



Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Engenharia de Software

Auto-Localização e Mapeamento de Ambientes: Uma Abordagem para Robôs Simples

Autor: Rafael Fazzolino P. Barbosa

Orientador: Prof. Dr. Maurício Serrano

Coorientadora: Profa. Dra. Milene Serrano

Brasília, DF

2016



Rafael Fazzolino P. Barbosa

Auto-Localização e Mapeamento de Ambientes: Uma Abordagem para Robôs Simples

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software .

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. Maurício Serrano

Coorientador: Profa. Dra. Milene Serrano

Brasília, DF

2016

Rafael Fazzolino P. Barbosa

Auto-Localização e Mapeamento de Ambientes: Uma Abordagem para Robôs Simples/ Rafael Fazzolino P. Barbosa. – Brasília, DF, 2016-
80 p. : il. ; 30 cm.

Orientador: Prof. Dr. Maurício Serrano

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB
Faculdade UnB Gama - FGA , 2016.

1. Robótica. 2. Auto-Localização. I. Prof. Dr. Maurício Serrano. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Auto-Localização e Mapeamento de Ambientes: Uma Abordagem para Robôs Simples

CDU 02:141:005.6

Rafael Fazzolino P. Barbosa

Auto-Localização e Mapeamento de Ambientes: Uma Abordagem para Robôs Simples

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software .

Trabalho aprovado. Brasília, DF, 01 de junho de 2013:

Prof. Dr. Maurício Serrano
Orientador

Profa. Dra. Milene Serrano
Coorientadora

Prof. Ms. Giovanni Almeida Santos
Convidado 1

Profa. Dra. Carla Silva Rocha Aguiar
Convidado 2

Brasília, DF
2016

*Este trabalho é dedicado às crianças adultas que,
quando pequenas, sonharam em se tornar cientistas.*

Agradecimentos

Agradeço, primeiramente, à minha mãe, Monica Fazzolino Pinto, por todo o apoio e compreensão durante minha caminhada até aqui. À minha família como um todo e a todos os meus amigos da universidade, assim como os amigos que obtive durante a vida e à minha namorada, por toda a força que me deram.

Agradeço ao prof. Dr. Maurício Serrano por todo o apoio durante disciplinas, projeto e TCC. E a profa. Dra. Milene Serrano por toda a paciência e ajuda durante o curso como um todo. Agradeço ainda, aos dois professores, pela oportunidade e apoio durante a participação no projeto de pesquisa orientado diretamente por eles.

Agradeço a todos os professores do curso de Engenharia de Software, em especial, ao prof. Dr. Sergio Freitas, por ampliar minhas visões em relação a projetos de pesquisa e extensão na universidade. Entre outros, destaco a profa. Dra. Rejane Costa, por todo o apoio durante o curso.

Resumo

O resumo deve ressaltar o objetivo, o método, os resultados e as conclusões do documento. A ordem e a extensão destes itens dependem do tipo de resumo (informativo ou indicativo) e do tratamento que cada item recebe no documento original. O resumo deve ser precedido da referência do documento, com exceção do resumo inserido no próprio documento. (...) As palavras-chave devem figurar logo abaixo do resumo, antecidas da expressão Palavras-chave:, separadas entre si por ponto e finalizadas também por ponto. O texto pode conter no mínimo 150 e no máximo 500 palavras, é aconselhável que sejam utilizadas 200 palavras. E não se separa o texto do resumo em parágrafos.

Palavras-chaves: latex. abntex. editoração de texto.

Abstract

This is the english abstract.

Key-words: latex. abntex. text editoration.

Lista de ilustrações

Figura 1 – Utilização do filtro de Kalman	30
Figura 2 – Microprocessador central	35
Figura 3 – Sensor RGB - Lego	35
Figura 4 – Atuador - Lego	36
Figura 5 – Kit Mindsotm completo - Lego	36
Figura 6 – Funções básicas RoboMind	40
Figura 7 – Funções básicas Scratch	41
Figura 8 – Ambiente de desenvolvimento em Logo	42
Figura 9 – Arquitetura do robô	47
Figura 10 – Montagem do robô	48
Figura 11 – Ambiente utilizado	49
Figura 12 – Página utilizada para versionamento do trabalho	50
Figura 13 – Processo metodológico	56
Figura 14 – Processo de revisão sistemática	59
Figura 15 – Processo geral de revisão sistemática	61
Figura 16 – Arquitetura de comunicação	69
Figura 17 – Emissão do sinal sonoro - sonar	72
Figura 18 – Sensor de toque.	76
Figura 19 – Sensor ultrasônico.	76
Figura 20 – Emissão do sinal ultrasônico.	77
Figura 21 – Trabalho desenvolvido	79

Lista de tabelas

Tabela 1 – Metodologia de pesquisa.	55
Tabela 2 – Cronograma de atividades TCC_1.	55
Tabela 3 – Cronograma de atividades TCC_2.	56
Tabela 4 – Processo de revisão sistemática	60
Tabela 5 – Resultados obtidos com a <i>string</i> inicial	64
Tabela 6 – Exemplo de registro de material	65
Tabela 7 – Comparação das <i>strings</i> inicial e final	68
Tabela 8 – Contabilização de estratégias (SLAM)	73

Lista de abreviaturas e siglas

MIT	Massachusetts Institute of Technology
API	Application Program Interface
BPMN	Business Process Model and Notation
IDE	Integrated Development Environment
RAM	Random Access Memory
USB	Universal Serial Bus
SLAM	Simultaneous Localization and Mapping

Sumário

1	INTRODUÇÃO	21
1.1	Contextualização	21
1.2	Problema de pesquisa	23
1.3	Justificativa	23
1.4	Objetivos	24
1.4.1	Objetivo geral	24
1.4.2	Objetivos específicos	24
1.5	Organização do trabalho	25
2	REFERENCIAL TEÓRICO	27
2.1	Robótica e a auto-localização	27
2.2	O problema de SLAM	29
2.2.1	Filtro de Kalman	30
2.2.2	Filtro de partículas	31
2.3	Robótica educacional	32
2.3.1	Lego Mindstorms NXT	34
2.4	Considerações parciais	36
3	SUPORTE TECNOLÓGICO	39
3.1	Engenharia de software	39
3.1.1	GIT	39
3.1.2	Github	39
3.1.3	Bonita BPMN	39
3.1.4	Linux Mint	39
3.1.5	LaTeX	40
3.1.6	Sublime Text 3	40
3.2	Robótica educacional	40
3.2.1	RoboMind 6.0	40
3.2.2	Scratch	41
3.2.3	Linguagem Logo	41
3.2.4	Kodu Game Labs	42
3.2.5	Alice 2.0	43
3.2.6	Mindstorms Lego	43
3.3	Considerações parciais	43
4	PROPOSTA	45

4.1	Obtenção da base teórica	45
4.2	Adaptação e implementação	45
4.2.1	Arquitetura do Robô	46
4.2.2	Montagem do robô	47
4.2.3	Ambiente utilizado	48
4.2.4	Condução do trabalho	49
4.3	Considerações parciais	50
5	METODOLOGIA	53
5.1	Classificação da pesquisa	53
5.1.1	Abordagem da pesquisa	53
5.1.2	Natureza da pesquisa	54
5.1.3	Objetivos da pesquisa	54
5.1.4	Procedimentos	54
5.2	Processo metodológico	55
5.2.1	Revisão sistemática	58
5.2.1.1	Processo de revisão sistemática	58
6	RESULTADOS PARCIAIS	61
6.1	Revisão sistemática	61
6.1.1	Planejamento da revisão	61
6.1.1.1	Objetivos e questão de pesquisa	61
6.1.1.2	Estratégia de pesquisa	62
6.1.1.3	Procedimento de seleção	64
6.1.1.4	Avaliação da qualidade	64
6.1.1.5	Extração de dados	65
6.1.2	Condução da revisão	65
6.1.3	Publicação dos resultados	68
6.1.3.1	Arquitetura da solução	69
6.1.3.2	Técnica probabilística utilizada	70
6.1.3.3	Informações disponíveis	71
6.2	Desenvolvimento prático	73
6.2.1	Planejamento e condução	73
6.2.2	Características técnicas	74
6.2.2.1	Seleção da linguagem e ambiente de desenvolvimento	74
6.2.2.2	Atuadores	75
6.2.2.3	Sensores	75
6.2.3	Computador central NXT (Brick)	77
6.3	Considerações parciais	77

7	CONSIDERAÇÕES FINAIS	79
---	--------------------------------	----

1 Introdução

A robótica é um termo que gera bastante curiosidade para qualquer pessoa, desde uma criança aprendendo a ler, até idosos surpresos com as novidades. Por esse e outros motivos, a robótica vem sendo pesquisada e evoluída em diversos contextos, como a robótica educacional, por exemplo.

Entre as áreas de pesquisa que se enquadram na robótica, uma delas merece atenção especial: a robótica móvel. A mobilidade robótica é um dos grandes desafios para a comunidade devido à complexidade e à imprecisão dos dados obtidos, o qual será apresentado durante este trabalho de conclusão de curso. As próximas seções deste capítulo buscam apresentar a contextualização da pesquisa, a questão de pesquisa a ser investigada, a justificativa para realização da mesma, e os objetivos a serem alcançados.

1.1 Contextualização

O nascimento da robótica se deu no contexto industrial, no qual ferramentas autônomas foram desenvolvidas para executar atividades de forma repetitiva e incansável, maximizando a qualidade dos produtos e minimizando o custo e o tempo para produção dos mesmos [?]. Segundo [?], a palavra robô é derivada da palavra *robota*, de origem eslava, que significa *trabalho forçado*, ou seja, robôs podem ser considerados ferramentas incansáveis que apóiam o trabalho humano.

Segundo [?], a autonomia de um robô é condicionada pela sua capacidade de perceber o ambiente de navegação, interagindo com o meio e realizando tarefas com o mínimo de precisão. Este mínimo, de acordo com [?], seria a navegação sem colisão em obstáculos.

Para que robôs sejam capazes de navegar em um ambiente desconhecido sem que haja colisão em objetos e obstáculos, os mesmos necessitam de informações sobre este ambiente. Estas informações são adquiridas utilizando sensores. Como foi apresentado por [?], no livro de Robótica Industrial, os sensores possuem o dever de fornecer informações ao sistema de controle do robô sobre distâncias de objetos, posição do robô, contato do robô com objetos, força exercida sobre objetos, cor e textura dos objetos, entre outras.

Além de obter dados sobre o ambiente, o robô precisa se auto-localizar para processar as informações obtidas e traçar rotas sem colisões até o ponto de destino. Para isso, foram desenvolvidas muitas formas de auto-localização, algumas delas são citadas por [?], como:

- **Utilização de Mapas:** O robô conhece o mapa onde realizará a navegação à priori, conhecendo os obstáculos e os caminhos possíveis. Possuindo essas informações, o robô irá traçar as rotas mais eficientes para chegar em seu objetivo.
- **Localização Relativa em Grupos:** Esta técnica utiliza a navegação simultânea de muitos robôs, cada robô sabe a posição relativa dos outros robôs, podendo calcular sua posição relativa.
- **Utilização de Pontos de Referência:** Conhecendo pontos de referência que estão distribuídos pelo mapa de navegação, o robô consegue calcular sua posição através da técnica de triangulação.
- **Localização Absoluta com GPS:** A partir desta técnica, é fácil obter a posição absoluta do robô em relação à terra. O grande problema desta técnica é a margem de erro presente no sistema de GPS, inviável para navegações internas.
- **Utilização de Bússolas:** É uma técnica interessante para reconhecimento da orientação do robô, o que facilita muito na navegação do mesmo. Entretanto, as Bússolas são muito frágeis a interferências externas, como por exemplo, a proximidade de materiais ferro-magnéticos ou as fugas magnéticas dos motores presentes no próprio robô.
- **Odometria:** Consiste na medição da distância relativa percorrida pelo robô, utilizando sensores presentes nas rodas do mesmo. Necessita do conhecimento do ponto de origem.

As formas apresentadas anteriormente, para se trabalhar com auto-localização, possuem características únicas que as adequam para diferentes contextos de navegação. Por exemplo, segundo [?], a Utilização de Mapas é uma técnica bastante útil quando se está trabalhando com um ambiente conhecido e estático, porém, em ambientes mutáveis e não conhecidos, essa estratégia se torna um problema. A Localização Relativa em Grupos é a técnica adequada quando a navegação envolve muitos robôs, a qual não necessita de conhecimento prévio do mapa. A Utilização de Pontos de Referência é uma técnica comumente utilizada, sendo útil quando não se conhece o ambiente de navegação. Entretanto, os pontos de referência, nesse caso, precisam ser conhecidos.

Quando se têm ambientes abertos e amplos, a técnica de Localização Absoluta com GPS é a mais utilizada. Entretanto, sua margem de erro torna a navegação em ambientes pequenos ou fechados inviável, como apresenta [?]. A Navegação com utilização de Bússolas garante um apoio muito útil para orientação do robô. Contudo, essa técnica gera problemas relacionados a interferências externas, como materiais eletromagnéticos próximos à bússola [?].

A técnica de Odometria é muito utilizada em navegações curtas, em ambientes com o piso regular e plano. Entretanto, segundo [?], esta técnica se caracteriza pela adição de erros a cada centímetro percorrido, por meio de derrapagens e falhas no giro das rodas.

Desse modo, é fácil perceber que cada técnica possui características que se adaptam melhor para diferentes situações. Ao longo deste trabalho, o principal foco de interesse é a navegabilidade de robôs simples e baratos, como os Kit Lego Mindstorms¹. Tais kits possuem poucas opções de sensores e características limitadas [?]. Os sensores do kit que serão utilizados neste trabalho são: *Sensor de proximidade*, *sensor RGB*, *sensor de contato* e *sensor odométrico*.

1.2 Problema de pesquisa

Todas as técnicas de auto-localização apresentadas na seção anterior já foram testadas, comparadas e refatoradas em diferentes contextos, desde a navegação marítima até questões relacionadas à tecnologia aeroespacial [?]. Porém, sabe-se que, nestes contextos, o *hardware* utilizado para navegação era de alta tecnologia, possuindo processadores de alto desempenho, com a utilização de processamento paralelo [?], por exemplo, e sensores precisos.

Já em um contexto educacional, a infraestrutura disponível nem sempre engloba os critérios necessários para aplicação das técnicas de auto-localização. Um exemplo disso é a utilização, por [?], de uma câmera omnidirecional para obter informações sobre o ambiente.

Desse modo, vê-se a necessidade da adaptação de técnicas de auto-localização para o contexto da Robótica Educacional, utilizando os kits de robótica *Mindstorms* da LEGO. Nesses kits, a capacidade de processamento e a precisão dos sensores e dos atuadores são limitadas. A intenção é tomar como base a técnica de mapeamento do ambiente e auto-localização simultâneos (conhecida como problema de SLAM - *Simultaneous Location and Mapping*) [?].

A questão de pesquisa que será discutida durante este trabalho é "*Como tratar o problema de SLAM no contexto de robôs simples?*".

1.3 Justificativa

A utilização da Robótica como uma forma de ensinar programação em escolas e faculdades, conhecida como Robótica Educacional [?], traz alguns benefícios para o aluno. Conforme colocado pelos autores [?], [?], [?] e [?], alguns desses benefícios são:

¹ mindstorms.lego.com

- maior interesse pelos conteúdos estudados em aula;
- capacidade de trabalhar em grupo;
- aplicação prática do conhecimento teórico, e
- multidisciplinaridade.

A Universidade de Brasília utiliza esta abordagem de ensino/aprendizagem durante a disciplina de Introdução à Robótica Educacional, ministrada em 2016, pelo professor Dr. Maurício Serrano. A disciplina utiliza os Kits de robótica Mindstorms, da Lego, para desenvolvimento de soluções dos problemas presentes em um tapete de missões. A organização da disciplina se inspira nos campeonatos de robótica, como o *ciber-rato* [?] ou *micro-rato* [?]. Neste tipo de campeonato, a navegação é o quesito mais importante [?], a qual deve possuir a menor margem de erro para solucionar as missões.

As missões utilizadas durante a disciplina da UnB são referentes a problemas recorrentes no contexto da robótica mundial. A solução dos problemas é uma adaptação das técnicas existentes para o contexto limitado da disciplina, onde são utilizadas apenas ferramentas presentes no kit *Mindstorms* da Lego. Esta adaptação exige um conhecimento específico sobre a técnica, para que o estudante possa identificar características relevantes e adaptá-las de acordo com o *hardware* disponível.

A disciplina já influenciou dois estudantes de Engenharia de Software a se aprofundarem no contexto da Robótica e Robótica Educacional, gerando dois trabalhos de conclusão de curso, [?], que desenvolveu um *framework* de definição de trajetórias para robôs móveis, e [?], que desenvolveu um algoritmo para tomada de decisões estratégicas em robótica educacional. O trabalho atual vem para complementar os trabalhos realizados pelos dois alunos.

1.4 Objetivos

1.4.1 Objetivo geral

Adaptar técnicas de resolução do problema de SLAM para o contexto de robôs simples, utilizando os kits de robótica *Mindstorms* da Lego, em um primeiro momento.

1.4.2 Objetivos específicos

- Identificar uma solução para o problema de SLAM em um contexto simplificado;
- Propor adaptação para o contexto de robôs simples, na Robótica Educacional;
- Implementar adaptação.

1.5 Organização do trabalho

Este trabalho de conclusão de curso está organizado nos capítulos:

- Introdução: Capítulo referente à contextualização, levantamento da questão de pesquisa, justificativa e definição dos objetivos do trabalho;
- Referencial teórico: O objetivo deste capítulo é fornecer ao leitor o conhecimento necessário para compreender a pesquisa realizada. O capítulo é sub-dividido nas seções *robótica e a auto-localização*, *o problema de SLAM* e *robótica educacional*;
- Suporte tecnológico: Apresenta as ferramentas e tecnologias utilizadas para auxiliar o desenvolvimento desta pesquisa, desde a pesquisa bibliográfica e documentação, até o desenvolvimento da prova de conceito e apresentação;
- Metodologia: Este capítulo busca apresentar as técnicas utilizadas para a realização da pesquisa, definindo as atividades a serem desempenhadas para conclusão do trabalho, e
- Resultados parciais: Neste capítulo, são apresentados os resultados obtidos durante o desenvolvimento da primeira etapa deste trabalho de conclusão de curso.
- Considerações finais: Capítulo final do TCC_1 que tem como objetivo apresentar o status atual do trabalho, assim como o que se espera para a próxima etapa do trabalho.

2 Referencial teórico

Parte da teoria relacionada à robótica móvel e à robótica educacional será descrita, brevemente, ao longo deste capítulo, visando facilitar o entendimento dos termos utilizados durante a realização desta pesquisa.

2.1 Robótica e a auto-localização

O nascimento da robótica se deu no contexto industrial, onde a automação de atividades repetitivas garantiu maior eficiência e, conseqüentemente, maior lucro [?]. Porém, com o passar dos anos, a robótica vem se expandindo e fazendo parte da vida cotidiana de muitos [?]. A robótica é uma importante ferramenta que apóia o trabalho humano em diversos contextos, seja para a limpeza de uma casa [?] ou até para explorar novos planetas, por exemplo [?].

No contexto da robótica móvel, existem, por exemplo, os robôs de serviço. Esses robôs vêm sendo largamente evoluídos pela comunidade de robótica [?]. Ainda segundo [?], estes robôs, geralmente, desempenham ações que contemplam aplicações, como:

- Aplicações que envolvam risco de vida significativo para humanos [?];
- Funções economicamente desvantajosas no uso de trabalhadores humanos [?];
- Uso humanitário (cadeiras de rodas autônomas, por exemplo);
- Uso educacional [?].

Robôs de serviço capazes de se deslocar livremente pelo ambiente, conhecendo-o, poderão realizar suas atividades de forma mais eficiente [?], o que garante sua autonomia.

Segundo [?], a palavra *automação* traz à mente a noção de que a máquina será capaz de sentir e interagir com o ambiente, conseguindo se localizar e navegar por ele, executando suas atividades. Para que esta navegação seja possível, o robô precisa obter informações sobre o ambiente, as quais são obtidas a partir da utilização de sensores [?]. Segundo [?], existem inúmeros tipos de sensores, desde sensores de toque até sensores de visão ou de som. Os sensores mais utilizados em robôs móveis, segundo [?], são:

- *Odômetro*:

São sensores de implementação simples e de baixo custo. Este tipo de sensor conta a quantidade de rotações de cada roda do robô, o que permite calcular o trajeto

percorrido pelo mesmo. Esta técnica é conhecida como *dead-reckoning*, como é apresentado por [?]. Segundo [?], técnicas como o *dead-reckoning* são bastante suscetíveis a erros, graças ao não alinhamento das rodas, derrapagens das mesmas e até erros no sinal dos sensores.

- *Câmera:*

A utilização de câmeras pode ser bastante útil quando se deseja navegar em um ambiente fechado, construído pelo homem e com características bem definidas, como afirma [?]. Porém, sua utilização possui uma exigência computacional [?] que muitas vezes pode inviabilizá-la.

- *Sonar:*

É um tipo de sensor de proximidade barato e de fácil utilização e, por esse motivo, é bastante utilizado em robôs móveis para ambientes fechados [?]. Ainda segundo [?], em ambientes abertos, este é um sensor falho, devido ao seu alcance limitado e por não ser direcionado.

- *Infravermelho:*

É um tipo de sensor muito semelhante aos sonares, porém, estes são direcionados, ou seja, são capazes de identificar a direção do objeto [?].

- *Laser:*

Também é um tipo de sensor semelhante aos sonares e sensores de infravermelho, entretanto, esses, além de serem direcionados, são mais precisos. Tal característica os tornam equipamentos mais caros, como apresenta [?].

A utilização de cada tipo de sensor se dá de acordo com o contexto em que se deseja navegar, levando em consideração a maneira mais viável de entender o ambiente ao seu redor [?]. Utilizando os sensores, o robô obterá informações sobre o ambiente, e precisa processá-las para *entender* o mesmo. A maneira de processar (analisar) essas informações também depende do contexto da navegação, como mostra [?].

Para que a navegação ocorra sem colisões em obstáculos, o robô precisa, além de informações sobre o ambiente, informações relacionadas a sua posição em relação a este ambiente [?]. Para solucionar este problema, uma técnica bastante difundida é a utilização de mapas, como mostra [?]. Nesta técnica, o robô recebe o mapa do ambiente que se deseja navegar a priori e, a partir deste mapa, traça sua trajetória sem obstáculos. Utilizando este mapa e informações do ambiente, o robô é capaz de se auto-localizar no ambiente, navegando com maior precisão [?].

Por outro lado, esta técnica possui requisitos que, muitas vezes, são inviáveis, como o conhecimento prévio do mapa do ambiente. Caso o objetivo seja navegar em

um ambiente desconhecido, onde não há mapa nem pontos de referência já conhecidos, deve-se buscar formas de se auto-localizar e navegar utilizando apenas as informações obtidas pelos sensores. Para isso, vem sendo discutida na comunidade de pesquisadores em robótica móvel, a resolução do problema de SLAM (Auto-Localização e Mapeamento de Ambientes Simultâneos). Segundo [?], SLAM é considerado, pela comunidade, como o *Santo Graal* da robótica móvel.

2.2 O problema de SLAM

Auto-localização e Mapeamento de Ambientes Simultâneos (SLAM) é uma técnica bastante utilizada para navegação em diferentes contextos, como na navegação marítima, por exemplo [?]. Esta técnica garante a possibilidade do robô navegar em um ambiente desconhecido, construindo um mapa do ambiente e, simultaneamente, utilizar este mapa para se auto-localizar em relação ao ambiente [?]. Ou seja, a técnica de SLAM, utilizando conceitos de inteligência artificial, garante autonomia móvel à máquina.

Segundo [?], uma máquina capaz de partir de um ponto de origem desconhecido em um ambiente desconhecido e, utilizando seus sensores, mapear o ambiente, utilizando este mapa, simultaneamente, para se auto-localizar no ambiente faz jus à palavra *robô*. Desse modo, com a vontade de criar verdadeiros *robôs*, pesquisadores em robótica móvel vêm desenvolvendo e evoluindo diversas soluções para o problema de SLAM, nos mais diferentes contextos [?]. Esta busca por uma solução elegante ao problema de SLAM possui o intuito de maximizar a efetividade da navegação autônoma [?].

A possibilidade de se auto-localizar em um ambiente garante façanhas importantes para a robótica, como a resolução do problema do *sequestro do robô*, por exemplo, como mostra [?]. Este é um problema especial de localização global no campo da robótica móvel, onde o desafio deste problema, segundo [?], é fazer com que o robô seja capaz de se localizar em um mapa após ser sequestrado.

A ocorrência do *sequestro* se dá quando, o robô, enquanto navega em um ambiente, é retirado do seu local e levado a outro totalmente desconhecido [?]. Em uma situação normal, o robô não seria capaz de entender que não está mais no local onde estava, fazendo com que o mesmo se perdesse. Diversas técnicas para solucionar este problema já foram desenvolvidas, geralmente utilizando como apoio o filtro de partículas, que será apresentado mais à frente [?].

De acordo com [?], graças à natureza imperfeita presente nos sensores, à falta de previsibilidade em ambientes reais de atuação e à necessidade de aproximações para viabilizar a análise de decisões computacionais, a robótica é uma ciência dependente de algoritmos probabilísticos. Desse modo, as diversas soluções desenvolvidas, geralmente, diferem umas das outras pela utilização dos algoritmos probabilísticos. Entre os algoritmos

mais utilizados, destacam-se o *filtro de kalman* e o *filtro de partículas*, como mostram as seções 2.2.1 e 2.2.2.

2.2.1 Filtro de Kalman

De acordo com [?], a abordagem probabilística mais utilizada para resolução do problema de SLAM é a utilização do filtro de Kalman. [?] descreve o filtro de Kalman como: "*Um algoritmo recursivo de processamento de informações, proporcionando a estimativa ótima do estado de um sistema dinâmico com ruído linear*". Sua utilização está representada de forma clara na Figura 1.

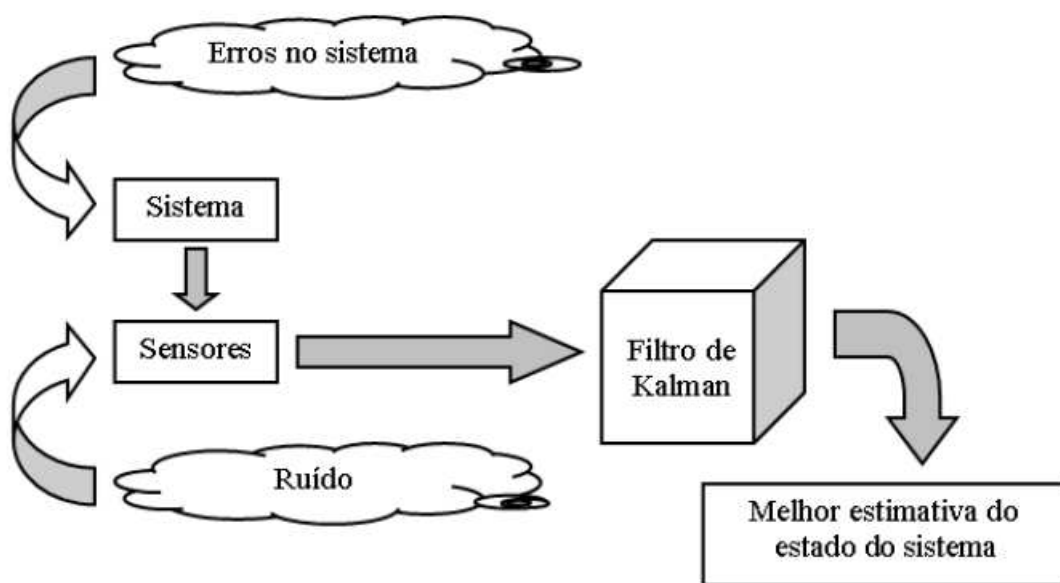


Figura 1 – Utilização do filtro de Kalman. Fonte: [?].

Como mostra [?], entre as disciplinas presentes na utilização deste filtro, encontram-se:

- Mínimos quadrados;
- Teoria das probabilidades;
- Sistemas dinâmicos;
- Sistemas estocásticos, e
- Álgebra.

Apesar das colocações, a utilização da abordagem do filtro de Kalman, geralmente, exige requisitos computacionais de alto custo, dificultando sua utilização em um contexto de robôs simples, onde baixo processamento e pouca memória fazem parte da realidade [?].

Em seu trabalho, [?] exemplifica este problema, quando quantifica a relação entre o crescimento do requisito computacional necessário para a quantidade de pontos de referência no ambiente, com a utilização do filtro de Kalman. Segundo ele, enquanto a quantidade de pontos de referência aumenta em N , os requisitos computacionais e de armazenamento necessários aumentam em N^2 . [?] mostra que este problema pode ser solucionado, em parte, utilizando técnicas de aproximação delimitada, por exemplo. Entretanto, estas técnicas minimizam o problema, mas não o solucionam completamente [?].

2.2.2 Filtro de partículas

Outra abordagem bastante utilizada para resolução do problema de SLAM envolve a utilização do filtro de Partículas, que é uma técnica para implementação de um *Filtro Bayesiano* de forma recursiva, utilizando o método de *Monte Carlo* [?]. O método de Monte Carlo baseia-se em amostragens aleatórias em grande quantidade, com o objetivo de estabelecer o valor de uma grandeza que não é disponível através de uma expressão matemática [?] e [?].

O funcionamento desta abordagem baseia-se em subdividir o ambiente em partículas espalhadas uniformemente, que representam o robô munido de seus sensores [?]. Cada partícula é uma hipótese da posição atual do robô [?]. O robô e as partículas obtêm informações do ambiente; o robô utilizando seus sensores e as partículas utilizando equações matemáticas para geração destes dados [?]. As informações obtidas pelo robô real são comparadas com as informações de cada partícula, excluindo as partículas que não oferecem informações semelhantes às do robô real [?]. Com o passar dos ciclos, as partículas que ainda continuam no ambiente representam a posição atual do robô [?].

A partir da comparação entre estas duas abordagens (filtro de Kalman e filtro de Partículas) [?], observou-se que as duas possuem vantagens e desvantagens. O filtro de Kalman converge mesmo com um estado inicial impreciso, fornecendo informações sobre as incertezas presentes em cada estágio e permitindo, ainda, a incorporação de toda informação disponível (sensores) para minimizar a margem de erro da estimativa [?]. Porém, ele só garante a efetividade da estimativa para sistemas lineares com ruídos gaussianos [?]. O mesmo não resolve o problema do *sequestro do robô*, apresentado anteriormente.

Já o filtro de Partículas, permite sua utilização em sistemas sem ruído gaussiano, e soluciona o problema do sequestro do robô [?]. Entre suas desvantagens, encontram-se o custo computacional e a dificuldade da definição da quantidade ideal de partículas a serem utilizadas [?].

As duas abordagens apresentadas anteriormente são amplamente utilizadas em diversos contextos do mundo real, contemplando questões importantes a serem estudadas e refinadas por estudiosos de engenharia de todo o mundo, desde estudantes que buscam

ingressar nesta área até profissionais da área [?]. Desse modo, vê-se a necessidade da adaptação de técnicas de resolução do problema de SLAM, seja a partir de uma abordagem baseada no filtro de Kalman, ou de uma abordagem baseada no filtro de Partículas, para um contexto Educacional. Nesse contexto, sensores, capacidades de processamento e memória são bastante limitados [?].

2.3 Robótica educacional

A utilização da robótica como uma ferramenta de apoio ao aprendizado vem sendo ampliada com o passar dos anos [?]. Principalmente, graças ao reconhecimento, por muitos autores, dos benefícios da utilização destas tecnologias como abordagem de ensino, onde os alunos são inseridos no problema real e instigados a solucioná-lo [?], [?], [?].

Entre os benefícios apresentados por [?] e [?], pode-se destacar, o maior interesse dos alunos sobre o tema, a abordagem facilitadora para o relacionamento entre aluno, professor e conteúdo, a experiência com trabalho em grupo, a multidisciplinariedade e a construção do conhecimento por parte do aluno.

Referente ao aperfeiçoamento do interesse dos alunos sobre o tema, [?] e [?] apresentam motivos deste aperfeiçoamento, como:

- Curiosidade sobre a tecnologia;
- Reconhecimento do problema real como um problema cotidiano do aluno, e
- Utilização do princípio *Hands-on* [?]

Sobre o aperfeiçoamento da relação entre alunos, professor e conteúdo, [?] destaca a reformulação do padrão de ensino e aprendizagem durante a aula. De acordo com [?], no contexto do ensino tradicional, o aluno busca *clonar* o conhecimento do professor, decorando informações para, no futuro, utilizá-los no contexto real. [?] acrescenta que, esta abordagem, além de limitar o aprendizado do aluno ao conhecimento do professor, minimiza a capacidade de aprendizado do aluno, já que o conhecimento não é construído, e sim repassado.

Já na abordagem da utilização de tecnologias no contexto educacional, a resolução prática do problema atacado faz com que alunos e professores trabalhem lado-a-lado, muitas vezes realizando troca de papéis aluno/professor, como apresentam [?] e [?]. Desse modo, em muitas ocasiões, os alunos se encontram explicando uma possível solução do problema ao professor, o que acaba com o aprendizado limitado ao conhecimento do professor [?]. [?] caracteriza o professor como um fio condutor do conhecimento, e não a fonte do mesmo.

Já, de acordo com a relação da experiência do aluno em trabalhos em grupo, a abordagem *Hands-on* garante que todos os envolvidos na solução devem interagir entre si, trocando conhecimento e ideias [?]. Qualquer contexto em que se deseja trabalhar, atualmente, envolve inúmeras atividades de trabalho em grupo [?]. Desse modo, o aperfeiçoamento da capacidade de trabalhar em grupo é uma atividade essencial para o futuro profissional do aluno [?].

Outra característica importante da utilização da tecnologia como ferramenta de aprendizado é a multidisciplinariedade [?], já que conteúdos referentes a diversas disciplinas são trabalhados e adaptados para buscar a solução do problema em investigação. [?] sugere a utilização de atividades multidisciplinares para aprendizado de conteúdos base, como matemática e física, minimizando o problema do conhecimento parcial de certos conteúdos.

Além de todos os benefícios apresentados acima, uma forte qualidade da abordagem *Hands-on* é referente à construção do conhecimento por parte do aluno [?]. Esta construção é gerada utilizando conhecimentos básicos de diferentes disciplinas para solucionar um problema que exige a integração de diversos conteúdos, seguindo uma filosofia construcionista, como é apresentado por [?].

De acordo com [?], quando o aluno se sente imerso no problema trabalhado, a maneira com que o mesmo aprende é aperfeiçoada, maximizando as relações entre aluno, professor e conteúdo. Este pensamento segue uma filosofia construcionista, a qual, de acordo com [?], implica no objetivo de ensinar, de forma a produzir o máximo de aprendizagem a partir do mínimo de ensino.

A utilização de aplicações rotineiras do mundo real é uma forma bastante eficaz de obter máximo interesse dos alunos nos conteúdos apresentados [?]. Desse modo, inserir alunos na resolução de problemas presentes no contexto da robótica móvel, por exemplo, fará com que os mesmos aprendam, com eficiência, diversos temas recorrentes no contexto mundial da robótica móvel [?], além de conteúdos presentes em diversas disciplinas, como matemática e física [?].

O construcionismo é baseado em uma filosofia construtivista [?], que afirma que o conhecimento não deve ser uma cópia da realidade, e sim uma construção, realizada pelo aluno a partir da sua interação com o contexto do problema [?].

Segundo [?], a utilização da robótica no meio da educação traz inúmeros benefícios que vão além dos objetivos diretos da melhoria da aprendizagem. De acordo com ele, sua utilização garante a introdução dos estudantes nos problemas recorrentes do contexto de Engenharia mundial. Esta introdução possui grande importância para a evolução do país, já que a demanda por profissionais qualificados na área de Engenharia, no mundo atual, é ampla [?]. De acordo com [?], em 2014 no Brasil, haviam cerca de 78 mil vagas

relacionadas à área de Tecnologia da Informação e apenas 33 mil foram preenchidas, o que comprova a falta de profissionais na área. [?] afirma, ainda, que o motivo para esta falta de profissionais é resultado advindo do baixo interesse por parte dos estudantes brasileiros por ciências exatas.

Grande parte deste baixo interesse, segundo [?], é devido à metodologia tradicional de ensino, onde os alunos buscam copiar o conhecimento do professor, sem atividades com abordagem construcionista de aprendizado, por exemplo. Uma forma de aumentar o interesse dos alunos, segundo [?], é a utilização da robótica como uma ferramenta de apoio ao aprendizado, seguindo uma abordagem construcionista.

Atualmente, de acordo com [?], no Brasil, o ensino de computação e robótica se restringe aos níveis superiores e técnicos, exceto alguns projetos realizados por universidades em escolas de nível fundamental e médio, como [?]. Esta introdução tem como objetivo introduzir noções lógicas e computacionais na Educação Básica, despertando o interesse dos alunos nas ciências exatas e estimulando-os a ingressar na carreira de Engenharia [?].

Diversas ferramentas já foram desenvolvidas com o objetivo de apoiar a inserção de alunos da Educação Básica no contexto da computação e robótica. Na seção 3.2, são apresentadas algumas utilizadas por [?], [?], [?]. Ferramentas como essas, são utilizadas em diversos contextos, desde a utilização como apoio ao aprendizado, como mostra [?], até a utilização em campeonatos e jogos, como mostra [?].

De acordo com [?], algumas abordagens utilizam ferramentas virtuais para facilitar o aprendizado. Por outro lado, segundo [?], a utilização de ferramentas reais, equipadas com sensores e atuadores, garante maior interesse e introdução do aluno no contexto da engenharia do mundo real. Desse modo, a ferramenta utilizada neste trabalho será o kit de robótica educacional Mindstorm, da Lego, que tem suas características especificadas na seção 2.3.1.

2.3.1 Lego Mindstorms NXT

Com o passar dos anos, os componentes de *hardware* vêm sendo evoluídos continuamente [?]. Esta evolução faz com que o preço de componentes simples tenda a diminuir, como é o caso de pequenos processadores poderosos e praticamente descartáveis [?]. Desse modo, esta evolução dos componentes de *hardware* vem permitindo a utilização dos mesmos em diversos contextos, como educação e entretenimento.

Com o crescimento de demandas relacionadas a estes componentes, nascem os kits de robótica que integram componentes simples, porém, poderosos para utilização no contexto educacional e de entretenimento [?]. Entre os kits existentes atualmente no mercado, destaca-se o kit de robótica Mindstorm, da Lego, o qual foi desenvolvido na década de 80 por pesquisadores do MIT. O kit utiliza peças de montagem no padrão Lego

para construir a estrutura do robô, diversos tipos de sensores e alguns atuadores, para garantir a mobilidade e interação com o ambiente [?].

Desde o lançamento do kit, na década de 80, o mesmo é utilizado por instituições de ensino como ferramenta de apoio ao aprendizado. O sucesso do kit o garantiu premiações devido a sua reusabilidade, modularidade e simplicidade, associado ao seu baixo custo [?].

O cérebro do kit pode ser observado na Figura 2.3.1, onde se encontra um microprocessador disposto de diversos acessos para sensores e atuadores. Em alguns modelos do kit, o cérebro possui uma tela LCD e alguns botões para interação do usuário. Além do cérebro, o kit possui alguns sensores, como o apresentado na Figura 3, e atuadores, como na Figura 4



Figura 2 – Microprocessador central



Figura 3 – Sensor RGB - Lego



Figura 4 – Atuador - Lego

Além da obtenção do kit completo, como mostra na Figura 5, é possível adquirir componentes separados, como sensores e atuadores específicos, o que garante a possibilidade de adquirir apenas os componentes necessários para o contexto de aplicação, minimizando os custos da sua utilização.



Figura 5 – Kit Mindsotm completo - Lego

2.4 Considerações parciais

A utilização da tecnologia como ferramenta de apoio ao aprendizado, como foi apresentado durante o capítulo, garante diversos benefícios ao aluno e até ao professor. Principalmente, quando o contexto trabalhado envolve problemas reais da comunidade de

robótica, como o problema de SLAM. A partir desta linha de pensamento, este capítulo buscou apresentar informações importantes para viabilizar a compreensão da pesquisa realizada, como a apresentação do problema de SLAM, os filtros probabilísticos mais utilizados e a abordagem educacional da Robótica.

A partir do conteúdo apresentado durante este capítulo, este trabalho irá buscar aplicar técnicas de auto-localização no contexto limitado da robótica educacional, aproximando os estudos em sala de aula dos contextos da robótica mundial.

3 Suporte Tecnológico

Nesta seção, serão apresentadas ferramentas e tecnologias utilizadas para auxiliar o desenvolvimento deste projeto, desde a organização e definição da metodologia de pesquisa, até o desenvolvimento dos projetos pilotos durante o trabalho. Esta seção está dividida em *Engenharia de Software* e *Robótica Educacional*.

3.1 Engenharia de software

Neste tópico, serão apresentadas ferramentas e tecnologias voltadas ao contexto da Engenharia de Software que são utilizadas durante este trabalho, como, por exemplo, ferramentas para gerência de configuração e versionamento dos artefatos gerados.

3.1.1 GIT

A ferramenta GIT¹ foi desenvolvida por Linus Torvalds, mesmo criador do Linux, e assim como ele, é *open-source*. Disponibiliza uma eficiente forma de versionamento e gerenciamento de projetos.

3.1.2 Github

O Github² é uma ferramenta utilizada para hospedagem remota de projetos GIT. A ferramenta contempla uma *Wiki* para documentação do projeto e sistemas de *Issues* e *Milestones*³ para gerenciamento de atividades.

3.1.3 Bonita BPMN

Ferramenta para modelagem de processos *BPMN*⁴, o Bonita⁵ 7 foi escolhido graças a sua facilidade de utilização e portabilidade para o sistema operacional Linux.

3.1.4 Linux Mint

O sistema operacional utilizado durante este trabalho é o Linux Mint⁶ e o Windows 7 Ultimate⁷.

¹ <https://git-scm.com/>

² <https://github.com>

³ <https://guides.github.com/features/issues/>

⁴ <http://www.bpmn.org/>

⁵ <http://www.bonitasoft.com/>

⁶ <https://www.linuxmint.com/>

⁷ <https://www.microsoft.com/pt-br/software-download/windows7>

3.1.5 LaTeX

O LaTeX⁸ 3.14 é um sistema para criação de documentos utilizando textos *tex*, foi inicialmente desenvolvido por Leslie Lamport, na década de 80. O LaTeX oferece diversos comandos avançados para organização de alto nível de documentos, incluindo facilitadores para citações, bibliografias, fórmulas matemáticas, figuras e tabelas.

3.1.6 Sublime Text 3

O Sublime Text 3⁹ é um editor de texto bastante utilizado por programadores, por possuir apoio para diversas linguagens de programação, incluindo textos em LaTeX.

3.2 Robótica educacional

3.2.1 RoboMind 6.0

É um ambiente de desenvolvimento proposto por Arvid Halma¹⁰, da Universidade de Amsterdam, que tem como objetivo facilitar o ensino de programação para estudantes do ensino básico. Utiliza um robô virtual para exercitar os conceitos de inteligência artificial e lógica de programação. A linguagem utilizada é chamada Robo/Roo, e permite implementação da movimentação do robô em um ambiente 2D (duas dimensões). As funções básicas da ferramenta, são *ver*, *andar*, *pegar*, *pintar*, como mostra a Figura 6.

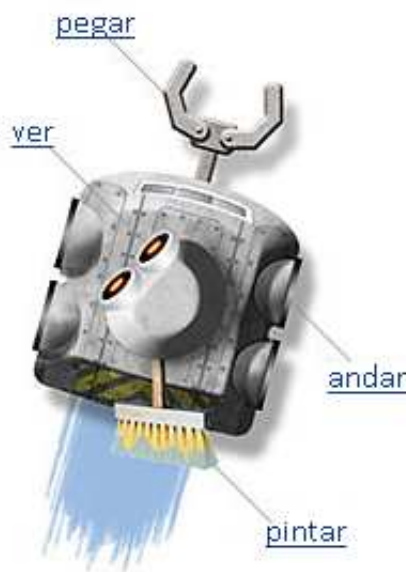


Figura 6 – Funções básicas RoboMind

⁸ <https://www.latex-project.org/>

⁹ <https://www.sublimetext.com/3>

¹⁰ <http://www.robomind.net/pt/index.html>

3.2.2 Scratch

É um ambiente de desenvolvimento criado por Lofelong Kindergarten Group(LLK)¹¹, que é um grupo de pesquisa do MIT. Também tem como objetivo introduzir conceitos de programação a alunos da educação básica. Com a utilização desta ferramenta, é possível desenvolver jogos, animações e histórias interativas, de modo visual.

A utilização da ferramenta pode ser visualizada na Figura 7, onde são apresentadas funções básicas no modo visual de desenvolvimento do Scratch.

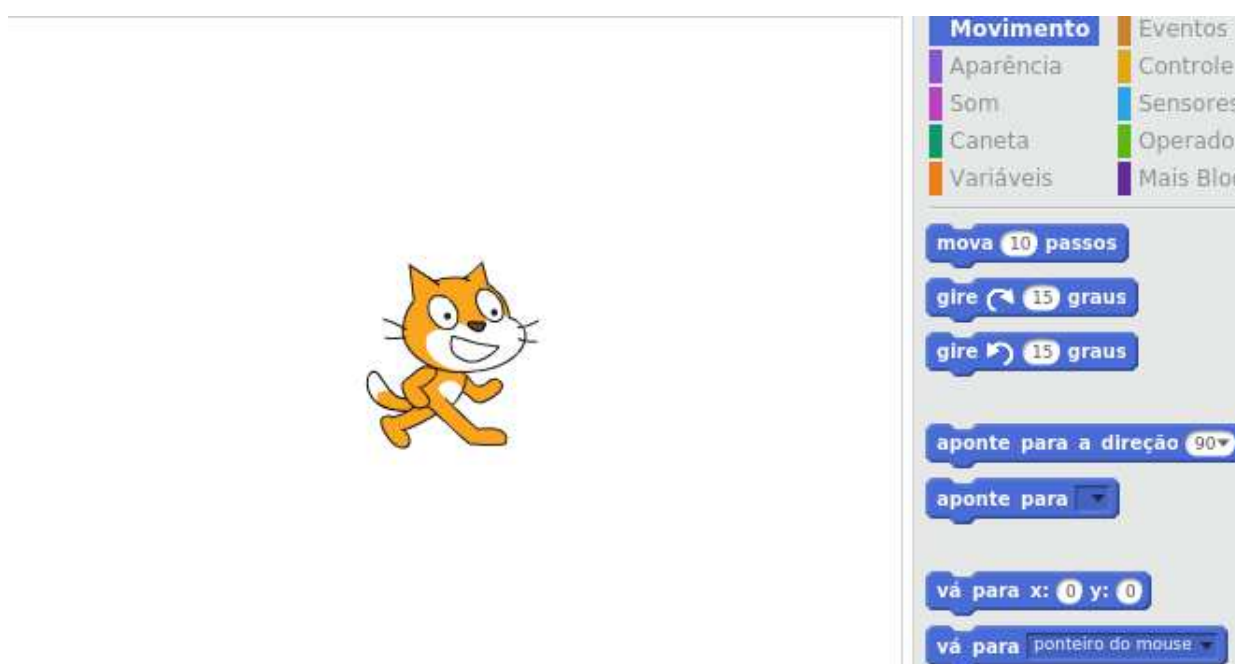


Figura 7 – Funções básicas Scratch

3.2.3 Linguagem Logo

A linguagem Logo¹² também foi desenvolvida em 1967, também pelo MIT. É uma linguagem interpretada, bastante utilizada para o desenvolvimento de inteligência artificial no contexto educacional. Sua utilização também é de forma visual, como apresenta a Figura 8. A caixa de texto visível na Figura 8 recebe os comandos para movimentação do robô virtual em um ambiente 2D (duas dimensões).

¹¹ <https://llk.media.mit.edu/projects/>

¹² <https://logo.codeplex.com/>

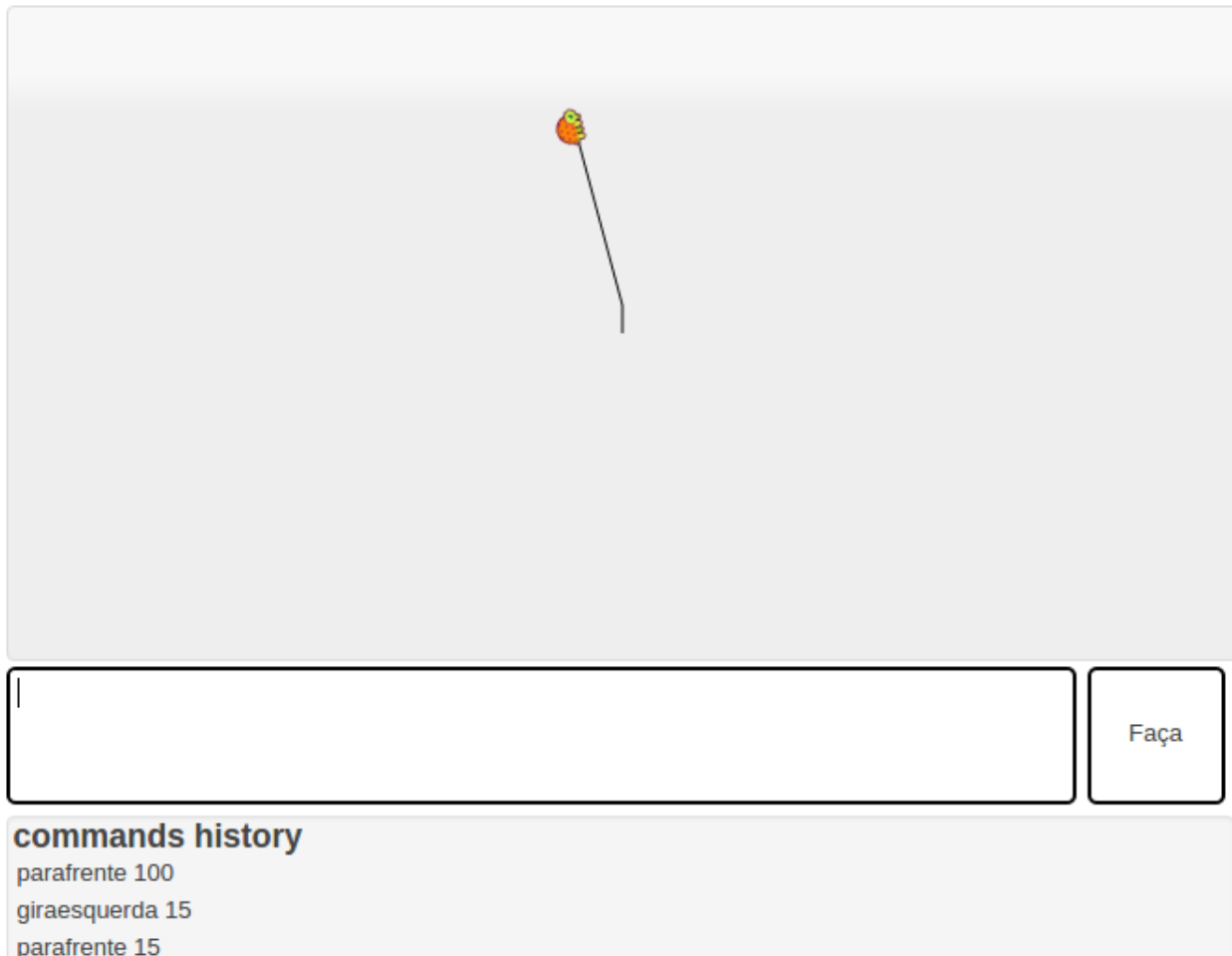


Figura 8 – Ambiente de desenvolvimento em Logo

3.2.4 Kodu Game Labs

Foi desenvolvido pelo laboratório de pesquisas FUSE (Future Social Experiences)¹³, que é mantido pela Microsoft. Esta ferramenta possui uma complexidade que não estava presente nas ferramentas apresentadas anteriormente. Seu objetivo é desenvolver jogos em ambiente 3D (três dimensões) no contexto educacional, facilitando o aprendizado de programação orientada a objetos.

Os jogos desenvolvidos se baseiam na criação de um mundo completo, com personagens, ações e objetos. Os personagens utilizados podem ser inseridos com funções já pré-definidas, ou o usuário pode programar o personagem da forma que desejar, garantindo a flexibilidade e quantidade de possibilidades da ferramenta.

¹³ <https://fuse.microsoft.com/>

3.2.5 Alice 2.0

Assim como o Kodu Game, a ferramenta Alice¹⁴ também disponibiliza um ambiente tridimensional para desenvolvimento de animações e jogos interativos. Foi desenvolvida por um grupo de pesquisa liderado por Randy Pausch¹⁵, da Universidade de Virgínia e Universidade de Carnegie Mellon. Diferentemente do Kodu Game, a ferramenta Alice possui apenas um *Mundo*, onde serão implementadas todas as histórias e relações da animação ou jogo. Após análise da ferramenta, [?] cita problemas de instabilidade e *bugs*.

3.2.6 Mindstorms Lego

Os kits de robótica Mindstorms da Lego, diferentemente de todas as ferramentas apresentadas acima, disponibiliza um ambiente real de desenvolvimento. Envolve a implementação de ações dos atuadores presentes em cada robô, onde estas ações serão executadas baseando-se nas informações advindas dos sensores. A especificação desta ferramenta se encontra na seção 2.3.1.

3.3 Considerações parciais

Este capítulo buscou apresentar as ferramentas e tecnologias utilizadas para apoiar o desenvolvimento desta pesquisa, desde a realização da revisão sistemática e documentação do projeto, até a análise de abordagens utilizadas na Robótica Educacional. Analisando as ferramentas utilizadas, observa-se que a base tecnológica do trabalho é resumida em componentes gratuitos e, se possível, *open source*.

A análise das ferramentas citadas se deu com o objetivo de analisar maneiras de se trabalhar o contexto educacional, utilizando ferramentas de apoio e abordagens de ensino. Dentre as ferramentas apresentadas, se encontram ferramentas que apoiam o ensino de diversas áreas de conhecimento, como programação e matemática. Sendo elas ferramentas virtuais ou reais, como é o caso do kit Mindstorm, foco deste trabalho.

¹⁴ <http://www.aliceprogramming.net/>

¹⁵ <http://www.cs.cmu.edu/~pausch/news/>

4 Proposta

Este capítulo apresenta detalhes sobre a pesquisa realizada durante o desenvolvimento do TCC_1 e do TCC_2. O capítulo foi organizado nas seções de *Obtenção da base teórica*, a partir da utilização de revisão sistemática, e *Adaptação e implementação*, onde serão especificados os objetivos práticos da pesquisa.

4.1 Obtenção da base teórica

Como qualquer trabalho científico, para alcançar os objetivos da pesquisa, faz-se necessária a realização de uma pesquisa bibliográfica buscando obter uma base teórica suficiente para sustentar o trabalho realizado. Durante este trabalho de conclusão de curso, como uma forma de obtenção da base teórica, foi utilizada a técnica de revisão sistemática.

O grande foco da revisão sistemática tem sido as técnicas de resolução do problema de SLAM utilizadas atualmente, em diferentes contextos. Vale ressaltar que o foco será na utilização das mesmas em um contexto simplificado e educacional. A identificação destas técnicas possibilita a seleção e adaptação das mesmas para viabilizar sua implementação no contexto simplificado.

4.2 Adaptação e implementação

Com a identificação e análise das diferentes técnicas de resolução do problema de SLAM, fruto da revisão sistemática, serão realizadas provas de conceito ao início da segunda etapa do trabalho. Estas provas de conceito têm como objetivo selecionar, primeiramente, qual filtro probabilístico será utilizado neste trabalho.

De acordo com o resultado obtido com a revisão sistemática, a grande maioria das soluções do problema de SLAM identificadas utilizam o filtro de Kalman, como mostra a tabela 8. Por outro lado, a utilização do filtro de partículas possibilita a implementação, com maior facilidade, do problema do sequestro do robô, como foi apresentado na seção 2.2.

Desse modo, a escolha do filtro a ser utilizado ainda depende de estudos mais específicos sobre a implementação dos mesmos, levando em consideração sua complexidade e suas vantagens em relação ao concorrente.

Ao final deste trabalho, espera-se obter uma solução onde o robô seja inserido em um ambiente desconhecido, em um ponto desconhecido e, a partir das informações

obtidas pelos sensores, o mesmo construa um mapa do local, utilizando este mesmo mapa, simultaneamente para se auto-localizar no ambiente. Para isso, estarão disponíveis os sensores de distância (sonar), odométricos e, possivelmente, uma bússola.

Como as técnicas identificadas durante a revisão sistemática utilizam, em sua maioria, sensores como *câmera de vídeo* e sensores a *laser*, adaptações serão necessárias para viabilizar a resolução do problema com a utilização dos sensores citados anteriormente (sonar, odometria e bússola).

A resolução do problema será realizada de maneira simplificada, possibilitando sua conclusão durante o período previsto para o desenvolvimento da segunda etapa deste trabalho de conclusão de curso. Desse modo, questões referentes ao desempenho e máxima precisão não fazem parte do foco deste trabalho.

A arquitetura da solução que será desenvolvida durante este trabalho segue o apresentado ao percorrer da revisão sistemática, como apresenta a figura 16. Nesta arquitetura, o robô é responsável apenas por obter as informações do ambiente, a partir de seus sensores, e *obedecer* as ordens do computador. Já o computador, será responsável por processar as informações obtidas pelo robô e, a partir deste processamento, enviar decisões referentes à movimentação e localização ao robô. Esta arquitetura foi destacada, durante a revisão sistemática, como uma arquitetura de processamento remoto, onde nenhum processamento é feito localmente (no próprio robô).

Durante as seções a seguir, serão apresentadas informações referentes à solução e desenvolvimento durante a segunda etapa deste trabalho.

4.2.1 Arquitetura do Robô

De acordo com [?], a arquitetura de robôs móveis pode ser sub-dividida cinco camadas: *percepção*, *decisão*, *planejamento de caminho*, *geração de trajetória* e *sistema de controle*.

A primeira camada, denominada *camada de percepção*, será o grande foco deste trabalho. Onde a identificação de obstáculos para mapeamento do ambiente é uma atividade essencial para a resolução do problema de SLAM. Esta camada é responsável por adquirir informações sobre o ambiente ao seu redor, viabilizando a navegação e auto-localização no mesmo.

Já a segunda camada, *decisão*, que foi o foco de trabalho de [?], tem como responsabilidade processar decisões do robô. Nesta camada se encontra o verdadeiro cérebro do robô, como afirma [?]. Este trabalho também atuará nesta camada, para realização do processamento das informações e geração das decisões referentes a auto-localização do robô.

Na terceira camada, *planejamento de caminho*, será utilizado, como ferramenta de apoio, o *framework* Traveller, desenvolvido por [?]. Sua utilização se refere ao planejamento da navegação no ambiente após o mapeamento, mesmo que parcial, do ambiente. Ou seja, enquanto o robô navega e mapeia o ambiente, os locais já percorridos (conhecidos e mapeados) possibilitarão a utilização do *framework*. Por outro lado, em locais ainda desconhecidos, o planejamento do caminho será feito de forma aleatória, buscando mapear o ambiente como um todo.

A quarta camada, *geração da trajetória*, utiliza o planejamento realizado na camada anterior para definir quais ações devem ser realizadas no *hardware* do robô, levando em consideração as características físicas do mesmo. Com isso, a quinta camada, *sistema de controle*, é chamada para verificar o recebimento adequado dos sinais, assim como sua execução nos atuadores.

A figura 9 apresenta as cinco camadas, destacando as camadas que são o principal foco deste trabalho.



Figura 9 – Arquitetura do robô. Fonte [?].

4.2.2 Montagem do robô

A montagem utilizada durante o início deste trabalho, mais especificamente durante a realização da prova de conceito, é baseada em esteiras. O robô é equipado com um sonar, para identificar a distância de obstáculos, e um sensor de toque. Para movimentação, foram utilizados dois motores, A e B, dispostos na direita e esquerda, respectivamente. A Figura 10 apresenta a montagem utilizada.

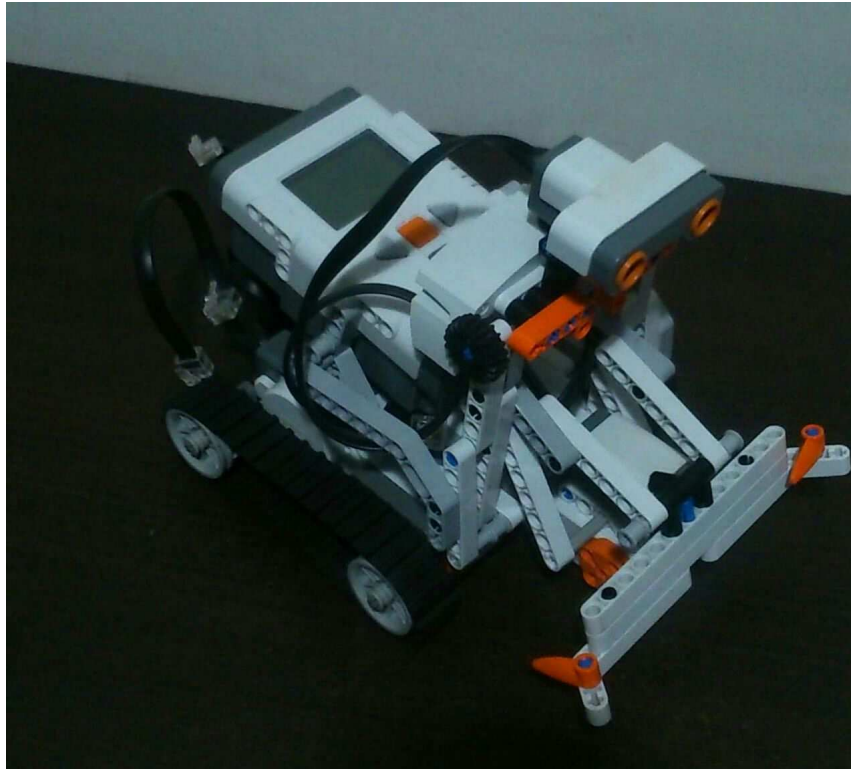


Figura 10 – Montagem do robô.

4.2.3 Ambiente utilizado

O ambiente utilizado para navegação do robô é o tapete de missões *Nature's Fury*, utilizado como um dos desafios do torneio *First Lego League*¹. Como já existem alguns locais pré-definidos para possuírem obstáculos, o tapete será utilizado, ao longo deste trabalho para teste, avaliando a qualidade da geração do mapa por parte do robô. O mapa pode ser visualizado na Figura 11.

¹ <http://www.firstlegoleague.org/>



Figura 11 – Ambiente utilizado.

Serão adicionados obstáculos neste tapete, os quais deverão ser mapeados pelo robô possibilitando sua auto-localização no ambiente, mesmo que de maneira simplista. A validação será a partir da comparação entre o verdadeiro mapa e o mapa gerado, levando em consideração a margem de erro comum nesse tipo de navegação.

4.2.4 Condução do trabalho

O desenvolvimento deste trabalho terá como base, durante o TCC_2, a metodologia ágil, utilizando o conceito de *releases*, *sprints* e *histórias* para apoiar o mapeamento e gerenciamento dos requisitos, assim como o acompanhamento da evolução do trabalho ao longo do tempo. Para representar as *releases*, *sprints* e *histórias*, serão utilizadas, respectivamente, as ferramentas de criação de *tags*, *milestones* e *issues* presentes no GitHub².

Com o objetivo de acompanhar o desenvolvimento, a ferramenta *waffle.io*³, em integração com o GitHub, será utilizada. Possibilitando maior visibilidade e organização do *status* do trabalho.

Ao início de cada *sprint*, serão analisadas as histórias que farão parte da mesma, levando em consideração o *velocity* do pesquisador e o período fixo para cada *sprint* de duas semanas. Ao final de cada *sprint*, o *velocity* deverá ser atualizado para melhor acompanhamento do projeto.

De acordo com a utilização do conceito de *velocity*, as histórias deverão ser pontuadas a partir de sua complexidade de implementação, em relação ao conhecimento do desenvolvedor. A pontuação será realizada ao início de cada *sprint*.

² <https://github.com>

³ <https://waffle.io>

A gerência de configuração e versionamento de todo o trabalho se baseia na utilização da ferramenta GitHub, baseada em projetos GIT. Desse modo, para organização dos artefatos do trabalho, foi criada uma Organização⁴ no GitHub, a qual possui dois repositórios, um referente ao versionamento de toda a documentação do projeto⁵ (*slam_lego_doc*), e um referente ao versionamento do código⁶ (*slam_lego*). A Figura 12 apresenta a organização utilizada para versionamento deste trabalho.

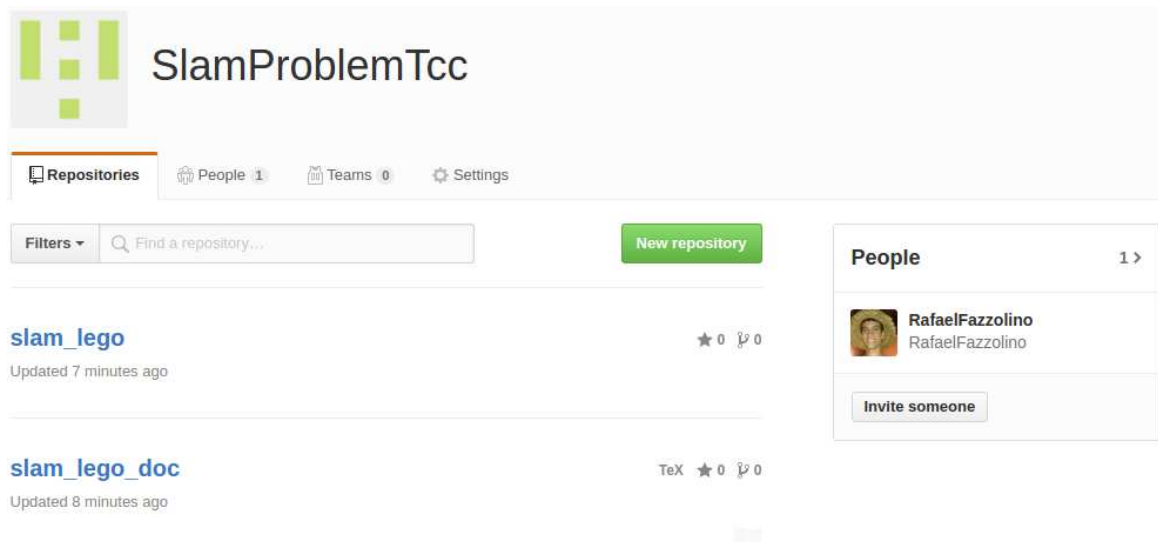


Figura 12 – Página utilizada para versionamento do trabalho

4.3 Considerações parciais

A proposta deste trabalho é baseada na identificação e adaptação de técnicas de resolução do problema de SLAM. Para isso foi realizada uma revisão sistemática para encontrar e registrar diferentes técnicas de SLAM utilizadas atualmente. Com a identificação destas técnicas, durante o TCC_2, o foco do trabalho será adaptar algumas destas técnicas para o contexto simplificado da robótica educacional.

Este contexto simplificado é baseado no kit de robótica Mindstorm, que, como já foi explicado, possui características bastante limitadas, como diversidade de sensores e capacidade de memória e processamento. A adaptação será implementada utilizando a linguagem Java, a partir da ferramenta *leJOS NXJ*⁷.

Esta solução busca complementar uma sequência de soluções relacionadas à robótica educacional que vêm sendo desenvolvidas na Universidade de Brasília, como [?] e [?]. O *framework* Traveller, desenvolvido por [?], será utilizado para navegação em ambientes já mapeados pelo algoritmo desenvolvido durante este trabalho. Desse modo, este

⁴ <https://github.com/SlamProblemTcc>

⁵ https://github.com/SlamProblemTcc/slam_lego_doc

⁶ https://github.com/SlamProblemTcc/slam_lego

⁷ <http://www.lejos.org/nxj.php>

trabalho tem como objetivo incorporar um módulo novo do *framework*, possibilitando a construção de mapas e auto-localização nos mesmos.

5 Metodologia

De acordo com [?], a metodologia de pesquisa adotada neste trabalho pode ser classificada nas seguintes categorias: classificação quanto aos objetivos da pesquisa, quanto à natureza da pesquisa, quanto à escolha do objeto de estudo, quanto à técnica de coleta de dados, e quanto à técnica de análise de dados. A classificação escolhida em cada categoria busca garantir conhecimento amplo sobre diferentes técnicas de solução do problema de SLAM, possibilitando a adaptação de técnicas para um contexto de robôs simples. Nas seções seguintes, são apresentadas as categorias e suas respectivas classificações de pesquisa adotadas para este trabalho.

5.1 Classificação da pesquisa

De acordo com [?], uma pesquisa científica pode ser classificada nas categorias *abordagem da pesquisa*, *natureza da pesquisa*, *objetivos da pesquisa*, e *procedimentos*, que são apresentadas nas seções 5.1.1, 5.1.2, 5.1.3 e 5.1.4, respectivamente.

5.1.1 Abordagem da pesquisa

A classificação deste trabalho em relação à abordagem da pesquisa é definida como *qualitativa*. Segundo [?], abordagens de cunho qualitativo buscam o significado dos dados obtidos, capturando, além da aparência do fenômeno estudado, suas essências. Dessa forma, a pesquisa qualitativa envolve a obtenção e a análise dos dados descritivos por meio do contato direto entre o pesquisador e o fenômeno estudado [?].

A abordagem qualitativa não busca definir resultados de maneira quantificável, como mostra [?], buscando compreender de forma subjetiva o evento estudado. Nesse trabalho, os dados serão coletados em ambiente controlado e analisados de forma não quantificável, buscando caracterizar os resultados como, por exemplo, satisfatórios ou não, caracterizando, assim, segundo [?], uma pesquisa qualitativa.

Sua utilização, durante este trabalho, tem como objetivo, entender detalhadamente o funcionamento de diversas técnicas de resolução do problema de SLAM, viabilizando a adaptação das mesmas para um contexto limitado. Além disso, a proximidade do pesquisador com o fenômeno a ser estudado possibilita uma análise descritiva dos dados, de forma que viabilize uma adaptação de técnicas para contextos limitados.

5.1.2 Natureza da pesquisa

A classificação deste trabalho em relação à natureza da pesquisa é definida como *pesquisa aplicada*, devido ao fato de buscar solucionar um problema recorrente no contexto mundial da robótica. Segundo [?], "*a pesquisa aplicada busca gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos*".

5.1.3 Objetivos da pesquisa

A classificação deste trabalho em relação aos objetivos da pesquisa é definida como *exploratória*. De acordo com [?], a pesquisa exploratória tem como objetivo ampliar os conhecimentos do pesquisador sobre o tema. A pesquisa exploratória é, normalmente, o primeiro passo quando o pesquisador não conhece suficientemente a área que pretende abordar [?].

O uso da pesquisa exploratória, neste trabalho, se dá pela necessidade do conhecimento sobre as diversas técnicas de resolução do problema de SLAM, com o objetivo de possibilitar adaptações das mesmas para o contexto de robôs simples.

5.1.4 Procedimentos

Quanto aos procedimentos da pesquisa, a mesma foi dividida em duas etapas. Durante a primeira etapa, o procedimento seguido foi *pesquisa bibliográfica*. Nesse caso, segundo [?], esse procedimento é feito a partir do levantamento de referências teóricas publicadas por meio de livros, artigos, páginas da web, entre outros.

Já na segunda etapa do trabalho, o procedimento a ser seguido será o de *pesquisa-ação*. Como [?] afirma, "*a pesquisa-ação pressupõe uma participação planejada do pesquisador na situação problemática a ser investigada*". Além da participação do pesquisador, o objeto de estudo da pesquisa-ação não pode ser representado por um conjunto de variáveis que poderiam ser analisadas separadamente [?].

A utilização da *pesquisa-ação* durante este trabalho é devido ao conhecimento que será obtido pelo pesquisador durante a participação ativa na situação estudada, possibilitando análises amplas de todo o contexto.

De acordo com a classificação da pesquisa nas diferentes categorias apresentadas acima, a Tabela 1 apresenta, de maneira resumida, a abordagem definida para realização deste trabalho.

Tabela 1 – Metodologia de pesquisa.

Abordagem da pesquisa	Natureza da pesquisa	Objetivos da pesquisa	Procedimentos
Qualitativa	Aplicada	Exploratória	Pesquisa bibliográfica/ Pesquisa-ação

De acordo com [?], a partir da classificação da pesquisa, fica fácil estabelecer o *caminho* a ser seguido para realização da mesma. Portanto, definiu-se o processo metodológico, o qual é apresentado na seção 5.2.

5.2 Processo metodológico

O TCC, segundo as regras da UnB, é realizado em duas fases: uma chamada de TCC_01 e outra chamada de TCC_02. Durante o TCC_01, o foco desse trabalho foi estabelecer os pilares teóricos para embasamento do projeto como um todo, bem como desenvolver uma prova de conceito visando estudar a viabilidade da proposta. Já durante o TCC_02, o trabalho estará focado em atingir o objetivo geral *"Adaptar técnicas de resolução do problema de SLAM para o contexto de robôs simples, utilizando os kits de robótica Mindstorms da Lego, em um primeiro momento."*, com base nas fundamentações teóricas e nos resultados acordados na prova de conceito, ambos - fundamentações e resultados - conquistados ao longo do TCC_01.

O TCC_01 pôde ser sub-dividido em oito atividades: *selecionar tema, realizar pesquisa bibliográfica, definir proposta, escrever referencial teórico, estabelecer suporte tecnológico, evoluir metodologia, realizar prova de conceito e apresentar TCC 1*. As mesmas estão distribuídas de acordo com o cronograma disposto na Tabela 2.

Tabela 2 – Cronograma de atividades TCC_1.

Cronograma	Março	Abril	Maio	Junho
Selecionar Tema	X			
Realizar pesquisa bibliográfica	X	X	X	X
Definir proposta	X			
Escrever referencial teórico		X	X	
Estabelecer suporte tecnológico		X	X	X
Evoluir metodologia		X	X	
Realizar prova de conceito				X
Apresentar TCC 1				X

Já a segunda etapa do trabalho, durante a realização do TCC_2, o cronograma de atividades segue o apresentado na Tabela 3.

Tabela 3 – Cronograma de atividades TCC_2.

Cronograma	Agosto	Setembro	Outubro	Novembro	Dezembro
Desenvolver provas de conceito	X				
Selecionar tecnologias de apoio	X	X			
Desenvolver solução		X	X	X	
Testar solução		X	X	X	
Relacionar com o âmbito educacional		X	X	X	X
Analisar resultados				X	X
Apresentar TCC_2					X

Com o objetivo de definir e acompanhar o processo de desenvolvimento da primeira etapa do trabalho de conclusão de curso, foi modelado um processo metodológico, o qual se encontra na Figura 5.2.

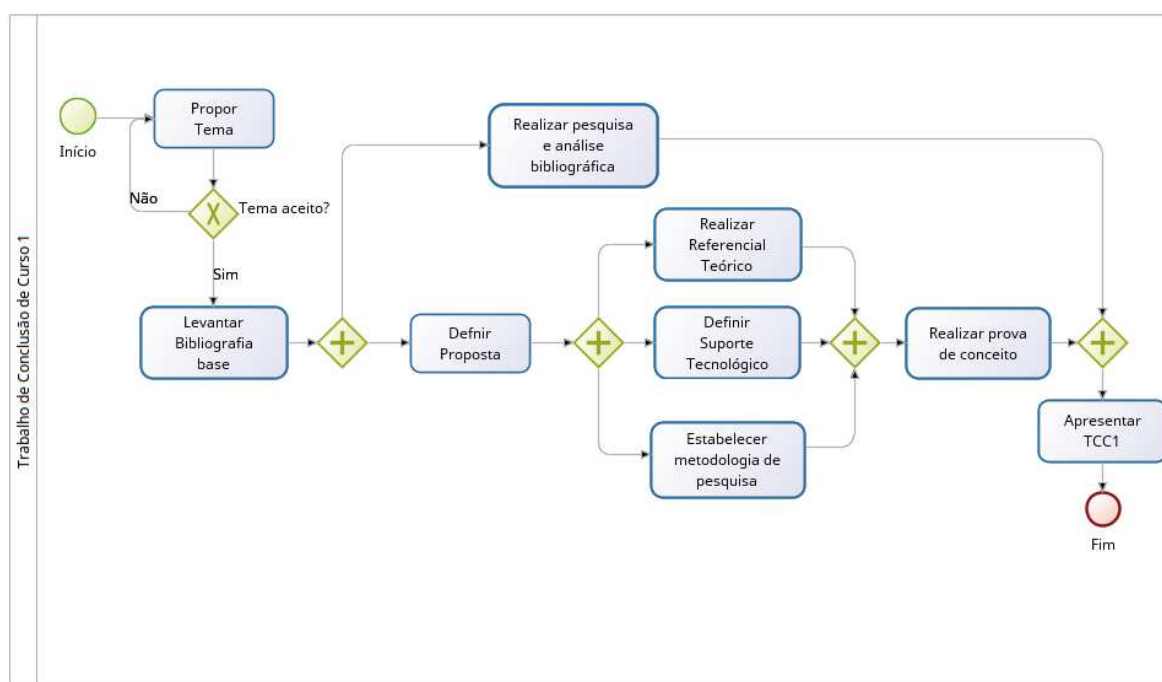


Figura 13 – Processo metodológico

• Propor tema:

A atividade de propor tema engloba desde a escolha do contexto em que se deseja trabalhar, até a definição dos orientadores do trabalho. Após a escolha do contexto e

dos orientadores, buscou-se definir um escopo que será abordado durante o trabalho, ou seja, o tema. Os orientadores devem validar o tema escolhido para concluir a atividade.

- **Levantar bibliografia base:**

Esta atividade refere-se à definição de pilares para o estudo proposto, ou seja, estabelecer o marco teórico do trabalho. Este levantamento garante o entendimento do contexto trabalhado e as possibilidades de atuação, especificando mais adequadamente o escopo.

- **Definir proposta:**

Documentar a proposta de pesquisa para este trabalho. A proposta inclui, não apenas, mas, principalmente, uma introdução com a contextualização do tema, o objetivo geral e específicos, a justificativa e uma metodologia de pesquisa.

- **Realizar pesquisa e análise bibliográfica:**

A pesquisa bibliográfica foi feita a partir da utilização da técnica de *revisão sistemática*, com o objetivo de ampliar os conhecimentos em relação ao tema, conhecendo pesquisas em diferentes contextos e de diversas bases científicas, como *IEEE*, *Springer* e *CAPES*. O detalhamento da revisão sistemática encontra-se no capítulo 6.

- **Realizar referencial teórico:**

Trata-se da escrita do capítulo dois deste trabalho. O mesmo descreve o referencial teórico do trabalho em andamento. Como insumos para esta atividade, encontram-se todas as pesquisas bibliográficas obtidas durante a atividade de *Realizar pesquisa e análise bibliográfica*. No segundo capítulo deste trabalho, o tema é especificado com mais detalhe.

- **Definir suporte tecnológico:**

Nesta atividade, são definidas as principais ferramentas e tecnologias utilizadas para a execução deste trabalho.

- **Estabelecer metodologia de pesquisa:**

Durante esta atividade, a metodologia de pesquisa inicial, apresentada durante a *proposta*, foi evoluída, com o objetivo de adequar as formas de atuação ao longo da realização do trabalho proposto.

- **Realizar prova de conceito:**

Durante esta atividade, foi realizada a implementação de uma prova de conceito que buscou avaliar a viabilidade da realização deste trabalho. Durante a prova de conceito, que se encontra detalhada na seção 6.2, ferramentas e maneiras de mapear ambientes foram estudadas.

- **Apresentar TCC 1:**

Apresentar os resultados obtidos até o momento para a banca examinadora.

As atividades de *Levantar bibliografia base* e *Realizar pesquisa e análise bibliográfica* foram complementadas pelo emprego da técnica de revisão sistemática, que segundo [?], tem como objetivo identificar, avaliar e interpretar o máximo de pesquisas disponíveis relacionadas a um tema em específico. O planejamento e condução da revisão sistemática é detalhado no capítulo 6.

5.2.1 Revisão sistemática

Durante décadas, as pesquisas se encontravam carentes de métodos científicos que detalhassem o processo de revisão de literatura, como mostra [?]. Esta carência acaba por impossibilitar a realização futura da mesma busca, já que sem detalhamento da pesquisa, o interessado não poderá aplicar a busca, realizando comparações, por exemplo. Desse modo, entre as décadas de 1970 e 1980, psicólogos e cientistas sociais buscaram definir métodos de sistematizar revisões de literatura, as chamadas *revisões sistemáticas*.

Com o objetivo de realizar uma ampla pesquisa bibliográfica, buscando conhecer diferentes técnicas de auto-localização e, mais especificamente, o problema de SLAM, utilizou-se da técnica de revisão sistemática. Uma revisão sistemática busca identificar e analisar o máximo de pesquisas relacionadas a um tema em específico, como define [?, p. 8]:

"Uma revisão literária sistemática é um meio de identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes a uma determinada questão de pesquisa, ou área de um tópico, ou fenômeno de interesse. Estudos individuais que contribuem para uma revisão sistemática são chamados estudos primários; uma revisão sistemática é uma forma de estudo secundário."

Como afirma [?], a técnica de revisão sistemática é bastante utilizada quando se busca identificar soluções propostas para resolver o problema levantado. Neste trabalho, a revisão sistemática busca identificar técnicas de resolução do problema de SLAM em diferentes contextos, inclusive educacional.

5.2.1.1 Processo de revisão sistemática

O processo utilizado durante a pesquisa segue o processo de revisão sistemática apresentado por [?]. Desse modo, o processo é dividido em três etapas: *planejamento da revisão*, *condução da revisão* e a *documentação da revisão*. As atividades que estão distribuídas entre essas etapas podem ser observadas na Figura 14.

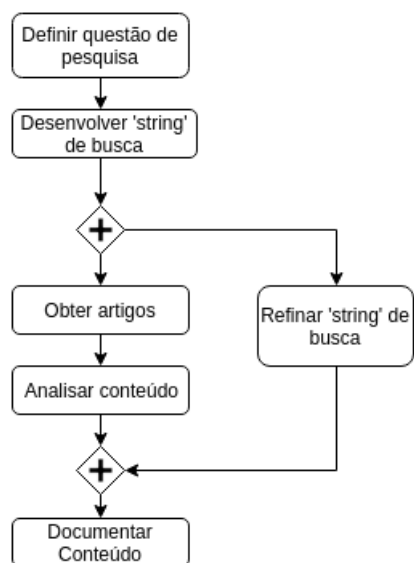


Figura 14 – Processo de revisão sistemática. Processo adaptado de [?]

A descrição de cada atividade do processo da Figura 14 encontra-se na Tabela 4.

Tabela 4 – Processo de revisão sistemática

Atividade	Descrição
Definir questão de pesquisa	Atividade relacionada à definição do objetivo final da pesquisa, o que o pesquisador deseja alcançar com a mesma. A definição da questão de pesquisa possibilita o estabelecimento de critérios de busca durante os ciclos de revisão.
Desenvolver <i>string</i> de busca	Atividade de desenvolvimento. Nessa atividade, o objetivo é definir uma <i>string</i> de busca que represente todo o trabalho pesquisado. A <i>string</i> será refinada ao longo de toda pesquisa, buscando maximizar a efetividade das buscas.
Obter artigos	A partir da definição de um critério de busca, os artigos que se enquadram neste critério são selecionados para análise. A obtenção deve ser realizada em diferentes bases científicas, como IEEE, Springer e CAPES.
Analisar conteúdo	A análise do conteúdo busca, além de estudar o tema pesquisado, conhecer novas pesquisas e, com isso, refinar a <i>string</i> de busca.
Refinar a <i>string</i> de busca	Esta atividade é considerada uma das atividades mais importantes do processo de revisão sistemática. Busca maximizar a efetividade do processo de busca, identificando artigos importantes para a pesquisa.
Documentar conteúdo	A documentação do conteúdo, basicamente, é definida como o registro das informações necessárias à pesquisa, obtidas com os ciclos de pesquisa. Os resultados da revisão sistemática são frutos desta atividade.

Todo o processo de revisão sistemática é visto como um resultado parcial do trabalho em andamento. Dess modo, o detalhamento do desenvolvimento da revisão sistemática, como os ciclos de busca e o processo de refinamento da *string*, consta no capítulo 6.

6 Resultados Parciais

Dado o cronograma de atividades planejado para este trabalho, algumas atividades já foram realizadas. Neste capítulo apresentam-se o emprego da *Revisão sistemática* e os ganhos referentes ao *Desenvolvimento prático*, com a realização da prova de conceito.

6.1 Revisão sistemática

A revisão bibliográfica deste trabalho foi complementada pelo emprego da técnica de revisão sistemática. Nesta seção apresentam-se as três fases básicas da revisão sistemática definidas por [?], 6.1.1 *planejamento da revisão*, 6.1.2 *condução da revisão* e 6.1.3 *reporte dos resultados da revisão*, como mostra a Figura 15.

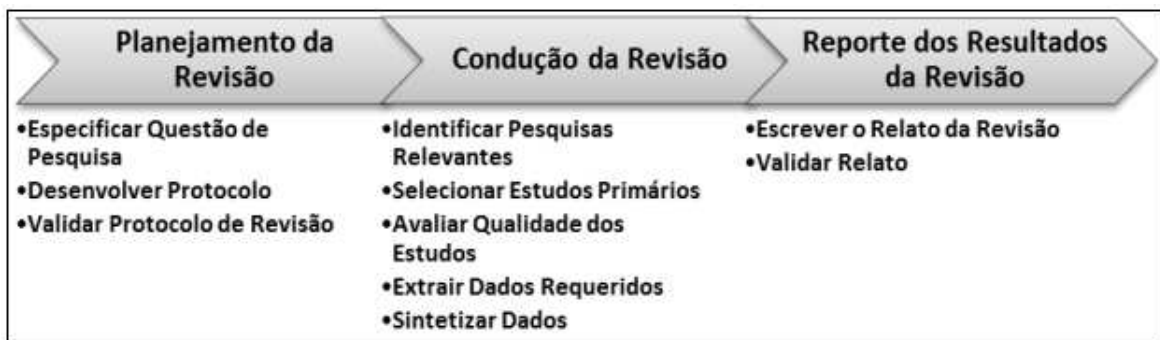


Figura 15 – Processo geral de revisão sistemática, segundo [?]

6.1.1 Planejamento da revisão

A revisão sistemática foi realizada entre os meses de março e junho de 2016, utilizando como fonte de busca as bases *IEEE*, *CAPE*s e *Springer*. A partir dos modelos de revisão sistemática apresentados por [?], foi desenvolvido um protocolo de revisão, o qual possibilita, a outros pesquisadores, a repetição da pesquisa.

6.1.1.1 Objetivos e questão de pesquisa

O objetivo inicial do trabalho é estudar a problemática da auto-localização na robótica, identificando diferentes soluções em diversos contextos. Afirmando isso, foi possível realizar uma pesquisa bibliográfica com o objetivo de identificar diferentes linhas de pesquisa nesta área. Após uma análise superficial de cada linha de pesquisa, foi selecionada a linha de pesquisa que adota, como solução para a auto-localização, a utilização da técnica de SLAM, o que pode ser considerado como o marco teórico do trabalho.

Com a identificação do marco teórico do trabalho, a definição do foco da pesquisa torna-se uma tarefa menos árdua. Como o marco teórico deste trabalho baseia-se nas linhas de pesquisa que buscam utilizar a técnica de SLAM para realizar navegação autônoma, esta revisão sistemática teve como objetivo identificar diferentes técnicas utilizadas atualmente para solucionar o problema de SLAM em diferentes contextos, desde contextos simplificados até contextos altamente complexos.

A definição deste objetivo da revisão dá-se pela necessidade de conhecimento amplo em relação a diferentes técnicas para solucionar o problema de SLAM. Como este trabalho buscará adaptar técnicas para um contexto simplificado, ou educacional, adicionou-se aos objetivos da revisão itens relacionados à Robótica Educacional e robôs simples.

Segundo [?], o primeiro passo para se realizar uma revisão sistemática é definir a sua questão de pesquisa. Desse modo, a partir da realização de uma pesquisa bibliográfica inicial, foram identificadas as seguintes questões de pesquisa:

- **Q1.** *Quais técnicas são mais utilizadas para solucionar o problema de SLAM? e*
- **Q2.** *"Como tratar o problema de SLAM no contexto simplificado da robótica educacional?"*

Além das questões de pesquisa, de acordo com [?], alguns outros itens devem ser destacados, como:

- **População:** comunidade acadêmica e de robótica.
- **Intervenção:** adaptação de técnicas para um contexto de robótica simplificado (educacional).
- **Controle:** utilização do *Quasi-gold standard* [?], que sera explicado mais à frente.
- **Resultados:** obtenção de técnicas adaptáveis ao contexto simplificado.
- **Aplicação:** servir de base para a implementação da segunda etapa deste trabalho de conclusão de curso, onde técnicas serão adaptadas buscando solucionar, de maneira simplificada, o problema de SLAM.

A partir da definição das questões de pesquisa e dos objetivos da revisão, buscou-se definir a estratégia de pesquisa, apresentada no tópico 6.1.1.2.

6.1.1.2 Estratégia de pesquisa

A estratégia de pesquisa adotada para esta revisão segue recomendações de diversos autores, como [?] e [?], utilizando o conceito de *quasi-gold standard*.

Segundo [?], *gold standard* representa o conjunto completo de estudos primários referentes a uma questão de pesquisa, com máxima precisão e sensibilidade. Já o *quasi-gold standard*, representa um subconjunto do *gold standard*, o qual vai sendo evoluído ao longo dos ciclos de busca, com o objetivo de se aproximar do *gold standard*.

É utilizado para definir os valores de precisão e sensibilidade da busca, o que possibilita a avaliação da busca realizada, verificando a necessidade de refinamento da *string*, por exemplo. [?] define precisão e sensibilidade da busca da seguinte forma:

- $precisão = \frac{ERO}{EO}$
- $sensibilidade = \frac{ERO}{TER}$

onde ERO = número de estudos relevantes obtidos,

EO = número de estudos obtidos e

TER = número total de estudos relevantes.

Esta definição possibilita a criação de critérios que avaliem a qualidade da busca, ou seja, da *string* de busca utilizada. Porém, a seleção dos materiais deve seguir critérios relacionados à qualidade do material, os quais são divididos em *critério de inclusão (CI)* e *critério de exclusão (CE)*, como se pode observar a seguir:

- CI 1 - Os artigos devem estar escritos em inglês ou português;
- CI 2 - Artigos referentes à auto-localização e ao mapeamento de ambientes simultâneos (SLAM);
- CI 3 - Artigos com acesso gratuito, disponíveis na *web* para *download* ou leitura;
- CE 1 - Artigos que buscam solucionar o problema de auto-localização sem a utilização da técnica de SLAM, e
- CE 2 - Artigos que não possuem o problema de SLAM como foco principal.

A partir da definição da questão de pesquisa e dos objetivos da revisão, assim como a definição dos critérios de inclusão e exclusão, foi possível desenvolver uma *string* de busca inicial. O idioma escolhido para a *string* de busca foi o inglês, devido a sua ampla utilização nas bases de conhecimento selecionadas. Buscou-se utilizar a mesma *string* de busca em todas as bases de dados pesquisadas, exceto em alguns casos em que houve a necessidade da adaptação da *string* de acordo com os padrões adotados pela base.

Com o objetivo de identificar pesquisas relacionadas à auto-localização utilizando mapeamento de ambientes simultaneamente, a *string* de busca definida foi: *auto-localization AND environment mapping*. Na Tabela 5 são apresentados os resultados obtidos a partir desta busca.

Tabela 5 – Resultados obtidos com a *string* inicial

String	IEEE, Springer e CAPES	
	Nº. Artigos	Nº. Artigos relevantes
<i>Auto-localization AND environment mapping</i>	29	6

Esta primeira busca foi de extrema importância para se obter uma visão inicial da pesquisa, identificando novas palavras-chave e iniciando os ciclos de busca.

6.1.1.3 Procedimento de seleção

As buscas foram realizadas com a mesma *string* de busca (na maioria dos casos, como já foi explicado anteriormente) nas três bases de conhecimento científico utilizadas. A cada busca realizada, os artigos foram registrados, para que, posteriormente, os mesmos pudessem ser submetidos à avaliação da qualidade e, se comprovada a relevância do mesmo, à extração de dados.

A seleção dos artigos deu-se a partir da leitura dos títulos, resumos e palavras-chave, classificando o artigo como relevante ou não, em um primeiro momento. Caso fosse confirmada a relevância do mesmo, o artigo passaria por uma avaliação mais profunda, a *avaliação da qualidade*, como mostra o tópico 6.1.1.4.

6.1.1.4 Avaliação da qualidade

A avaliação da qualidade do artigo deu-se a partir da análise do conteúdo do mesmo, focando principalmente na introdução, nos resultados e conclusões dos artigos. A avaliação positiva do artigo significa uma resposta positiva para as seguintes perguntas:

1. O estudo é interessante? (em relação aos objetivos da pesquisa)
2. As evidências apresentadas são válidas?
3. As evidências apresentadas são importantes?
4. As evidências apresentadas não contradizem autor algum selecionado como pilar da pesquisa?

Com a confirmação da qualidade do material, o mesmo foi exposto à extração de dados, ou seja, à leitura completa e detalhada do artigo.

6.1.1.5 Extração de dados

Com o objetivo de organizar os dados obtidos, facilitando o manuseio das informações, os dados extraídos de cada artigo foram registrados conforme o modelo apresentado na Tabela 6.

Tabela 6 – Exemplo de registro de material

Título	Autor(es)	Data de publicação	Fonte da publicação	Listagem das informações importantes
<i>Integration of Vision based SLAM and Nonlinear Filter for Simple Mobile Robot Navigation</i>	Dae Hee Won, Young Jae Lee, Sangkyung Sung, Taesam Kang	2008	IEEE	-Utilização de sensor de visão e encoders. -Filtro de partículas

6.1.2 Condução da revisão

Durante a condução da revisão, a busca efetiva dos materiais é realizada, os ciclos de busca são documentados e a *string* de busca é refinada, como afirma [?].

Esta pesquisa foi realizada entre os meses de março e junho de 2016. A *string* de busca apresentada no tópico 6.1.1.2 resultou em uma visão considerada fraca, pelo pesquisador e seus orientadores, sobre a auto-localização e o mapeamento de ambientes, peças chave da técnica de SLAM. Além disso, essa busca possibilitou a criação do *quasi-gold standard* inicial, com apenas um artigo: *Auto-localização e construção de mapas de ambiente para robôs móveis baseados em visão omnidirecional estéreo* [?].

Desse modo, foi necessária a realização de uma pesquisa manual para obter maior conhecimento sobre o tema e as palavras-chave a serem usadas para garantir maior qualidade dos resultados obtidos. Com a realização desta pesquisa, os seguintes artigos foram adicionados ao *quasi-gold standard*:

- *The Cleaning Robot Project: Aplicação do Filtro de Kalman na Auto-Localização de um Sistema Robótico Autônomo* [?],
- *Integration of Vision based SLAM and Nonlinear Filter for Simple Mobile Robot Navigation* [?] e
- *A Solution to the Simultaneous Localization and Map Building (SLAM) Problem* [?].

A partir da análise destes artigos iniciais, foi possível identificar diversas palavras-chave que levavam à pesquisa desejada, possibilitando refinamento da *string* de busca. Adicionando à mesma novos termos, como "*SLAM problem*" e "*simultaneous*", evoluindo a *string* e obtendo o seguinte resultado: "*simultaneous AND auto-localization AND environment mapping AND SLAM problem*".

O ciclo de busca utilizando esta *string* gerou poucos resultados, selecionando apenas um para análise: *Improved global localization of an indoor mobile robot via fuzzy extended information filtering* [?]. Com o objetivo ampliar a abrangência da busca, optou-se por modificar a palavra-chave *auto-localization* por apenas *localization*, obtendo a seguinte *string* de busca: "*simultaneous AND localization AND environment mapping AND SLAM problem*".

Com a realização deste novo ciclo de busca, diversos artigos relevantes foram identificados e adicionados ao *quasi-gold standard*. Os artigos presentes no *quasi-gold* corrente foram encontrados com esta nova busca, evidenciando uma certa qualidade da *string* de busca. Os artigos adicionados ao *quasi-gold standard* durante este ciclo são:

- *A Simultaneous Localization and Mapping Algorithm in Complex Environments: SLASEM* [?],
- *A Neuro-Fuzzy Assisted Extended Kalman Filter-Based Approach for Simultaneous Localization and Mapping (SLAM) Problems* [?],
- *Map Management for Efficient Simultaneous Localization and Mapping (SLAM)* [?] e
- *Simultaneous Localization and Map Building by Integrating a Cache of Features* [?].

Com o intuito de especificar mais a busca, foi adicionada, devido a uma dica do orientador prof. Dr. Maurício Serrano, a palavra-chave "*simple robots*" à *string* de busca, chegando a seguinte *string*: "*simultaneous AND localization AND environment mapping AND SLAM problem OR ("simple mobile robots"AND slam)*".

Com esta mudança, obteve-se outra visão desta pesquisa como um todo. Dessa forma, foram identificadas diversas pesquisas que buscam solucionar problemas de locomoção, como o problema de SLAM, em contextos limitados, da mesma forma que objetivo geral deste trabalho. Por fim, foram adicionados ao *quasi-gold standard*, os seguintes artigos:

- *BatSLAM: Simultaneous Localization and Mapping Using Biomimetic Sonar* [?],
- *Neural Network-Based Multiple Robot Simultaneous Localization and Mapping* [?] e

- *Visual simultaneous localization and mapping: a survey* [?].

Como é possível observar, os artigos selecionados para adição no *quasi-gold standard* não envolviam temas referentes a robôs simples, apesar da afirmação de que esta mudança havia modificado a visão da pesquisa. Artigos referentes a este tema não foram adicionados ao *quasi-gold standard* devido ao fato dos mesmos serem *barrados* pelo critério de exclusão "*CE 2 - Artigos que não possuam o problema de SLAM como foco principal*".

Em contrapartida, este ciclo possibilitou o conhecimento de novos termos, viabilizando um refinamento eficiente para o próximo ciclo de busca.

No próximo ciclo de busca, foram adicionadas palavras-chave referentes à Robótica Educacional e às estratégias de resolução do problema de SLAM, como mostra a *string*: "*(Simple mobile robot? AND (SLAM OR auto-localization)) AND (map* OR education* robot*) AND strateg**". Além de adicionar estes novos termos, utilizou-se de técnicas disponíveis nas bases, como a utilização de '?', que representa qualquer caractere, e '*', que significa que quaisquer caracteres precedidos dos caracteres anteriores ao '*' serão considerados.

Para realização desta busca, foi necessária a adaptação da *string* de busca durante a pesquisa na base de dados *Springer*, devido a diferenças nos padrões de definição da *string*. Para esta adaptação, novas palavras-chave precisaram ser selecionadas, as quais foram obtidas a partir dos resultados advindos das outras bases, com a mesma *string*.

A *string* adaptada para a base *Springer* foi: "*(Simple AND mobile AND robot AND SLAM AND localization AND mapping) AND (educational AND navigation AND simultaneous) AND strategies*".

Ao realizar esta busca, foi observado que os resultados atendiam, em sua maioria, ao desejado pela pesquisa, sendo muitos artigos selecionados para análise e avaliação e alguns adicionados ao *quasi-gold standard*, como:

- *Incremental SLAM with Backtracking Data Association for Mobile Robots* [?],
- *Mapping and Pursuit-Evasion Strategies For a Simple Wall-Following Robot* [?] e
- *A Simple and Parallel Algorithm for Real-Time Robot Localization by Fusing Monocular Vision and Odometry/AHRS Sensors* [?].

Com a realização desta busca, foram obtidos 27 artigos, restando apenas 20 após a avaliação da relevância dos mesmos para a pesquisa. Desde o primeiro ciclo de busca, diversos artigos foram considerados relevantes para a pesquisa, chegando a um número de 42 (quarenta e dois) artigos relevantes.

Ou seja, aplicando os conceitos de *precisão* e *sensitividade*, temos que:

- $sensitividade = \frac{20}{42}$
- $precisao = \frac{20}{27}$

Desse modo, foi obtida uma *sensitividade* de 47% e uma precisão de 74%, dando fim aos ciclos de busca com 42 artigos selecionados e analisados. A Tabela 7 apresenta, de maneira resumida, a evolução da *string* de busca, comparando a *string* inicial com a *string* final da pesquisa.

Tabela 7 – Comparação das *strings* inicial e final

String	Fonte de busca	Resultados		Observações
		Total	Relevantes	
Inicial	<i>IEEEExplore</i>	2	1	String construída sem o conhecimento necessário para utilização das palavras-chave que representam a pesquisa.
	<i>Springer</i>	19	2	
	<i>CAPES</i>	8	3	
Refinada	<i>IEEEExplore</i>	17	13	String refinada, ao longo de diversos ciclos de busca, adicionando novas palavras-chave e obtendo resultados mais específicos.
	<i>Springer</i>	6	3	
	<i>CAPES</i>	4	4	

Durante a realização de cada ciclo, os artigos foram analisados, avaliados e seus dados foram extraídos como fonte de estudo para a realização deste trabalho de conclusão de curso. Na seção 6.1.3, serão apresentados os resultados de maneira organizada e simplificada.

6.1.3 Publicação dos resultados

A primeira etapa deste trabalho de conclusão de curso pode ser vista como o resultado geral desta revisão sistemática, onde conceitos, termos, abordagens e qualquer informação utilizada no trabalho é fruto, seja direta ou indiretamente, desta revisão sistemática.

Com o objetivo de documentar os resultados diretos desta revisão sistemática, estes foram registrados como as técnicas utilizadas atualmente para solucionar o problema de SLAM em diferentes contextos.

Com a realização desta revisão, foi possível identificar *áreas* mutáveis nas diferentes soluções do problema de SLAM. As áreas identificadas são "*arquitetura da solução*" 6.1.3.1, "*técnica probabilística utilizada*" 6.1.3.2 e "*informações disponíveis*" 6.1.3.3. Em cada solução, os autores buscaram se adequar ao contexto trabalhado, seja a partir da disponibilidade de sensores específicos ou da capacidade computacional disponível.

6.1.3.1 Arquitetura da solução

A arquitetura da solução, na grande maioria dos estudos, foi definida a partir do requisito computacional. Ou seja, a limitação computacional presente nos robôs simples levou os autores a buscarem arquiteturas que contornassem esse problema, como mostra [?]. Entre as diversas arquiteturas, as mais comumente utilizadas são as que buscam processar as informações em um computador, utilizando o robô apenas para obtenção das informações, como ilustra a Figura 16.

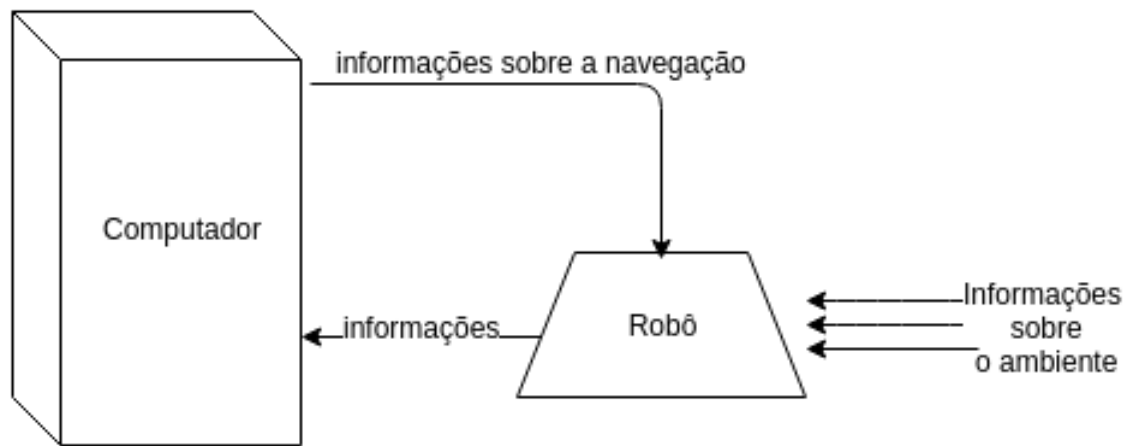


Figura 16 – Arquitetura de comunicação

De acordo com a Figura 16, o robô será responsável apenas por obter informações do ambiente, ou seja, recuperar os dados obtidos a partir dos sensores disponíveis. O computador, em posse das informações sobre o ambiente, ou seja, os pontos de referência, a quantidade de rotações em cada roda, a distância e cores de objetos, por exemplo, será responsável por processar toda a informação, construindo um mapa lógico para possibilitar a localização do robô em relação ao ambiente, como apresenta [?].

Geralmente, são utilizados dois mapas simultâneos, um presente no robô (local) e outro, mais completo, presente no computador (remoto). O mapa local é, basicamente, um vetor de pontos ' p ' em relação a um tempo ' t ', como explica [?]. O computador utiliza este mapa local, que é disponibilizado pelo robô, para completar, corrigir e atualizar o mapa remoto, mesclando informações e utilizando, geralmente, filtros probabilísticos para maximizar sua precisão [?].

As decisões referentes à navegação são geradas a partir da análise do mapa remoto, já que o mapa local é incompleto e inconsistente, como afirma [?]. A utilização desta arquitetura de mapeamento remoto e local torna prática a realização de navegações com múltiplos robôs, como apresenta, [?], em seu trabalho sobre rede de comunicação sem fio para navegação de múltiplos robôs.

Seguindo esta arquitetura, [?] e [?] também desenvolveram sistemas de resolução

do problema de SLAM com a utilização de múltiplos robôs, mostrando a viabilidade da sua utilização. Neste tipo de trabalho, os robôs colaboram entre si, pois, como a informação obtida é centralizada em um computador único, as decisões referentes à navegação de um determinado robô são resultados do processamento das informações obtidas por todos os robôs, maximizando a visão global de cada robô [?].

De qualquer forma, independente da arquitetura da solução utilizada, os erros advindos dos sensores sempre serão um problema sério a ser resolvido [?]. Com o objetivo de solucionar este problema, a comunidade de robótica vê-se presa à utilização de estruturas matemáticas probabilísticas, como afirma [?]. As seções 2.2.1, 2.2.2 e 6.1.3.2 apresentam as estruturas probabilísticas mais utilizadas atualmente, assim como suas vantagens e desvantagens.

6.1.3.2 Técnica probabilística utilizada

Para a realização de uma navegação específica, primeiramente, o robô deverá obter informações sobre o ambiente [?]. Para isso, é necessária a utilização de sensores que captem estas informações, seja a partir de odometria, infra-vermelho ou vídeo. Entretanto, estes sensores são munidos de uma margem de erro que, muitas vezes, prejudica a navegação e a auto-localização como um todo [?]. De acordo com [?]:

Devido à natureza imperfeita dos sensores, à falta da previsibilidade em ambientes reais e à necessidade de aproximações para alcançar decisões computacionais, a robótica é uma ciência que depende de algoritmos probabilísticos.

Desse modo, a comunidade de robótica vem buscando, na matemática, soluções probabilísticas que minimizem esta margem de erro, como afirma [?]. Entre as soluções mais utilizadas, encontram-se, no topo da lista, os filtros de *Kalman* e *partículas*, como pode ser observado nas seções 2.2.1 e 2.2.2, respectivamente.

Alguns autores, como, por exemplo, [?], vêm estudando as duas técnicas e comparando-as com o intuito de selecionar a "*melhor*" técnica. Entretanto, o termo "*melhor*" é relativo, neste caso, a depender do contexto em que será aplicado o filtro probabilístico, como afirma [?].

O estudo de [?] exemplifica, com clareza, as comparações entre os filtros de partícula e de Kalman, ao utilizar as duas técnicas para solucionar o problema de SLAM com um robô móvel munido de sensores a laser. De acordo com a relatividade do termo "*melhor*", nesta ocasião, o autor buscou apresentar apenas algumas vantagens e desvantagens dos dois filtros. As quais podem ser visualizadas nas seções 2.2.1 e 2.2.2.

6.1.3.3 Informações disponíveis

Na robótica móvel, existem diversas maneiras de se obter informações sobre o ambiente, a partir da utilização de sensores específicos para informações específicas [?]. Entre os sensores mais utilizados, encontram-se os *sonares*, *sensores infra-vermelho*, *câmeras de vídeo*, *sensores de distância*, *sensores RGB* e *sensores odométricos*. A partir do ambiente em que se deseja navegar, faz-se necessária a seleção dos sensores adequados para o mesmo.

Além do tipo de sensor escolhido, deve-se levar em consideração algumas características do mesmo, como o alcance, desempenho e precisão, por exemplo. Conhecer a margem de erro dos sensores é essencial, como afirma [?].

Como foi explicado no tópico 6.1.3.2, os sensores possuem margens de erro que podem prejudicar a navegação e a auto-localização do robô. Em busca de tentar solucionar este problema, além da utilização de filtros probabilísticos, diversos autores buscam utilizar múltiplos sensores, integrando as informações dos mesmos para minimizar a margem de erro na informação, como mostram [?], [?] e [?].

De acordo com [?], há três tipos de integração entre sensores: *complementar*, onde os sensores têm apenas uma visão parcial do ambiente, unindo as informações para obter uma visão mais completa; *competitiva*, onde dois ou mais sensores competem para obter a informação com maior precisão, como a medição da distância de um único objeto utilizando sonar e sensor a laser, por exemplo; e *cooperativa*, quando os sensores cooperam entre si para obter informações que não seriam possíveis com a utilização de apenas um, como a geração de informações em três dimensões, por exemplo.

Entre os sensores citados anteriormente, o sensor odométrico, geralmente, é utilizado em qualquer sistema de navegação sobre eixos, de acordo com [?]. Este sensor recupera os dados sobre rotações realizadas nas rodas em que o mesmo se encontra, possibilitando a mensuração da distância percorrida, assim como a identificação de curvas, por exemplo [?].

Entretanto, como afirma [?], sensores odométricos geram erros acumulativos, ou seja, quanto maior for a navegação, maior será o erro total do sensor. Trabalhar com sensores assim gera uma certa dificuldade. Entretanto, com a utilização de filtros probabilísticos, como apresentado na seção 6.1.3.2, este erro pode ser reduzido, viabilizando sua utilização.

Já a utilização de câmeras de vídeo, segundo [?], além de necessitar de *landmarks* verificáveis através do processamento de imagens, as mesmas exigem requisitos computacionais que, muitas vezes, acabam por inviabilizar sua utilização.

O sensor ultrassônico, que será utilizado durante a segunda etapa deste trabalho, diferentemente das câmeras de vídeo, são simples, rápidos e baratos, de acordo com [?].

Entretanto, sua utilização acaba por limitar o contexto de navegação, devido ao seu limite de alcance. Ou seja, em um ambiente aberto, o sensor ultrasônico não será capaz de identificar obstáculo algum. Além da sua limitação de alcance, [?] apresenta algumas características que podem problematizar sua utilização:

1. *pobre direcionalidade*: não são capazes de identificar a direção do obstáculo identificado, apenas sua distância;
2. *passíveis de ruídos*: ruídos são comuns durante a utilização de sonares, seja devido a interferências de outros sensores, ou a partir da sua margem de erro padrão, por exemplo.
3. *reflexão especular*: O sensor emite o sinal sonoro em formato de cone, possibilitando a obtenção de erros relacionados à angulação de incidência, como ilustra a Figura 17.

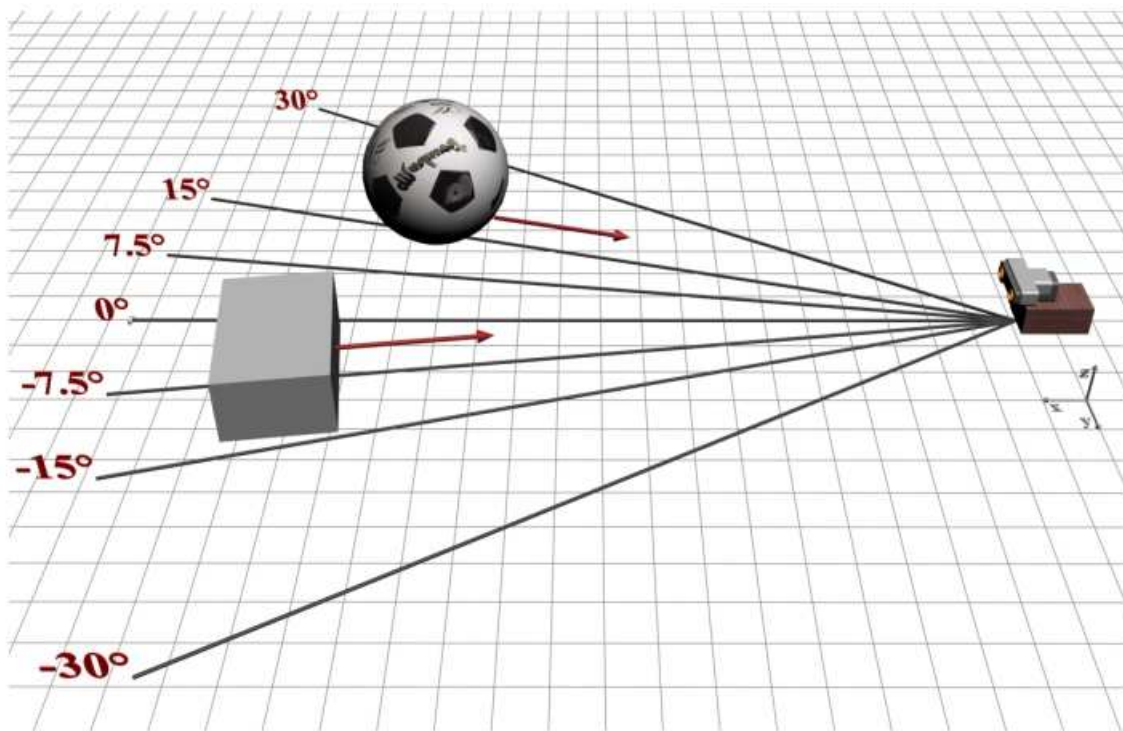


Figura 17 – Emissão do sinal sonoro - sonar

Outro detalhe importante a ser analisado, segundo [?], é a velocidade de emissão do sinal. Esta deve ser de acordo com a velocidade atual do robô, ou seja, se o robô se movimenta rapidamente, devem ser gerados sinais a uma frequência alta, possibilitando a reação do robô a eventos inesperados [?].

Com a emissão de sinais sonoros enquanto o robô se movimenta, deve-se levar em consideração o atraso no mapeamento, já que, ao obter o retorno do sinal, o robô já não se encontra no mesmo local [?].

A Tabela 8 apresenta de maneira organizada a contabilização dos resultados da revisão, de acordo com as áreas destacadas anteriormente. Encontram-se nesta Tabela apenas os artigos que solucionaram, de alguma maneira, o problema de SLAM.

Tabela 8 – Contabilização de estratégias (SLAM)

Arquitetura de processamento	Filtro	Inputs	Nº. Artigos
Processamento remoto	Kalman	laser, vídeo, odometria, sonar, infra-vermelho, câmera térmica, kinect RGB, sonar biomimético.	12
	Partículas	Câmera de vídeo, odometria, laser, sonar, magnetômetro, sensor de rádio.	5
Processamento local	Kalman	sensor de distância (laser), odometria, identificador de landmarks (simulado) e sonar.	3
	Partículas	Sonar e odometria	1
Processamento remoto + múltiplos robôs	Kalman	laser (range), vídeo e odometria	3
	Partículas	0	0
Total			24

6.2 Desenvolvimento prático

Com o objetivo de obter conhecimento referente à solução prática proposta durante este trabalho, foi desenvolvida uma pequena prova de conceito durante a primeira etapa deste trabalho (TCC_1). Nesta seção estão apresentadas as informações sobre o planejamento e condução da prova de conceito, assim como as informações importantes obtidas com a realização da mesma, caracterizadas como *resultados da prova de conceito*.

Esta seção está dividida entre *planejamento e condução* da prova de conceito, 6.2.1, e *características técnicas* 6.2.2.

6.2.1 Planejamento e condução

Esta prova de conceito foi pensada com o objetivo de identificar algumas características básicas sobre a implementação de sistemas de navegação inteligente utilizando os

kits de robótica Mindstorm, da Lego. Com este objetivo, a prova de conceito envolveu a análise de ferramentas utilizadas, como sensores e atuadores, o estudo e seleção da linguagem de programação a ser utilizada e alguns detalhes referentes à localização relativa de obstáculos.

Para isso, buscou-se simular uma atividade básica na navegação e auto-localização na robótica. Atividade esta, que tem como objetivo identificar obstáculos ao redor do robô, levando em consideração a margem de erro, assim como todas as características presentes nos sensores e atuadores do kit. Ou seja, o robô é colocado em um ambiente desconhecido, onde o mesmo busca identificar obstáculos à sua volta, com o objetivo de definir uma saída, onde não se encontra nenhum obstáculo.

O ambiente utilizado para a prova de conceito pode ser observado na Figura ???. A montagem do robô utilizada nesta prova seguiu o estabelecido na seção 4.2.2, utilizando esteiras, um sonar para identificar a distância de obstáculos, um sensor de toque e sensores odométricos em cada motor.

6.2.2 Características técnicas

Neste tópico serão apresentadas algumas das características importantes identificadas durante a realização da prova de conceito, como pode ser observado nos sub-tópicos a seguir.

6.2.2.1 Seleção da linguagem e ambiente de desenvolvimento

Para realização desta prova de conceito, assim como o desenvolvimento de toda a solução proposta, ao longo do TCC_2, optou-se pela utilização da linguagem Java. Esta escolha se deu não só devido à possibilidade, com mais facilidade, da integração desta solução com o *framework* desenvolvido por [?], como também por todo o apoio técnico advindo da utilização da ferramenta *leJOS NXJ*¹.

A ferramenta leJOS NXJ engloba, basicamente, uma máquina virtual Java (JVM) desenvolvida em C, sendo multiplataforma, ou seja, é portátil para sistemas Linux, Windows e Macintosh [?]. O material de estudo sobre a ferramenta leJOS NXJ se encontra disponível em toda a *web*, de maneira livre, e no livro [?], que foi utilizado como fonte de informação durante o desenvolvimento.

O sistema operacional utilizado para realização desta prova de conceito foi o *windows*, devido a sua facilidade de configuração da comunicação entre robô/PC, diferentemente do observado em sistemas Linux. O *passo-a-passo* para instalação e configuração do ambiente utilizado para desenvolvimento se encontra em [?, p. 6].

¹ <http://www.lejos.org/nxj.php>

A seleção desta ferramenta se deu, além dos motivos apresentados anteriormente, pelo fato da existência de uma *API* que disponibiliza diversas funcionalidades referentes à navegação dos robôs. A *API* do leJOS NXJ engloba desde funções referentes ao controle de motores, até o apoio a criação de mapas e sistemas complexos de navegação.

Em paralelo à escolha da linguagem de programação, buscou-se identificar um ambiente de desenvolvimento que disponibilizasse ferramentas de apoio que facilitem o desenvolvimento da solução. Com este objetivo, optou-se pela utilização da *IDE* Eclipse², por possuir um *plugin* da ferramenta leJOS NXJ, como apresenta [?]. O tutorial para instalação e configuração do ambiente utilizado se encontra em [?, p. 14].

6.2.2.2 Atuadores

Os atuadores, ou motores, contemplam a base da robótica móvel. Desse modo, o leJOS NXT disponibiliza diversas funcionalidades referentes ao controle destes atuadores. Um exemplo disso é a possibilidade de controlar a aceleração do motor, possibilitando um arranque com aceleração gradual, para minimizar as chances de derrapagem, por exemplo.

Estas possibilidades de controle de rotação são devido a existência de *encoders* óticos em cada motor, como apresenta [?]. Cada *encoder* tem como objetivo registrar as rotações de cada eixo, possibilitando a navegação por odometria, como já foi explicado ao longo do trabalho.

Para acessar os atuadores, o leJOS NXJ oferece, como principal fonte de acesso, a classe *Motor*, que possui três instâncias estáticas: *Motor.A*, *Motor.B* e *Motor.C*. O livro [?] faz uma análise detalhada de todos os métodos presentes nesta classe, os quais são, principalmente, voltados à aceleração e velocidade de rotação dos eixos.

6.2.2.3 Sensores

Cada computador central do kit Mindstorm possui quatro portas para sensores, ou seja, a solução deste trabalho só pode envolver um máximo de quatro sensores, que fazem parte do kit Mindstorm. Para esta prova de conceito, foram utilizados os sensores de *odometria*, a partir dos *encoders* em cada atuador, um sensor ultrasônico, como sensor de distância e um sensor de toque.

- **Sensor de toque:**

É o sensor mais básico do kit, que retorna um valor *booleano* que indica se o sensor está pressionado ou não, a partir do botão laranja apresentado na figura 18.

² <https://eclipse.org/>



Figura 18 – Sensor de toque.

A classe *TouchSensor*, que implementa a interface deste sensor, contém apenas um simples método: *boolean isPressed()*.

- **Sensor ultrasônico:**

Como apresenta a Figura 19, o sensor ultrasônico lembra bastante um par de olhos, apesar de possuir muitas características em comum com um sensor de som, em vez de uma câmera, por exemplo. Isso se dá pela estratégia de funcionamento do sensor, o qual emite um sinal sonoro que reflete em obstáculos à frente, retornando ao sensor. A partir da análise do tempo percorrido pelo sinal sonoro, é possível estimar a distância do objeto analisado, em relação ao robô.



Figura 19 – Sensor ultrasônico.

Este sensor é capaz de identificar distâncias de até dois metros e meio, mais especificamente 255 centímetros. Porém, sua utilização em distâncias tão grandes não é recomendada por [?], devido a grande margem de erro presentes em medições como esta. De acordo com [?], este sensor possui, em distâncias de aproximadamente 180 centímetros, uma margem de erro de mais ou menos três centímetros. Sua margem de erro é proporcional à distância entre robô e obstáculo.

Outra característica importante do funcionamento deste sensor é a emissão de sinais no formato de cone, como apresenta a Figura 20³.

³ <http://arcbotics.com/products/sparki/parts/ultrasonic-range-finder/>

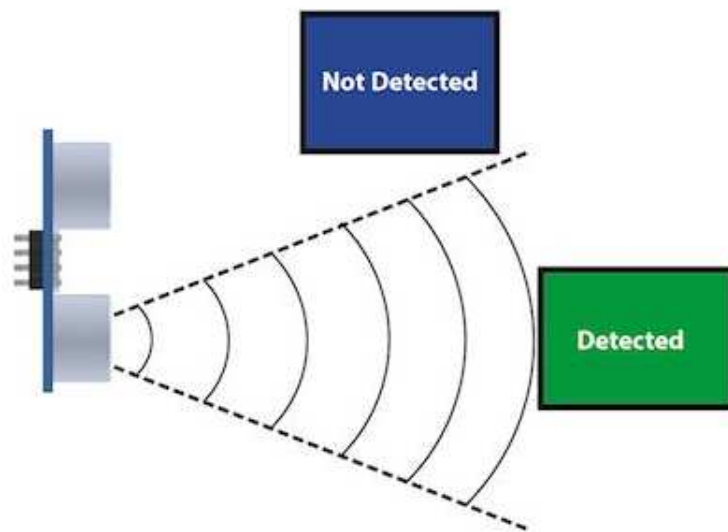


Figura 20 – Emissão do sinal ultrassônico.

O cone formado pela emissão do sinal segue uma angulação de 30° , ou seja, em uma distância de 180 centímetros, o cone possui um diâmetro de 90 centímetros. Desse modo, deve-se levar em consideração a incapacidade de identificar pequenas irregularidades em obstáculos, como fendas e buracos.

6.2.3 Computador central NXT (Brick)

O robô utilizado durante esta prova de conceito e durante o desenvolvimento da solução proposta, é da família Mindstorm NXT, da Lego. O computador central, de acordo com [?], possui uma área de 7,2 x 11,2 centímetros, com um processador *Atmel 32-bit ARM* de 48 MHz de frequência, memória RAM de 64 KB e 256 KB de memória *flash*.

Estas limitações de memória e processamento foram contornadas a partir da utilização da arquitetura de processamento remoto, como foi definido na seção 4.2, já como um resultado da revisão sistemática apresentada na seção 6.1.

6.3 Considerações parciais

Durante a segunda etapa do trabalho, pretende-se evoluir a prova de conceito apresentada durante esta primeira etapa, buscando solucionar, de maneira simplificada, o problema de SLAM. A partir dos ciclos de desenvolvimento, serão realizadas análises referentes às peculiaridades encontradas ao implementar técnicas de auto-localização em robôs simples, assim como o impacto desta implementação em um contexto educacional.

A escolha da técnica que será utilizada será feita durante a realização da segunda etapa do trabalho, a partir de análises da exigência computacional e complexidade de

implementação.

7 Considerações finais

Como foi apresentado ao longo deste trabalho, a comunidade de robótica busca solucionar o problema de SLAM utilizando diversas técnicas e estratégias. Entretanto, boa parte da comunidade concorda na utilização de algumas estratégias, como o processamento remoto, apresentado na tabela 8. Visto isso, este trabalho buscará seguir as estratégias mais utilizadas e mais úteis para o contexto de pesquisa, como a utilização da arquitetura de processamento remoto e filtros probabilísticos.

A escolha do filtro probabilístico a ser utilizado depende de etapas de provas de conceito que serão realizadas durante as primeiras sprints de desenvolvimento do TCC_2. A escolha será entre a utilização do filtro de Kalman ou do filtro de Partículas, ambos detalhados ao decorrer deste trabalho.

A escolha da linguagem de programação utilizada, Java, se dá pela necessidade de integração com o *framework* Traveller, além da facilidade e apoio disponíveis com a utilização da ferramenta *Lejos*. Os conhecimentos obtidos durante a realização da prova de conceito comprovam que a escolha da utilização da linguagem Java possibilita maior suporte técnico, como materiais e conteúdos na *web*, o que facilita o desenvolvimento do trabalho como um todo.

Ao longo do TCC_1, o foco do trabalho foi levantar técnicas e estratégias existentes para resolução do problema de SLAM, assim como identificar conceitos e informações referentes à auto-localização na robótica e robótica educacional. Desse modo, levando em consideração os objetivos específicos, a Figura 7 apresenta o *status* do trabalho.

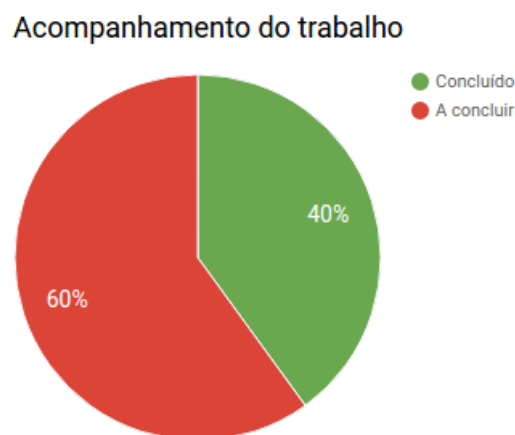


Figura 21 – Trabalho desenvolvido

Os objetivos específicos foram registrados como:

- Identificar uma solução para o problema de SLAM em um contexto simplificado;
- Propor adaptação para o contexto de robôs simples, na Robótica Educacional;
- Implementar adaptação.

Desse modo, durante a primeira etapa da pesquisa (TCC_1), foi possível identificar uma solução para o problema de SLAM em um contexto simplificado, a partir da análise do resultado da revisão sistemática. A proposta de adaptação foi realizada em parte, já que já foi definida a utilização da arquitetura de processamento remoto e a seleção de dois filtros probabilísticos que serão utilizados na adaptação. Ou seja, aproximadamente 40% do trabalho foi concluído, o que está dentro do previsto, pois o período destinado para a realização da segunda etapa (TCC_2) é maior, já que é referente a uma disciplina de seis créditos, diferentemente da primeira etapa.