



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: Dr. Maurício Serrano  
Coorientadora: 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: Dr. Maurício Serrano

Coorientador: Dra. Milene Serrano

Brasília, DF

2016



# 1 Introdução

## 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 tempo para produção dos mesmos (ROMANO, 2002). Segundo (ROMANO, 2002), a palavra robô é derivada da palavra *robota*, de origem eslava, que significa *trabalho forçado*, ou seja, robôs podem ser considerados ferramentas incansáveis que apoiam o trabalho humano.

Segundo (OLIVEIRA, 2008), 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 (OLIVEIRA, 2008), 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 (COSTA; Okamoto Jr., 2002), 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 (SANTOS; SILVA; ALMEIDA, 2002), 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 conhecimento 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 (SANTOS; SILVA; ALMEIDA, 2002), 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, a qual é 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 (SANTOS; SILVA; ALMEIDA, 2002). A Navegação com utilização de Bússulas garante um apoio muito útil para orientação do robô. Entretanto, essa técnica gera problemas relacionados a interferências externas, como materiais eletromagnéticos próximos à bússula (SANTOS; SILVA; ALMEIDA, 2002).

A técnica de Odometria é muito utilizada em navegações curtas, em ambientes com o piso regular e plano. Entretanto, segundo (SANTOS; SILVA; ALMEIDA, 2002), 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 Mindstorms<sup>1</sup>. Tais kits possuem poucas opções de sensores e características bem limitadas (HÁMORI; LENGYEL; RESKÓ,

---

<sup>1</sup> mindstorms.lego.com

2011). 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 (OLIVEIRA, 2008). Porém, sabe-se que, nestes contextos, o *hardware* utilizado para navegação era de alta tecnologia, possuindo processadores de altíssimo desempenho e sensores bem 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 (OLIVEIRA, 2008), 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*) (DISSANAYAKE et al., 2001).

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, a chamada Robótica Educacional (MALIUK, 2009), traz alguns benefícios para o aluno. Conforme colocado pelos autores (GALVAN et al., 2006), (ZHAO et al., 2008), (MALIUK, 2009) e (WAHAB1; AZAHARI2; TAJUDDIN1, 2015), alguns desses benefícios são:

- Maior interesse pelos conteúdos estudados em aula;
- capacidade de trabalhar em grupo;
- aplicação prática do conhecimento teórico;
- 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* (LAU et al., 2002) ou *micro-rato* (SANTOS; SILVA; ALMEIDA, 2002). Neste tipo de campeonato, a navegação é o quesito mais importante (LAU et al., 2002), 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, (RINCON, 2015), que desenvolveu um *framework* de definição de trajetórias para robôs móveis, e (RAMALHO, 2015), 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

- Resolucionar do problema de SLAM;
- Propor adaptação pro contexto de robôs simples, na robótica educacional;
- Implementar adaptação.

## 1.5 Metodologia

Este trabalho segue uma metodologia de pesquisa exploratória e com uma abordagem qualitativa, onde são selecionadas, comparadas e analisadas técnicas de resolução



do problema de SLAM. De acordo com (OLIVEIRA, 2011), a pesquisa exploratória tem como objetivo ampliar os conhecimentos do pesquisador sobre o tema. O uso da pesquisa exploratória, neste trabalho, se dá pela necessidade do conhecimento sobre as técnicas de resolução do problema de SLAM, com o intuito de realizar adaptações das mesmas para o contexto de robôs simples. Já a abordagem qualitativa, tem como objetivo entender o funcionamento e as técnicas existentes, para buscar uma forma de adaptação para um contexto simplificado, não se importando com dados quantificáveis.

Com o objetivo de identificar o máximo de técnicas possível, com os mais diferentes tipos de *hardwares*, será realizada uma pesquisa bibliográfica sobre os temas *Auto-localização na Robótica, o Problema de SLAM e as características dos robôs simples presentes na Robótica Educacional*. As principais fontes de dados em estudo são as bases da CAPES, IEEE e Scopus.

A primeira fase deste trabalho é focada em estabelecer pilares teóricos que sustentem um projeto de adaptação de técnicas durante a segunda fase. Esta primeira fase pode ser sub-dividida em três atividades: *Realizar Pesquisa Bibliográfica, selecionar Técnicas e Propor Adaptações*. As mesmas estão distribuídas de acordo com o cronograma disposto na tabela 1.

Tabela 1 – Cronograma TCC 1

Cronograma	Março	Abril	Maior	Junho
Realizar Pesquisa Bibliográfica	X	X	X	X
Selecionar técnicas		X	X	
Propor adaptação			X	X



## 2 Referencial Teórico

Durante esta seção, questões referentes a todo o contexto abordado neste trabalho serão apresentadas e descritas de forma prática para facilitar o total entendimento do tema trabalhado.

### 2.1 A Robótica e a Auto-Localização

O nascimento da robótica se deu no contexto industrial, onde a automação de atividades repetitivas garantiu maior eficiência e, consequentemente maior lucro (ROMANO, 2002). Porém, com o passar dos anos, a robótica vem se expandindo e fazendo parte da vida cotidiana de muitos (GALVAN et al., 2006). A robótica é uma importante ferramenta que apóia o trabalho humano em diversos contextos, seja para a limpeza de uma casa (MARTINS, 2008) ou até para explorar novos planetas, por exemplo (MAIMONE; CHENG; MATTHEWS, 2007).

No contexto da robótica móvel, existem por exemplo, os robôs de serviço. Os quais vêm sendo largamente evoluídos pela comunidade de robótica (PINTO, 2008). Ainda segundo (PINTO, 2008), estes robôs, geralmente, desempenham ações que contemplam diversas aplicações, como:

- Aplicações que envolvam risco de vida significativo para humanos (MAIMONE; CHENG; MATTHEWS, 2007);
- Funções economicamente desvantajosas no uso de trabalhadores humanos (ROMANO, 2002);
- Uso humanitário (cadeiras de rodas autônomas, por exemplo);
- Uso educacional (FARROKHSHIAR; KRYSS; NAJJARAN, 2010).

Robôs de serviço capazes de se deslocar livremente pelo ambiente, conhecendo-o, poderão realizar suas atividades de forma mais eficiente (PINTO, 2008), o que garante sua autonomia.

Segundo (ROMANO, 2002), 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 (CHEW et al., 2009). Segundo (MACHADO, 2003), existem inúmeros tipos de sensores,

desde sensores de toque até sensores de visão ou de som. Os sensores mais utilizados em robôs móveis, segundo (MACHADO, 2003), 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 (WON et al., 2008). Segundo (MACHADO, 2003), 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 (MACHADO, 2003). Porém, sua utilização possui uma exigência computacional (OLIVEIRA, 2008) 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 (MACHADO, 2003). Ainda segundo (MACHADO, 2003), 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 (PINTO, 2008).

- *Laser:*

Também é um tipo de sensor semelhante aos sonares e sensores de infravermelho, porém estes, além de serem direcionados, são mais precisos. O que os tornam equipamentos mais caros (MACHADO, 2003).

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 (MACHADO, 2003). 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 (SANTOS; SILVA; ALMEIDA, 2002).

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 (PINTO, 2008). Para solucionar este problema, uma técnica bastante difundida é a utilização de mapas, como mostra (SANTOS; SILVA; ALMEIDA, 2002). 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 (SANTOS; SILVA; ALMEIDA, 2002).

Porém, 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 entre toda a comunidade de pesquisadores em robótica móvel, a resolução do problema de SLAM (Auto-Localização e Mapeamento de Ambientes Simultâneos). Segundo (DISSANAYAKE et al., 2001), 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 muito utilizada para navegação em diferentes contextos, como na navegação marítima, por exemplo (DISSANAYAKE et al., 2001). Segundo (DISSANAYAKE et al., 2001), 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 juz a palavra *robô*. Desse modo, pesquisadores em robótica móvel vêm buscando soluções elegantes para o problema de SLAM (WON et al., 2008).

Inúmeras soluções para o problema de SLAM, em diferentes contextos, já foram desenvolvidas e vêm sendo refinadas, com o intuito de maximizar a efetividade da navegação (PINTO, 2008). Segundo (ADAMS; MULLANE; VO, 2013), graças a natureza imperfeita presente nos sensores, a falta de previsibilidade em ambientes reais de atuação e a necessidade de aproximações para viabilizar a análise de decisões computacionais, a robótica é uma ciência dependente de algoritmos probabilísticos. E por esse motivo, diversas abordagens de análise probabilística podem ser seguidas (ADAMS; MULLANE; VO, 2013).

De acordo com (DISSANAYAKE et al., 2001), a abordagem mais utilizada para resolução do problema de SLAM, é a utilização do filtro de Kalman. (PINTO, 2008) descreve o filtro de Kalman como: "*Um algoritmo recursivo de processamento de informações, proporcionando a estimação ótima do estado de um sistema dinâmico com ruído linear*".

Este filtro utiliza diversas disciplinas, como mostra (PINTO, 2008):

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

Porém, 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 (MACHADO, 2003).

Em seu trabalho, (DISSANAYAKE et al., 2001) 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  $N^2$ . (DISSANAYAKE et al., 2001) 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 (DISSANAYAKE et al., 2001).

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* (WON et al., 2008). 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 (MOONEY, 1997) e (NETO, 2015).

O funcionamento desta abordagem se baseia em subdividir o ambiente em partículas espalhas uniformemente, que representam o robô munido de seus sensores (NETO, 2015). Cada partícula é uma hipótese da posição atual do robô (GUIZILINI et al., 2004). O robô e as partículas obtêm informações do ambiente, o robô utilizando seus sensores e as partículas utilizando equações matemáticas para geração destes dados (NETO, 2015). 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 (NETO, 2015). Com o passar dos ciclos, as partículas que ainda continuam no ambiente representam a posição atual do robô (NETO, 2015).

A partir da comparação entre estas duas abordagens (filtro de Kalman e filtro de Partículas) (NETO, 2015), 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 a

informação disponível (sensores) para minimizar a margem de erro da estimativa (NETO, 2015). Porém, ele só garante a efetividade da estimativa para sistemas lineares com ruídos gaussianos (NETO, 2015). O mesmo não resolve o problema do sequestro do robô.

Já o filtro de Partículas permite sua utilização em sistemas sem ruído gaussiano, e soluciona o problema do sequestro do robô (SEIFZADEH; WU; WANG, 2009). Entre suas desvantagens se encontra o alto custo computacional e a dificuldade da definição da quantidade ideal de partículas a serem utilizadas (NETO, 2015).

O sequestro do robô, citado acima, é um problema especial de localização global no campo da robótica móvel, onde o desafio deste problema, segundo (MAJDIK et al., 2010), é fazer com que o robô seja capaz de se localizar em um mapa após ser sequestrado (ser retirado de sua posição e colocado em uma posição desconhecida no mapa).

As duas grandes abordagens apresentadas acima 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 **REFERENCIAR**. 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. Onde os sensores, capacidades de processamento e memória são bastante limitados **REFERENCIAR**.

## 2.3 Robótica Educacional

### 2.3.1 Construcionismo

De acordo com (NUNES; SANTOS, 2013), 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. Esta filosofia, de acordo com (NUNES; SANTOS, 2013), 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 (NUNES; SANTOS, 2013). 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 **REFERENCIAR**, além de conteúdos presentes em diversas disciplinas, como matemática e física (MALIUK, 2009).

O construcionismo é baseado em uma filosofia construtivista (NUNES; SANTOS, 2013), 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 ambiente do problema

([BECKER, 2009](#)).

### 2.3.2 Mindstorm NXT



## 3 Suporte Tecnológico

Nesta seção serão apresentadas ferramentas e tecnologias utilizadas para suportar 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 *Projeto* e *Engenharia de Software*.

### 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 GIT<sup>1</sup> 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 Github<sup>2</sup> é uma ferramenta mais utilizada para hospedagem remota de projetos GIT. A ferramenta contempla uma *Wiki* para documentação do projeto e sistemas de *Issues* e *Milestones* para gerenciamento de atividades.

#### 3.1.3 Bonita BPMN

Ferramenta para modelagem de processos *BPMN*<sup>3</sup>, o Bonita<sup>4</sup> foi escolhido graças a sua facilidade de utilização e portabilidade para sistema Linux.

#### 3.1.4 Linux Mint

O sistema operacional utilizado durante este trabalho será o Linux Mint<sup>5</sup>.

---

<sup>1</sup> <https://git-scm.com/>

<sup>2</sup> <https://github.com>

<sup>3</sup> <http://www.bpmn.org/>

<sup>4</sup> <http://www.bonitasoft.com/>

<sup>5</sup> <https://www.linuxmint.com/>

### 3.1.5 LaTeX

O LaTeX<sup>6</sup> 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 3<sup>7</sup>, é um editor de texto bastante utilizados por programadores, por possuir apoio para diversas linguagens de programação, incluindo textos em LaTeX.

---

<sup>6</sup> <https://www.latex-project.org/>

<sup>7</sup> <https://www.sublimetext.com/3>

# Referências

- ADAMS, M.; MULLANE, J.; VO, B.-N. Circumventing the feature association problem in slam. *IEEE Intelligent transportation systems magazine*, 2013. Citado na página 11.
- BECKER, F. O que é construtivismo? Universidade Federal do Rio Grande do Sul, 2009. Citado na página 14.
- CHEW, M. et al. Simple mobile robots for introduction into engineering. *International Instrumentation and Measurement Technology Conference*, 2009. Citado na página 9.
- COSTA, A. H. R.; Okamoto Jr., J. *Robótica Industrial - Interação de Robô no Ambiente*. [S.l.]: Edgard Blücher Ltda, 2002. ISBN 8521203152. Citado na página 3.
- DISSANAYAKE, G. et al. A solution to the simultaneous localization and map building (slam) problem. The University of Sydney, Australia, 2001. Citado 3 vezes nas páginas 5, 11 e 12.
- FARROKHSIAR, M.; KRYSS, D.; NAJJARAN, H. A teaching tool for the state-of-the-art probabilistic methods used in localization of mobile robots. Okanagan School of Engineering, University of British Columbia, Kelowna, B. C., Canada, 2010. Citado na página 9.
- GALVAN, S. et al. Innovative robotics teaching using lego sets. University of Verona, Italy, 2006. Citado 2 vezes nas páginas 5 e 9.
- GUIZILINI, V. C. et al. Implementação do dp-slam em tempo real para robôs móveis usando sensores. Universidade Estadual de Campinas, 2004. Citado na página 12.
- HÁMORI Ákos; LENGYEL, J.; RESKÓ, B. 3dof drawing robot using lego-nxt. Óbuda University, Hungary, 2011. Citado na página 5.
- LAU, N. et al. Ciber-rato: Uma competição robótica num ambiente virtual. 2002. Citado na página 6.
- MACHADO, K. F. Módulo de auto-localização para um agente exploratório usando filtro de kalman. Universidade Federal do Rio Grande do Sul - Instituto de Informática, 2003. Citado 3 vezes nas páginas 9, 10 e 12.
- MAIMONE, M.; CHENG, Y.; MATTHIES, L. Two years of visual odometry on the mars exploration rovers. *Journal of Field Robotics*, Wiley Online Library, v. 24, n. 3, p. 169–186, 2007. Citado na página 9.
- MAJDIK, A. et al. New approach in solving the kidnapped robot problem. Technical University of Cluj-Napoca, Romania, 2010. Citado na página 13.
- MALIUK, K. D. Robótica educacional como cenário investigativo nas aulas de matemática. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2009. Citado 2 vezes nas páginas 5 e 13.

- MARTINS, J. M. F. Melhoramento do desempenho do robot de serviço de limpeza. Faculdade de Engenharia da Universidade do Porto, 2008. Citado na página 9.
- MOONEY, C. Z. *Monte carlo simulation*. [S.l.]: Sage Publications, 1997. Citado na página 12.
- NETO, A. M. M. da S. Comparação entre o filtro de kalman e filtro de partículas aplicados na robótica móvel. Universidade Estadual de Campinas, 2015. Citado 2 vezes nas páginas 12 e 13.
- NUNES, S. da C.; SANTOS, R. P. dos. O construcionismo de papert na criação de um objeto de aprendizagem e sua avaliação segundo a taxionomia de bloom. IX Encontro Nacional de Pesquisa em Educação em Ciências, 2013. Citado na página 13.
- OLIVEIRA, M. F. de. Metodologia científica: um manual para a realização de pesquisas em administração. Universidade Federal de Goiás, Catalão GO, 2011. Citado na página 7.
- OLIVEIRA, P. R. G. D. Auto-localização e construção de mapas de ambiente para robôs móveis baseados em visão omnidirecional estéreo. São Paulo, Brasil, 2008. Citado 3 vezes nas páginas 3, 5 e 10.
- PINTO, F. P. V. F. The cleaning robot project aplicação do filtro de kalman na auto-localização de um sistema robótico autónomo. Faculdade de Engenharia da Universidade do Porto, 2008. Citado 3 vezes nas páginas 9, 10 e 11.
- RAMALHO, C. B. Máquina de raciocínio lógico para tomada de decisões estratégicas em robótica educacional. 2015. Citado na página 6.
- RINCON, R. L. Traveller: Um framework de definição de trajetórias para robôs móveis. 2015. Citado na página 6.
- ROMANO, V. F. *Introdução à Robótica Industrial*. [S.l.]: Edgard Blücher Ltda, 2002. ISBN 8521203152. Citado 2 vezes nas páginas 3 e 9.
- SANTOS, F. M.; SILVA, V. F.; ALMEIDA, L. Auto-localização em pequenos robôs móveis e autônomos: O caso do robô bulldozer iv. 2002. Citado 5 vezes nas páginas 3, 4, 6, 10 e 11.
- SEIFZADEH, S.; WU, D.; WANG, Y. Cost-effective active localization technique for mobile robots. International Conference on Robotics and Biomimetics, 2009. Citado na página 13.
- WAHAB<sup>1</sup>, A. F. A.; AZAHARI<sup>2</sup>, M. H.; TAJUDDIN<sup>1</sup>, R. M. An evaluation of robotic education scale in enhancing science achievement. Universiti Teknologi MARA, 2015. Citado na página 5.
- WON, D. H. et al. Integration of vision based slam and nonlinear filter for simple mobile robot navigation. Konkuk University Seoul, Korea, 2008. Citado 3 vezes nas páginas 10, 11 e 12.
- ZHAO, S. et al. Research on robotic education based on lego bricks. School of Information Science and Engineering, Northeastern University, 2008. Citado na página 5.