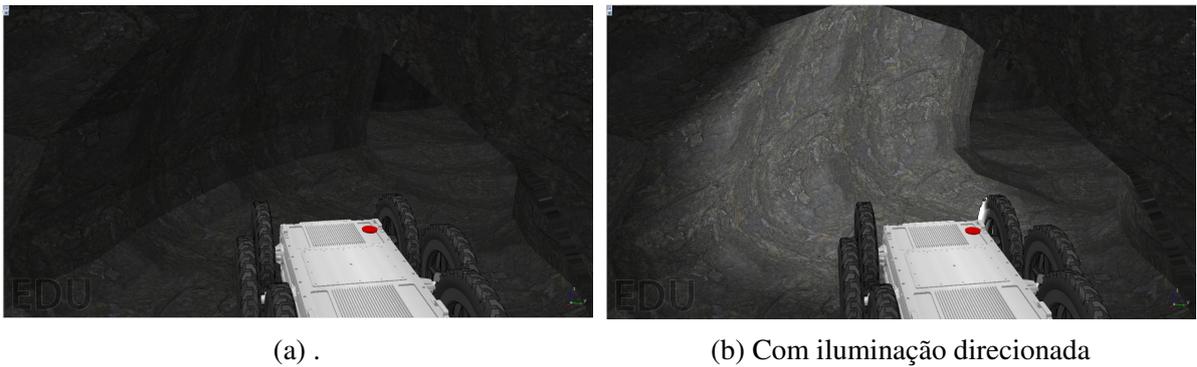


Figura 10 – Sistema de iluminação em ambiente simulado de mina subterrânea: (a) sem iluminação do robô, e (b) sistema de iluminação ativado.

3.3.4 Velodyne VPL16

Este sensor já está implementado no simulador, portanto, foram realizadas apenas as alterações para comunicação com o ROS, conforme Figura 12. O mesmo procedimento pode ser realizado com outros tipos de LiDAR, como o Hokuyo. Outra modificação realizada foi acoplar o sensor 2D a uma junta de rotação, permitindo assim o mapeamento tridimensional do ambiente.

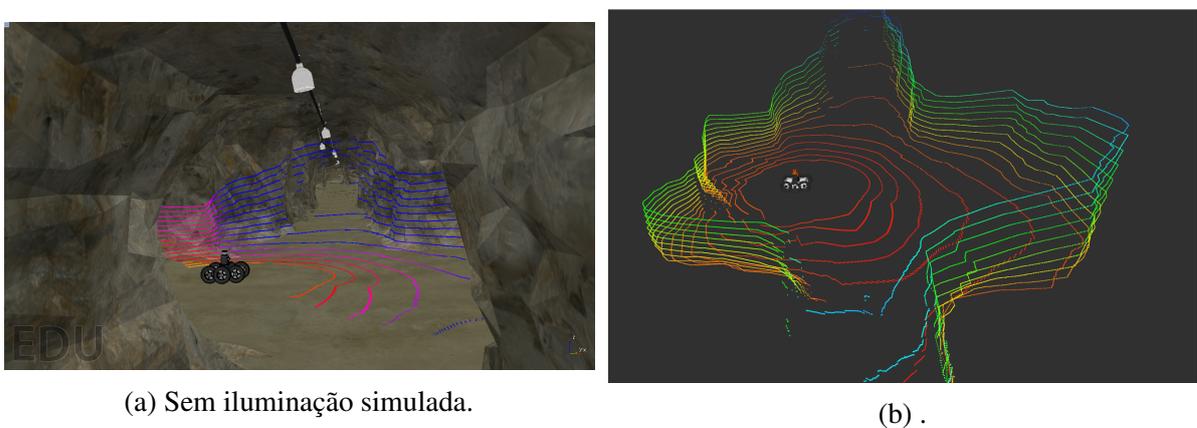


Figura 11 – Simulação do Velodyne no CoppeliaSim (a) e visualização no RVIZ dos pontos lidos pelo Velodyne (b).

3.3.5 IMU

A simulação da IMU (Inertial Measurement Unit) é realizada a partir da fusão de dois sensores presentes no CoppeliaSim: acelerômetro e giroscópio. As informações coletadas pelos sensores são publicadas em dois tópicos ROS.

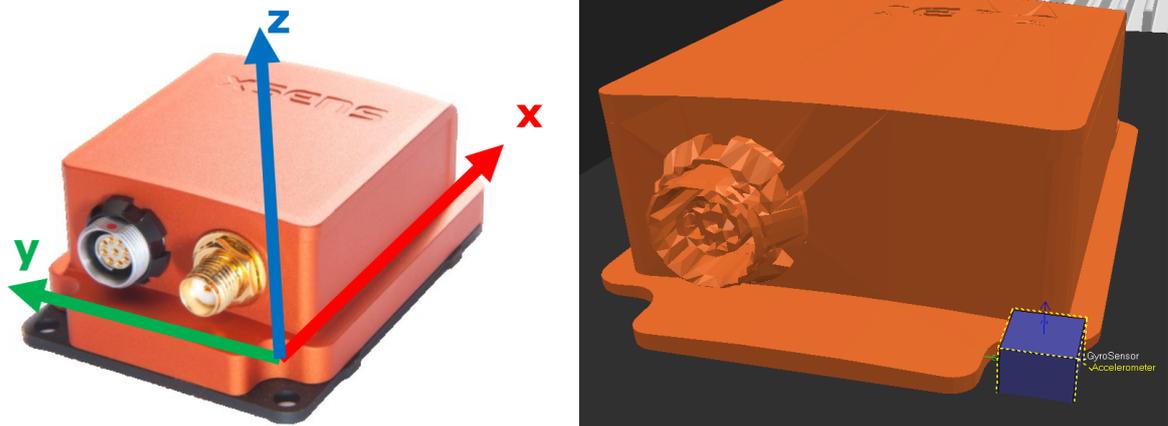


Figura 12 – IMU da Xsens simulada respeitando o sistema de coordenadas da Xsens real (XSENS, 2019). O bloco azul do sensor simulado representa a origem do sistema de coordenadas do sensor real.

3.3.6 Motores

Os motores do robô são representados por juntas rotativas na simulação, conforme Figura 13. Para que o EspeleoRobô simulado se comporte de maneira similar ao real, os valores das reduções de cada motor do robô real são adicionados à simulação. Deste modo, os mesmos códigos e métodos utilizados para a locomoção do robô real podem ser utilizados no simulador, assim como programas desenvolvidos para o ambiente simulado podem ser exportados para a plataforma real. Valores como torque instantâneo e corrente dos motores também foram adicionados à simulação, com base nas características fornecidas pelo manual dos motores MCD EPOS, com o intuito de deixar a teleoperação mais próxima possível do robô real.

Primeiramente, o valor das reduções de cada motor são lidas através do servidor de parâmetro do ROS. Atualmente o EspeleoRobô possui uma redução de valor igual para todos os motores (devido a engrenagem planetária) e, além desta redução, os motores centrais possuem redução de 50/26 e os externos de 44/21, devido às respectivas correias dentadas. Logo, para que os motores se comportem igual aos motores reais, a velocidade lida no tópico do ROS é convertida de rotações por minuto para radianos por segundo. Em seguida, os valores de cada redução são aplicados de acordo com cada motor, conforme equações 3.1 e 3.2, onde $Velocidade_i$ e $Velocidade_e$ são as velocidades, em rotações por minuto, a serem enviadas aos motores internos e externos, respectivamente. Já Red_p , Red_i e Red_e são os valores das reduções

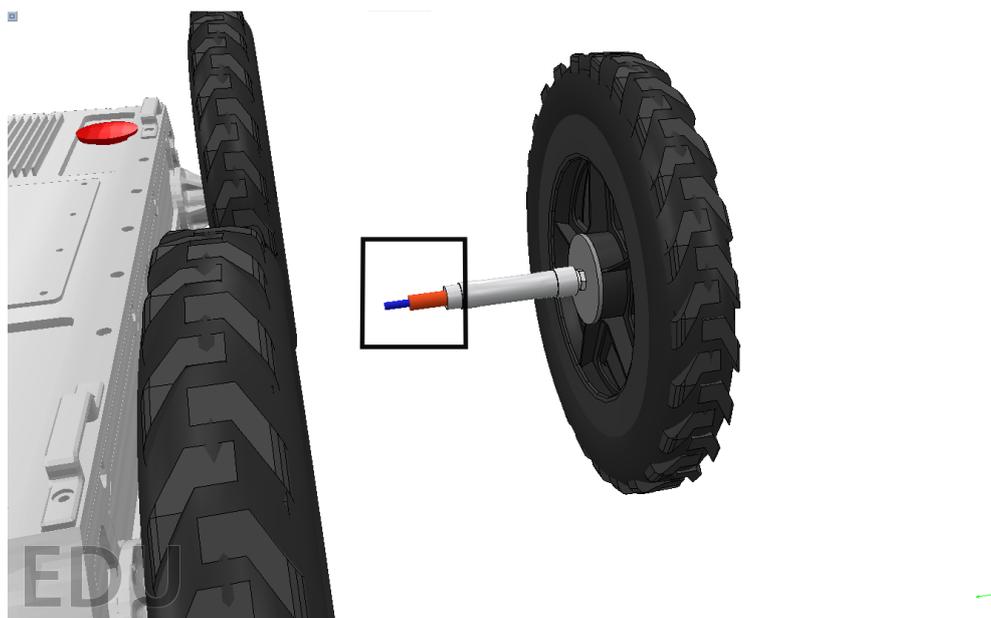


Figura 13 – Junta de rotação para simular motores do EspeleoRobô.

planetária, redução dos motores internos e redução dos motores externos, respectivamente.

$$Velocidade_i = \frac{2\pi}{60} \cdot Red_p \cdot Red_i \quad (3.1)$$

$$Velocidade_e = \frac{2\pi}{60} \cdot Red_p \cdot Red_i \quad (3.2)$$

3.4 Ambientes Modelados

Cenas virtuais modeladas manualmente via CAD oferecem ambientes representativos para testes, mas, para aumentar a fidelidade das simulações, podem ser usados também reconstruções 3D. Ambientes virtuais reconstruídos são ideais para simulação pois apresentam características típicas, estruturas geométricas e texturas de uma possível aplicação real. Técnicas de reconstrução vêm sendo usadas em várias áreas como em preservação histórica (EL-HAKIM et al., 2004), jogos (SZWOCH; KACZMAREK; BARTOSZEWSKI, 2017) e medicina (CIOBANU; ROTARIU, 2014). Para a elaboração de mapas virtuais realísticos existem técnicas como fotogrametria e mapeamento por LiDAR. Como o Espeleorobô tem como finalidade ser utilizado em diferentes ambientes, como usinas, cavidades naturais e tubulações, diversos ambientes foram projetados para a simulação.

A fotogrametria consiste em usar imagens 2D, adquiridas por câmeras digitais, e algoritmos de *structure from motion* para estimar as estruturas tridimensionais. Este método produz cenas realísticas em termos de forma e aparência. Já a reconstrução utilizando laser rotativos é feita utilizando o concatenamento da nuvens de pontos obtidas pelos sensores. O cenários reconstruídos por LiDAR tem alta fidelidade em forma porém não apresentam textura enquanto

a fotogrametria requer somente imagens, adquiridas por câmeras digitais comuns, e a cena reconstruída apresenta texturas reais.

3.5 Avaliação do ambiente simulado

Além de uma representação fiel do robô real, a simulação deve ser otimizada para garantir a viabilidade de uso quando o tempo de resposta é crítico. O atraso de controle de uma simulação lenta afeta a experiência de teleoperação e prejudica o treinamento e a operação adequada da plataforma. Dois fatores afetam muito o tempo de resposta do simulador: O número de interações dinâmicas de objetos e o tamanho dos dados dos sensores e cálculos do atuador. Portanto, ao reduzir a complexidade da cena e melhorar a eficiência do sensor, a velocidade da simulação aumentará.

O modelo dinâmico do robô pode ser representado por: (i) Um modelo detalhado, que corresponde à representação mais próxima do robô real; (ii) Um modelo simplificado, onde o número de faces na malha 3D é reduzido ao mínimo possível para que o modelo dinâmico não perca as características de sua forma; e (iii) um modelo utilizando apenas primitivas geométricas visando menor custo computacional para a simulação.

O fator de simulação em tempo real pode ser utilizado para verificar o desempenho de cada modelo, estimando a média e o desvio padrão para 30 execuções dos modelos do robô seguindo um mesmo caminho. Este fator corresponde à razão entre o tempo total simulado ($time_{sim}$) e o tempo real decorrido ($time_{real}$), onde valores próximos a um representam uma simulação mais próxima do tempo real:

$$RT_{factor} = \frac{time_{sim}}{time_{real}}. \quad (3.3)$$

Conforme mostrado na Figura 15, foram comparados diferentes modelos dinâmicos para o robô de inspeção em três cenários representativos: (i) um mapa com poucas interações dinâmicas com o robô (mapa plano), (ii) um mapa maior, com um número considerável de interações (Mapa DARPA), e (iii) finalmente um mapa com malhas complexas e não otimizadas, com muitas interações (mapa do moinho de bolas). Os experimentos foram realizados em um computador com processador Intel I7-7700HQ de 2,8 GHz, 8 GB de RAM e placa de vídeo GTX 1050TI.

É possível observar na Figura 15 que em mapas com poucas interações dinâmicas, todos os modelos têm bom desempenho (fator de simulação próximo a 1,0). No entanto, a medida que essas interações aumentam, o modelo que usa primitivas geométricas se destaca, conforme mostrado pelo terceiro mapa, onde apenas o modelo dinâmico geométrico foi próximo a 1,0.

Há uma compensação entre a eficiência e a precisão dos modelos de simulação dinâmica,



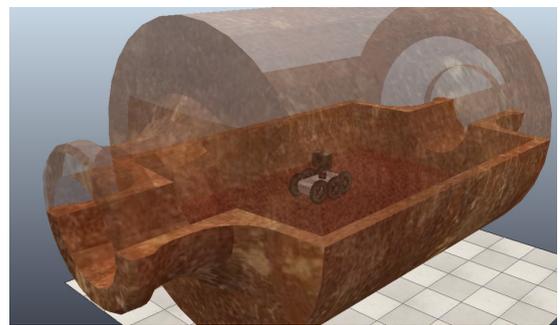
(a)



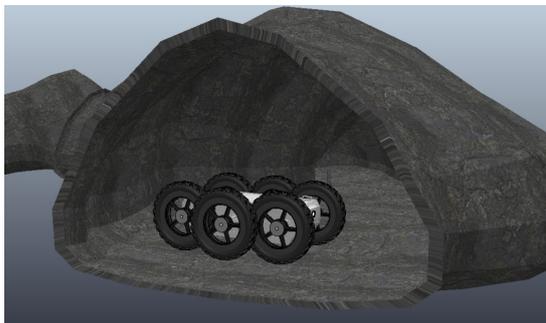
(b)



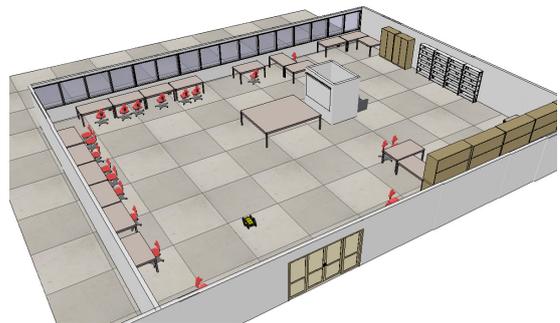
(c)



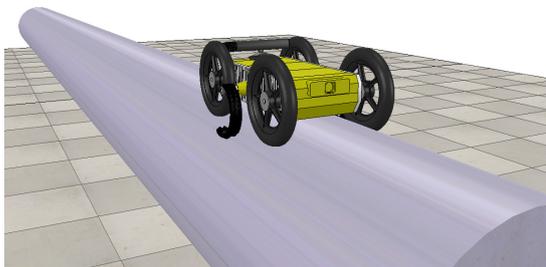
(d)



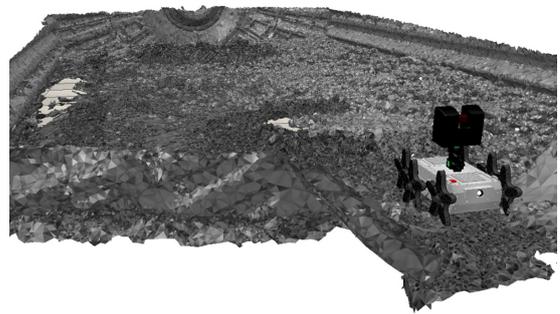
(e)



(f)



(g)



(h)

Figura 14 – Ambientes projetados para a simulação: (a) pista de motocross, (b) mina de Ouro, (c) cavidade natural, (d) moinho de bolas, (e) sistema de tuncis da DARPA, (f) laboratório ITVRoC, (g) tubulação e (h) reconstrução interna de moinho com LiDAR.

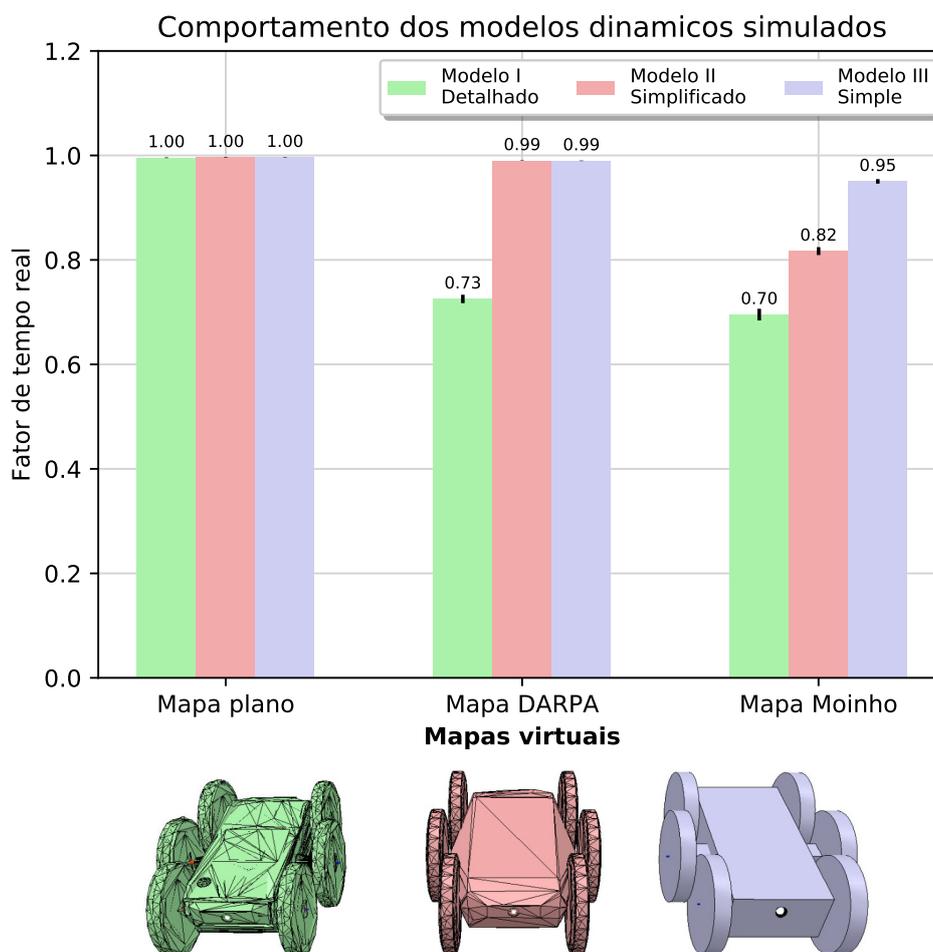


Figura 15 – Análise de desempenho do modelo dinâmico do robô no CoppeliaSim em diferentes cenários. Linhas pretas em cada barra representam o desvio σ .

porém, por meio de vários testes, nenhuma diferença de precisão foi observada no comportamento dinâmico usando o modelo mais simples. No caso de teleoperação ou treinamento, modelos simplificados são necessários para garantir a agilidade da simulação, no entanto, para casos de uso específicos, como testes de mobilidade, os modelos mais complexos e fiéis podem ser utilizados.

Também foi observado que o uso combinado de vários sensores reduz significativamente o desempenho da simulação, mostrando que o sensor Velodyne padrão foi o mais custoso de todos (Figura 16). Assim, uma alternativa é usar apenas o conjunto de sensores necessários para aplicações específicas. Outra forma de otimizar os sensores é alterar sua frequência de processamento durante a simulação. Com isso, é possível priorizar sensores específicos, de acordo com a aplicação. Por exemplo, câmeras RGB podem ter sua taxa de aquisição reduzida para 1HZ nos momentos em que a teleoperação não está sendo usada, resultando em uma melhoria de até 20% no desempenho. Outra opção é usar plugins para que a comunicação do sensor com o ROS seja feita externamente ao simulador (Barra Velodyne adaptada na Figura 16).

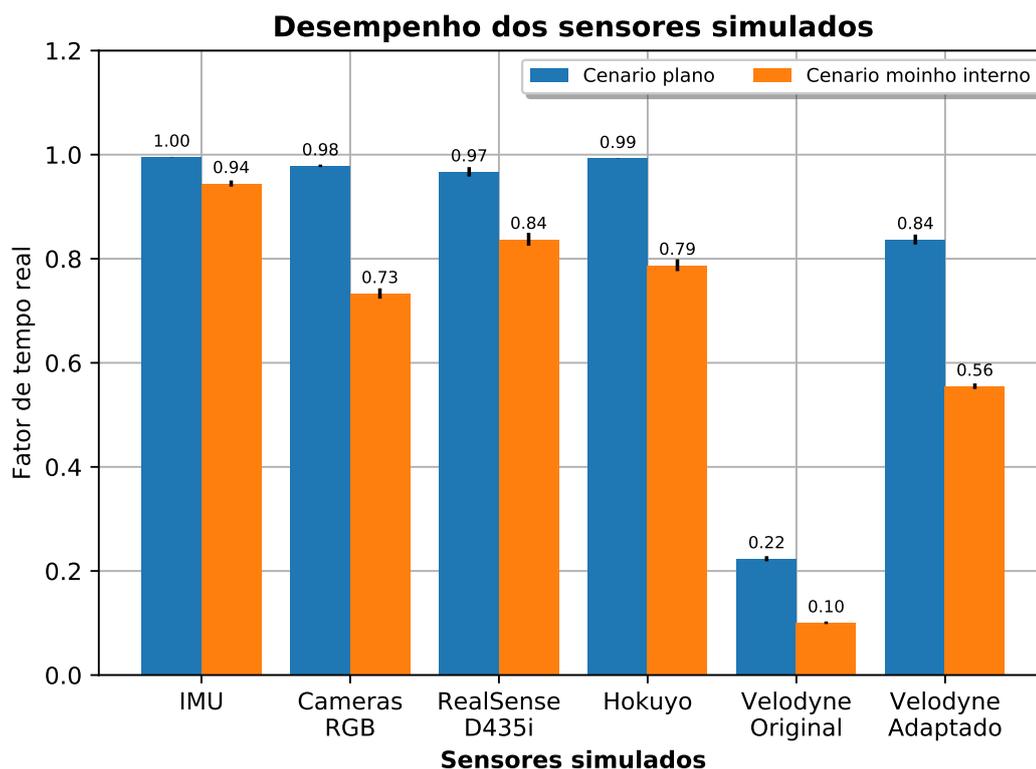


Figura 16 – Análise de desempenho dos sensores embarcados no EspeleoRobô utilizados no CoppeliaSim. Linhas pretas em cada barra representam o desvio σ .

3.6 Teleoperação e Treinamento

Com o intuito de aprimorar a utilização da GUI (Graphical User Interface), a simulação do EspeleoRobô foi adaptada para fornecer as informações internas do robô de modo similar ao real (Figura 17). Desta forma, a interface gráfica pode ser utilizada na simulação para treinamento de pilotos em missões ou para execução e testes de algoritmos sem precisar da plataforma real.

O sistema de teleoperação é o responsável por receber a entrada de dados do usuário e convertê-la para comandos de velocidade, controle de iluminação e outros acionamentos. De forma a garantir a teleoperação do robô, a entrada de dados acontece de duas formas: via *joystick* ou teclado. Ambos possuem funcionamento análogo nas versões virtual e real do robô (Figura 18).

3.7 Análise de Mobilidade

Usando os dados provenientes da IMU simulada, e conhecendo as dimensões físicas do dispositivo, é possível analisar a mobilidade do robô em diferentes terrenos considerando o polígono de sustentação e os ângulos de tombamento (PAPADOPOULOS, 1996). A Figura 19 ilustra o comportamento do EspeleoRobô no mapa do moinho de bolas, assim como seu polígono de sustentação, indicando se o robô se encontra próximo do capotamento. Desta forma é possível

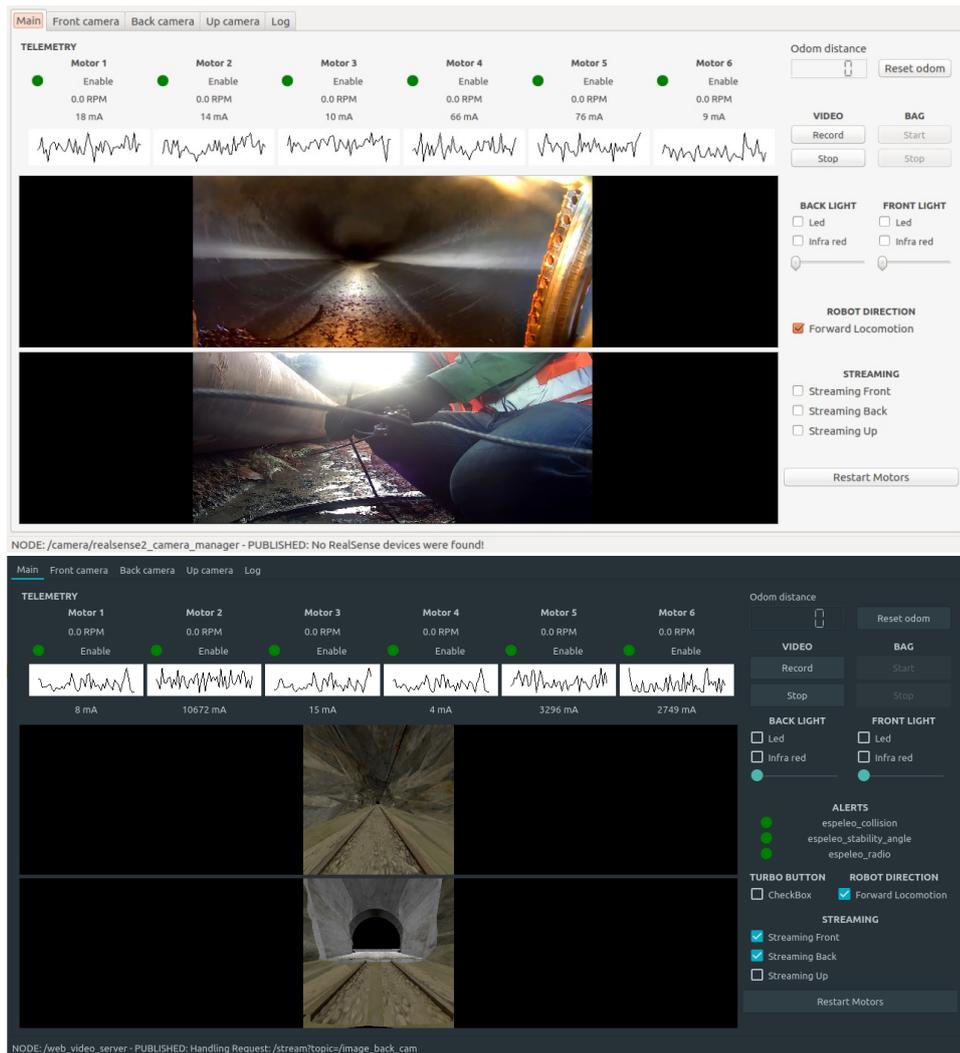


Figura 17 – Interface gráfica do EspeleoRobô. A figura superior mostra a GUI numa inspeção de dutos com o robô real. Embaixo é mostrado a mesma GUI num cenário de mina simulado disponibilizado pela DARPA.

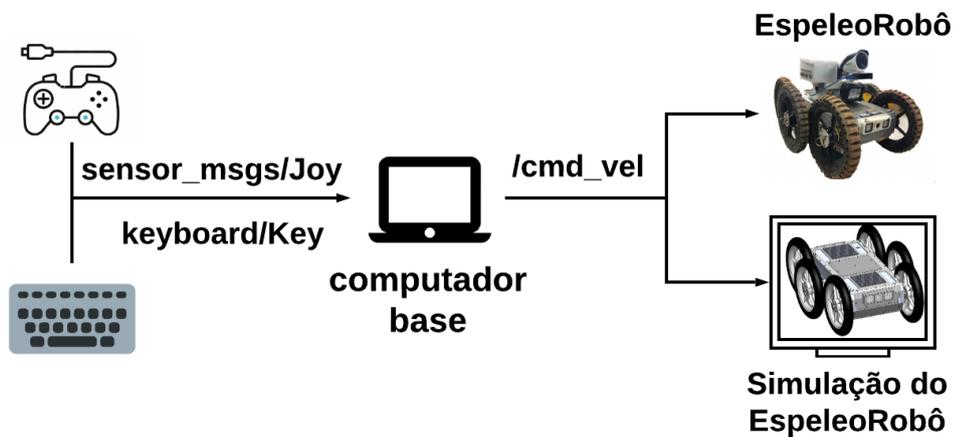


Figura 18 – Diagrama de teleoperação do EspeleoRobô através do ROS usando a plataforma real ou simulada.

prever o tombamento do robô num determinado local antes de submetê-lo a um possível risco operacional (AMARAL et al., 2020). Outras análises também podem ser realizadas levando em conta a eficiência energética ou tamanho máximo de superação de obstáculos (ROCHA et al., 2017).

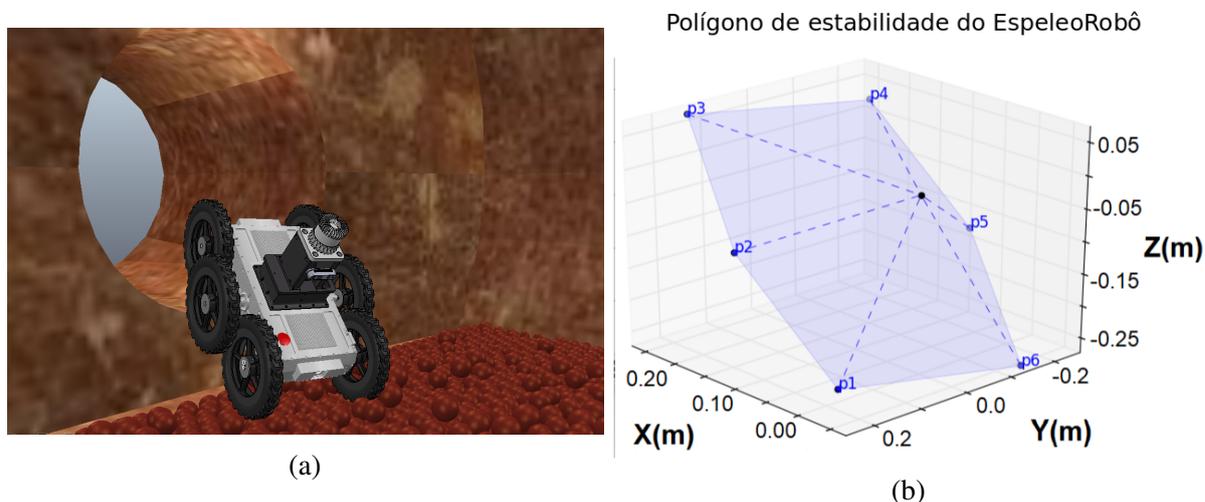


Figura 19 – Plataforma robótica (a) com avaliação de estabilidade calculada com o polígono de sustentação (b).

3.7.1 Planejamento de Caminhos, Mapeamento e Controle

Uma nuvem de pontos 3D do ambiente pode ser gerada a partir do LiDAR ou dos dados de profundidade. Ao mesclar essas informações com outros sensores é possível realizar a localização e mapeamento simultâneos. Foi utilizado o algoritmo OctoMap como estrutura base para a reconstrução do mapa (HORNUNG et al., 2013b).

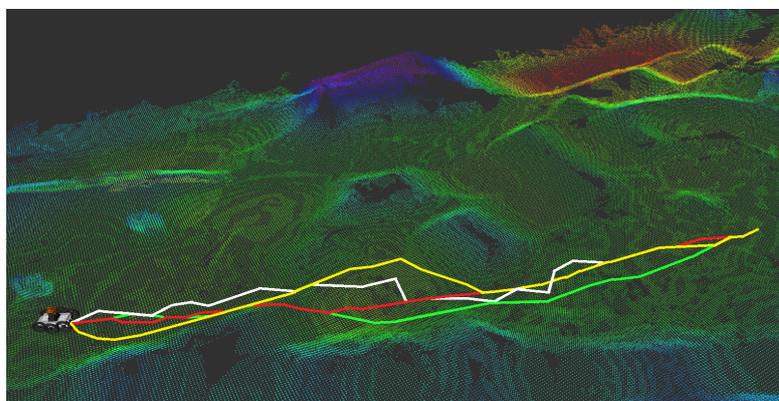
A estratégia de planejamento de caminhos implementada é baseada em uma abordagem onde uma malha 3D gerada a partir da nuvem de pontos do ambiente é modelada como um grafo ponderado (SANTOS, 2019; AZPÚRUA et al., 2021). Os pesos das arestas são definidos por meio de diferentes métricas: distância percorrida, energia consumida, transversalidade do terreno e sua combinação por meio de valores normalizados. Assim, o algoritmo de Dijkstra é executado no gráfico para encontrar a rota ótima garantida de acordo com a métrica adotada.

O caminho ideal selecionado é percorrido usando um algoritmo de controle de campos vetoriais. Esse controle é baseado na definição de uma velocidade de referência composta por dois componentes: (i) convergente; e (ii) tangente. O componente de convergência é responsável por controlar o robô na direção normal para o caminho ideal. A tangente é responsável por guiar o robô ao longo desse caminho. O controle de navegação usa a estrutura de campo vetorial definida em (GONÇALVES et al., 2010).

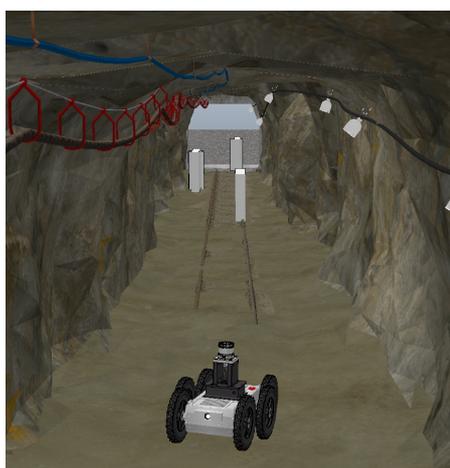
O resultado do mapeamento completo, planejamento de caminho e processo de controle de navegação pode ser visto na Figura 20



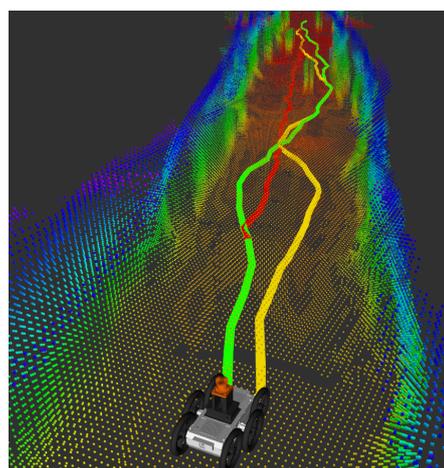
(a) Terreno acidentado (pista de motocross).



(b) Mapa 3D e caminhos gerados para o terreno acidentado.



(c) Sistema de túneis DARPA.



(d) Mapa e caminhos 3D do túnel.

Figura 20 – Pipeline de navegação em dois mapas representativos: pista de motocross (a-b) e sistema de túneis DARPA (c-d). As cores dos caminhos definem a métrica usada: Menor distância (vermelho), transversalidade do terreno (branco), menor consumo de energia (amarelo) e métricas combinadas (verde).

4 MAPA DE INTENSIDADE DO SINAL DE RÁDIO

Neste capítulo serão apresentados os procedimentos relacionados a geração de um mapa da intensidade do sinal de rádio, utilizando a versão simulada do robô, com o intuito de facilitar operações de planejamento de caminhos com base na qualidade de sinal e a geração de alertas dada a potência que este sinal chega ao robô.

4.1 Sistema de navegação

Antes de realizar o mapeamento do sinal de rádio é preciso que o robô consiga gerar um mapa de ocupação do ambiente, assim como se localizar nesse mapa. Logo, a utilização de um método de SLAM que permita isto é essencial para a tarefa.

4.1.1 Odometria

Por possuir encoder nas rodas, permitindo realizar a leitura de velocidade instantânea em cada motor, uma alternativa para a localização do robô é a utilização das fórmulas 2.3 e 2.4 para obter sua posição e orientação. Para isto, foi desenvolvido um algoritmo, conforme Figura 21.

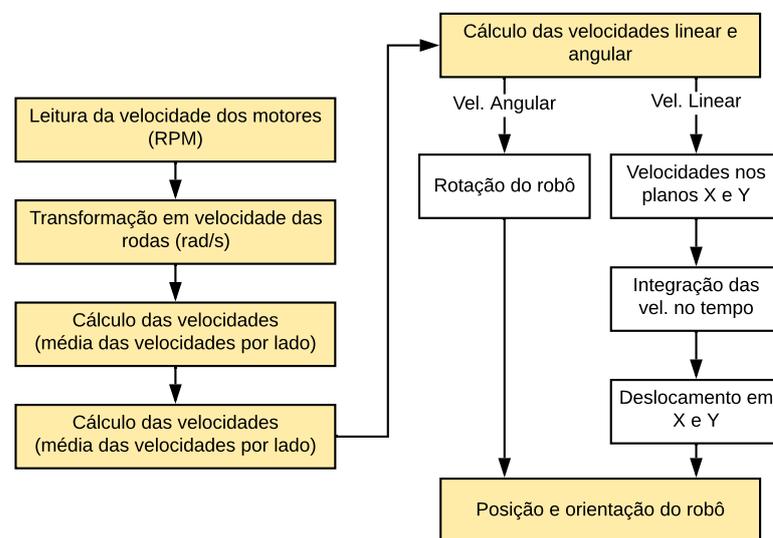


Figura 21 – Fluxograma para utilização das velocidades dos motores para localização do robô.

Para estudo de viabilidade da odometria de rodas, foi realizado um teste de navegação no simulador, obtendo a posição do robô no plano X,Y percorrendo um pequeno trajeto, como visto na Figura 22.

Conforme Cota (2019) e Anousaki e Kyriakopoulos (2004), a utilização do modelo diferencial para cálculo de odometria de um robô Skid-steer gera erros na localização, principalmente

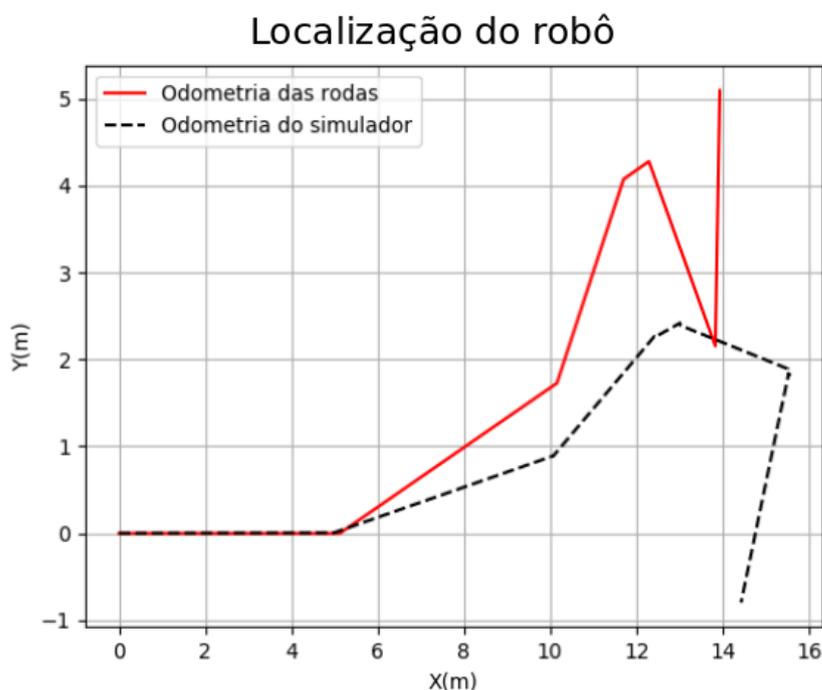


Figura 22 – Teste de odometria de rodas em comparação com localização verdadeira do robô (fornecida pelo simulador).

em movimentos de rotação. Devido a isso, os testes realizados nas próximas seções utilizaram apenas da localização e orientação absoluta fornecida pelo simulador.

4.1.2 Mapeamento e localização simultâneos

Dado os algoritmos de mapeamento citados na secção 2.4.1, optou-se pela utilização do Gmapping para a realização dos testes no simulador. O Google cartographer apresentou melhores resultados (YAGFAROV; IVANOU; AFANASYEV, 2018) por utilizar da fusão de dados de IMU, LiDAR e odometria para gerar uma localização precisa, porém, por utilizar um simulador que possui a localização exata do robô, o Gmapping atende à proposta inicial (ZHANG et al., 2020).

Os testes de mapeamento foram realizados em um cenário que houvesse obstáculos a serem desviados assim como na Figura 23. Logo, é necessário utilizar do espaço de configuração do robô. A primeira etapa consiste em gerar uma representação 2D do robô para ser utilizada como base nos desvios de obstáculos. São atribuídas duas áreas para essa representação para garantir que não haja colisão com os objetos: Uma área interna e outra externa, como visto na Figura 24.

- Caso algum obstáculo se encontre na área interna, é garantido a colisão.
- Caso o obstáculo esteja entre a área interna e externa, há uma possível colisão, a depender da orientação do robô.



Figura 23 – Laboratório de controle e robótica do Instituto Tecnológico Vale, utilizado para teste de mapeamento.

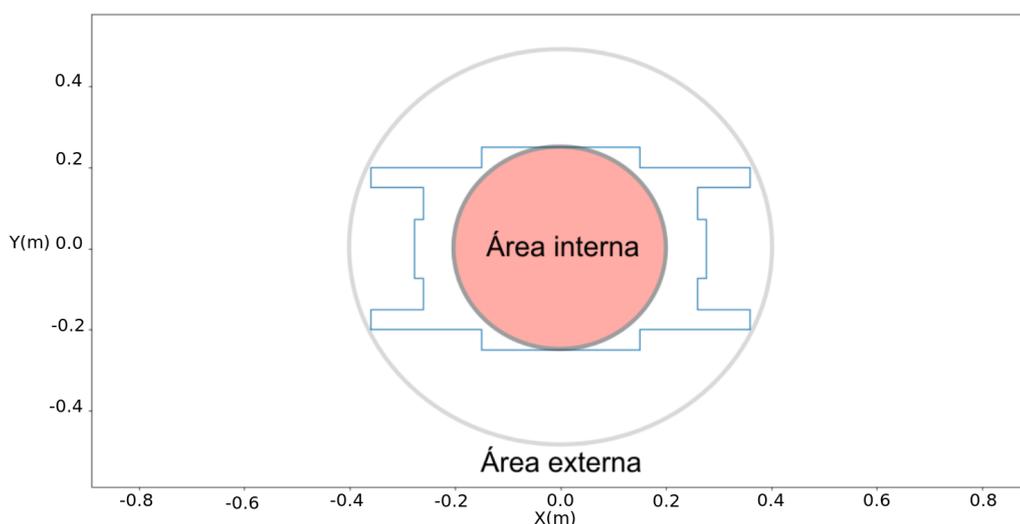


Figura 24 – Representação 2D do EspeleoRobô com áreas de colisão.

- Caso o obstáculo esteja externo a área externa, não há risco de colisão.

Além da representação 2D do EspeleoRobô, todos os obstáculos detectados pelo LiDAR sofrem um processo de inflação, conforme as dimensões do robô, para garantir que não haja colisão. Para isso, o centro da representação do robô não pode ultrapassar a área inflada do obstáculo. O processo de mapeamento pode ser visto na Figura 25, onde o mapa parcial pode ser observado na Figura 25a e o mapa final gerado na Figura 25b.

4.1.3 Localização

Caso o processo de aquisição dos dados do sinal de rádio não aconteça simultâneo ao mapeamento, será preciso de algum método que permita a localização do robô no mapa conhecido.

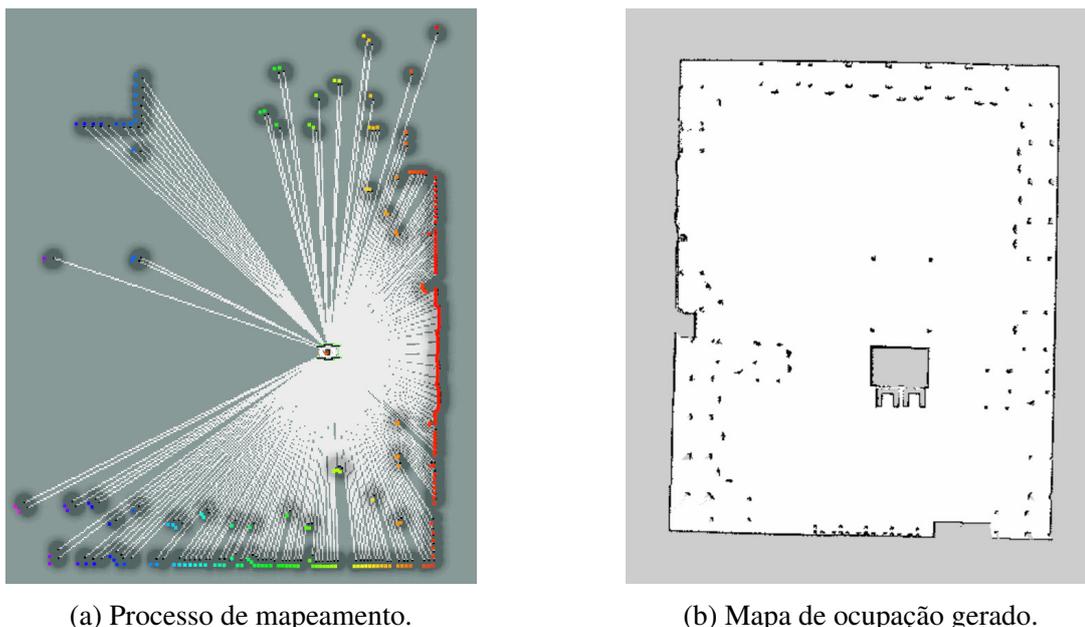


Figura 25 – EspeleoRobô mapeando o laboratório simulado: (a) Processo de mapeamento, onde os pontos coloridos representam as leituras do LiDAR e as áreas escuras ao redor destes pontos são os obstáculos inflados. (b) Mapa de ocupação final do laboratório.

Para isto, é utilizado um algoritmo probabilístico de localização: O AMCL (DELLAERT et al., 1999). Este algoritmo utiliza de um filtro de partículas para relacionar os dados do LiDAR, as transformadas do robô e a posição inicial para estimar a posição e orientação do robô, conforme Figura 26.

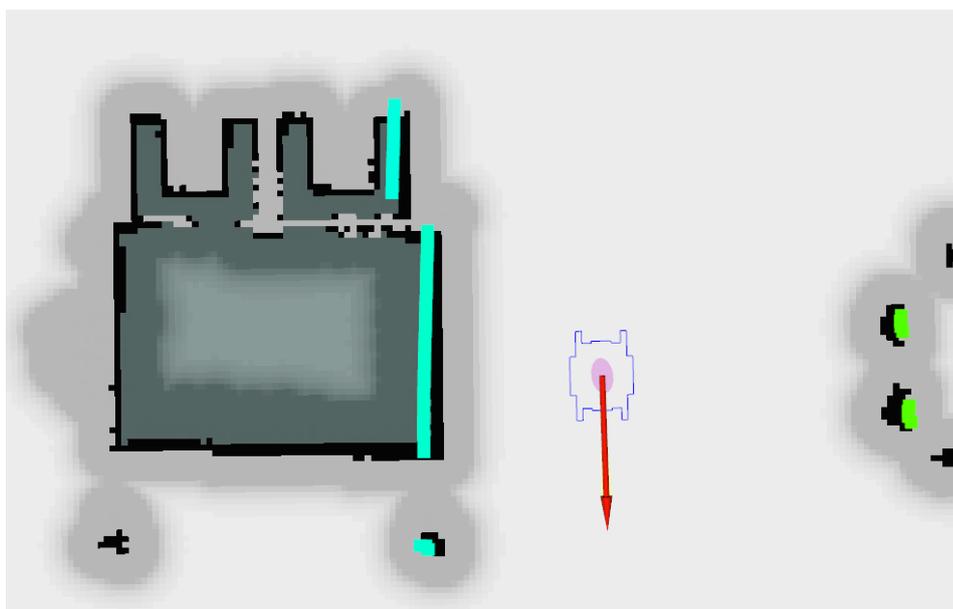


Figura 26 – Posição e orientação do robô dados pelo filtro de partículas. A elipse rosa representa a covariância da posição e o triângulo amarelo (abaixo da seta vermelha) a covariância da orientação

4.2 Simulação do sinal de rádio

Para a realização dos testes de mapeamento do sinal de rádio, é necessário primeiro simular o seu sinal. Para tal, foi adotado o modelo do logaritmo da distância (NAJNUDEL, 2004), que serve como uma aproximação da propagação do sinal de rádio em ambientes fechados. A fórmula utilizada pode ser visualizada em 4.1, e relaciona a perda do sinal de rádio em decibéis (P_{dB}), a uma distância arbitrária (d), com a perda em um local conhecido (P_0), a distância deste local (d_0), o expoente de perda do ambiente (n) e uma variável caso haja sombreamento do sinal (x).

$$P_{dB} = P_0 + 10 \cdot n \cdot \log_{10}\left(\frac{d}{d_0}\right) + X \quad (4.1)$$

Para fins de simplificação na simulação, foi considerado que à distâncias menores que um metro da fonte de sinal de rádio, não há perda na potência do sinal e que as antenas são omnidirecionais. Também foi considerando que o expoente de perda de sinal do ambiente a ser testado é o mesmo que o de uma sala fechada, com os receptores transmitindo a uma frequência de 900MHz, assim como os receptores de rádio presentes no EspeleoRobô. O sombreamento do sinal também foi desconsiderado.

Assim, a fórmula final utilizada para simular o sinal de rádio pode ser definida como a diferença entre o sinal transmitido e o recebido, conforme equação 4.2.

$$P_{total} = P_{transmissor} - 10 \cdot n \cdot \log_{10}(d) \quad (4.2)$$

Outro fator importante na simulação do sinal de rádio é se o robô possui ou não visada direta para a antena, pois este afeta diretamente na intensidade do sinal que chega ao receptor. Para isto, foi adaptado um sensor de obstáculos, posicionado acima do transmissor de rádio, para que este siga o EspeleoRobô no mapa. Acoplado na base desse sensor, há uma junta de revolução que recebe o ângulo de rotação necessário para que o sensor foque no robô, conforme Figura 27 e equação 4.3, adaptada para os diferentes quadrantes no plano (X, Y). Se um objeto que não seja o EspeleoRobô for detectado, então uma variável booleana é registrada no servidor de parâmetros do ROS, informando a presença de um obstáculo.

$$\alpha = \arctg\left(\frac{|X_0 - X_E|}{|Y_0 - Y_E|}\right) \quad (4.3)$$

Caso o robô possua visada direta com a antena, a distância entre os dois é calculada utilizando a fórmula da distância entre dois pontos, conforme equação 4.4. Caso tenha um obstáculo entre o robô e a antena, a fórmula 4.5 é utilizada, conforme Figura 28.

$$d = \sqrt{(X_0 - X_E)^2 + (Y_0 - Y_E)^2} \quad (4.4)$$

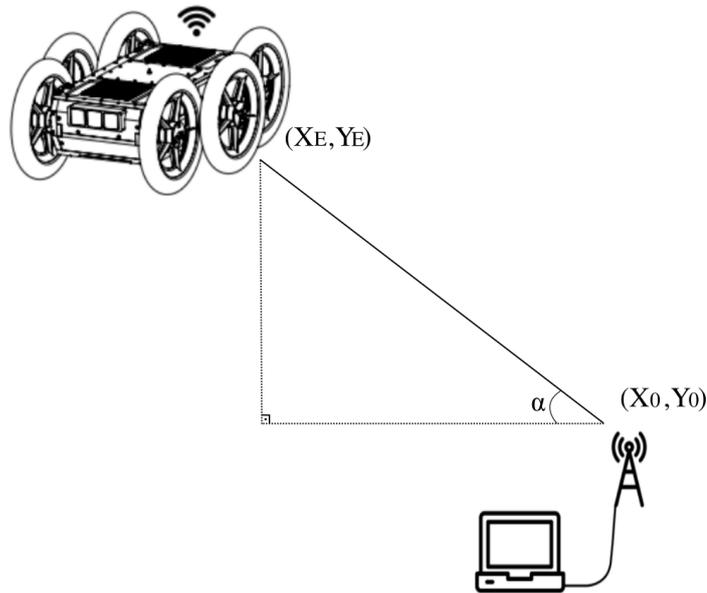
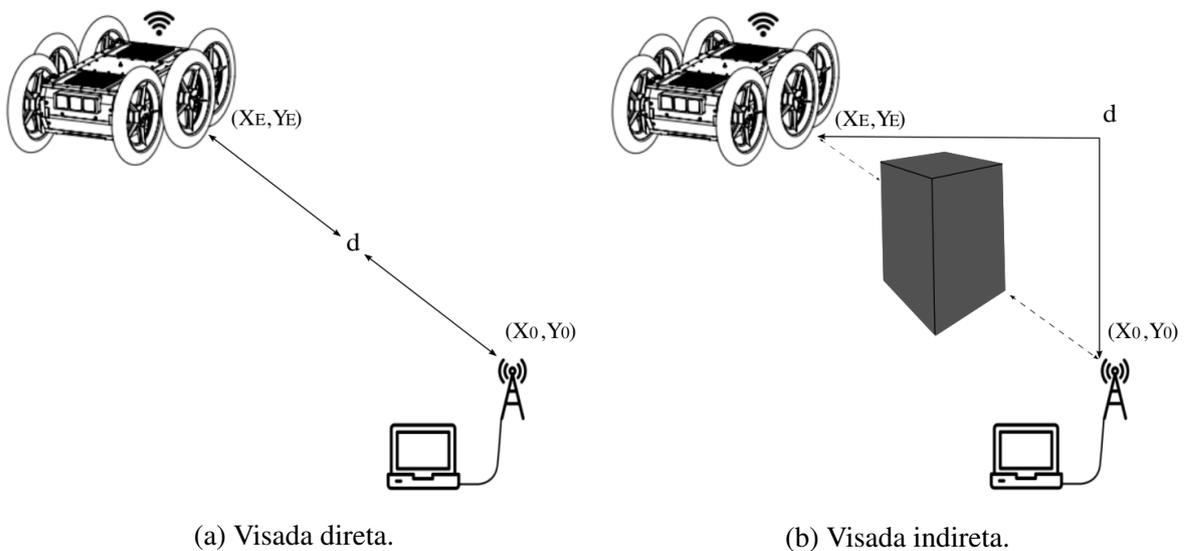


Figura 27 – Ângulo para manter o sensor de obstáculos apontado para o EspeleoRobô

$$d = |(X_0 - X_E)| + |(Y_0 - Y_E)| \quad (4.5)$$



(a) Visada direta.

(b) Visada indireta.

Figura 28 – Distância entre o robô e a antena nos casos de visada direta (a) e a presença de um obstáculo (b).

4.3 Mapeamento do sinal de rádio

Após escolher um local para a base do sinal de rádio, conforme Figura 29, começou o processo de mapeamento. Este consiste no robô percorrer por toda a cena, colhendo os dados da intensidade do sinal de rádio, assim como a sua posição, a cada intervalo de tempo pré definido, ou quando o operador achar necessário. Dados os métodos citados na seção 2.5.1, os métodos lineares apresentam erro consideravelmente baixo.

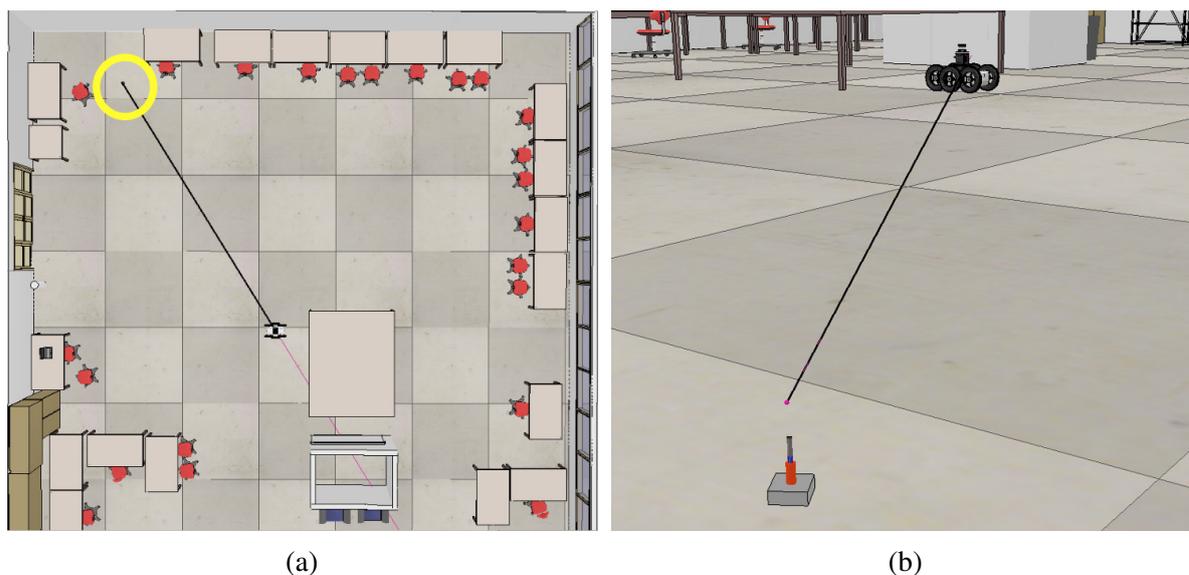


Figura 29 – Ambiente simulado do laboratório: (a) Posição da antena de rádio no mapa (circulado em amarelo), e (b) sensor de obstáculos apontado para o EspeleoRobô.

Racko, Machaj e Brida (2017) compararam dois métodos diferentes de interpolação: linear e Dalaunay. Ambos apresentaram erro menor que 5dBm em 70% dos pontos interpolados e apresentaram valor médio de erro próximos, mostrando-se assim como métodos viáveis de interpolação de sinal de rádio.

Para a criação do mapa principal de rádio, utilizado para futura comparação entre métodos, foram colhidos 460 pontos na simulação do laboratório do Instituto Tecnológico Vale (ITV). Também foi utilizado o mapa de ocupação fornecido pelo mapeamento 2D da região, criado com o algoritmo de SLAM Gmapping, para que os dados interpolados sejam aplicados apenas em locais que o robô possa transitar. Para isto, a imagem, no formato *.PGM*, e suas informações como origem e resolução (ROS, 2018a), são utilizadas para fazer a relação entre os pixels da imagem e a localização do robô no plano (X, Y). O resultado do mapeamento pode ser visto na Figura 30, onde nota-se o decaimento do sinal de rádio conforme a distância entre sua origem (ponto superior esquerdo da imagem) vai aumentando, assim como quando não há visada direta entre a antena e o robô (atrás dos objetos como pilastras, mesas e cadeiras).

4.3.1 Comparação entre métodos de interpolação

Para a comparação entre os métodos de interpolação, foi criado um caminho igual a ser percorrido pelo robô onde ele irá colher 30 vezes o sinal de rádio. Para cada modelo, dois mapas são gerados: Um contendo os valores interpolados dentro do polígono formado pelos pontos lidos; Outro realizando a interpolação pelo mapa todo, na tentativa de prever o sinal de rádio conforme os pontos lidos. Os métodos utilizados foram a interpolação bilinear e cúbica. Os resultados da interpolação de 30 pontos são mostrados na Figura 31, e os mapas resultantes da predição do sinal podem ser vistos na Figura 32.

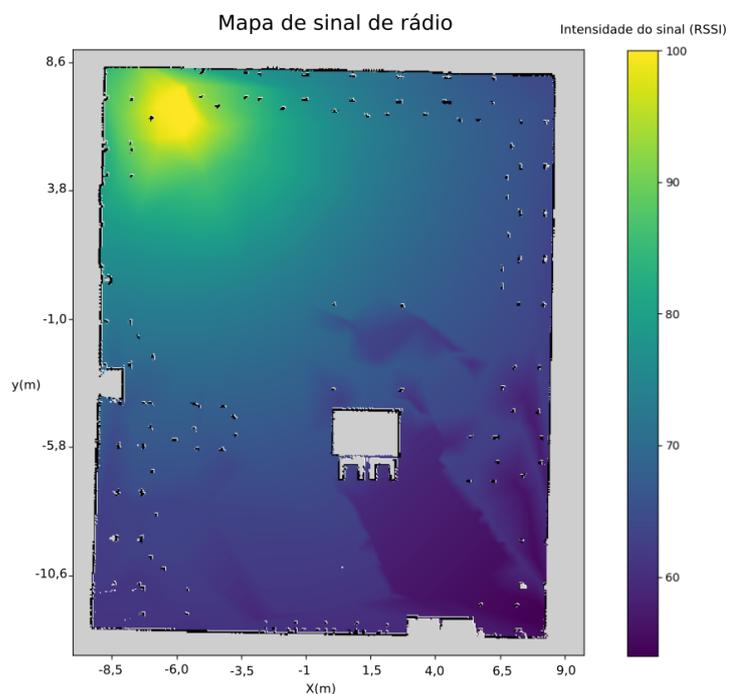


Figura 30 – Mapa de rádio do laboratório simulado.

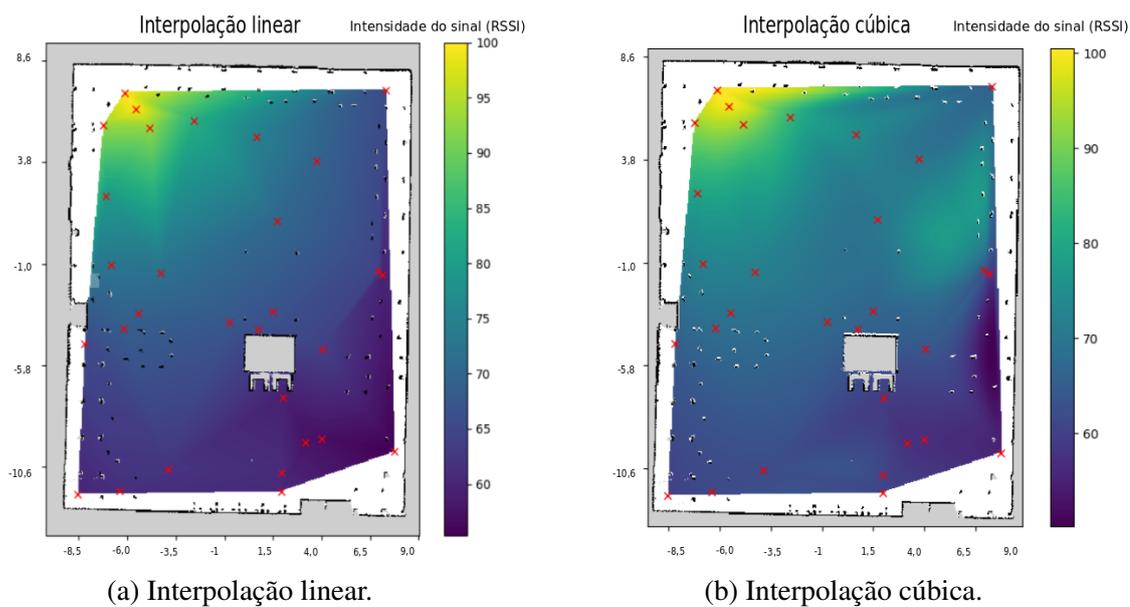
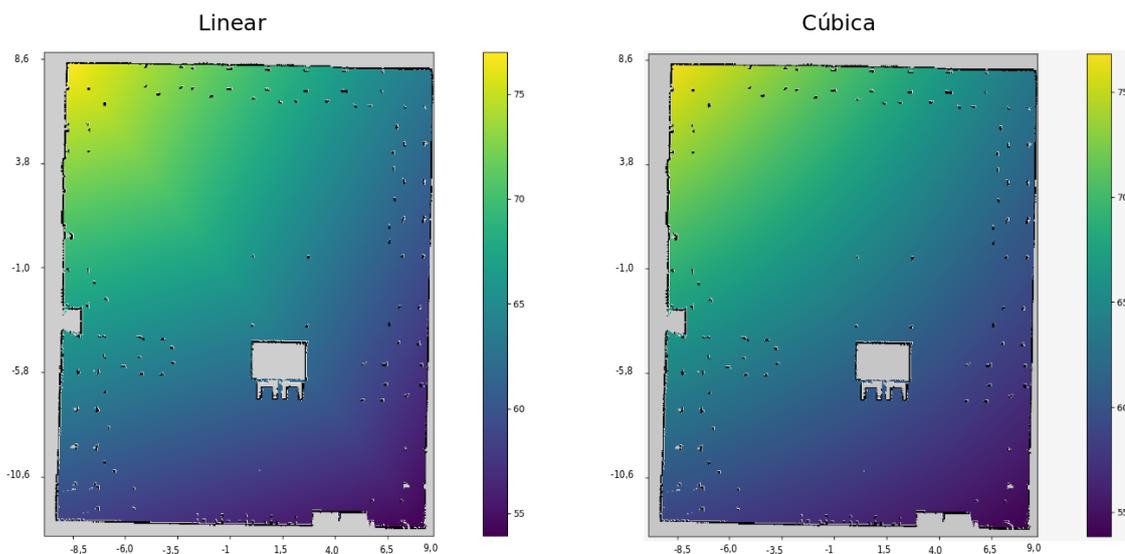


Figura 31 – Mapa do sinal de rádio pelo método da interpolação linear (a) e cúbica (b).



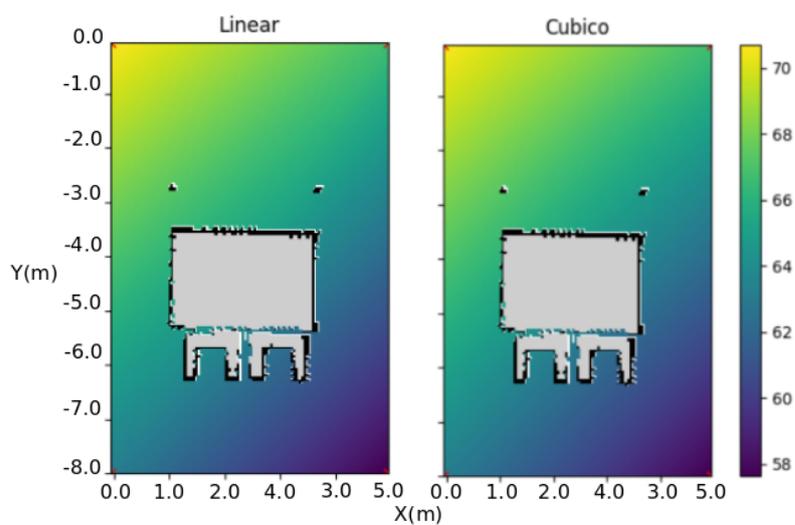
(a) Predição com interpolação linear.

(b) Predição com interpolação cúbica.

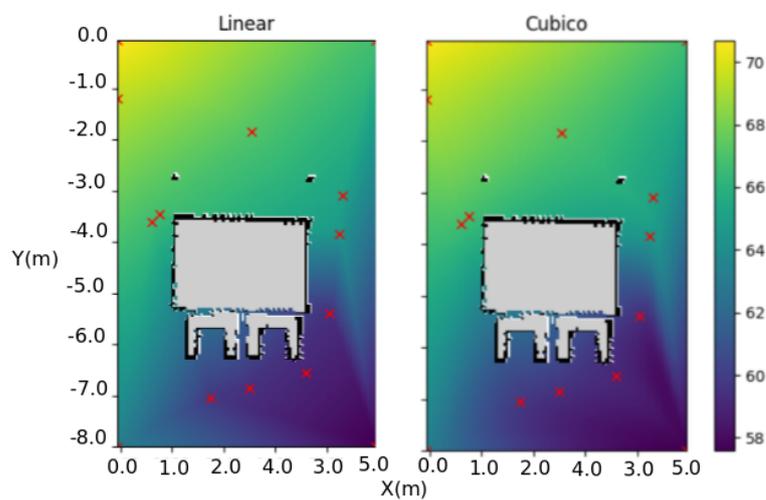
Figura 32 – Predição do mapa do sinal de rádio completo pelo método da interpolação linear (a) e cúbica (b).

Comparando o uso dos dois métodos de interpolação utilizados para gerar o mapa de rádio, percebe-se que o método linear apresentou melhores resultados. Tal análise pode ser vista de maneira explícita na Figura 33, onde o método cúbico apresentou maior discrepância no mapa com 15 pontos, principalmente nas bordas (onde o sinal varia de forma repentina). Como esperado, sistemas de interpolação simples não conseguem prever com precisão o mapa completo com poucos pontos.

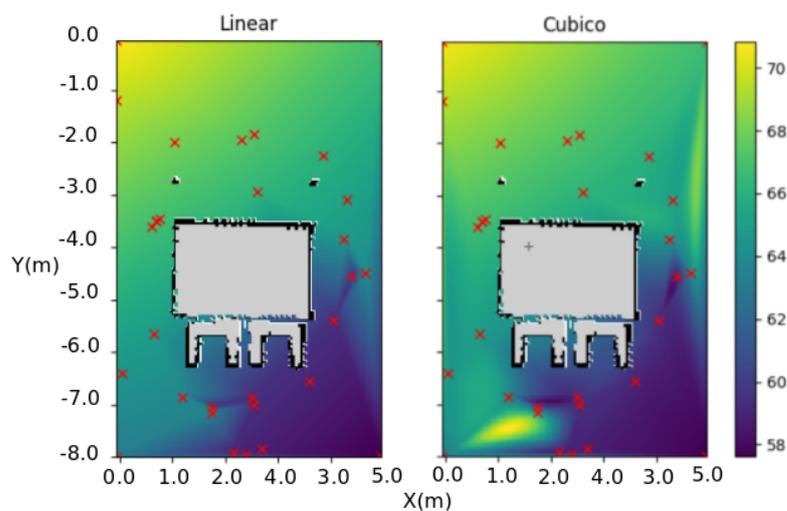
Para realizar o mapeamento, é importante levar em consideração o número de pontos obtidos. Conforme visto na Figura 33a, poucos pontos não são capazes de descrever o ambiente de maneira apropriada. A medida que há mais valores da intensidade do sinal de rádio, medidos em diferentes pontos, o mapa vai se aproximando do esperado. Neste sentido, para a análise da viabilidade do mapeamento de sinal de rádio no EspeleoRobô, foi também realizado um teste em um ambiente confinado representativo das atividades do EspeleoRobô (Figura 34), extraído do desafio DARPA Challenge SubT (ALLEN, 2018). O procedimento foi o mesmo adotado nos experimentos do ambiente *indoor* de laboratório, consistindo no mapeamento 2D, e em seguida, o mapeamento do sinal de rádio, havendo apenas a alteração do expoente de perda de sinal para simular um transmissor de rádio mais robusto. O resultado pode ser observado na Figura 34.



(a) 4 pontos de amostragem.

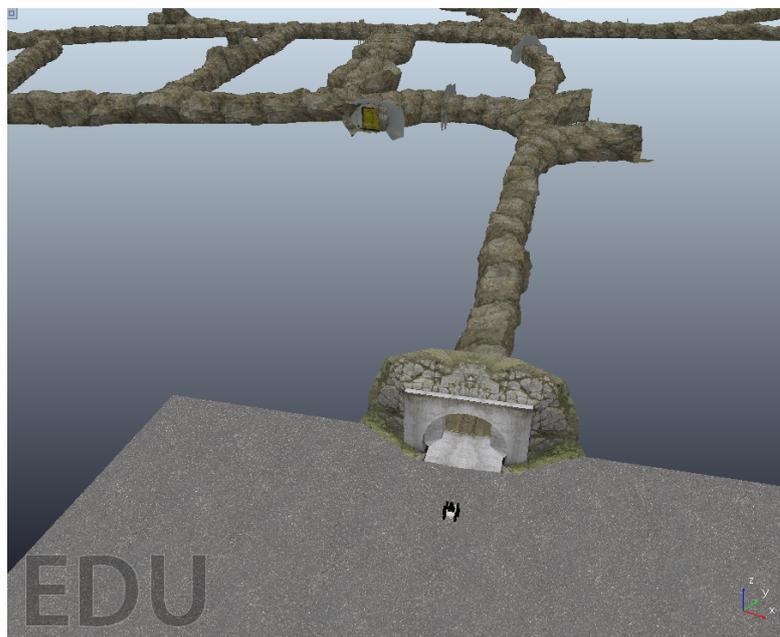


(b) 15 pontos de amostragem.

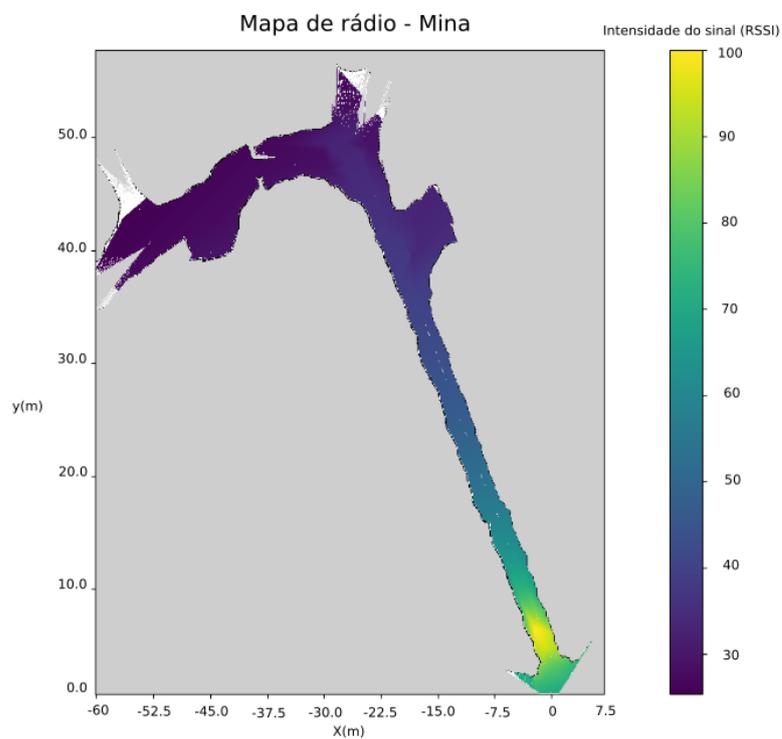


(c) 25 pontos de amostragem.

Figura 33 – Resultado do mapeamento de uma seção do mapa, considerando diferentes pontos colhidos: (a) 4 pontos, (b) 15 pontos, (c) 25 pontos.



(a) Mina Edgar do DARPA Challenge SubT.



(b) Interpolação de sinal dentro da mina.

Figura 34 – Simulação de interpolação de sinal de rádio em ambiente representativo: (a) mina Edgar do DARPA challenge SubT, e (b) mapa de ocupação de uma seção do mapa e seu sinal de rádio.

5 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

A utilização de robôs para tarefas de inspeção é uma ótima alternativa para não expor os funcionários a riscos. A implementação de um ambiente que consiga simular robôs de inspeção é essencial pois evita possíveis danos físicos ao robô, agiliza a etapa de testes e ainda pode ser utilizada como plataforma de treino. O ambiente desenvolvido no Capítulo 3 se mostrou robusto para diversas situações, como testes de teleoperação, análise de mobilidade e até mesmo operações autônomas. Isto se deve a versatilidade do simulador escolhido, pois o mesmo permite alterar o modelo dinâmico do robô, suas propriedades, sensores e até mesmo as formas do ambiente. Com isso, há a possibilidade de direcionar o processamento das simulações para áreas específicas, de acordo com a finalidade da operação.

Experimentos demonstram as vantagens de uso de modelos simplificados na simulação na maioria dos casos. Também foi apresentado o desempenho dos sensores simulados para diferentes ambientes virtuais. Estes sensores foram adaptados da maneira mais próxima aos utilizados no robô real. Além dos sensores nativos ao simulador, há a alternativa de utilizar plugins externos para otimizar o processamento de dados de um sensor custoso para o desempenho do simulador. Finalmente, é descrito um processo de mapeamento, planejamento de caminhos e controle do robô, que funciona inteiramente dentro do ambiente simulado e que pode ser usado no robô real. Esta demonstração é importante pois demonstra que a simulação funcionou conforme o esperado tanto em procedimentos operados quanto autônomos.

Trabalhos futuros relacionados ao ambiente de simulação terão como foco a validação de mais parâmetros da simulação, como o motor de física utilizado e o tempo de cada iteração da simulação. Será realizado também o aprimoramento da eficiência da configuração interna dos sensores do robô e também a adaptação apropriada das diversas configurações de roda do EspeleRobô. Sensores que demandam maior processamento também serão estudados a fim de encontrar uma alternativa viável, como a feita com o Velodyne adaptado. Além da simulação, serão avaliados novos algoritmos de navegação e controle e a viabilidade do CoppeliaSim para a simulação de ambientes de cooperação e colaboração com outros tipos de robôs.

Quanto ao mapeamento do sinal de rádio, foi observado que a implementação do pacote de navegação é uma ferramenta importante para ser adaptada ao robô. Por ser um robô de inspeção de ambientes confinados, o processo de mapeamento deve ser realizado com cautela, logo, a implementação do sistema de navegação auxilia principalmente por considerar o espaço de configuração do robô.

Foi visto que utilizar apenas a odometria de roda adaptada de um modelo diferencial não é o suficiente para o robô, isto por ele ser um robô do tipo skid-steering. Apesar de possuir erro considerável nos dados, a odometria de roda ainda pode ser utilizada em filtros probabilísticos,

desde que seja atribuído os valores corretos de covariância para o sistema.

Já o sistema de simulação do sinal de rádio mostrou ser uma boa alternativa para diversos testes, apesar de utilizar um fórmula relativamente simples. Uma forma de aprimorar tal sistema é analisando se há algum modo de utilizar a tecnologia de ray-tracing no CoppeliaSim. Com isto, artifícios como o uso de um sensor de proximidade para verificação de visada direta não serão mais necessários.

No ambiente confinado, o mapeamento do sinal de rádio se mostrou eficiente e aplicável ao robô real. Em conjunto com o sistema de mapeamento, o ideal é a utilização de um sistema de segurança que informe quando o sinal de rádio chegou a um valor limite como apresentado em [Amaral et al. \(2020\)](#).

Em trabalhos futuros, é recomendado o estudo da aplicabilidade de métodos de interpolação que permitam a previsão do sinal de rádio, como a utilização de processos gaussianos ou outros algoritmos baseados em aprendizado de máquina. Outras propostas interessantes são a utilização do mapa de rádio para auxiliar na localização do robô ([WANG et al., 2020](#)) ou no planejamento de caminhos.

REFERÊNCIAS

- ALLEN, B. Unearthing the Subterranean Environment. DARPA, 2018. Disponível em: <<https://www.subtchallenge.com/>>. Citado na página 50.
- AMARAL, I. et al. Sistema de alertas e operação assistida de um robô para a inspeção de ambientes confinados-espeleorobô. Anais da Sociedade Brasileira de Automática, v. 2, n. 1, 2020. Citado 2 vezes nas páginas 40 e 54.
- ANOUSAKI, G.; KYRIAKOPOULOS, K. J. A dead-reckoning scheme for skid-steered vehicles in outdoor environments. In: IEEE. IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004. [S.l.], 2004. v. 1, p. 580–585. Citado 2 vezes nas páginas 21 e 42.
- ARAI, K.; TOLLE, H. Color radiomap interpolation for efficient fingerprint wifi-based indoor location estimation. International Journal of Advanced Research in Artificial Intelligence, Citeseer, v. 2, n. 3, p. 10–15, 2013. Citado na página 26.
- ARVIN, F.; SAMSUDIN, K.; RAMLI, A. Development of ir-based short-range communication techniques for swarm robot applications. Advances in Electrical and Computer Engineering, v. 10, 11 2010. Citado na página 14.
- AZPÚRUA, H. et al. Towards semi-autonomous robotic inspection and mapping in confined spaces with the espeleorobô. Journal of Intelligent & Robotic Systems, Springer, v. 101, n. 4, p. 1–27, 2021. Citado na página 40.
- AZPURUA, H. et al. EspeleoRobô - a robotic device to inspect confined environments. In: 2019 19th Int. Conf. on Advanced Robotics (ICAR). [S.l.]: IEEE, 2019. Citado na página 19.
- BI, J. et al. A method of radio map construction based on crowdsourcing and interpolation for wi-fi positioning system. In: IEEE. 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN). [S.l.], 2018. p. 1–6. Citado na página 27.
- BULLET. Home. 2020. Disponível em: <<https://pybullet.org/>>. Acesso em: 2020-05-09. Citado na página 22.
- BUTNARU, T. et al. Mobile robot system controlled through mobile communications. In: _____. [S.l.: s.n.], 2005. p. 433–442. Citado na página 14.
- CID, A. et al. A simulated environment for the development and validation of an inspection robot for confined spaces. In: 2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE). [S.l.: s.n.], 2020. p. 1–6. Citado 2 vezes nas páginas 9 e 19.
- CIOBANU, O.; ROTARIU, M. Photogrammetric scanning and applications in medicine. In: TRANS TECH PUBL. Applied Mech. and Materials. [S.l.], 2014. v. 657. Citado na página 34.
- CISZEWSKI, M. et al. Modeling and simulation of a tracked mobile inspection robot in matlab and v-rep software. Journal of Automation, Mobile Robotics & Intelligent Systems, v. 11, p. 5–11, 06 2017. Citado na página 28.

- CM-LABS. Vortex Studio. 2020. Disponível em: <<https://www.cm-labs.com/vortex-studio/software/vortex-studio-academic-access/>>. Acesso em: 2020-05-09. Citado na página 22.
- COPPELIAROBOTICS. CoppeliaSim. 2019. Disponível em: <<https://www.coppeliarobotics.com/>>. Acesso em: 2020-05-04. Citado na página 23.
- COTA, E. Implementação e avaliação de técnicas de odometria aplicadas a um dispositivo robótico móvel. 2019. Citado 2 vezes nas páginas 21 e 42.
- CYBERBOTICS. Getting Started with Webots. 2020. Disponível em: <<https://cyberbotics.com/doc/guide/getting-started-with-webots>>. Acesso em: 2020-05-25. Citado na página 23.
- DELLAERT, F. et al. Monte carlo localization for mobile robots. In: IEEE. Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C). [S.l.], 1999. v. 2, p. 1322–1328. Citado na página 45.
- DOURADO, W. B. Avaliação de técnicas de interpolação de imagens digitais. Universidade Estadual Paulista (UNESP), 2014. Citado na página 27.
- EL-HAKIM, S. F. et al. Detailed 3d reconstruction of large-scale heritage sites with integrated techniques. IEEE computer graphics and applications, IEEE, v. 24, n. 3, p. 21–29, 2004. Citado na página 34.
- FILHO, A. M. d. S. T. Implementação da plataforma tecnológica embarcada em um dispositivo robótico móvel para inspeção de ambientes confinados: Espeleorobô. 2019. Citado 3 vezes nas páginas 9, 20 e 22.
- GONÇALVES, V. M. et al. Vector fields for robot navigation along time-varying curves in n -dimensions. IEEE Transactions on Robotics, IEEE, v. 26, n. 4, p. 647–659, 2010. Citado na página 40.
- HÄHNEL, B. F. D.; FOX, D. Gaussian processes for signal strength-based location estimation. In: Proceeding of robotics: science and systems. [S.l.: s.n.], 2006. Citado na página 26.
- HESS, W. et al. Real-time loop closure in 2d lidar slam. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). [S.l.: s.n.], 2016. p. 1271–1278. Citado na página 25.
- HICHRI, B. et al. Cooperative lifting and transport by a group of mobile robots. In International Symposium on Distributed Autonomous Robotic Systems, DARS 20, nov. 2014. Citado na página 13.
- HORNUNG, A. et al. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. Autonomous Robots, 2013. Disponível em: <<http://octomap.github.com>>. Citado na página 25.
- HORNUNG, A. et al. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. Autonomous Robots, 2013. Disponível em: <<http://octomap.github.com>>. Citado na página 40.
- KOENIG, N.; HOWARD, A. Design and use paradigms for gazebo, an open-source multi-robot simulator. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. Sendai, Japan: [s.n.], 2004. p. 2149–2154. Citado na página 22.

- KOHLBRECHER, S. et al. A flexible and scalable slam system with full 3d motion estimation. In: IEEE. 2011 IEEE international symposium on safety, security, and rescue robotics. [S.l.], 2011. p. 155–160. Citado na página 25.
- KRUMM, J.; PLATT, J. Minimizing calibration effort for an indoor 802.11 device location measurement system. Microsoft Research, Citeseer, 2003. Citado na página 27.
- LABBÉ, M.; MICHAUD, F. Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation: Labbé and michaud. Journal of Field Robotics, v. 36, 10 2018. Citado na página 25.
- LI, X.; GAO, X. Acceleration of the development for motion planning algorithms using v-rep. In: . [S.l.: s.n.], 2019. Citado na página 28.
- LIDORIS, G. et al. The autonomous city explorer (ace) project – mobile robot navigation in highly populated urban environments. In IEEE International Conference on Robotics and Automation, 2009. Citado na página 13.
- MARTZ, J.; AL-SABBAN, W.; SMITH, R. N. Survey of unmanned subterranean exploration, navigation, and localisation. IET Cyber-systems and Robotics, IET, v. 2, n. 1, p. 1–13, 2020. Citado na página 15.
- MELO, M. S. P. de et al. Analysis and comparison of robotics 3d simulators. In: IEEE. Symp. on Virtual and Aug. Reality (SVR). [S.l.], 2019. Citado na página 28.
- MILFORD, M. J. Robotic mapping methods. In: _____. Robot Navigation from Nature: Simultaneous Localisation, Mapping, and Path Planning based on Hippocampal Models. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. p. 15–28. ISBN 978-3-540-77520-1. Disponível em: <https://doi.org/10.1007/978-3-540-77520-1_3>. Citado na página 24.
- NAJNUDEL, M. Estudo de propagação em ambientes fechados para o planejamento de wlangs. Rio de Janeiro, v. 136, 2004. Citado na página 46.
- NASA. Mars Rover Communications. 2021. Disponível em: <<https://mars.nasa.gov/mars2020/spacecraft/rover/communications/>>. Citado na página 25.
- ODE. Home Page. 2020. Disponível em: <<https://www.ode.org/>>. Acesso em: 2020-05-09. Citado na página 22.
- PAOLANTI, M. et al. Mobile robot for retail surveying and inventory using visual and textual analysis of monocular pictures based on deep learning. In European Conference on Mobile Robots (ECMR), 2017. Citado na página 13.
- PAPADOPOULOS, D. A. R. E. G. A new measure of tipover stability margin for mobile manipulators. Proceedings of IEEE International Conference on Robotics & Automation, v. 4, p. 3111–3116, 1996. Citado na página 38.
- PINCIROLI, C. et al. ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems. Swarm Intelligence, Springer, Berlin, Germany, v. 6, n. 4, p. 271–295, 2012. Citado na página 23.
- PITONAKOVA, L. et al. Feature and performance comparison of the v-rep, gazebo and argos robot simulators. In: . [S.l.: s.n.], 2018. Citado na página 28.

- QUIGLEY, M. et al. Ros: an open-source robot operating system. In: KOBE, JAPAN. ICRA workshop on open source software. [S.l.], 2009. v. 3, n. 3.2, p. 5. Citado na página 22.
- RACKO, J.; MACHAJ, J.; BRIDA, P. Wi-fi fingerprint radio map creation by using interpolation. Procedia engineering, Elsevier, v. 192, p. 753–758, 2017. Citado na página 48.
- ROBERTS, J. M. et al. Autonomous control of underground mining vehicles using reactive navigation. In: IEEE. Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065). [S.l.], 2000. v. 4, p. 3790–3795. Citado na página 13.
- ROCHA, F. et al. Análise de mobilidade de um dispositivo robótico para inspeção remota de cavidades. SBAI 2017, 2017. Citado 3 vezes nas páginas 19, 20 e 40.
- ROS. About ROS. 2010. Disponível em: <<https://www.ros.org/about-ros/>>. Citado na página 20.
- ROS. TF. 2017. Disponível em: <<http://wiki.ros.org/tf>>. Acesso em: 2020-05-06. Citado na página 30.
- ROS. Map Server. 2018. Disponível em: <http://wiki.ros.org/map_server>. Acesso em: 2021-02-02. Citado na página 48.
- ROS. Rviz. 2018. Disponível em: <<http://wiki.ros.org/rviz>>. Acesso em: 2020-06-05. Citado na página 22.
- SANTOS, A. S. Planejamento de caminhos para robôs móveis em ambientes acidentados. Machine Vision and Applications, 2019. Citado na página 40.
- SERRALHEIRO, A. Elementos de Telecomunicações e Rádio-Propagação. Brasil: Instituto Universitario Militar, 2017. Citado na página 25.
- SHIH, R. Learning SOLIDWORKS 2020. [S.l.]: SDC Pub., 2020. Citado na página 29.
- SZWOCH, M.; KACZMAREK, A.; BARTOSZEWSKI, D. Sterio-reconstruction of 3d scenery for video games using stereo-photogrammetry. In: Proceedings of the Conference on Game Innovations (CGI), Lodz. [S.l.: s.n.], 2017. Citado na página 34.
- TALVITIE, J.; RENFORS, M.; LOHAN, E. S. Distance-based interpolation and extrapolation methods for rss-based localization with indoor wireless signals. IEEE transactions on vehicular technology, IEEE, v. 64, n. 4, p. 1340–1353, 2015. Citado na página 26.
- THRUN, S. Probabilistic robotics. Communications of the ACM, ACM New York, NY, USA, v. 45, n. 3, p. 281–308, 2002. Citado na página 25.
- THRUN, S. Robotic mapping: A survey. In: LAKEMEYER, G.; NEBEL, B. (Ed.). Exploring artificial intelligence in the new millennium. [S.l.]: Morgan Kaufmann, 2003. cap. 1, p. 1–35. Citado na página 23.
- TINTO, R. Pioneering Automation and Robotics in Mining. 2021. Disponível em: <<https://www.riotinto.com/about/innovation/automation>>. Citado na página 13.

- VALE. Vale terá a primeira mina operando somente com caminhões autônomos no Brasil. 2018. Disponível em: <<http://www.vale.com/brasil/pt/aboutvale/news/paginas/vale-tera-a-primeira-mina-operando-somente-com-caminhoes-autonomos-no-brasil.aspx>>. Citado na página 13.
- VALE, I. T. Espeleorobô. 2016. Disponível em: <<http://www.itv.org/inovacao/espeleorobo/>>. Citado na página 19.
- WANG, X. et al. Indoor radio map construction and localization with deep gaussian processes. IEEE Internet of Things Journal, IEEE, v. 7, n. 11, p. 11238–11249, 2020. Citado na página 54.
- WILLS, M. HTTP Streaming of ROS Image Topics in Multiple Formats. 2019. Disponível em: <https://wiki.ros.org/web_video_server>. Acesso em: 2020-05-07. Citado na página 31.
- XSENS. MTI User Manual. 2019. Disponível em: <https://www.xsens.com/hubfs/Downloads/usermanual/MTi_usermanual.pdf>. Acesso em: 2021-01-05. Citado 2 vezes nas páginas 9 e 33.
- YAGFAROV, R.; IVANOU, M.; AFANASYEV, I. Map comparison of lidar-based 2d slam algorithms using precise ground truth. In: IEEE. 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV). [S.l.], 2018. p. 1979–1983. Citado na página 43.
- ZHANG, X. et al. 2d lidar-based slam and path planning for indoor rescue using mobile robots. Journal of Advanced Transportation, Hindawi, v. 2020, 2020. Citado na página 43.
- ZHAO, H.; HUANG, B.; JIA, B. Applying kriging interpolation for wifi fingerprinting based indoor positioning systems. In: IEEE. 2016 IEEE Wireless Communications and Networking Conference. [S.l.], 2016. p. 1–6. Citado na página 27.