

**UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS**

Alisson Lovatto

Controle de navegação de um robô autônomo móvel

São Carlos

2021

Alisson Lovatto

Controle de navegação de um robô autônomo móvel

Monografia apresentada ao Curso de Engenharia Elétrica com Ênfase em Eletrônica, da Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Engenheiro Eletricista.

Orientador: Prof^o. Dr. Marcelo Becker

São Carlos
2021

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

L896c Lovatto, Alisson
Controle de navegação de um robô autônomo
móvel / Alisson Lovatto; orientador Marcelo Becker. São
Carlos, 2021.

Monografia (Graduação em Engenharia Elétrica com
ênfase em Eletrônica) -- Escola de Engenharia de São
Carlos da Universidade de São Paulo, 2021.

1. Controle Digital. 2. Robótica. 3. Modelagem. 4.
Motor. 5. Discretização. 6. Autônomo. 7. Encoder. 8.
Simulink. I. Título.

FOLHA DE APROVAÇÃO

Nome: Alisson Lovatto

Título: “Controle de navegação de um robô autônomo móvel”

**Trabalho de Conclusão de Curso defendido e aprovado em
29/11/2021,**

com NOTA NOVE (9,0), pela Comissão Julgadora:

Prof. Associado Marcelo Becker - Orientador - SEM/EESC/USP

Prof. Associado Daniel Varela Magalhães - SEM/EESC/USP

Prof. Associado Valdir Grassi Junior - SEL/EESC/USP

**Coordenador da CoC-Engenharia Elétrica - EESC/USP:
Prof. Associado Rogério Andrade Flauzino**

*Este trabalho é dedicado a todos os membros da Equipe SEMEAR
que buscam por meio da pesquisa o desenvolvimento interno e constante
crescimento.*

AGRADECIMENTOS

Meus agradecimentos primeiramente à Deus, por ter me dado saúde e persistência para a conclusão de mais este projeto na minha vida.

Aos meus pais, João e Sandra, pelo seus incentivos, carinho e educação que fizeram chegar até aonde eu cheguei hoje.

À minha irmã, Bruna, que, embora distante, sempre me incentivou com exemplo de dedicação aos meus estudos.

À minha namorada, Ana Paula, que com seu apoio, dedicação e valores, me faz ser uma pessoa melhor a cada dia com o seu exemplo.

Ao corpo docente da Escola de Engenharia de São Carlos pelo profissionalismo demonstrado na transmissão de conhecimentos durante as aulas e demais atividades curriculares.

Ao meu orientador Marcelo Becker, expresso o meu reconhecimento e admiração pela sua competência profissional e minha gratidão pela paciência na condução das orientações para a conclusão deste trabalho.

E por fim, agradecer aos demais colegas por terem participado da minha caminhada construindo nosso conhecimento e por permitirem que eu também fizesse parte das suas caminhadas demonstrando assim serem pessoas especiais que, por diversas vezes, me mostraram que o companheirismo faz a nossa vida valer mais a pena.

*A felicidade é a pretensão ilusória de converter
um instante de alegria em eternidade.
Clóvis de Barros Filho*

RESUMO

LOVATTO, A. **Controle de navegação de um robô autônomo móvel**. 2021. 66p. Monografia (Trabalho de Conclusão de Curso) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2021.

Nas duas últimas competições presenciais da *Latin American and Brazilian Robotics Competition* na categoria Instituto de Engenheiros Eletrônicos e Eletricistas (IEEE) na modalidade *Open* nos anos de 2018 e 2019, o grupo Soluções em Engenharia Mecatrônica e Aplicação a Robótica (SEMEAR) da Universidade de São Paulo no campus de São Carlos teve dificuldades ao executar as rotinas de trajetória do robô autônomo da equipe. Desta forma, neste trabalho, buscou-se explorar os conceitos acerca desta problemática, uma vez que o robô deslocava-se realizando curvas, a partir dos quais foi possível compreender que o problema era causado devido ao desequilíbrio da distribuição nas forças de apoio do robô. Tal realidade afeta diretamente na velocidade. Logo, este trabalho foi uma sugestão de controle digital para corrigir a trajetória de um robô móvel autônomo, o qual foi modelado em diagramas de blocos que representaram fielmente a realidade encontrada na produção de um robô, identificando cada peça do *hardware* utilizado.

Palavras-chave: Controle Digital, Robótica, Modelagem, Motor, Discretização, Autônomo, Encoder, Simulink, Proporcional, Integrativo, Derivativo

ABSTRACT

LOVATTO, A. **Navigation control of an autonomous mobile robot**. 2021. 66p. Monograph (Conclusion Course Paper) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2021.

In the last two presential competitions of Latin American and Brazilian Robotics Competition in the category Institute of Electronic Engineers and Electricians (IEEE) in the Open modality in 2018 and 2019, the group Solutions in Mechatronics Engineering and Application to Robotics (SEMEAR) from University of São Paulo, campus of São Carlos, had difficulties executing the trajectory routines of the team's autonomous robot. Thus, in this work, we sought to explore the concepts about this problem, as it was occurring, as the robot moved around performing curves, from which it was possible to understand that the problem was caused due to an imbalance in the distribution on the supporting forces of the robot. Such reality directly affects speed. Therefore, the work involved in implementing a suggestion of digital control, in which block diagrams were modeled that faithfully represent the reality found in the production of a robot, identifying each piece of hardware used.

Keywords: Digital Control, Robotics, Modeling, Motor, Discretization, Autonomous, Encoder, Simulink, Proportional, Intergrative, Derivative

LISTA DE FIGURAS

Figura 1 – Modelos de rodas utilizadas em robôs móveis veiculares	24
Figura 2 – Representação eletromecânica do motor CC com ímã permanente	27
Figura 3 – Circuito elétrico do motor CC com ímã permanente	27
Figura 4 – Disposição do funcionamento de um sensor óptico	28
Figura 5 – Simplificação de um circuito PWM para geração de ondas quadradas	29
Figura 6 – Saída PWM com variação de ciclo de trabalho.	29
Figura 7 – Representação simplificada do circuito ponte H. (a) Configuração "H". (b) corrente da esquerda para a direita; (c) corrente da direita para a esquerda.	30
Figura 8 – Circuito da Ponte H com a utilização de MOSFETs como chaves eletrônicas.	31
Figura 9 – Possíveis combinações de movimentação de um robô com quatro <i>mecanum wheels</i>	32
Figura 10 – Esquemático geral para sistema de controle e programação de Robôs Móveis	32
Figura 11 – Representação do mapeamento ($S \rightarrow Z$) pela aproximação diferencial	34
Figura 12 – Diagrama de blocos de um sistema de controle digital utilizando a <i>RaspBerry Pi</i> como microprocessador	34
Figura 13 – Diagrama de blocos do controlador proporcional integrativo derivativo.	35
Figura 14 – Modelo 3D do robô realizado no SolidWorks	37
Figura 15 – Diagrama de corpo livre de um bloco em uma rampa	38
Figura 16 – Cenário da competição da IEEE Open de 2019	39
Figura 17 – Eixo coordenado com referencial no robô	39
Figura 18 – VGPlan - Cálculo da força em cada apoio	40
Figura 19 – Momento de Inércia de um cilindro maciço	43
Figura 20 – Sistema dinâmico do motor de corrente contínua	43
Figura 21 – Resposta da função de transferência do motor corrente contínua para entrada degrau unitário	44
Figura 22 – Representação esquemática do motor de corrente contínua com redução e acoplado a uma carga que possui força de atrito	45
Figura 23 – Diagrama de blocos do motor acoplado com a roda e com a força de atrito	46
Figura 24 – Diagrama de blocos do motor acoplado com a roda e com a força de atrito	46
Figura 25 – Comparação da resposta ao degrau de cada motor com seu respectivo peso no apoio	47
Figura 26 – Velocidades lineares x e y do robô e velocidade angular z	48
Figura 27 – Diagrama de blocos para calculo da trajetória do robô com referência o plano cartesiano	49
Figura 28 – Trajetória do robô com referencial no centro do robô	49
Figura 29 – Trajetória do robô com referencial no plano cartesiano	50

Figura 30 – Diagrama de blocos do sistema de controle utilizado no Simulink para aplicar as equações diferenças	51
Figura 31 – Resposta da função de transferência do motor contínua e discretizada para duas condições mecânicas.	53
Figura 32 – Método dos lugares das raízes aplicado para definir q_0 para que o sistema seja estável e não obtenha sobressinal	55
Figura 33 – Diagrama de blocos do sistema dinâmico completo e controlado com a finalidade de obter-se a trajetória do robô autônomo e suas velocidades lineares no eixo X e Y e angular em Z	57
Figura 34 – Ação de controle, erro e velocidade final do motor quando aplicado o controlador	59
Figura 35 – Comparação das velocidades de cada robô com e sem carga e discretizados	60
Figura 36 – Velocidade linear em cada eixo do robô e velocidade de rotação	61
Figura 37 – Posição linear em cada eixo do robô e posição angular do robô no tempo . .	61
Figura 38 – Trajetória do robô com e sem a implementação de um controlador	62

LISTA DE TABELAS

Tabela 1	–	Características construtivas do robô	37
Tabela 2	–	Força de reação no apoio de cada roda do robô em função ao	40
Tabela 3	–	Característica eletromecânica do motor do robô	41
Tabela 4	–	Índices de desempenho do motor CC calculados no MatLab	44
Tabela 5	–	Desempenho do motor <i>Front Left</i> acoplado com roda e carga.	47
Tabela 6	–	Escolha intervalo de amostragem de forma prática	51
Tabela 7	–	Código implementado no bloco function do Simulink	56

SUMÁRIO

1	INTRODUÇÃO	21
1.1	Motivação	21
1.2	Objetivos	21
1.3	Organização	21
2	FUNDAMENTAÇÃO TEÓRICA	23
2.1	O que é um robô?	23
2.2	Modelagem Dinâmica e Cinemática de Robôs Móveis	24
2.2.1	Cinemática	24
2.2.2	Dinâmica	25
2.3	Arquitetura de controle	25
2.3.1	Arquitetura de controle reativa	25
2.3.2	Arquitetura de controle deliberativa	25
2.3.3	Arquitetura de controle híbrida	26
2.4	Componentes necessários do robô	26
2.4.1	Motores CC escovados com ímãs	26
2.4.1.1	Encoder	28
2.4.2	Modulação por largura de pulso - PWM	28
2.4.3	Ponte H	30
2.4.4	Rodas Mecanum Wheels	31
2.5	Sistema de Controle	32
2.6	Controladores de Ordem Fixa	33
2.6.1	Controle Digital	33
2.6.2	Controladores PID	34
2.6.2.1	Ações de controle	35
3	METODOLOGIA	37
3.1	Definir a dinâmica do motor	38
3.2	Modelagem do sistema eletromecânico	41
3.3	Modelagem de sistema dinâmico do robô e sua trajetória	48
3.4	Definição de <i>Hardware</i>	50
3.5	Elaboração do controlador	53
4	RESULTADOS E DISCUSSÃO	59
5	CONCLUSÃO	63

REFERÊNCIAS 65

1 INTRODUÇÃO

1.1 Motivação

O grupo Soluções em Engenharia Mecatrônica e Aplicação a Robótica (SEMEAR) é uma equipe extracurricular da Universidade de São Paulo, no campus de São Carlos, que possui como missão o desenvolvimento de projetos inovadores em engenharia e robótica, promovendo o desenvolvimento individual e coletivo dos membros. O SEMEAR possui três núcleos de desenvolvimento na robótica: Robótica Móvel Controlada, Robótica Móvel Autônoma e a Robótica Aérea.

O núcleo de Robótica Móvel Autônoma possui expressivos resultados nos dois últimos anos sendo atuais bi-campeões em etapas simuladas da *Latin American and Brazilian Robotics Competition* (LARC). Entretanto, nos anos de 2018 e 2019, a equipe teve problemas recorrentes na locomoção do robô autônomo. Sem uma locomoção livre precisa torna-se difícil para o robô saber a sua localização durante a performance de um desafio, podendo até perder pontos por avançar em áreas proibidas do desafio.

Nesses robôs, a implementação de um sistema de controle tem o principal papel de reduzir o erro na locomoção, caso ele comece a desviar de sua trajetória planejada. Tendo em mente essa problemática, apresenta-se a seguir os objetivos desse trabalho.

1.2 Objetivos

- Objetivo Geral:

1. Analisar a influência de um robô descompensado¹ em sua trajetória.

- Objetivos Específicos:

1. Descrever o sistema físico do robô;
2. Definir os equipamentos eletrônicos para o hardware; e
3. Construir um controlador para corrigir a trajetória;

1.3 Organização

O estudo desenvolvido neste projeto foi distribuído nos seguintes capítulos:

No Capítulo 2, constam a fundamentação teórica sobre arquitetura de controle, descrição de funcionamento dos componentes que serão utilizados, modelagem cinemática e dinâmica e uma síntese de controle digital;

¹ Robô descompensado é aquele que o peso está distribuindo diferente em cada apoio.

No Capítulo 3, há a descrição da série de etapas dos procedimentos da escolha dos 2 pares de motor CC, cálculos de força de apoio nas rodas e aplicação do controle digital;

No Capítulo 4, são expostos os resultados finais gerados após a aplicação do controlador digital. Trata-se também da discussão acerca dos resultados atingidos.

No Capítulo 5, encontram-se as considerações finais sobre o estudo realizado neste projeto, sugerindo também as proposições de trabalhos futuros para este ramo de estudo.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 O que é um robô?

Antigamente a noção de robótica estava interligada a questão de máquinas eletromecânicas que conseguiam ajudar as pessoas a fazer suas atividades, como por exemplo uma máquina de costura, máquina de lavar roupa entre outros. Entretanto com o avanço da ciência e da tecnologia com o surgimento dos microprocessadores na década de 70, a utilização desses componentes nas máquinas transformou e sofisticou a noção da robótica.

Com a utilização de um hardware no qual conseguia elaborar uma programação a ser seguida, a noção de robô passou a incluir pensamento, raciocínio, resoluções de problemas e consistências, os robôs passaram a ser representações mais próximas de seres vivos como insetos, animais e seres humanos. Nesse sentido, os robôs são sistemas autônomos no qual existem um mundo físico que consegue sentir seu ambiente e agir sobre ele para alcançar seus objetivos (MATARIC, 2007).

Existem basicamente quatro classificações para descrever um robô: tipo de modalidade, tipo de mecanismo de locomoção, tipo de local de atuação e tipo de autonomia.

1. Tipo de autonomia:

- Robôs com ações autônomas, isto é, aqueles tomam as decisões sem nenhuma intervenção humana;
- Robôs com ações semi-teleoperadas que são aqueles que executam ações autônomas após um comando enviado pelo seu operador; e
- Robôs com ações teleoperadas os quais são utilizados algum dispositivo de comunicação comanda as ações.

2. Tipo de local de atuação:

- interno: locais fechados e internos
- externo: estruturados (estradas) ou não-estruturados.

3. Tipo de mecanismo de locomoção: o robô pode utilizar pernas, rodas, esteiras, propulsão e hélices para se locomover.

4. Tipo de mobilidade:

- Base fixa: robôs que não possuem locomoção, como os robôs manipuladores ou braços robóticos
- Base móvel: com restrição (grua) e sem restrição (veículo)

No sistema de locomoção existem ainda dois tipos de rodas que vale a pena se atentar, que são:

- Tradicionais: as rodas tradicionais são aquelas encontradas em carros, conforme apresentado na Figura 1a.
- Suecas: as rodas suecas possuem em sua estrutura rolamentos que permitem que o robô possa se movimentar na combinação dos vetores x e y em relação ao centro geométrico, de acordo com a Figura 1b.

Figura 1 – Modelos de rodas utilizadas em robôs móveis veiculares

(a) Rodas tradicionais



Fonte: Extraída de (ROBÓTICA, 2021)

(b) Rodas suecas: com enrolamento a 45°



Fonte: Extraída de (INDIA, 2021)

Visto que um robô precisa sentir seu ambiente, são colocados sensores que servem para retornar algumas percepções - o toque, a visão, o ouvir e o cheiro. Entretanto, o robô não precisa necessariamente ter sensores para todos os sentidos humanos, geralmente a equipe que desenvolve utiliza apenas aqueles necessários para executar seus objetivos. Portanto, quando o robô percebe as situações pré-programadas, ele sempre executa uma tomada de decisão referente a aquela situação e com isso o robô pretende alcançar os objetivos de origem. (MATARIC, 2007).

2.2 Modelagem Dinâmica e Cinemática de Robôs Móveis

A cinemática e a dinâmica são dois pontos fundamentais na análise do movimento dos robôs. A aplicação em conjunto das mesmas apresenta resultados satisfatórios, tanto na análise do comportamento do robô quanto na análise de controle (MIRANDA, 2017).

2.2.1 Cinemática

A cinemática está relacionada à correspondência entre o movimento dos atuadores com a movimentação final do robô. Desse modo, a cinemática descreve a trajetória da movimentação do robô pelas variáveis de posição, velocidade e aceleração. A cinemática clássica aborda tanto o estudo de partículas quanto de um corpo rígido, ambas possuem como foco na análise das posições, velocidades e aceleração em função do tempo. Para o caso de estudo de robôs móveis,

pelo fato das suas dimensões não serem desprezíveis na análise, será abordado a cinemática de corpos rígidos. Neste caso, será estudado a movimentação de todas as partículas do robô de tal forma que a distância entre elas são fixas. Na movimentação de um corpo rígido, possui duas análises diferentes de deslocamento: translação e rotação.

- A translação ocorre quando todas as partículas do corpo rígido possuem em sua análise em cinemática vetorial a mesma magnitude, direção e sentido. Ou seja, todas as partículas do corpo possuem velocidades e acelerações na mesma direção com a mesma magnitude.
- Já a rotação possibilita que o corpo gire em torno do seu eixo de rotação, o que caracteriza que as partículas possuem variáveis lineares diferentes para cada partícula que varia conforme se afasta do eixo de rotação.

2.2.2 Dinâmica

A dinâmica se refere às propriedades de movimentação e energia do robô. Nesse ponto será estudado o movimento, velocidade, aceleração do robô na ótica das Leis de Newton, desse modo será considerado resolver os problemas relacionados a força e momento envolvidos (MIRANDA, 2017).

2.3 Arquitetura de controle

Uma arquitetura de software é o conjunto de ações que o microprocessador irá executar conforme o *loop* do sistema embarcado, desta forma os sensores enviarão dados para o microprocessador que irá interpretar as informações recebidas e decidirá uma ação futura. Os *hardwares* localizados nas portas programáveis (entrada e saída) são reconhecidos pelo microprocessador do robô. Desse modo, o microcontrolador é responsável por perceber e agir sobre o ambiente. A arquitetura de controle determinará o comportamento do robô. Dentre as arquiteturas destacam-se como principais: reativa, deliberativa e híbrida (KLIPP, 2013).

2.3.1 Arquitetura de controle reativa

Nesse tipo de arquitetura o robô obedece um sistema de ação e reação, ou seja os sensores enviam as informações e após a leitura dos sinais o microprocessador acaba decidindo qual será a próxima ação a ser tomada. Não há necessidade de armazenar os dados para que o robô utilize para ações futuras e nem um mapeamento do ambiente, entretanto precisa-se de uma constante captação de informações dos sensores em tempo real suficiente para o robô conseguir definir a sua próxima ação (KLIPP, 2013).

2.3.2 Arquitetura de controle deliberativa

A deliberação possui um estágio intermediário de planejamento, baseado em um modelo do mundo real onde o robô interage. Para isso é necessário obter informações do ambiente e das

ações já realizadas e dependendo da programação até os estados anteriores em que o robô estava (KLIPP, 2013). Dessa forma, é interessante que o robô tenha pré-processado o ambiente em sua memória e executar as ações conforme o planejamento. Nesses modelos é comum encontrar redes neurais e inteligências artificiais que são as responsáveis em planejar e executar a seleção de comportamento (KLIPP, 2013).

2.3.3 Arquitetura de controle híbrida

A arquitetura híbrida é a intersecção dos dois modelos anteriores, entretanto há na elaboração do projeto algumas estruturas de interrupções. Desse modo, o robô tem como arquitetura principal a deliberativa em que ele vai tomando as decisões conforme o planejamento de dados pré-processados e de tomadas de decisões anteriores, entretanto quando algum sensor que esteja conectado em alguma porta de interrupção encaminha alguma informação, o robô para de executar o que já tinha sido programado e executa uma programação reativa até que esse pino de interrupção pare de receber a informação, voltando para as ações planejadas (KLIPP, 2013).

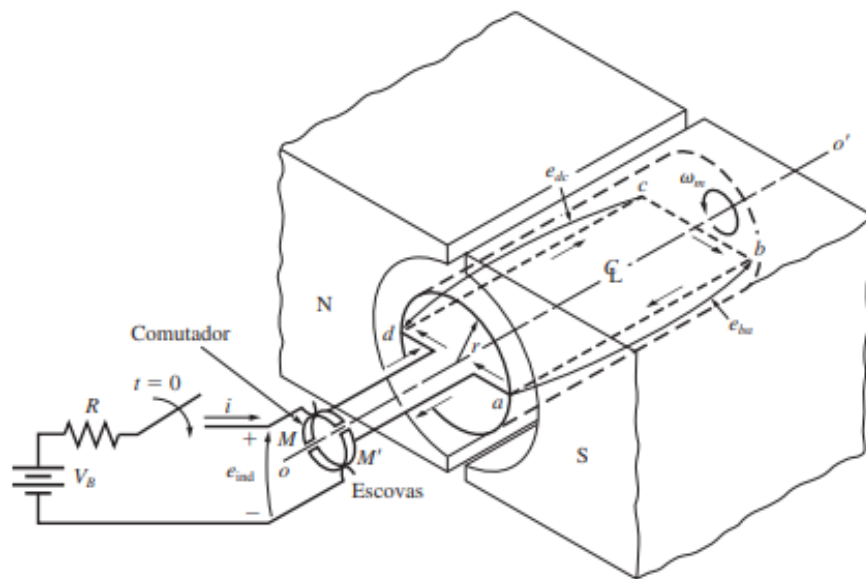
2.4 Componentes necessários do robô

2.4.1 Motores CC escovados com ímãs

O motor elétrico é um dos principais atuadores da robótica por sua capacidade de transformar energia elétrica em energia mecânica (torque e velocidade). O motor de corrente contínua (CC) é amplamente utilizado na robótica devido ao seu controle de velocidade e torque, precisão e simplicidade de acionamento, além de possuir variedade de modelos disponíveis no mercado e ser menores do que outros modelos de motores. Os utilizados neste trabalho são motores CC com ímãs permanentes, compostos por duas partes: uma fixa chamada de estator, na qual estão localizados os dois ímãs, o comutador e os apoios do eixo; e outra móvel girante, denominada rotor, onde fica o enrolamento de armadura e o eixo do motor. Este componente é mostrado na Figura 2.

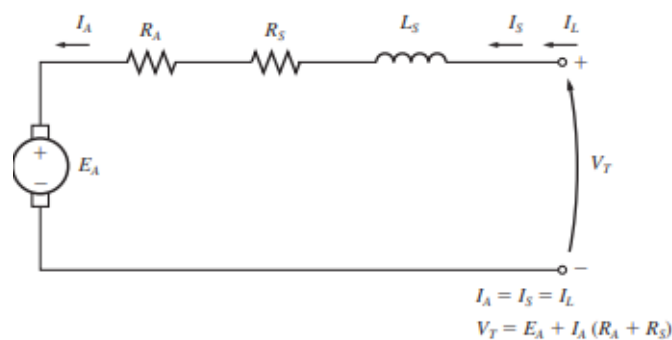
Desse modo, o controle de torque e velocidade do motor fica a cargo da corrente de armadura, o enrolamento presente no rotor. A fim de modelarmos o comportamento do motor a partir da tensão aplicada, o circuito equivalente é apresentado na Figura 3.

Figura 2 – Representação eletromecânica do motor CC com ímã permanente



Fonte: Extraída de (CHAPMAN, 2013)

Figura 3 – Circuito elétrico do motor CC com ímã permanente



Fonte: Extraída de (CHAPMAN, 2013)

A partir do circuito equivalente e utilizando a Lei de Kirchoff das tensões, obtém-se a seguinte Equação 2.1:

$$v_a(t) = R_a i_a(t) + L_a \frac{d}{dt} i_a(t) + K_e \omega(t) \quad (2.1)$$

A partir da Equação 2.1 observa-se que quanto maior a velocidade angular do motor $\omega(t)$, maior a força eletromotriz $e(t)$ produzida e menor a corrente $i_a(t)$ consumida, portanto será visto um maior consumo de corrente do motor com eixo travado ou durante a partida. Sem entrar nos detalhes mecânicos, sabe-se que o torque do motor é proporcional a corrente de armadura, como

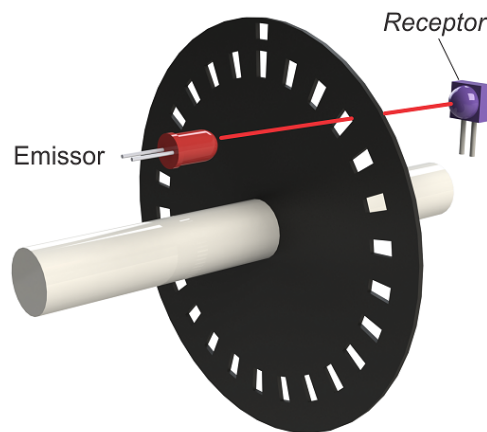
pode ser visto na Equação 2.2

$$\tau(t) = K_t i_a(t) \quad (2.2)$$

2.4.1.1 Encoder

Os *encoders* são sensores eletromecânicos que possuem a função de transformar a posição do eixo em sinal elétrico. Portanto, com a utilização dos *encoders* é possível verificar a posição do eixo do motor, controlar a velocidade, medir a angulação do motor e o número de rotações.

Figura 4 – Disposição do funcionamento de um sensor óptico



Fonte: Extraída de (ALMEIDA, 2020)

O *encoder* é composto por um disco com marcações, um componente emissor e um receptor. As marcações que tem no disco possuem a funcionalidade de bloquear ou não o feixe de luz do emissor para o receptor, desta forma na saída do *encoder* será lido ondas quadradas, proporcional ao número de marcações presentes no disco do dispositivo, que equivale a quantidade de ondas quadradas, ou *clock*, gerado em uma volta do *encoder*. Logo, a frequência de rotação retornado do encoder será utilizada para calcular a velocidade angular do motor em *rad/s* pela Equação 2.3.

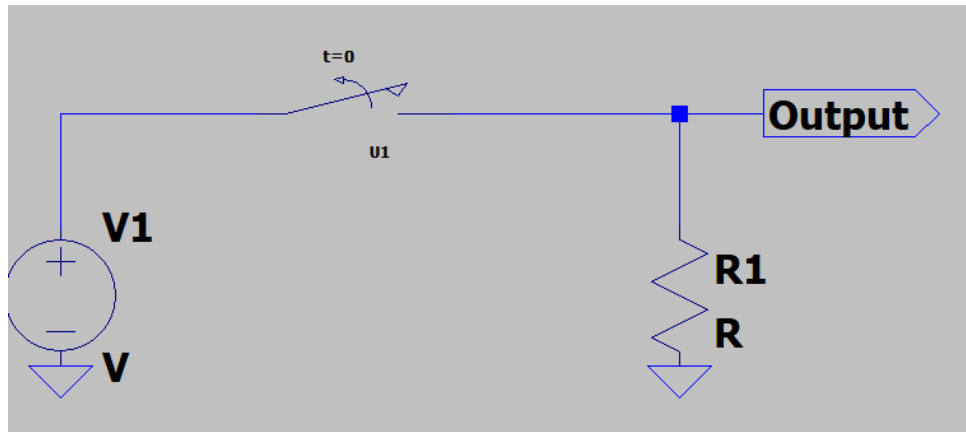
$$\omega_m = \frac{f_e}{n_r} \quad (2.3)$$

2.4.2 Modulação por largura de pulso - PWM

Uma forma de transformar um sinal digital em um sinal analógico é usando um circuito de modulação por largura de pulso.

Para melhor entendimento da modulação por largura de pulso, considere o circuito da Figura 5 com um sinal de saída de sempre 100ms. O controle dos tempos (*on/off*) efetivamente controla o ciclo de trabalho do sinal de saída, desse modo tem-se a relação do ciclo de trabalho

Figura 5 – Simplificação de um circuito PWM para geração de ondas quadradas



Fonte: própria do autor

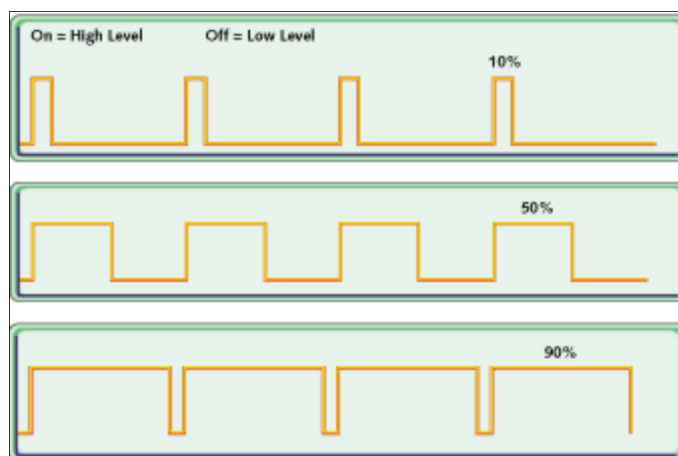
na Equação 2.4.

$$\delta = \frac{T_{on}}{T_{total}} \quad (2.4)$$

A saída do módulo PWM é dada, então, por uma onda quadrada com ciclo de trabalho dependente da tensão de entrada, desse modo, tem-se a relação de entrada e saída do sinal PWM na Equação 2.5.

$$V_{saida} = \delta \cdot V_{entrada} = \frac{T_{on}}{T_{Total}} V_{entrada} \quad (2.5)$$

Figura 6 – Saída PWM com variação de ciclo de trabalho.



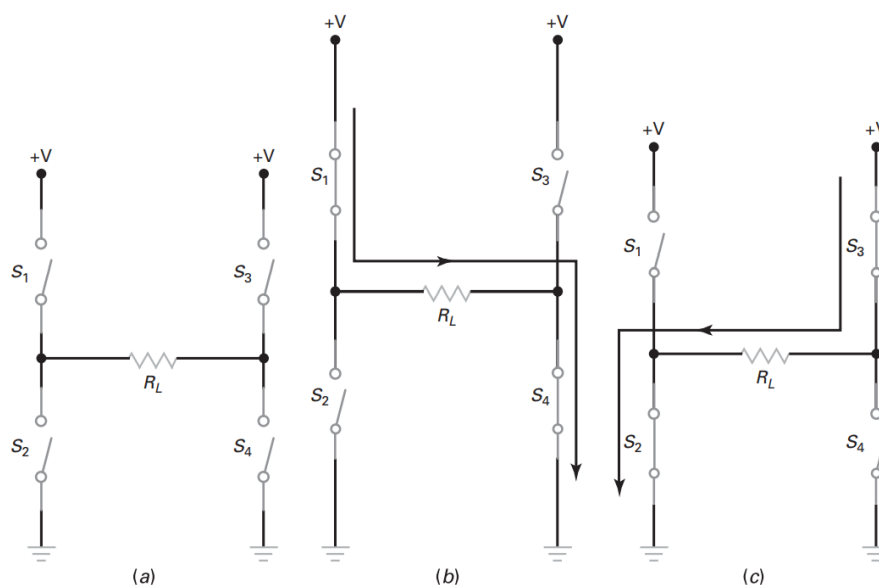
Fonte: Extraída de (BARR,)

A saída do sinal pode ser visto na Figura 6.

2.4.3 Ponte H

Um circuito de ponte H simplificado é composto por quatro chaves eletrônicas. Duas chaves são conectadas em cada lado com a carga situada entre as junções centrais dos dois lados, conforme mostrado na Figura 7. Por meio do controle individual dos comutadores, a corrente através da carga pode variar nos dois sentidos e em intensidade.

Figura 7 – Representação simplificada do circuito ponte H. (a) Configuração "H". (b) corrente da esquerda para a direita; (c) corrente da direita para a esquerda.

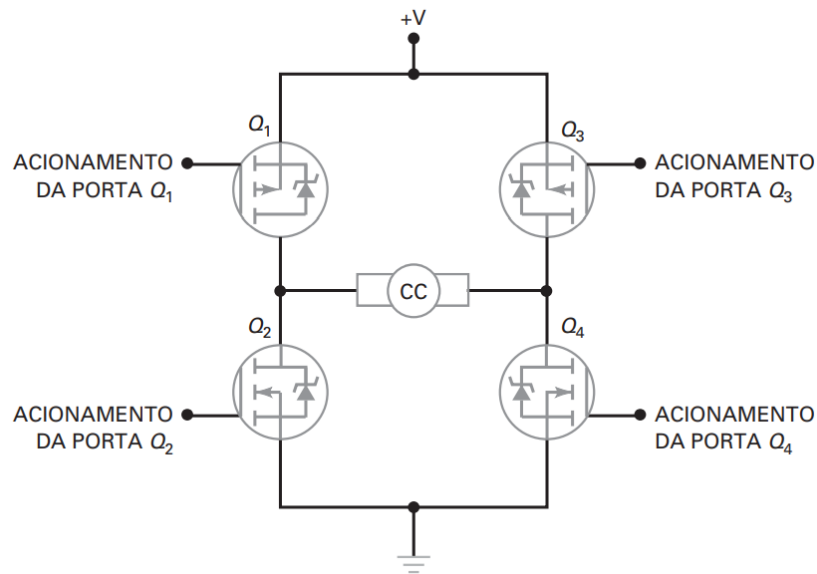


Fonte: Extraída de (MALVINO; BATES, 2016)

Na Figura 7(b), os comutadores S_1 e S_4 estão fechados. Isso faz com que a corrente circule através da carga da esquerda para a direita. Abrindo S_1 e S_4 , junto com o fechamento de S_2 e S_3 , a corrente circula através da carga no sentido oposto, como mostrado na Figura 7(c). A intensidade da corrente de carga pode ser alterada ajustando o nível de tensão $+V$ aplicado ou, de preferência, controlando o tempo de liga/desliga (*on/off*) nos diversos comutadores. Se um par de comutadores for fechado (*on*) metade do tempo e aberto (*off*) a outra metade do tempo, isso produziria um ciclo de trabalho de 50% resultando em uma corrente de carga metade da corrente normal total. O controle dos tempos *on/off* efetivamente controla o ciclo de trabalho dos comutadores e é conhecido como modulação por largura de pulso (PWM – pulse width modulation). Deve-se ter o cuidado de não fechar simultaneamente os comutadores de um mesmo lado da ponte. Por exemplo, se S_1 e S_2 forem fechados, isso resultaria em uma alta corrente fluindo através das chaves de $+V$ para o terra. Essa corrente de curto poderia danificar os comutadores ou a fonte de alimentação.

Geralmente os circuitos integrados utilizam MOSFET's nos lugares das chaves, como mostra a Figura 8. Além disso, nos circuitos integrados fazem uma lógica digital para que a porta

Figura 8 – Circuito da Ponte H com a utilização de MOSFETs como chaves eletrônicas.



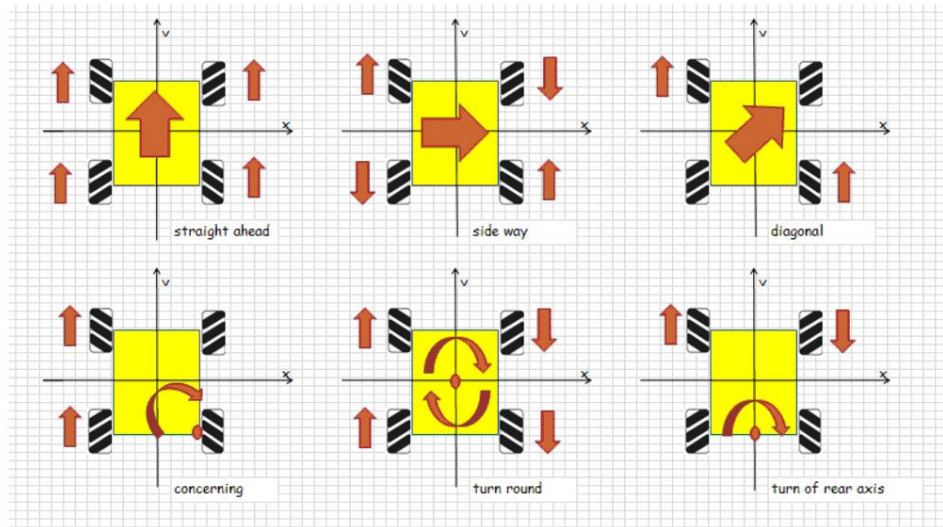
Fonte: Extraída de (MALVINO; BATES, 2016)

PWM da chave S_1 e S_4 sejam a mesma, o que acontece também para as chaves S_2 e S_3 , desse modo não há perigo da corrente passar diretamente do $+V$ para o terra do circuito.

2.4.4 Rodas Mecanum Wheels

Em busca da maior versatilidade na locomoção do robô, foram empregadas rodas *me-canum wheels* que possuem rolamentos dispostos a 45° . Essa característica faz com que a combinação quatro rodas que permite ao robô movimentação omnidirecional, trazendo os principais benefícios de permitir a rotação em torno do próprio eixo e movimentação lateral, como é mostrado na Figura 9.

Figura 9 – Possíveis combinações de movimentação de um robô com quatro *mecanum wheels*.

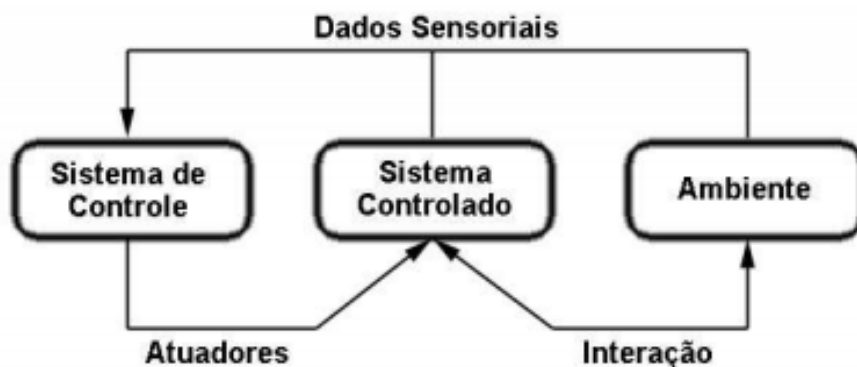


Fonte: Extraída de (US, 2020)

2.5 Sistema de Controle

Para definir o que é um sistema de controle, precisamos definir quais são os elementos que interagem com o sistema de controle. Ligado diretamente ao sistema de controle está o sistema controlado, que é o sistema que desejamos controlar. O ambiente não é, ou não pode ser, totalmente controlado diretamente pelo sistema de controle, mas interfere significativamente em seu funcionamento (HEISEN, 2002).

Figura 10 – Esquemático geral para sistema de controle e programação de Robôs Móveis



Fonte: Extraído de (HEISEN, 2002)

A tarefa de controle do sistema é fazer com que todo o sistema alcance um determinado estado. Alcançar este estado pode envolver ou depender de mudanças que ocorrem no ambiente,

no sistema controlado ou devido a interação entre os dois. Logo um sistema de controle é um processo que pode utilizar seus sensores para obter informações sobre o sistema controlado e sobre o ambiente. Ele pode utilizar este conhecimento para controlar seus atuadores fazendo com que todo o sistema alcance um determinado estado (HEISEN, 2002).

2.6 Controladores de Ordem Fixa

Grande parte dos sistemas de controle industriais operam com controladores de ordem fixa do tipo Proporcional-Integrativo (PI) e Proporcional-Integrativo-Derivativo (PID). O projeto de controladores envolve definir as especificações desejadas para a aplicação e a classe de controladores a ser executada. As especificações em geral são dadas no domínio do tempo ou da frequência. No primeiro caso, as especificações desejadas podem ser definidas em termos do máximo sobressinal ou coeficiente de amortecimento, do tempo de acomodação para uma entrada degrau e do erro de regime, as quais podem ser descritas em termos de polos e zeros do sistema de malha fechada. Pode-se, portanto utilizar o método do lugar das raízes para determinar a função de transferência do controlador para obter a configuração de polos e zeros desejada. No segundo caso, as especificações podem ser em termos de largura de banda, frequência de ressonância e margens de estabilidade (OLIVEIRA; AGUIAR; VARGAS, 2016).

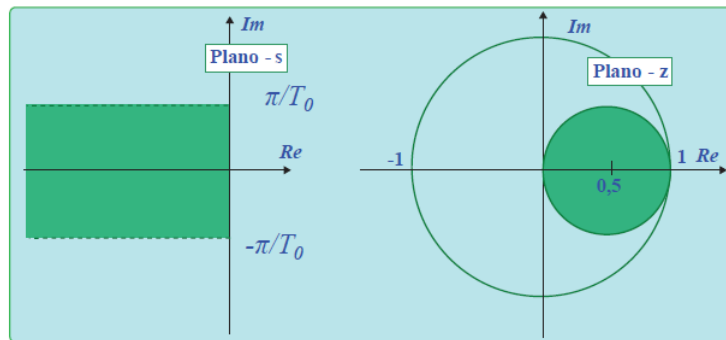
2.6.1 Controle Digital

No meio da robótica, dificilmente é encontrado um robô móvel utilizando controle analógico, uma vez que é complexo de modificar a referência desejada, o que não ocorre em um controlador digital, uma vez que a referência do sistema pode ser declarada como uma variável do código. Porém, ao projetar um controle digital existe mais medidas que precisam ser consideradas como a taxa de amostragem do conversor analógico-digital da porta de leitura dos sensores. Para realizar a transformada Z tendo como ponto de partida um sistema laplaciano, é feito aproximações da Equação 2.6.

$$s \approx \frac{z - 1}{T_0 z} \quad (2.6)$$

Esta proposta de discretização restringe, portanto a área de mapeamento dos polos/zeros do sistema contínuo do plano-Z. Este procedimento só será condizente com o caso exato para elevadas frequências de amostragem. Polos ou zeros que estariam no círculo unitário, Figura 11, mas fora da região mapeada seriam portanto mal representados e a discretização provocaria uma distorção no caso contínua.

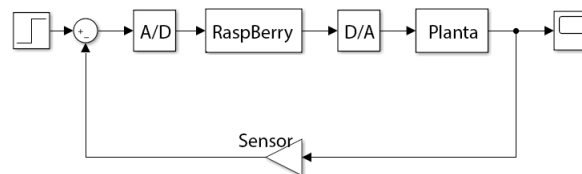
Figura 11 – Representação do mapeamento ($S \rightarrow Z$) pela aproximação diferencial



Fonte: Extraído de (OLIVEIRA; AGUIAR; VARGAS, 2017)

Em um sistema com controle digital, o microprocessador irá fazer a função de conversor analógico-digital da interpretação da leitura dos sensores e irá fornecer a ação de controle utilizando um conversor digital-analógico, como pode ser visto na Figura 12.

Figura 12 – Diagrama de blocos de um sistema de controle digital utilizando a *RaspBerry Pi* como microprocessador



Fonte: Adaptado de (OLIVEIRA; AGUIAR; VARGAS, 2017)

Portanto, o microprocessador será o responsável em discretizar a informação analógica conforme a taxa de amostragem de sua porta analógica. Logo, a escolha do microprocessador para realizar o controle do sistema pode ser considerado um dos gargalos do problema e por isso o projetista deve se atentar à sua escolha.

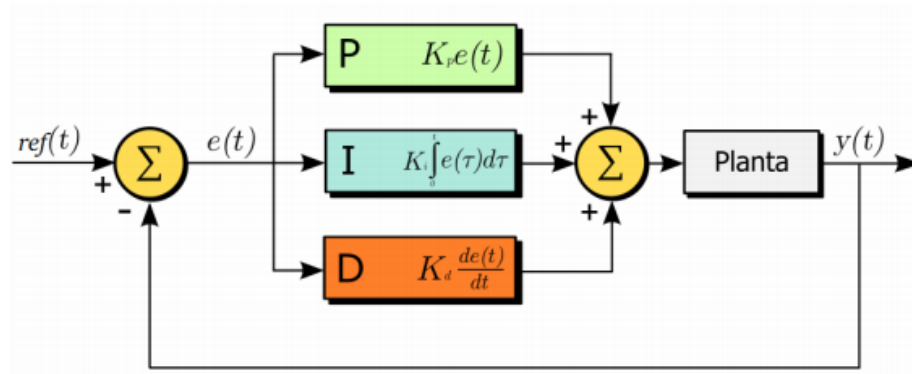
No capítulo 7 da obra de (OLIVEIRA; AGUIAR; VARGAS, 2017), é demonstrado como realizar a conversão de transformada Z para Equação diferença, informação relevante para o entendimento deste trabalho. Além das demais considerações sobre estabilidade do sistema digital.

2.6.2 Controladores PID

O controlador largamente utilizado em processos industriais é o controlador PID. A sua popularidade deve-se ao seu desempenho robusto em uma ampla gama de condições de funcionamento e em parte à sua simplicidade funcional, que permite aos engenheiros operá-los de uma forma simples e direta (MIRANDA, 2017). A técnica de controle PID (Figura 13) consiste em calcular um valor de atuação sobre o processo (planta) a partir das informações do valor desejado (sinal de referência) e do valor atual da saída do processo (realimentação). Este

valor de atuação sobre o processo é transformado em um sinal adequado ao atuador utilizado (exemplo um motor, um relé, etc), e deve garantir um controle estável e preciso. O controlador PID é composto por três coeficientes: proporcional, integral e derivativo, que são escolhidos (encontrados) de tal forma que o sistema apresente uma resposta ideal (esperada) (MIRANDA, 2017).

Figura 13 – Diagrama de blocos do controlador proporcional integrativo derivativo.



Fonte: Extraída de (OGATA, 2015)

2.6.2.1 Ações de controle

As ações de controle consideram que um sinal de controle será aplicado a uma planta a ser controlada. Este sinal de controle $U(s)$ é gerado pelo controlador tendo como entrada o sinal de erro $E(s)$. A Equação tradicional do controlador PID no domínio do tempo apresenta a relação matemática entre o sinal de erro e o sinal de controle e é dada pela Equação 2.7

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (2.7)$$

no qual $u(t)$ é a variável de controle e $e(t)$ é o erro de controle, ou seja, é a subtração da referência pela saída ($e = r - y$). A variável de controle $u(t)$ é portanto a soma de três termos: O termo P (que é proporcional ao erro), o termo I (que é proporcional à integral do erro) e o termo D (que é proporcional à derivada do erro). (MIRANDA, 2017).

- **Ação Proporcional:** é simplesmente proporcional ao erro de controle $e(t)$. Esse erro pode diminuir com o ganho, entretanto nunca é completamente zerado. Por outro lado quanto maior o ganho, mais oscilatório tende a ficar o comportamento transitório do sistema em malha fechada. Geralmente o aumento excessivo do ganho proporcional pode levar o sistema à instabilidade;
- **Ação Integral:** tem como principal função assegurar que a saída do processo $y(t)$ atinja o mesmo valor que o valor de referência $r(t)$ em regime estacionário, ou seja, assegura que o sistema tenha erro $e(t)$ igual a zero em regime estacionário.

- Ação Derivativa: tem como principal função o aumento da estabilidade do sistema em malha fechada.

Um problema a ser observado na introdução de um fator derivativo é o fato do mesmo ser um fator que aumenta o ganho do sistema com o aumento da frequência. Isso pode levar a uma amplificação de eventuais ruídos, normalmente de alta-frequências. A função de transferência do controlador PID é dado pela Equação 2.8. Os parâmetros do controlador K_p , K_i e K_d (ou K_p , T_i e T_d) são escolhidos em função do desempenho final esperado do sistema. A determinação desses parâmetros é chamada se sintonia do PID (MIRANDA, 2017).

$$\frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s \quad (2.8)$$

São objetivos da sintonia de um PID:

- Controle mais rápido: essencial em vários sistemas práticos. Porém muitas vezes conflita com a estabilidade e/ou o tempo de acomodação do sistema.
- Minimização do desgaste do equipamento: mais presente em sistemas mecânicos, deve-se evitar aplicar um sinal de controle fora do range de operação do componente. Ocorre principalmente pela saturação do controlador.
- Minimização da integral de erro: onde o objetivo é manter a área definida pelo sinal de entrada (*set-point*) e o sinal de saída (variável controlada) como mínima.
- Minimização da integral do erro quadrado: é possível que muito embora o erro definido pela área entre os sinais de entrada e saída seja aceitável o mesmo não ocorra para o instante inicial do transitório. Considerando-se que o erro instantâneo é multiplicado por uma constante obtém-se o erro quadrático.
- Minimizar os efeitos de perturbação: objetivo associado a capacidade do controlador em rejeitar perturbações que possam interferir no comportamento do sistema.
- Evitar sobressinal na partida do sistema: o objetivo é diminuí-lo quando o sistema é iniciado, em sistemas mecânicos e hidráulicos este pode danificar a planta.

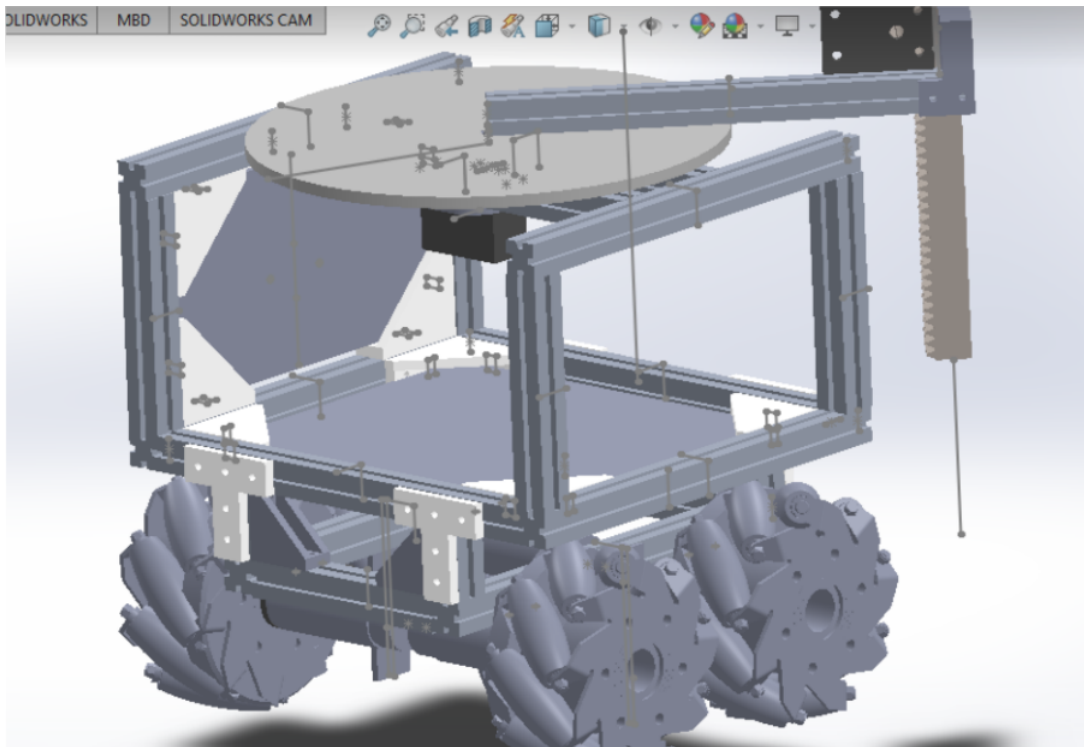
Nem sempre é possível atender a todos esses objetivos, sendo as vezes conflitantes. Logo devemos encontrar uma solução que melhor atenda as especificações do sistema (MIRANDA, 2017).

3 METODOLOGIA

Neste trabalho foi considerado o modelo de um veículo móvel com um guindaste, o qual fará com que o robô tenha o seu centro de massa deslocado em relação ao centro geométrico do mesmo. Esse robô não possui intervenção humana para executar as suas tarefas que são executadas em ambientes fechados. Embora sua lógica de programação é estruturada de forma reativa, não obtendo nenhuma informação do cenário em que vai atuar, o código de navegação do robô deverá funcionar em *thread*, desse modo a rotina principal do robô não será interrompida para realizar a leitura de sensores.

As dimensões do robô modelo, representado na Figura 14, e o peso estão descrito na Tabela 1.

Figura 14 – Modelo 3D do robô realizado no SolidWorks



Fonte: Própria.

Tabela 1 – Características construtivas do robô

Largura (m)	0,23
Comprimento (m)	0,21
Altura (m)	0,23
Peso do robô (kg)	4
Peso da cremalheira (kg)	0,5

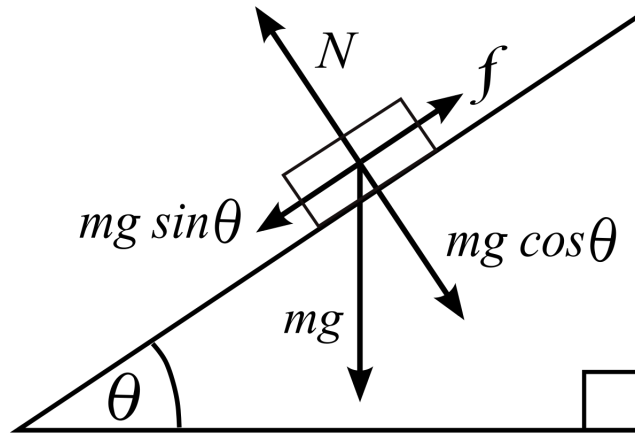
Fonte: Própria

3.1 Definir a dinâmica do motor

Uma das etapas primordiais na construção de um robô móvel é a escolha de motores para sua navegação, uma vez que são peças raramente trocadas devido ao seu alto custo. Para definir o motor, primeiramente é necessário calcular o torque mínimo no eixo do motor para que ele consiga se locomover.

Então, antes de começar a procura pelo motor ideal para o robô, é preciso definir o cenário em que esse robô será exposto. Ou seja, se esse robô será projetado para ambiente arenoso ou em um chão liso; rampa inclinada ou apenas em áreas planas.

Figura 15 – Diagrama de corpo livre de um bloco em uma rampa



Fonte: Wikipédia (2020)

A Figura 15 representa um corpo em plano com inclinação em θ . Dessa forma, temos em Y:

$$P_y - N = 0 \quad (3.1)$$

e em X:

$$F_o - P_x - F_{at} = F_r \quad (3.2)$$

sendo:

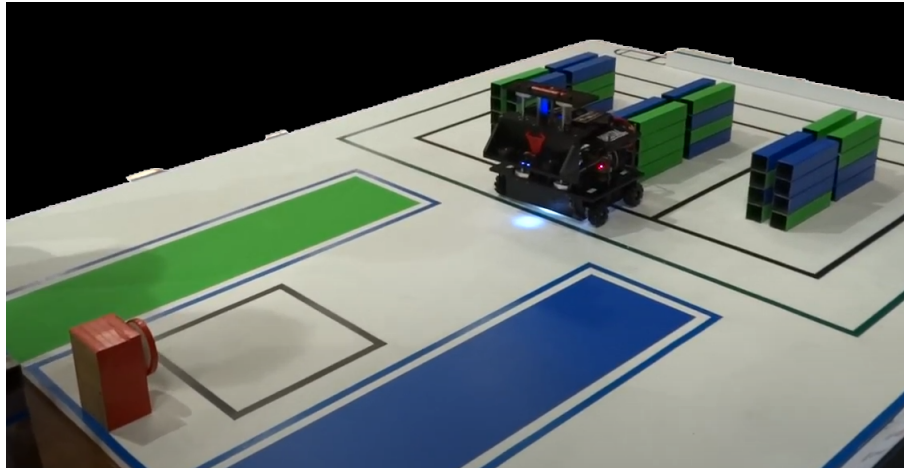
$$P_x = P \sin(\theta) \quad (3.3)$$

$$P_y = P \cos(\theta) \quad (3.4)$$

$$F_{at} = P_y \mu \quad (3.5)$$

Entretanto, neste trabalho será abordado o cenário da competição da IEEE Open da *Latin American Robotics Competition* que é plano ($\theta = 0$) e será considerado para estimar os motores que $\mu = 1$.

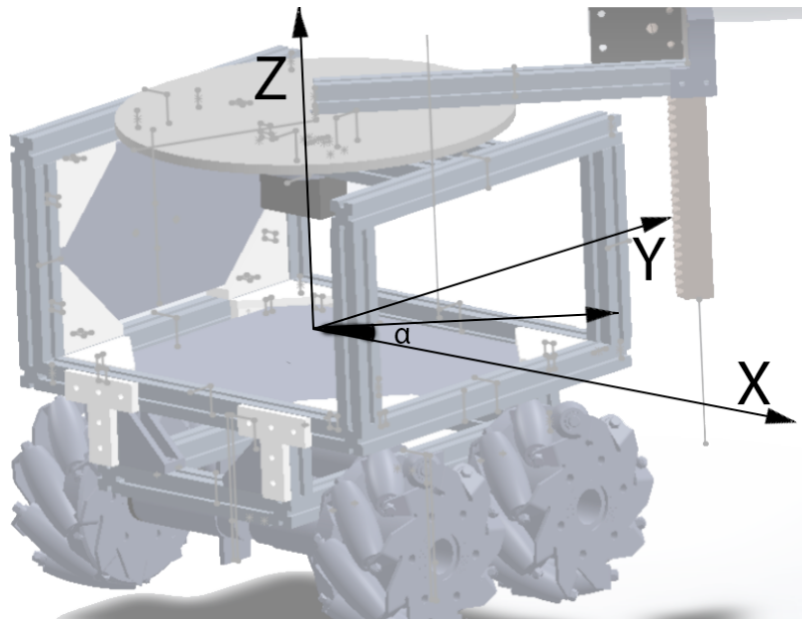
Figura 16 – Cenário da competição da IEEE Open de 2019



Fonte: Própria.

Nessas condições, o peso do robô, da Figura 14, é distribuído nas quatro rodas de apoio e varia conforme a posição de rotação da cremalheira, conforme é mostrado na Figura 17. Desse modo, foi utilizado o *software VGPlan* para calcular a força de reação das rodas do robô.

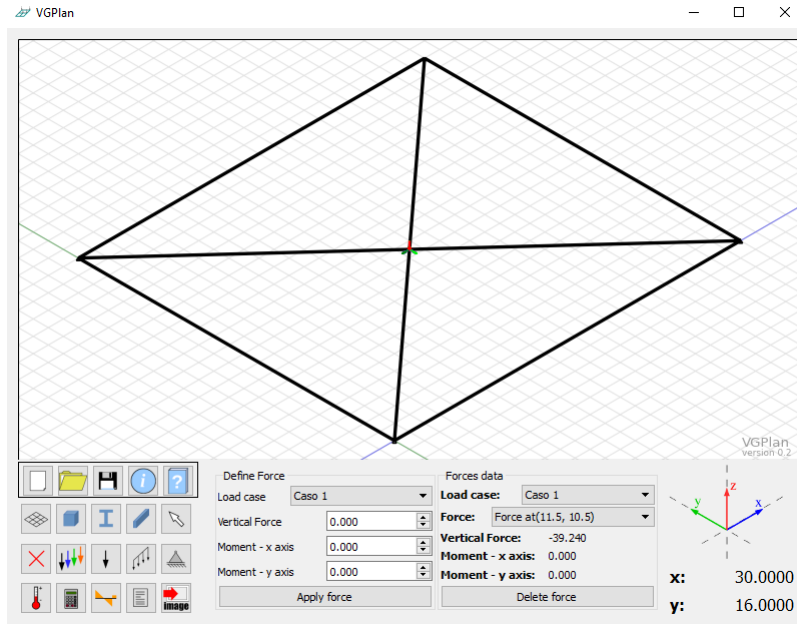
Figura 17 – Eixo coordenado com referencial no robô



Fonte: Própria.

Na utilização do VGPlan, Figura 18, são definidos os apoios do robô, seu centro de massa e o posicionamento da cremalheira em função de α , conforme as equações 3.6 e 3.7. O programa retorna para o usuário a força em cada roda do motor, que está descrito na Tabela 2.

Figura 18 – VGPlan - Cálculo da força em cada apoio



Fonte: Própria.

$$x_{cr} = cg_x + |cr| \cos(\alpha) \quad (3.6)$$

$$y_{cr} = cg_y + |cr| \sin(\alpha) \quad (3.7)$$

Sendo cr e cg a posição da cremalheira e o centro de gravidade do robô, respectivamente em x e y .

Tabela 2 – Força de reação no apoio de cada roda do robô em função ao

Caso	F_c	α (°)	R_a (N)	R_b (N)	R_c (N)	R_d (N)
1	4,905	0	9,22	12,85	12,85	9,22
2	4,905	15	8,77	12,27	13,30	9,80
3	4,905	30	8,47	11,61	13,60	10,46
4	4,905	45	8,35	10,91	13,72	11,16
5	4,905	60	8,41	10,22	13,66	11,85
6	4,905	75	8,65	9,59	13,42	12,48
7	4,905	90	9,05	9,05	13,03	13,03
8	4,905	105	9,59	8,65	12,49	13,42
9	4,905	120	10,22	8,41	11,85	13,66
10	4,905	135	10,91	8,35	11,16	13,72
11	4,905	150	11,61	8,47	10,46	13,60
12	4,905	165	12,27	8,77	9,80	13,30
13	4,905	180	12,85	9,22	9,22	12,85

Fonte: Própria

Nesse sistema, tem-se a maior força na roda R_c quando a cremalheira está em 45° , portanto será usado $13,72N$ para prosseguir o projeto.

Pela Equação 3.8, sabe-se a relação entre força e torque, o que pode ser aplicado e descobrir o torque mínimo do motor para locomover-se.

$$\tau_{eixo} = F x_{roda} \quad (3.8)$$

Observa-se pela 3.8 que a escolha da roda passa também pela definição do motor, uma vez que o raio da roda e o torque do motor possuem uma relação diretamente proporcional. Como o robô usado pelo grupo SEMEAR - EESC/USP na competição de 2019 foram quatro *mecanum wheels* de $100mm$ de diâmetro, ilustrado na Figura 17, será usado o seu raio para calcular o torque no eixo do motor.

$$\tau_{eixo} = 13,72 \times 0,05 \quad (3.9)$$

$$\tau_{eixo} = 0,686Nm \quad (3.10)$$

Portanto, o motor do robô deverá ter um torque maior do que $0,686N.m$ ou $0,07kgf.m$. A partir dessa informação, foi definido o motor para esse projeto, descrito na Tabela 3 as suas características.

Tabela 3 – Característica eletromecânica do motor do robô

Peso	88g
Redução	46,85:1
Tensão	12V
Indutância	1mH
Velocidade sem carga	220rpm
Corrente sem carga	0,3A
Corrente de Stall	5A
Torque de Stall	15kg.cm
Potência máxima	8,4W

Fonte: *Gearmotors* no site da Pololu (2019)

3.2 Modelagem do sistema eletromecânico

Para obter a melhor desempenho do sistema de locomoção, é importante saber o comportamento eletromecânico do modelo escolhido. Para isso, foi usado um estudo analítico das equações do motor com as Equações 3.11, 3.12 e 3.13.

$$V_m = Ri(t) + L \frac{di(t)}{dt} - k_v \omega_m \quad (3.11)$$

$$J_m \frac{d\omega_m}{dt} = \tau_m - K_{b_m} \omega_m \quad (3.12)$$

$$\tau_o = k_t i(t) \quad (3.13)$$

Observa-se que os parâmetros (R , k_v , k_b e k_t) não foram fornecidos na tabela 3. Entretanto, eles são essenciais para a modelagem do motor. Desse modo, foram usados os dados da tabela 3 e as equações 3.11, 3.12 e 3.13 quando o motor está em regime permanente, ou seja as derivadas de corrente e de velocidade são 0 para estimar os valores restantes.

Na condição de *Stall*, quando o motor tem o eixo travado, é possível estimar a resistência do motor (R), como é mostrado na Equação 3.14 a partir da Equação 3.11.

$$12 = R \times 5 + \frac{d}{dt} 5 - k_v \times 0 \therefore R = 2,4\Omega \quad (3.14)$$

E utilizando o motor sem carga, é possível estimar a constante eletromotriz conforme a Equação 3.15 a partir da Equação 3.11.

$$12 = 2,4 \times 0,3 + \frac{d}{dt} 0,3 - k_v \times 220 \times \left(\frac{2\pi}{60}\right) \therefore k_v = 489 \frac{mV}{s} \quad (3.15)$$

Aplicando as mesmas condições para as equações 3.12 e 3.13, calcula-se a constante de torque e o coeficiente de atrito viscoso, respectivamente nas equações 3.16 e 3.17

$$15 \times \frac{9,81}{100} = k_t \times (5 - 0,3) \therefore k_t = 0,313 \frac{Nm}{A} \quad (3.16)$$

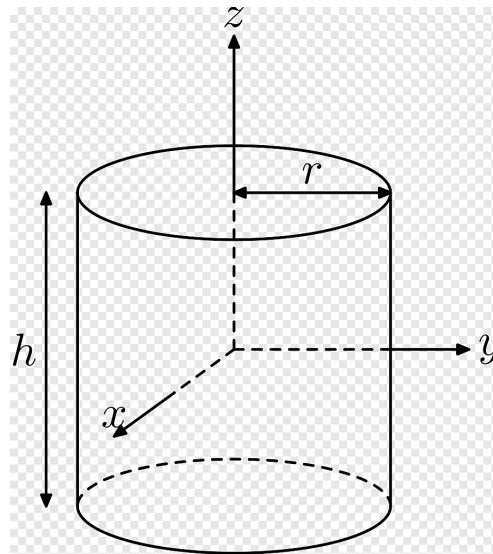
$$J_m \times \frac{220 \times \frac{2\pi}{60}}{dt} = 15 \times \frac{9,81}{100} - k_{b_m} \times 220 \times \frac{2\pi}{60} \therefore k_{b_m} = 189,55 \times 10^{-6} Nm/s \quad (3.17)$$

O momento de inércia do motor e da roda é aproximado para um cilindro maciço, descrito na Equação 3.18.

$$J = \frac{mr^2}{2} \quad (3.18)$$

Sendo J o momento de inércia da peça, m a massa e r , o raio; como pode ser visto na Figura 19.

Figura 19 – Momento de Inércia de um cilindro maciço

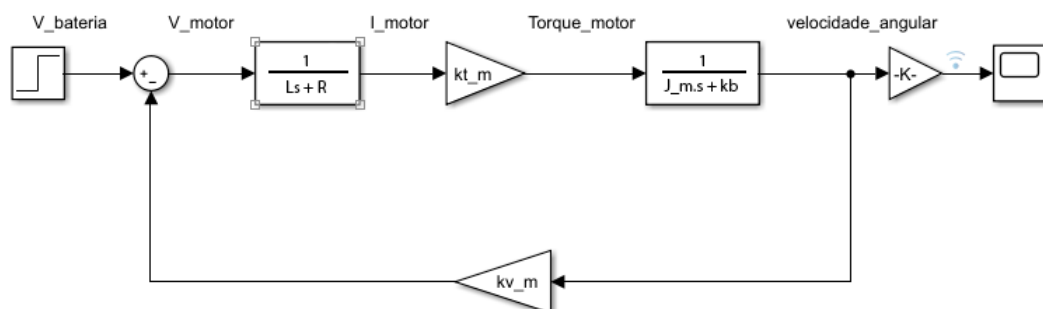


Assim sendo, o momento de inércia da roda e do motor é $150 \times 10^{-3} \text{kgm}^2$ e $62,5 \times 10^{-6} \text{kgm}^2$, respectivamente.

Vale destacar, que esse método de calcular o momento de inércia do motor é uma aproximação, uma vez que é considerado que o motor e a roda possuem uma massa igualmente distribuída, o que não ocorre na realidade. Por isso, a maneira mais eficaz de se obter a função transferência de um sistema motor com roda seria gerar uma função degrau nos terminais do motor e salvar no osciloscópio o seu comportamento. Com isso, utilizando funções do MatLab ou algoritmos de aproximações de curvas, foi possível obter uma aproximação real da função transferência, porém para continuar este estudo, usou-se a função transferência calculada com os parâmetros elétricos e mecânicos obtidos.

Com todas as variáveis do motor definidas, será definido o sistema dinâmico, representado na Figura 20, sendo a entrada do sistema a tensão de alimentação e saída a velocidade angular do motor aplicando a transformada de Laplace nas equações 3.11, 3.12 e 3.13, obtendo dessa forma a Equação 3.19.

Figura 20 – Sistema dinâmico do motor de corrente contínua



Fonte: Própria

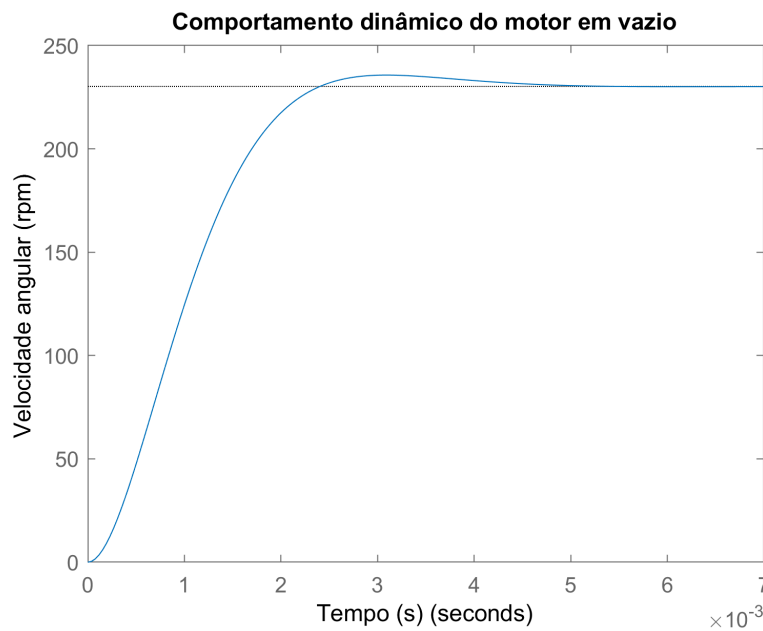
$$\frac{\Omega(s)}{V(s)} = \frac{k_t}{(J_m L)s^2 + (Lk_{b_m} + J_m R)s + (Rk_{b_m} - k_t k_v)} \quad (3.19)$$

Reescrevendo a Equação 3.19 substituindo os valores, tem-se a Equação 3.20

$$\frac{\Omega(s)}{V(s)} = \frac{5009361}{s^2 + 2417s + 2493629} \quad (3.20)$$

Vale ressaltar que o sistema montado na Equação 3.20 é do motor que possui as características na Tabela 3 a vazio, isto é, sem nada acoplado em seu eixo. Esse sistema possui o desempenho referente a uma entrada degrau registrado na Tabela 4 e sua resposta em degrau na Figura 21.

Figura 21 – Resposta da função de transferência do motor corrente contínua para entrada degrau unitário



Fonte: Própria

Tabela 4 – Índices de desempenho do motor CC calculados no MatLab

Tempo de subida	1,5ms
Tempo de acomodação	3,5ms
Máximo percentual	2,38%
Pico	235,69 rpm
Tempo de Pico	3ms

Fonte: Própria

Esse sistema seria útil para o projeto caso o projetista tivesse interessado no motor a vazio, entretanto essa não é a realidade quando se trata de motores de navegação de um robô. No

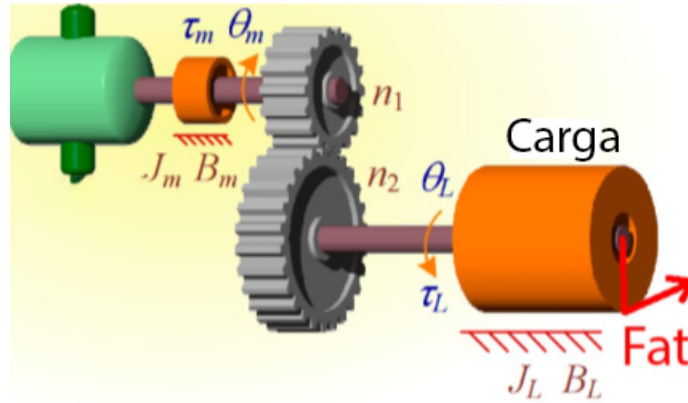
eixo do motor é acoplado uma roda *mecanum wheels* de 100mm de diâmetro, para esse trabalho será considerado um acoplamento perfeito, desse modo será considerado $k_{b_m} = k_{b_w}$. Além disso, deve-se considerar o acoplamento da roda no eixo do motor, representado na Equação 3.21, além da segunda lei de Newton no movimento de rotação da roda, Equação 3.22.

$$\omega_m = \omega_w \quad (3.21)$$

$$J_w \frac{d\omega_w}{dt} = \tau_w - k_{b_w} \omega_w \quad (3.22)$$

Logo, em cada roda do robô terá o torque gerado pelo motor, o torque torque da carga e o torque gerado pela força de atrito entre roda e chão, como pode ser visto na Figura 22.

Figura 22 – Representação esquemática do motor de corrente contínua com redução e acoplado a uma carga que possui força de atrito

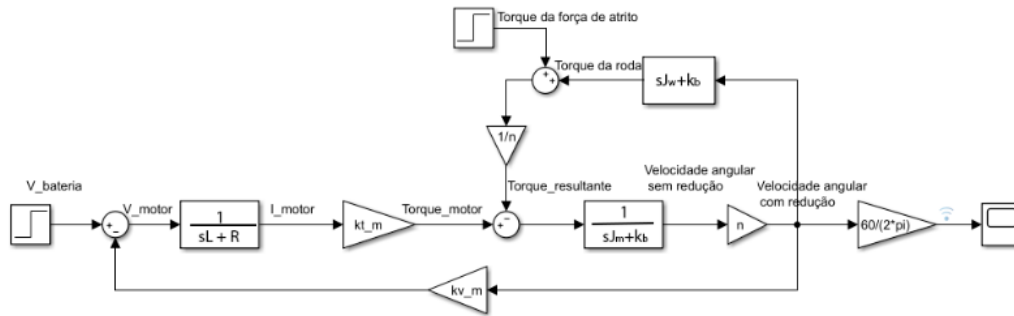


Fonte: Imagem modificada de (URREA; KERN, 2011)

Desse modo, segundo (URREA; KERN, 2011) apresenta em seu trabalho, o torque resultante do motor é dado na Equação 3.23 e o sistema dinâmico em cada roda do robô é representado na Figura 23.

$$\tau_r = \tau_o - \tau_w - \tau_{Fat} \quad (3.23)$$

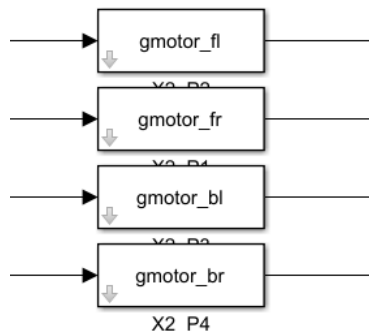
Figura 23 – Diagrama de blocos do motor acoplado com a roda e com a força de atrito



Fonte: Própria

Para simplificar o diagrama de blocos da Figura 23, foi utilizado um bloco apenas com o seu respectivo nome, *motor back-right*, *motor back-left*, *motor front-right* e *motor front-left*, como pode ser visto na Figura 24.

Figura 24 – Diagrama de blocos do motor acoplado com a roda e com a força de atrito



Fonte: Própria

Utilizando as forças de reação em cada apoio do robô, exibidos na Tabela 2, para quando o ângulo da cremalheira está em 0° , foi aplicada uma tensão de 12V em cada motor de forma degrau tanto o motor sem nenhum acoplamento quanto o sistema completo, como pode ser verificado na Figura 25.

A função de transferência do motor com roda acoplada e com carga, está representada na Equação 3.24.

$$\frac{\Omega(s)}{V(s)} = \frac{687,2}{0,0007057s^2 + 1,715s + 388,6} \quad (3.24)$$

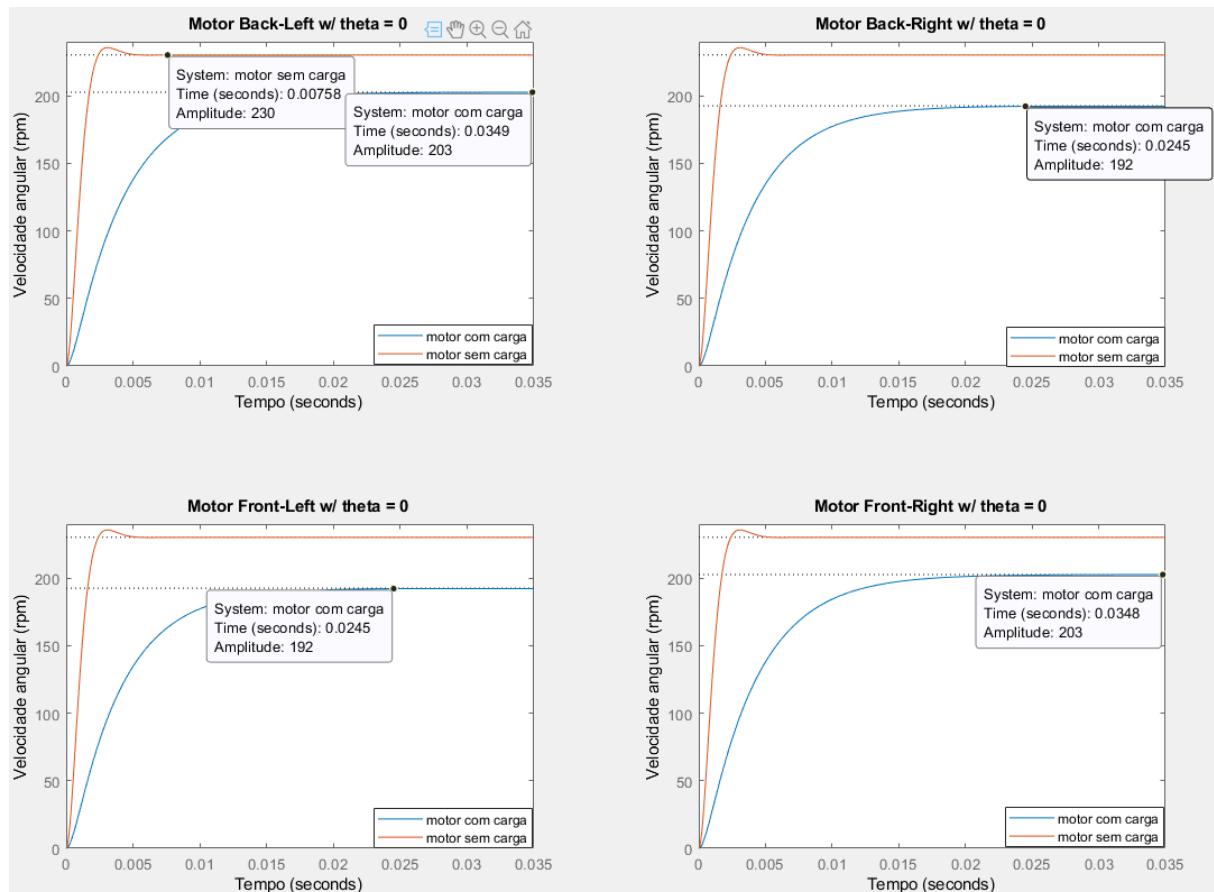
Tabela 5 – Desempenho do motor *Front Left* acoplado com roda e carga.

Tempo de subida	8,8 ms
Tempo de acomodação	16 ms
Máximo percentual	0
Pico	1,76
Tempo de Pico	35,1 ms

Fonte: Própria

Percebe-se que quando acoplado o motor a roda, o tempo de subida da função transferência aumenta, visto que o momento de inércia da roda é maior do que o motor.

Figura 25 – Comparação da resposta ao degrau de cada motor com seu respectivo peso no apoio



Fonte: Própria

Nota-se na Figura 25 que com o peso em cada roda do motor, a velocidade angular de cada roda diminui. Além do mais, percebe-se que para uma mesma tensão de entrada, no caso de 12V, as velocidades angulares dos motores podem ser diferentes conforme está a distribuição do peso do robô.

3.3 Modelagem de sistema dinâmico do robô e sua trajetória

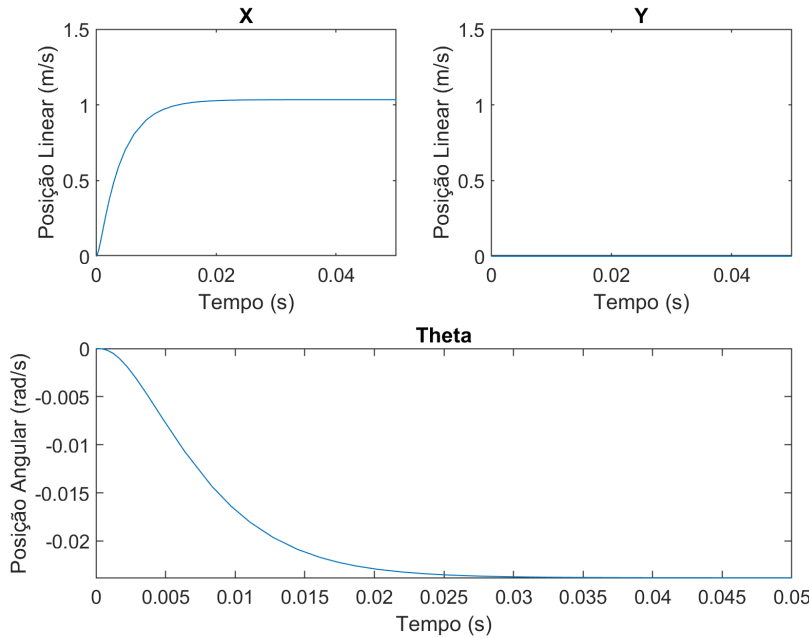
Nesta seção foi estudado o comportamento do robô que possui *mecanum wheels* quando suas velocidades angulares de cada motor não são iguais.

Como já visto, robôs que possuem rodas *mecanum wheels* possuem mais facilidade para se locomover para frente, lateral, diagonal e rotação em seu próprio eixo, conforme apresentado na Figura 9 e representado na Equação 3.25, que calcula a velocidade linear do robô no eixo X e Y , além da velocidade angular em Z .

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} = \frac{R}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ \frac{1}{l1-l2} & \frac{1}{l1-l2} & -\frac{1}{l1-l2} & \frac{1}{l1-l2} \end{bmatrix} \times \begin{bmatrix} \omega_{fl} \\ \omega_{fr} \\ \omega_{bl} \\ \omega_{br} \end{bmatrix} \quad (3.25)$$

Ao aplicar a Equação 3.25 nas mesmas condições da Figura 25 é possível obter a velocidade linear do robô com sua referência ao centro do robô, como ilustrado na Figura 26.

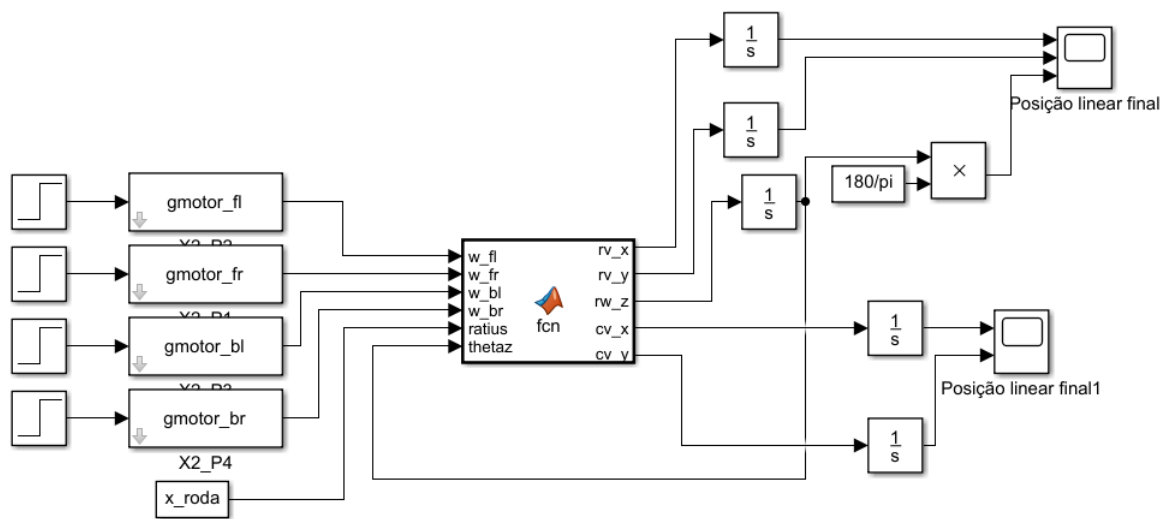
Figura 26 – Velocidades lineares x e y do robô e velocidade angular z



Fonte: Própria

Ao integrar as velocidades angulares em função do tempo com o ajuste da rotação do robô, pode-se estimar a trajetória do robô e entender o seu comportamento. Na Figura 27, é calculado a trajetória tendo o eixo cartesiano como referência, considerando que o robô inicia a sua trajetória no ponto $(0, 0)$.

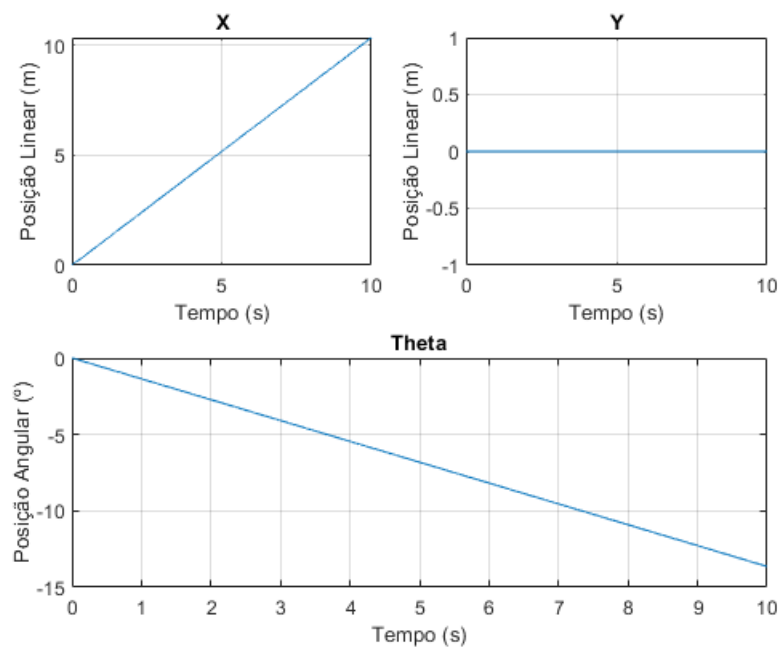
Figura 27 – Diagrama de blocos para calculo da trajetória do robô com referência o plano cartesiano



Fonte: Própria

A simulação do *simulink* da Figura 27 realiza a integral da velocidade linear e angular do robô, desse modo tendo a posição do robô no eixo X e Y , além da posição rotacionada, descrito na Figura 28. Todos os motores estão recebendo um degrau de 12 volts na entrada, isso indica que a trajetória esperada é uma trajetória reta em X .

Figura 28 – Trajetória do robô com referencial no centro do robô

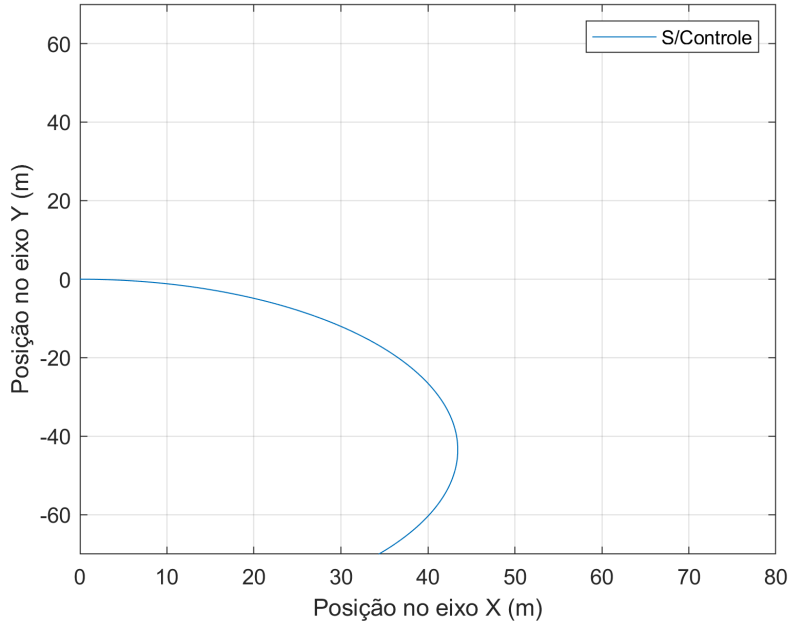


Fonte: Própria

A interpretação da Figura 28 é que o robô está indo para frente com uma rotação em seu eixo z , o que é melhor visualizado quando troca-se a referência do sistema para o plano

cartesiano, conforme é mostrado na Figura 29.

Figura 29 – Trajetória do robô com referencial no plano cartesiano



Fonte: Própria

Desse modo, é possível concluir pelas Figuras 26 e 28 que o erro no deslocamento é provocado pela rotação do robô, ou seja pela velocidade angular na rotação do eixo z que deveria ser zero, mas não é.

3.4 Definição de *Hardware*

Antes de elaborar um controlador de malha para zerar o erro de trajetória do robô, precisa-se definir o *hardware* que será utilizado para elaborar o controle com base na dinâmica do motor já descrito na seção 3.2.

Com a finalidade de utilizar dispositivos digitais para realizar o controle de malha fechada, é preciso discretizar o modelo do motor da Equação 3.20. Para isso, (OLIVEIRA; AGUIAR; VARGAS, 2017) destacam que o intervalo de amostragem da discretização deve ser menor ou igual a um décimo da constante de tempo dominante. Para o caso do método da resposta ao degrau, recomenda-se a condição imposta na Equação 3.26.

$$\frac{T_s}{V} \leq 0,1 \therefore T_s \leq 1,2s \quad (3.26)$$

Uma vez que a tensão da bateria é de $12V$. Além da condição calculada, a Tabela 6 lista algumas regras do ponto de vista prático para escolha adequada do tempo de amostragem.

Tabela 6 – Escolha intervalo de amostragem de forma prática

$T_0 \approx (1/16 - 1/8) \cdot (1/f_{bw})$	f_{bw} : banda passante
$T_0 \approx (1/8 - 1/4)T_t$	T_t : tempo morto
$T_0 \approx (0,35 - 1,2)T_u$	$0,1 \leq (T_u/T_0) \leq 1$
$T_0 \approx (0,22 - 0,35)T_u$	$1 \leq (T_u/T_0) \leq 10$
$T_0 \approx \pi/\omega_{max}$	Teorema da Amostragem
$T_0 \approx (1/15 - 1/6) \cdot T_{95\%}$	$T_{95\%}$: Tempo de subida

Fonte: extraído de (AGUIAR, 2018).

Para continuar o trabalho, será utilizado a regra do tempo de subida, no qual pode-se escolher entre $\frac{1}{6}$ até $\frac{1}{15}$ do tempo de subida. Pelos dados de desempenho do motor, descritos na Tabela 5, tem-se que o tempo de subida é de 8,8ms, portanto o tempo de amostragem do sistema discretizado é calculado na Equação 3.27,

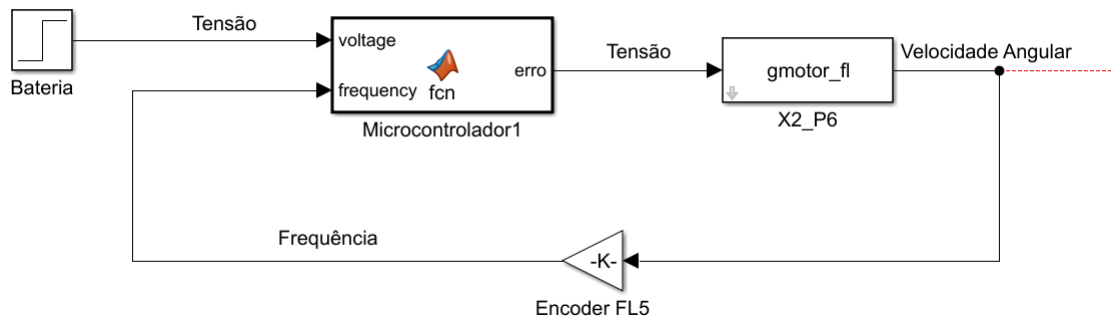
$$T_0 \approx \frac{1}{10} \times T_{95\%} \therefore T_0 \approx 880\mu s \quad (3.27)$$

Desse modo, os sensores e microcontroladores do sistema de controle tem que possuir uma frequência de amostragem duas vezes maior do sistema (Teorema da amostragem de Nyquist–Shannon), como é calculado na Equação 3.28.

$$f_s \geq 2 \frac{1}{T_0} \therefore f_s \geq 2,272kHz \quad (3.28)$$

O controle de malha fechada que será projetado esta esquematizado na Figura 30.

Figura 30 – Diagrama de blocos do sistema de controle utilizado no Simulink para aplicar as equações diferenças



Fonte: Própria

Assim sendo, será necessário um sensor que consiga interpretar a velocidade do motor como frequência para o microcontrolador. Um modelo de sensor que faz isso são sensores ópticos, mais conhecidos no mercado como *encoder*. Porém, a frequência por rotação tem que ser maior do que $2,272kHz$ que também serve para a taxa de amostragem do microcontrolador.

O microcontrolador selecionado para o projeto é uma *RaspBerry Pi 4B+* com 4GB de memória RAM. Esse microcontrolador possui uma taxa de amostragem em seus GPIO de $50kHz$. Já o *encoder*, será calculado a resolução dele na Equação 3.29. Para estimar a resolução desse sensor, será considerado que a velocidade final de projeto será de 140 rpm e a frequência por rotação será um pouco maior do que a taxa de amostragem mínima calculada na Equação 3.28.

$$\omega_m = f_e \frac{2\pi}{res_{min}} \therefore res_{min} = 985,7143 \quad (3.29)$$

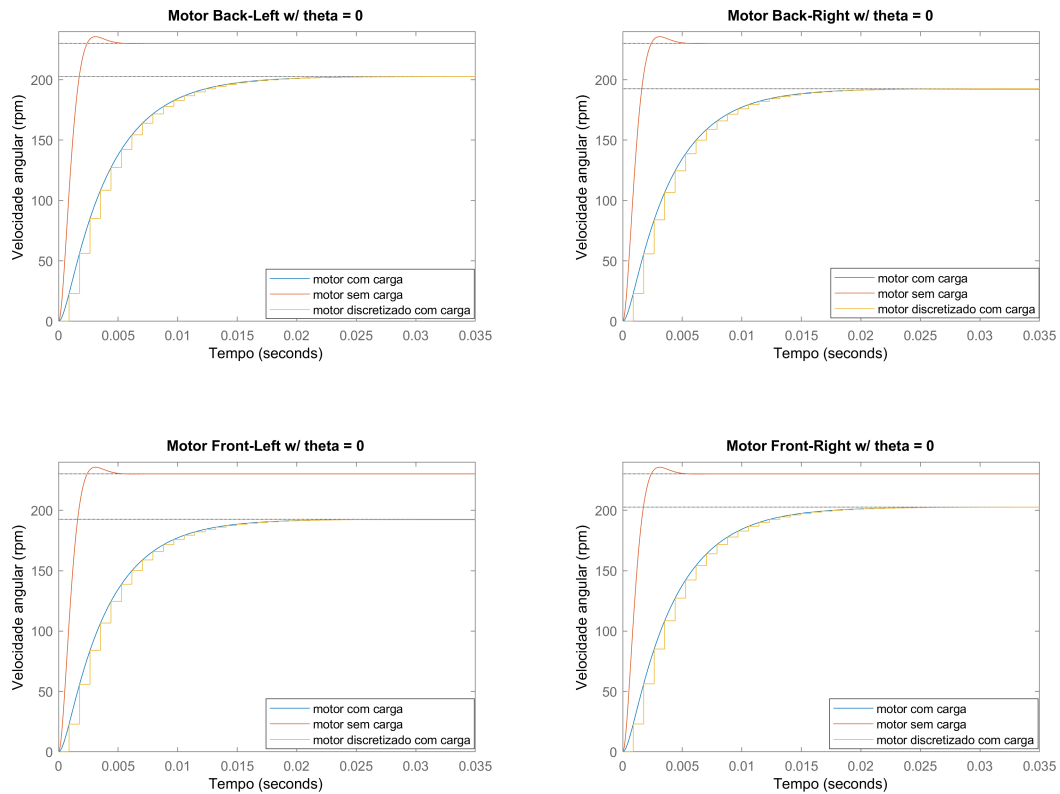
Sendo $f_e = 2,3kHz$ e $\omega_m = 140 \times (2\pi/60)rad/s$. A resolução mínima para a escolha do *encoder* é de 985,71. Ao analisar esse tipo de sensor no mercado, foi verificado que a resolução mais próxima para esse tipo de sensor é de uma resolução de 1024. Com a resolução definida e a taxa de amostragem fixada em $2,3kHz$, a velocidade mínima controlada será calculada na Equação 3.30.

$$\omega_{m_{min}} = f_e \cdot \frac{2\pi}{res_{1024}} \therefore \omega_{m_{min}} = 134,76rpm \quad (3.30)$$

Quando os *hardwares* possuem frequências de amostragem diferentes, como no caso do microprocessador e sensor, é considerado para o projeto a menor das frequências de amostragem do sistema, uma vez que ela é o gargalo do sistema.

Desse modo, a resposta do sistema discretizado está representado na Figura 31.

Figura 31 – Resposta da função de transferência do motor contínua e discretizada para duas condições mecânicas.



Fonte: Própria

3.5 Elaboração do controlador

Esta seção tem por finalidade descrever o controle de velocidade do motor CC com um controlador proporcional integrativo derivativo (PID), o qual permite um erro do regime permanente nulo. Para realizar esse controlador, foi necessário utilizar o modelo de segundo ordem representado na Equação 3.20 e o *simulink* para determinar os ganhos proporcional, integral e derivativo que fazem a resposta do sistema em malha fechada se adequar aos parâmetros da dinâmica desejada.

Para que o erro de regime seja nulo, será empregado um controlador PID digital desenvolvido em uma função do MatLab para o *Simulink*, que representou o microcontrolador escolhido na seção 3.4. Foi mostrado o passo a passo de como foi escolhido os ganhos do controlador. O *encoder* e microcontrolador foram adicionados ao circuito de potência do motor em um único esquemático do circuito.

Um controlador PID no caso discreto, com uma frequência de amostragem relativamente alta, pode ser descrito em forma de uma Equação diferenças, aproximando-se a integral e a derivada por somatórios e subtrações, obtendo-se pela aproximação a Equação 3.31.

$$u(k) = K_p e(k) + K_i \sum_{j=0}^{k-1} e(j) + K_d (e(k) - e(k-1)) \quad (3.31)$$

Para construir o controlador discretizado, é necessário saber os polos das funções transferência. Para exemplificar, foi utilizado a função discretizada do motor de trás a direita, representado na Equação 3.32. Vale ressaltar que método de aproximação da conversão de contínuo para o discreto foi o *Zero-order hold*.

$$\frac{\Omega(z)}{V(z)} = \frac{0,06599z + 0,04639}{z^2 - 1,279z + 0,3457} \quad (3.32)$$

Para obter os polos da função transferência discretizada, basta apenas calcular as raízes, mostrado na Equação 3.33, do numerados da Equação 3.32.

$$z^2 - 1,279z + 0,3457 = 0 \quad (3.33)$$

$$\begin{cases} p_1 = 0,8907 \\ p_2 = 0,3882 \end{cases} \quad (3.34)$$

Um controlador PID discretizado padrão é dado pela Equação 3.35.

$$G_{PID}(z) = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{z^{-1}(1 - z^{-1})} = q_0 \frac{z^2 + \frac{q_1}{q_0}z + \frac{q_2}{q_0}}{z(z - 1)} \quad (3.35)$$

É preciso que os zeros do controlador precisam ser próximos do polos da planta, desse modo tem-se a Equação 3.36.

$$z^2 - (p_1 + p_2)z + p_1 p_2 = z^2 - 1,279z + 0,3457 \quad (3.36)$$

Para definir q_0 será feito a função transferência de malha aberta entre o controlador e a planta do motor, mostrado na Equação 3.37.

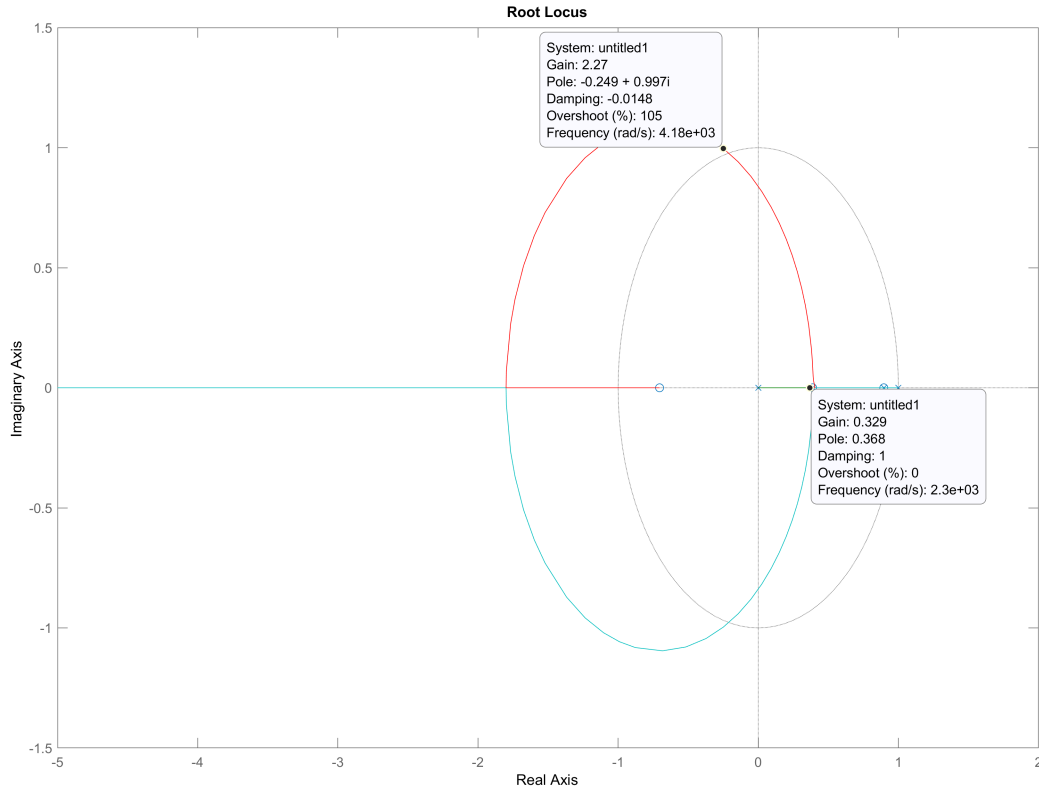
$$G_{MA} = G_{PID}(z) \cdot G_{br}(z) = q_0 \frac{0,06599z + 0,04639}{z(z - 1)} \quad (3.37)$$

Ao fechar a malha, tem-se a Equação 3.38.

$$G_{MF} = \frac{q_0(0,06599z + 0,04639)}{z^2 - (1 - 0,06599q_0)z + (0,1345q_0)} \quad (3.38)$$

A função transferência representa o método dos lugares das raízes, representado na Figura 32 produzida no MatLab®.

Figura 32 – Método dos lugares das raízes aplicado para definir q_0 para que o sistema seja estável e não obtenha sobressinal



Fonte: Própria

Para o sistema ser estável, os polos da função transferência de malha fechada deverá ficar dentro do círculo unitário. Desse modo, o ganho q_0 deverá ser menor do que 2, 27. Porém, para melhor eficiência no sistema, deseja-se que o sistema não possua sobressinal, desse jeito q_0 deverá ser, também, menor do que 0, 329. Assim sendo, foi escolhido ganho q_0 sendo 0, 2.

Com isso, os valores de q_1 e q_2 são cálculos na Equação 3.39.

$$\begin{cases} q_1/q_0 = -1,279 \therefore q_1 = -0,2558 \\ q_2/q_0 = 0,3457 \therefore q_2 = 0,06914 \end{cases} \quad (3.39)$$

Portanto, a Equação diferença, que será aplicada na RaspBerry Pi 4B, está expressa na Equação 3.40.

$$uc(k) = 0,2e(k) - 0,2558e(k-1) + 0,06914e(k-2) + uc(k-1) \quad (3.40)$$

Tabela 7 – Código implementado no bloco function do Simulink

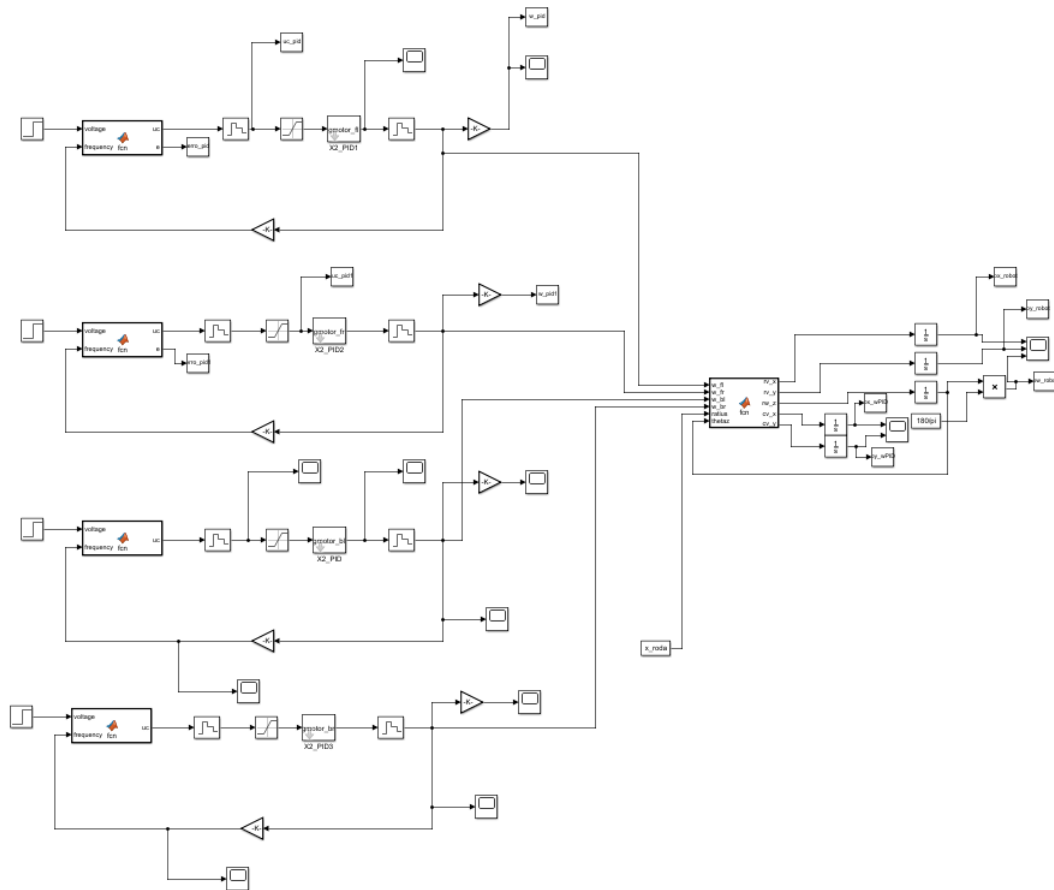
```
function uc = fcn(voltage , frequency )

persistent u0;
persistent e0;
persistent e00;
if isempty(u0)
    u0=0;
    e=0;
    e0=0;
    e00=0;
end

e = (1-frequency/(140/60))* voltage ;
uc = 0.2*e -0.2558*e0 + 0.06914*e00 + u0;
e00=e0;
e0=e;
u0=uc;
```

O código descrito na Tabela 7 foi aplicado para controlar todos os motores do robô. Nota-se que para aplicar o controlador, foi utilizado a função diferença, descrito na Equação 3.40 e que no cálculo de e (erro) foi selecionado a velocidade desejada de 140RPM. A Figura 33 ilustra como ficou o sistema dinâmico final em diagramas de blocos.

Figura 33 – Diagrama de blocos do sistema dinâmico completo e controlado com a finalidade de obter-se a trajetória do robô autônomo e suas velocidades lineares no eixo X e Y e angular em Z

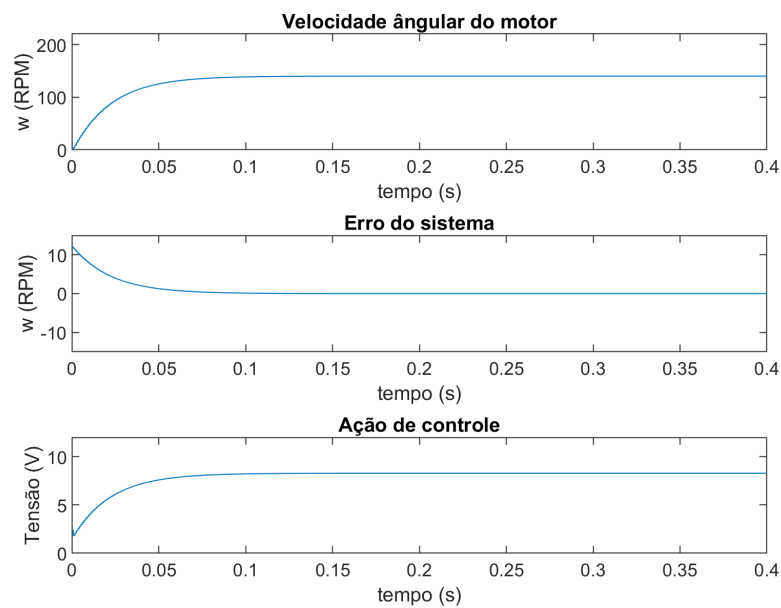


Fonte: Própria

4 RESULTADOS E DISCUSSÃO

Quando é controlado um sistema dinâmico, não basta verificar a saída apenas, é necessário verificar a ação de controle, o erro do sistema também para verificar se as características elétricas do motor foram atendidas, conforme a Tabela 3.

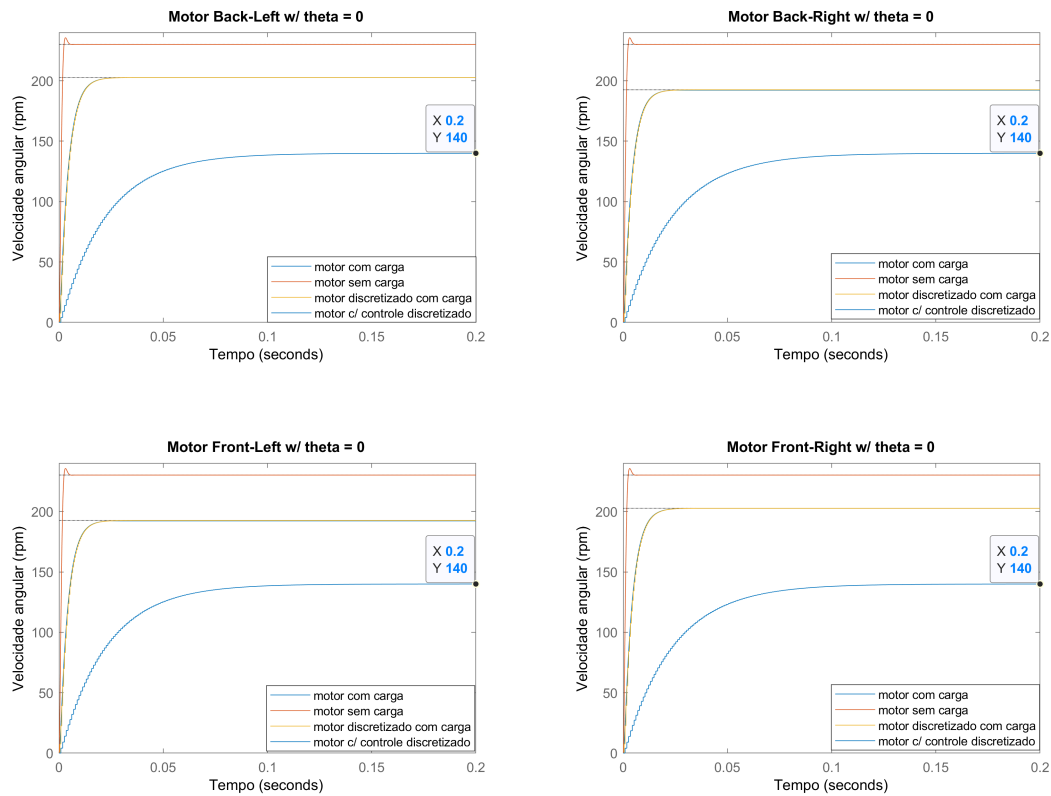
Figura 34 – Ação de controle, erro e velocidade final do motor quando aplicado o controlador



Fonte: Própria

Na Figura 34, mostra que as condições foram atendidas, uma vez que a tensão da ação de controle não ultrapassa em nenhum momento a tensão máxima do motor, que é de 12V. Além disso, a Figura 35 mostra que ao aplicar o mesmo controlador para todos os motores, foi possível determinar a velocidade de 140RPM para cada roda e isso faz com que o erro seja zero.

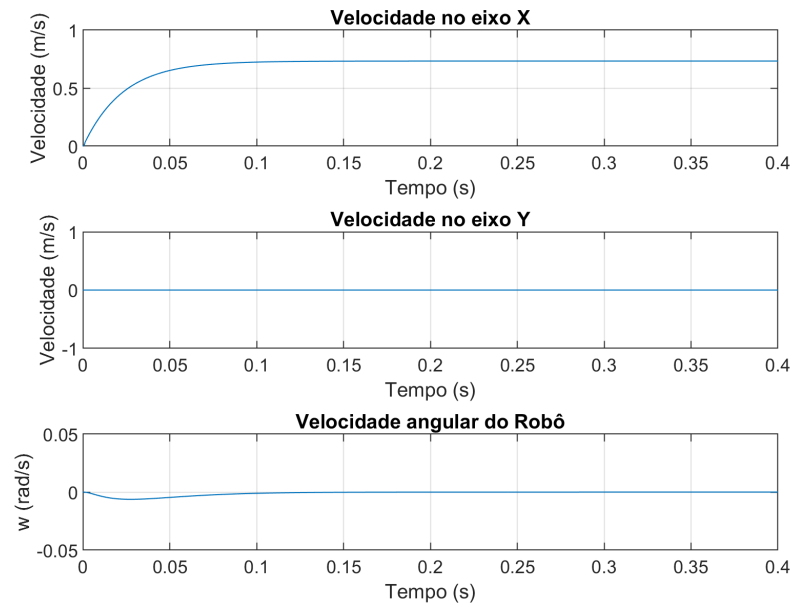
Figura 35 – Comparação das velocidades de cada robô com e sem carga e discretizados



Fonte: Própria

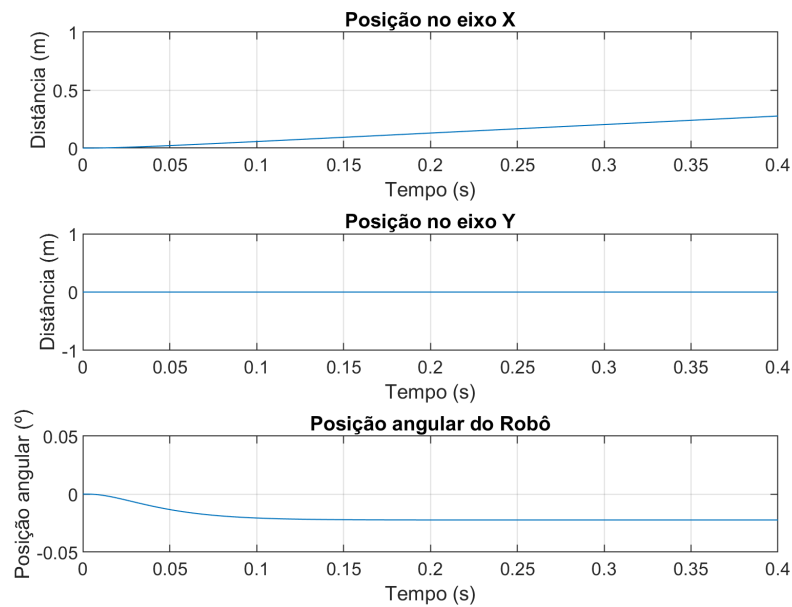
Já na Figura 36, nota-se que a partir de um instante a velocidade angular do robô é zerado, permanecendo apenas a velocidade no eixo X. O fato de o robô ter uma velocidade angular no eixo Z por algum instante poderia ser um grave problema, entretanto, na Figura 37, vê-se que o robô rotaciona apenas $0,02^\circ$ que é considerado ínfimo, uma vez que o competidor ao colocar o robô no chão, pode inserir um erro maior do que o gerado na movimentação dele com o controlador.

Figura 36 – Velocidade linear em cada eixo do robô e velocidade de rotação



Fonte: Própria

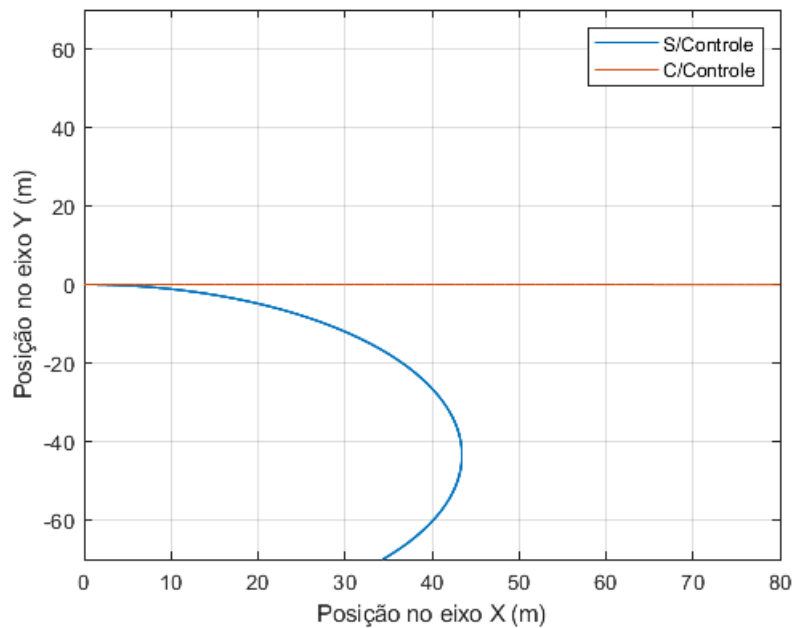
Figura 37 – Posição linear em cada eixo do robô e posição angular do robô no tempo



Fonte: Própria

Dessa forma, ao direcionar o robô para andar apenas no eixo X, a sua trajetória no eixo cartesiano é representado na Figura 38 que compara a trajetória do robô com e sem um controle de velocidade.

Figura 38 – Trajetória do robô com e sem a implementação de um controlador



Fonte: Própria

A Figura 38 indica que o controle proporcional integral derivativo (PID) de velocidade em cada motor já é o suficiente para consertar a trajetória de um robô descompensado. Entretanto, nota-se pela Figura 36 que teve uma pequena perturbação na velocidade angular do robô ainda, que foi corrigida rapidamente, mas também provocou uma pequena rotação de $0,2^\circ$ que pode ser desconsiderada, uma vez que o erro humano de colocar o robô no chão durante as baterias da competição é maior do que o erro gerado na rotação.

5 CONCLUSÃO

Este trabalho abordou as consequências de se elaborar um robô com seu centro de massa deslocado em relação ao centro geométrico e também mostrou uma solução muito conhecida na literatura para corrigir sistemas dinâmicos, empregando um controlador PID.

A trajetória do robô simulado neste projeto baseou-se em um modelo cinemático simples e foi dimensionado adequadamente para representar um robô de pequeno porte real com rodas suecas (Tabela 1) em uma aplicação prática de trajetórias simples de navegação. O fato desse objeto de estudo possuir uma cremalheira com um peso considerável em relação ao peso total causou o deslocamento do centro de massa em relação ao centro geométrico. Isso afeta na velocidade final dos motores e dificulta em tarefas simples de navegação, representado na seção 3.3.

Para consertar esse erro de trajetória, foi proposto um controlador digital PID para garantir que todos os motores possuam a mesma velocidade final e consequentemente corrigir a trajetória. Para isso, foi construída a função de transferência do motor com base nas variáveis elétricas e mecânicas utilizando a transformada de Laplace. Com a função transferência do motor, analisou-se os índices de desempenho do sistema (Tabela 4) para a escolha do microprocessador que irá fazer a leitura dos valores instantâneos, calcular o erro entre a velocidade atual e a de referência para ajustar a tensão aplicada no motor.

O controlador digital, projetado na seção 3.5, mostrou-se eficiente para corrigir a velocidade final e não teve prejuízo de desempenho pelo fato de ser discretizado. Com as velocidades de todos os motores ajustadas, foi possível corrigir a trajetória do robô, obtendo um erro ínfimo.

Embora esse trabalho tenha demonstrado toda a modelagem, construção do controlador e ter mostrado resultados bem coerentes com a realidade, é importante lembrar que esse projeto foi utilizado aproximações para se obter a função transferência do motor e da roda acoplada, que pode ser diferente na prática o que pode provocar um controlador diferente do apresentado nesse projeto. Uma vez que a metodologia implementada considera como partida os aspectos de desempenho do sistema dinâmico de uma modelagem aproximada.

Para trabalhos futuros, são sugeridos:

- Estender o estudo de controle de trajetória em sistemas de simulação mais próximos da realidade, como ROS (do inglês *Robot Operating System*), para aprimorar mais o ambiente de simulação;
- Implementar outra abordagem para corrigir a trajetória do robô, acrescentando mais sensores que não dependam de uma taxa de amostragem alta; e

- Comparar o sistema físico com o sistema simulado;

REFERÊNCIAS

- AGUIAR, M. **Principais notas de aula da disciplina de Controle Digital (SEL-359) da Escola de Engenharia de São Carlos**. [S.l.: s.n.], 2018.
- ALMEIDA, F. **O que é Encoder? Para que serve? Como escolher? Como interfacear?** 2020. <https://www.hitecnologia.com.br/blog/o-que-%C3%A9-encoder-para-que-serve-como-escolher-como-interfacear/>. Accessed: 2020-11-29.
- BARR, M. **Introduction to Pulse Width Modulation**. <https://www.embedded.com/introduction-to-pulse-width-modulation/#>. Accessed: 2020-11-29.
- CHAPMAN, S. J. **Fundamentos de Máquinas Elétricas**. Porto Alegre, Brazil: AMGH Editora Ltda, 2013.
- HEISEN, F. J. **Sistema de Controle Híbrido para Robôs Móveis Autônomos**. 2002. Dissertação de Mestrado (Computação Aplicada), UNISINOS (Universidade do Vale do Rio dos Sinos), São Leopoldo, Brazil.
- INDIA, P. **100mm Aluminium Mecanum wheels (Bearing type rollers) Right**. 2021. <https://www.prayogindia.in/product/100mm-aluminium-mecanum-wheels-bearing-type-rollers-right>.
- KLIPP, T. d. S. **Proposta de uma arquitetura para alocação de tarefas em grupos de robôs móveis baseada em acordo bizantino**. 2013. Monografia (Tecnologias de Informação e Comunicação), UFSC (Universidade Federal de Santa Catarina), Araranguá, Brazil.
- MALVINO, A.; BATES, D. **Eletrônica**. Porto Alegre, Brazil: AMGH Editora Ltda, 2016.
- MATARIC, M. **The Robotics Primer**. Cambridge, Massachusetts: The MIT Press, 2007.
- MIRANDA, R. C. de. **Sistema de Navegação para Robôs Móveis Baseado em PID-Fuzzy**. 2017. Monografia (Bacharel Engenharia da Computação), UFAM (Universidade Federal do Amazonas), Manaus, Brazil.
- OGATA, K. **Engenharia de Controle Moderno**. São Paulo, Brazil: Pearson, 2015.
- OLIVEIRA, V.; AGUIAR, M.; VARGAS, J. **Engenharia de Controle: Fundamentos e Aulas de Laboratório**. [S.l.: s.n.]: Elsevier Brasil, 2017.
- OLIVEIRA, V. A.; AGUIAR, M. L.; VARGAS, J. B. **Engenharia de controle: fundamentos e aulas de laboratório**. Rio de Janeiro, Brazil: Elsevier Editora Ltda, 2016.
- POLOLU. **47:1 Metal Gearmotor 25Dx52L mm HP 12V**. 2019. <https://www.pololu.com/product/3205>. Accessed: 2020-11-29.
- ROBÓTICA, A. **Roda Pneu 65mm Aro 12 para Chassi Robótica - Azul**. 2021. <https://www.autocorerobotica.com.br/roda-pneu-65mm-arro-12-para-chassi-robotica-azul>.
- URREA, C.; KERN, J. A new model for analog servo motors. simulations and experimental results. v. 2, p. 29–38, 03 2011.
- US, R. r. **A set of 152mm mecanum wheels basic (4 pieces)**. 2020. <https://robot-r-us.com.sg/p/a-set-of-152mm-mecanum-wheels-basic-4-pieces>. Accessed: 2020-11-29.

WIKIPÉDIA. **Plano inclinado**. 2020. https://pt.wikipedia.org/wiki/Plano_inclinado. Accessed: 2020-11-29.