

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

MAYCON POIATTO

**DESENVOLVIMENTO DE UM ROBÔ MÓVEL OMNIDIRECIONAL
COM CONTROLE CINEMÁTICO**

SÃO CARLOS
2023

MAYCON POIATTO

**DESENVOLVIMENTO DE UM ROBÔ MÓVEL OMNIDIRECIONAL
COM CONTROLE CINEMÁTICO**

Trabalho de Conclusão de Curso
apresentado ao Departamento de
Engenharia Mecânica da Universidade
Federal de São Carlos, como requisito
parcial para obtenção do título de
Bacharel em Engenharia Mecânica.

Orientador: Prof. Dr. Leonardo Mar-
quez Pedro

SÃO CARLOS
2023

Esta folha deve ser substituída pela folha
de aprovação digitalizada

AGRADECIMENTOS

Agradeço a todos que contribuíram direta e indiretamente para a conclusão deste trabalho, em especial, agradeço ao meu orientador, Prof. Dr. Leonardo Marquez Pedro, por me instruir e fornecer os recursos necessários para execução do projeto.

Agradeço o técnico do departamento, Djalma Aparecido Lingnari Durici, pela fabricação das peças para construção do robô; a Fundação CAPES por disponibilizar acesso ao seu acervo acadêmico; o Departamento de Engenharia Mecânica e a Universidade Federal de São Carlos em geral pela infraestrutura, recursos gastos e por todo conhecimento passado durante a graduação.

Agradeço ao Prof. Dr. Luis Antonio Oliveira Araujo pelo auxílio e conselhos dados em diversos dos problemas que tive relacionados ao curso.

Agradeço ao suporte, em todos os sentidos, da família e amigos.

RESUMO

Atualmente a robótica móvel apresenta, tanto em questão de mercado, quanto como área de estudo, uma tendência de aumento ainda maior de relevância nos próximos anos. Uma das problemáticas recorrentes nesse ramo é a mobilidade: em diversos ambientes, inclusive fabris, é frequente a limitação de espaço para movimentação das máquinas. Nesse contexto, os robôs omnidirecionais, definidos pela capacidade de se locomover a qualquer momento em qualquer direção desejada, ganham destaque. Com o intuito de estudar o tema e colocar em prática conhecimentos relacionados a robótica aprendidos na graduação de Engenharia Mecânica, foi construído um robô móvel terrestre omnidirecional com três graus de liberdade (três rodas/motores). Utilizando o método Ziegler-Nichols, foi feito o ajuste do controle PID de cada motor em Arduino Mega. Com a modelagem cinemática do robô, duas trajetórias em formato de quadrado foram definidas e executadas em testes práticos. Os resultados são satisfatórios, especialmente quando comparados a outros projetos semelhantes em que foi utilizado o controle de velocidade dos motores, ao invés do de posição, mas mostram a necessidade de utilizar um controle mais complexo, especialmente quando o robô realiza rotação em torno do próprio eixo. Implementando essa correção e adicionando um sensor para reconhecimento do ambiente se obterá um robô móvel utilizável em aplicações de transporte na indústria ou para tarefas de limpeza doméstica, por exemplo.

Palavras-chave: Robô móvel. Omnidirecional. Cinemática. Controle PID.

ABSTRACT

Currently, mobile robotics presents, both in the subject of market and as a research area, the tendency of growing an even greater relevance in the next years. One of the recurring issues in this field is mobility: in several environments, including factory ones, it is common to have space limitations to machines movement. In this context, omnidirectional robots, defined by the ability to move at any moment in any direction, stand out. With the intention of studying the subject and to put into practice the knowledge related to robotics learned on the Mechanical Engineering graduation course, it was constructed a omnidirectional land mobile robot with three degrees of freedom (three wheels/motors). Using the Ziegler-Nichols method, it was adjusted the PID control for each motor with Arduino Mega. With the kinematics modeling of the robot, two square shaped trajectories were defined and performed in practical tests. The results were satisfactory, especially when compared to other similar projects where it was used speed control of the motors, instead of position, but they still show the need for a more complex control, especially when the robot turn around its own axis. Implementing this correction and adding a sensor for environment monitoring, will result in a mobile robot usable in transport applications in the industry or for domestic cleaning tasks, for example.

Keywords: Mobile robot. Omnidirectional. Kinematics. PID Control.

LISTA DE FIGURAS

| | |
|----------------------------------------------------------------------------------------------------|----|
| Figura 1 – Exemplos de rodas omnidirecionais | 21 |
| Figura 2 – Exemplos de robôs móveis omnidirecionais | 22 |
| Figura 3 – Outra configuração bastante comum de robô omnidirecional | 23 |
| Figura 4 – Esquema de robô omnidirecional com três rodas | 23 |
| Figura 5 – Diagrama de um sistema de controle com malha fechada | 25 |
| Figura 6 – Curva de reação do processo | 26 |
| Figura 7 – Desenho esquemático do robô móvel | 27 |
| Figura 8 – Translação do robô na direção de y_R | 27 |
| Figura 9 – Translação do robô na direção de x_R | 28 |
| Figura 10 – Rotação do robô em torno do próprio eixo | 29 |
| Figura 11 – Translação de sistema de coordenadas | 30 |
| Figura 12 – Rotação de sistema de coordenadas | 31 |
| Figura 13 – Gráfico de uma função de trajetória no tempo | 31 |
| Figura 14 – Visual do robô | 33 |
| Figura 15 – Motoredutor CC utilizado | 34 |
| Figura 16 – Roda utilizada no robô | 34 |
| Figura 17 – Componentes do controlador | 35 |
| Figura 18 – Robô montado com o controlador | 35 |
| Figura 19 – Respostas com ganho crítico | 37 |
| Figura 20 – Identificação dos motores do robô | 38 |
| Figura 21 – Respostas com vários ajustes do controlador | 39 |
| Figura 22 – Esquema das trajetórias programadas | 40 |
| Figura 23 – Posições do robô para a trajetória 1 | 42 |
| Figura 24 – Posições do robô para a trajetória 2 no sistema de coordenadas global | 43 |
| Figura 25 – Posições do robô para a trajetória 2 no sistema de coordenadas local do robô | 43 |
| Figura 26 – Posições angulares dos motores na trajetória 1 | 44 |
| Figura 27 – Posições angulares dos motores na trajetória 2 | 45 |
| Figura 28 – Diagrama de blocos do controle do motor | 45 |
| Figura 29 – Posição dos motores durante trajetória 1 | 47 |
| Figura 30 – Posição cartesiana do robô durante trajetória 1 | 48 |
| Figura 31 – Posição angular do robô durante trajetória 1 | 48 |
| Figura 32 – Y_G versus X_G na trajetória 1 | 49 |
| Figura 33 – Posição dos motores durante trajetória 2 | 50 |
| Figura 34 – Posição cartesiana do robô durante trajetória 2 | 50 |
| Figura 35 – Posição angular do robô durante trajetória 2 | 51 |

Figura 36 – Y_G versus X_G na trajetória 2 51

LISTA DE TABELAS

| | |
|-------------------------------------------------------------------------|----|
| Tabela 1 – Ganhos pelo método Ziegler-Nichols | 26 |
| Tabela 2 – Valores dos ganhos e períodos críticos | 36 |
| Tabela 3 – Ganhos dos motores pelo método Ziegler-Nichols | 36 |
| Tabela 4 – Valores inicialmente definidos para a trajetória 1 | 40 |
| Tabela 5 – Valores inicialmente definidos para a trajetória 2 | 41 |
| Tabela 6 – Coeficientes da trajetória 1 | 41 |
| Tabela 7 – Coeficientes da trajetória 2 | 42 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|--------|-------------------------------------------------------------|
| 3D | Três dimensões |
| CAD | <i>Computer Aided Design</i> |
| CAPES | Coordenação de Aperfeiçoamento de Pessoal de Nível Superior |
| CC | Corrente Contínua |
| I2C | <i>Inter-Integrated Circuit</i> |
| PID | Proporcional, Integral e Derivativo |
| UFSCar | Universidade Federal de São Carlos |

LISTA DE SÍMBOLOS

| | |
|------------|--------------------------------------------------------------------------------------------------------------------------|
| G_c | Função de transferência do controlador PID |
| K_P | Ganho proporcional |
| K_I | Ganho integral |
| K_D | Ganho derivativo |
| K_U | Ganho proporcional crítico |
| P_U | Período crítico |
| T_I | Tempo integral |
| T_D | Tempo derivativo |
| s | Frequência de Laplace |
| t | Tempo |
| u | Saída do controlador PID |
| e | Erro |
| x_R | Eixo de coordenadas cartesianas x no referencial local do robô |
| y_R | Eixo de coordenadas cartesianas y no referencial local do robô |
| X_G | Eixo de coordenadas cartesianas X no referencial global |
| Y_G | Eixo de coordenadas cartesianas Y no referencial global |
| X_R | Eixo de coordenadas cartesianas X no referencial local do robô, mas as posições passadas são acumuladas ao longo tempo |
| Y_R | Eixo de coordenadas cartesianas Y no referencial local do robô, mas as posições passadas são acumuladas ao longo tempo |
| e_{XY} | Erro de posição cartesiana do robô |
| e_θ | Erro de posição angular do robô |
| M_i | Motoredutor i |
| r | Raio das rodas |
| R | Distância do centro do robô até meia largura das rodas |

e_{ϕ_i} Erro de posição angular do motor M_i

ϕ_i Posição angula do motor M_i

SUMÁRIO

| | | |
|----------|--------------------------------------------------------------------------|----|
| 1 | INTRODUÇÃO | 21 |
| 1.1 | PROPOSTA E OBJETIVO DO TRABALHO | 23 |
| 2 | FUNDAMENTAÇÃO TEÓRICA | 25 |
| 2.1 | MÉTODO ZIEGLER-NICHOLS PARA SINTONIZAÇÃO DE CON- TROLE PID | 25 |
| 2.2 | CINEMÁTICA | 27 |
| 2.3 | MUDANÇA DE COORDENADAS | 29 |
| 2.4 | PLANEJAMENTO DE TRAJETÓRIA | 31 |
| 3 | METODOLOGIA | 33 |
| 3.1 | CONSTRUÇÃO DO ROBÔ MÓVEL | 33 |
| 3.2 | AJUSTE DO CONTROLE PID | 36 |
| 3.3 | DETERMINAÇÃO DAS TRAJETÓRIAS E POSIÇÕES A SEREM PERCORRIDAS | 40 |
| 3.4 | PROGRAMAÇÃO DO CONTROLADOR | 44 |
| 3.5 | TESTES E COLETA DE DADOS | 46 |
| 4 | RESULTADOS | 47 |
| 4.1 | TRAJETÓRIA 1 | 47 |
| 4.2 | TRAJETÓRIA 2 | 49 |
| 5 | CONCLUSÃO | 53 |
| | REFERÊNCIAS | 55 |

1 INTRODUÇÃO

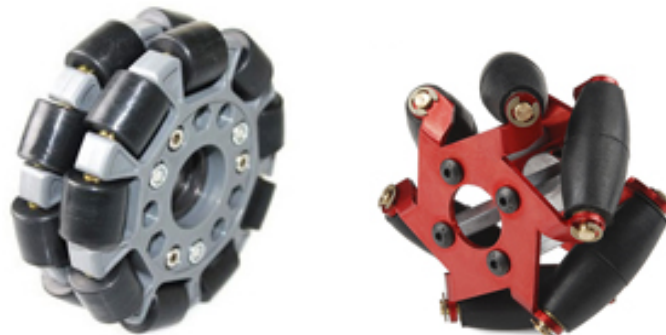
A robótica, especialmente no que diz respeito às aplicações industriais, trata da utilização de máquinas programáveis para realização autônoma de tarefas, até então, realizadas por humanos. Isso fica ainda mais claro ao saber que o termo robô tem origem na palavra tcheca *robot*, que significa trabalho. Foi utilizada a primeira vez numa peça de teatro em 1923, para designar operadores artificiais (DORF; BISHOP, 2001).

A robótica móvel, por sua vez, envolve os robôs que não operam fixos a um ponto do ambiente. É estimado que o valor de mercado de robôs móveis no ano de 2023 seja 20,3 bilhões de dólares americanos, com expectativa de crescimento para 40,6 bilhões em 2028 (MARKETSANDMARKETS, 2023). Há, portanto, um claro interesse na utilização desses equipamentos na indústria.

Dentre os diversos tipos de robôs móveis, há um que tem características que se alinham ao interesse, fundamental nas aplicações desse ramo, por mobilidade, especialmente em ambientes fechados com limitação de espaço. Tratam-se dos robôs omnidirecionais, também chamados de holonômicos, os quais, por definição, são capazes de se mover a qualquer momento em qualquer direção do plano sobre o qual estão (SIEGWART; NOURBAKHS; SCARAMUZZA, 2011).

Em se tratando de veículos terrestres, para conseguir essa mobilidade, é comum utilizar rodas com pequenos roletes distribuídos em torno de sua circunferência externa. Na Figura 1, são mostrados dois exemplos dos modelos mais usados desse tipo de roda.

Figura 1 – Exemplos de rodas omnidirecionais



(a) Universal: roletes a 90° (b) Mecanum: roletes a 45°

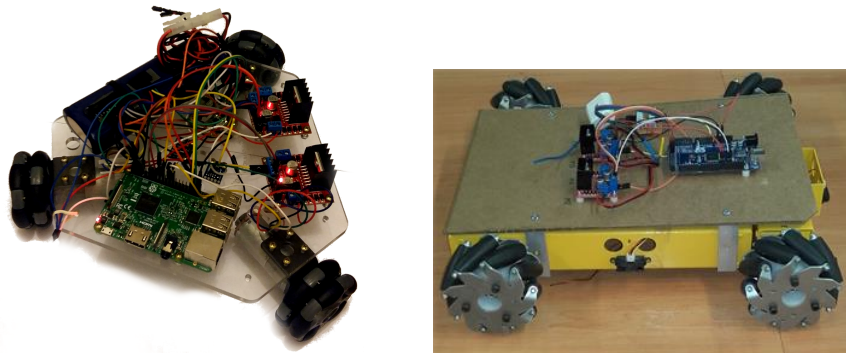
Fonte: Shabalina, Sagitov e Magid (2018)

As rodas com roletes a 45° , comumente chamadas de *mecanum*, movimentam-se com menos vibração e atrito no chão em comparação às que têm roletes a 90° , chamadas de universais. Para compensar este problema, mas ainda mantendo a simplicidade, utiliza-se mais de um conjunto de roletes no modelo universal, como visto na Figura 1a, que tem dois níveis de roletes. A desvantagem da roda *mecanum* é a sua maior sensibilidade à irregularidades no terreno (RITTER, 2016; TAHERI; ZHAO, 2020; SHABALINA;

SAGITOV; MAGID, 2018).

As rodas universais podem ser usados, como o nome indica, em diversas configurações. No entanto, a mais comum é com três rodas igualmente espaçadas a 120° . O modelo *mecanum*, por sua vez, é utilizado com frequência em robôs com quatro rodas com eixos paralelos (TAGLIAVINI et al., 2022). Na Figura 2 são mostrados fotos de exemplos desses robôs.

Figura 2 – Exemplos de robôs móveis omnidirecionais



(a) Robô com três rodas

(b) Robô com quatro rodas

Fonte: Cantú (2017) e Sarmiento et al. (2017)

Apesar da alta mobilidade dos robôs omnidirecionais, é importante salientar que, em comparação a configuração mais comum de veículos, chamada de Ackerman, como a que vemos em automóveis e motocicletas, eles apresentam desvantagem em relação a estabilidade. Num trajeto em linha reta, por exemplo, em que se deseja percorrer em alta velocidade, a configuração Ackerman garante melhor estabilidade lateral (SIEGWART; NOURBAKHS; SCARAMUZZA, 2011).

Sob a perspectiva da academia, os robôs omnidirecionais trazem evidentes pontos interessantes para pesquisa, inclusive bastante próximos a Engenharia Mecânica, como: análise dinâmica das rodas; estudo das melhores configurações quanto a energia, mobilidade e/ou estabilidade; estudo cinemático, dinâmico e, já entrando mais no ramo da computação, planejamento de trajetórias e otimização das rotinas de controle. Além, é claro, dos problemas práticos relacionados a construção dos robôs.

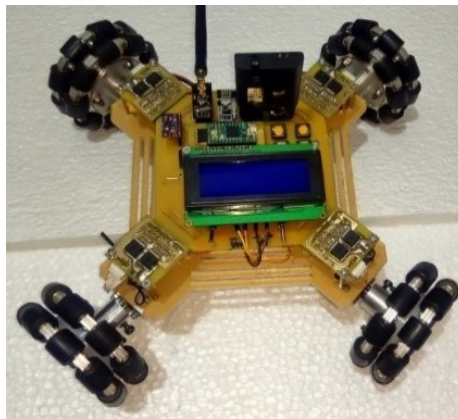
Nesse sentido os trabalhos de Ritter (2016) e Cantú (2017) exemplificam o estudo cinemático e dinâmico de robôs omnidirecionais com três rodas. Cantú (2017), ao contrário de Ritter (2016), também constrói o seu robô (Figura 2a) e realiza testes práticos com ele, trazendo, portanto, a questão da definição de trajetórias e do controle. Ritter (2016), por outro lado, realiza simulações em um *software* para essa finalidade. O grande diferencial deste último trabalho é disponibilizar todas as informações envolvidas no projeto com o propósito de propagar o estudo do tema.

Um outro trabalho muito interessante, por sua similaridade com o que é proposto aqui, é o de Sofwan et al. (2019), em que é desenvolvido um robô de quatro rodas, com

a mesma configuração do da Figura 3. O controle de Sofwan et al. (2019) é baseado na cinemática e na odometria, isto é, no cálculo de distância a partir de medidas incrementais. Além disso, assim como os demais autores citados no parágrafo anterior, utiliza controle PID (proporcional, integral e derivativo), com uso do método Ziegler-Nichols para ajuste dos ganhos. Um detalhe importante é que o controle dos motores foi feito diretamente na velocidade de rotação deles.

Por fim, Rijalusalam e Iswanto (2021) desenvolveu a modelagem cinemática do robô da Figura 3 e uma interface para o usuário inserir a trajetória desejada. Foi feito o controle, mas sem PID, o que levou a resultados não tão satisfatórios.

Figura 3 – Outra configuração bastante comum de robô omnidirecional

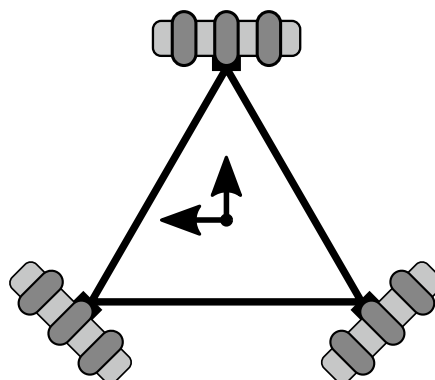


Fonte: Rijalusalam e Iswanto (2021)

1.1 PROPOSTA E OBJETIVO DO TRABALHO

Tendo em vista a tendência de crescimento da robótica, em especial, móvel, tanto na indústria como na academia, é proposto neste trabalho o desenvolvimento de um robô móvel terrestre omnidirecional. A configuração é a já bem estabelecida com três rodas universais, mostrada em esquema na Figura 4.

Figura 4 – Esquema de robô omnidirecional com três rodas



Fonte: Tagliavini et al. (2022)

A escolha é feita pela simplicidade e uso reduzido de material, que é, em sua maioria, o que está disponível no laboratório com o professor orientador.

Alguns pontos relevantes que podem divergir do que costuma ser feito nos trabalhos publicados são: o controle de posição dos motores (muitas vezes são feitos controles de velocidade) e o uso de polinômios de terceiro grau para definição das trajetórias, escolhidos com base em referências da literatura de robótica.

O objetivo geral do trabalho é projetar, construir e testar o robô holonômico em questão. Os objetivos específicos para alcançar essa meta são:

- Modelar as partes necessárias para construção do robô e fabrica-las;
- Ajustar o controle PID de posição dos motores;
- Fazer a modelagem cinemática e definir matematicamente as trajetórias;
- Programar e testar o robô, coletando os dados para análise.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, são apresentadas e descritas as principais considerações teóricas que fundamentam o controle do robô móvel.

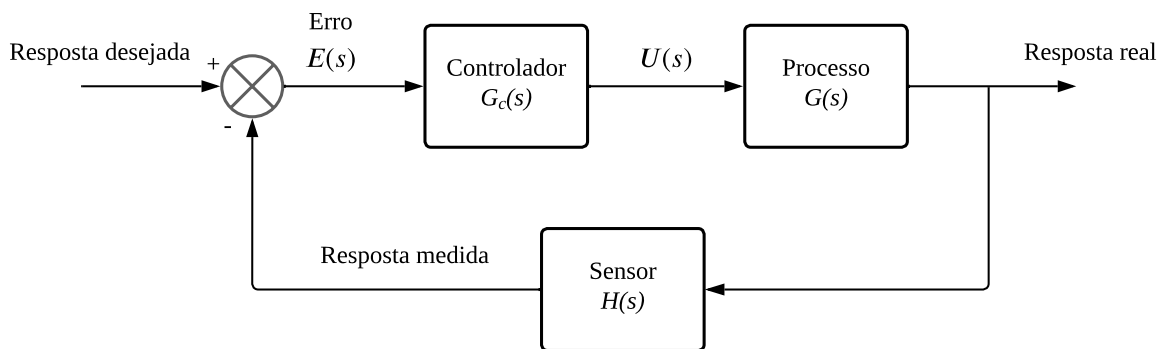
2.1 MÉTODO ZIEGLER-NICHOLS PARA SINTONIZAÇÃO DE CONTROLE PID

Uma forma comumente utilizada para controlar processos industriais é o chamado controle PID (proporcional, integral e derivativo). No domínio de Laplace, o controlador PID tem a seguinte função de transferência (DORF; BISHOP, 2001):

$$G_c(s) = K_P + \frac{K_I}{s} + K_D s, \quad (1)$$

em que K_P é o ganho proporcional, K_I é o ganho integral, K_D é o ganho derivativo e s é a variável de Laplace. Na Figura 5, é mostrado um diagrama de blocos de um sistema de controle, como o que será utilizado neste trabalho.

Figura 5 – Diagrama de um sistema de controle com malha fechada



Fonte: Próprio autor

A saída do controlador, $U(s)$, no domínio do tempo, t , é igual a (DORF; BISHOP, 2001)

$$u(t) = K_P e(t) + K_I \int e(t) dt + K_D \frac{de(t)}{dt}, \quad (2)$$

em que $e(t)$ é o erro, a diferença entre o valor desejado e o valor medido, no domínio do tempo.

Ziegler e Nichols (1942) sugerem duas formas de estimar os ganhos de um controlador PID. Na primeira, com todos os ganhos zerados, é aumentado gradativamente o valor de K_P até que o sinal de saída oscile com frequência e amplitude constantes. Acompanhando a nomenclatura de Franklin, Powell e Emami-Naeini (2018), o ganho proporcional correspondente é definido como K_U (“*ultimate gain*”) e o período do sinal como P_U (“*ultimate period*”), chamados neste trabalho, de ganho e período críticos, respectivamente.

Considerando $K_I = K_P/T_I$ e $K_D = K_P T_D$ é mostrado na Tabela 1 como são calculados os ganhos conforme o método Ziegler-Nichols. Nos casos em que faltam as equações dos tempos (T_I e T_D), os respectivos ganhos são zero.

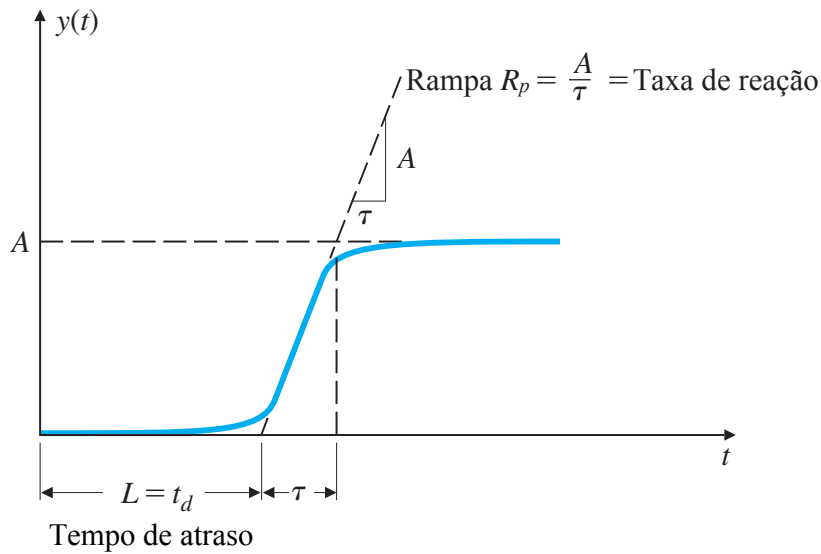
Tabela 1 – Ganhos pelo método Ziegler-Nichols

| Tipo de controle | K_P | T_I | T_D |
|------------------|-----------|-----------|------------|
| P | $0,5K_U$ | - | - |
| PI | $0,45K_U$ | $P_U/1,2$ | - |
| PID | $0,6K_U$ | $0,5P_U$ | $0,125P_U$ |

Fonte: Adaptado de Ziegler e Nichols (1942) e Franklin, Powell e Emami-Naeini (2018)

Na segunda forma proposta por Ziegler e Nichols (1942), o controlador é desligado do sistema e é feita uma entrada degrau “manualmente”. Espera-se que o sinal de resposta medido tenha o formato de “S”, como o mostrado na Figura 6.

Figura 6 – Curva de reação do processo



Fonte: Adaptado de Franklin, Powell e Emami-Naeini (2018)

Definindo

$$R_1 = \frac{R_p}{\Delta F}, \quad (3)$$

em que ΔF é a amplitude da entrada degrau, pode-se encontrar K_U e P_U com as Equações 4 e 5 (ZIEGLER; NICHOLS, 1942), e depois os ganhos com as fórmulas da Tabela 1.

$$K_U = \frac{2}{R_1 L}, \quad (4)$$

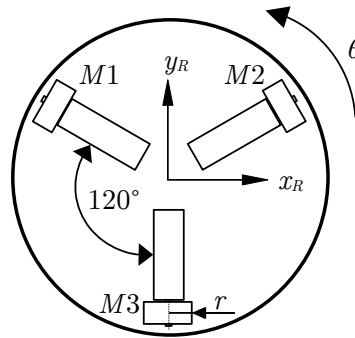
$$P_U = 4L. \quad (5)$$

2.2 CINEMÁTICA

Nesta seção, são apresentadas as equações usadas para encontrar a posição do robô sabendo as posições angulares dos motores (cinemática direta) e a situação contrária, em que deseja-se encontrar os ângulos que os motores precisam rotacionar para que o robô chegue numa posição desejada (cinemática inversa).

Na Figura 7, é mostrado um desenho esquemático do robô móvel visto de cima e com sistema de coordenadas com origem no centro da plataforma em que os motores estão fixados. O índice R é usado para indicar que o sistema é local do robô e se move junto com ele.

Figura 7 – Desenho esquemático do robô móvel



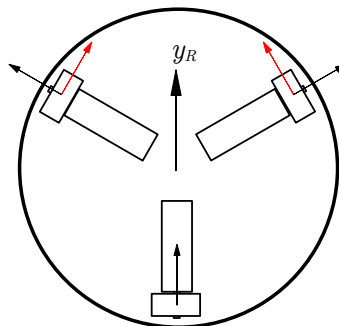
Fonte: Próprio autor

$M1$, $M2$ e $M3$ referem-se aos motores e suas respectivas rodas, cujos raios externos são iguais a r . No desenvolvimento a seguir, a posição angular do eixo de M_i é indicada como ϕ_i .

Para cada motor, é permitido o movimento tanto na sua direção axial, quanto na direção perpendicular de seu eixo, pois as rodas em questão são omnidirecionais. Entretanto, o movimento na direção axial de M_i não causa alteração em ϕ_i .

Dessa forma, considere que o robô é deslocado no sentido positivo de y_R , conforme mostrado na Figura 8.

Figura 8 – Translação do robô na direção de y_R



Fonte: Próprio autor

As setas sobre as rodas indicam os esforços que tendem a movimenta-las no sentido apontado. As de cor preta indicam os componentes que causam o movimento dos pequenos roletes situados na borda da roda, enquanto as setas de cor vermelha indicam os componentes que causam o giro dos motores. Por convenção, é adotado que o sentido positivo de giro dos motores é anti-horário, “olhando de frente” a roda.

O deslocamento de M_i na sua direção perpendicular é igual a $\phi_i r$. Fazendo a decomposição do deslocamento na direção de y_R , tem-se que as posições angulares são:

$$\begin{cases} \phi_1 = \frac{y^R \operatorname{sen}(\pi/3)}{r} \\ \phi_2 = -\frac{y^R \operatorname{sen}(\pi/3)}{r} \\ \phi_3 = 0 \end{cases} . \quad (6)$$

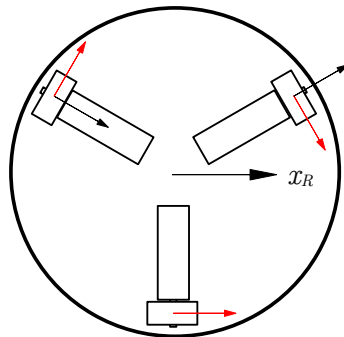
O índice sobrescrito, no caso R , indica que se trata de uma coordenada no sistema em questão, mas não dos eixos em si, que são indicados com um subíndice.

Da mesma forma, caso o robô se desloque da direção de x_R , conforme pode ser visualizado na Figura 9, tem-se que

$$\begin{cases} \phi_1 = \frac{x^R \operatorname{cos}(\pi/3)}{r} \\ \phi_2 = \frac{x^R \operatorname{cos}(\pi/3)}{r} \\ \phi_3 = -\frac{x^R}{r} \end{cases} . \quad (7)$$

Na Figura 10 é mostrado o desenho esquemático de rotação em torno do eixo que sai

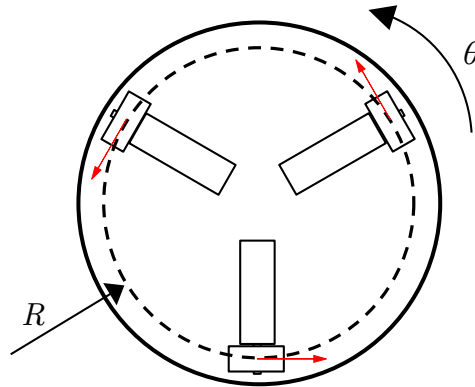
Figura 9 – Translação do robô na direção de x_R



Fonte: Próprio autor

do plano, isto é, uma variação, neste caso, um aumento, no ângulo θ . R é o raio da circunferência que passa no meio do comprimento das três rodas. O comprimento de arco que a plataforma rotaciona, $R\theta$, é igual ao deslocamento de cada roda. Sendo assim, as

Figura 10 – Rotação do robô em torno do próprio eixo



Fonte: Próprio autor

posições angulares, neste caso, são:

$$\begin{cases} \phi_1 = \frac{-R\theta}{r} \\ \phi_2 = \frac{-R\theta}{r} \\ \phi_3 = \frac{-R\theta}{r} \end{cases} . \quad (8)$$

Por fim, a soma das equações dos Sistemas 6, 7 e 8 resulta no caso geral, em que há deslocamentos e rotações. O Sistema de equações 9, é a solução explicita do problema da cinemática inversa.

$$\begin{cases} \phi_1 = \frac{x^R \cos(\pi/3) + y^R \sen(\pi/3) - R\theta}{r} \\ \phi_2 = \frac{x^R \cos(\pi/3) - y^R \sen(\pi/3) - R\theta}{r} \\ \phi_3 = \frac{-x^R - R\theta}{r} \end{cases} . \quad (9)$$

Por outro lado, sabendo ϕ_1 , ϕ_2 e ϕ_3 , também é possível encontrar x^R , y^R e θ resolvendo o Sistema 9. O Sistema 10 é a solução deste problema (cinemática direta):

$$\begin{cases} x^R = \frac{r(\phi_1 + \phi_2 - 2\phi_3)}{3} \\ y^R = \frac{\sqrt{3}r(\phi_1 - \phi_2)}{3} \\ \theta = \frac{-r(\phi_1 + \phi_2 + \phi_3)}{3R} \end{cases} . \quad (10)$$

2.3 MUDANÇA DE COORDENADAS

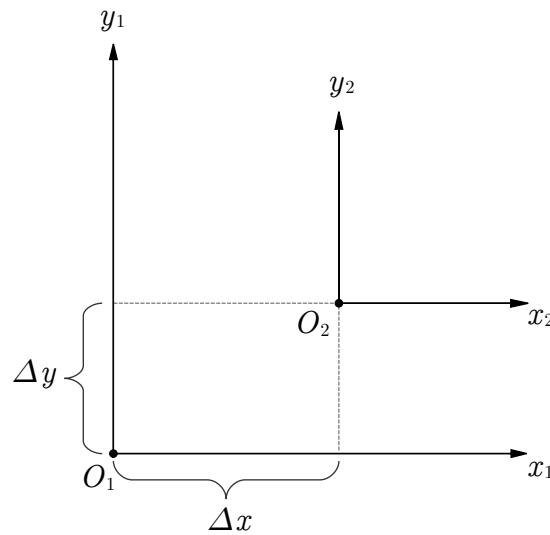
Além de determinar as posições que o robô deve percorrer em relação ao seu centro, é necessário ter estas coordenadas em relação a um sistema de coordenadas que

não se movimentará junto com o robô. Para isso, é feita uma mudança de sistema de coordenadas.

Neste caso, os dois sistemas, o que se movimenta junto com o robô (sistema local do robô) e o que está fixo em um local fora do robô (sistema global), são ortogonais, iguais ao da Seção 2.2, de forma que as mudanças envolvidas no controle serão de translação e rotação.

A translação é a mais simples. Considerando, por exemplo, o sistema de coordenadas $x_1O_1y_1$ da Figura 11, para transladá-lo até o ponto O_2 basta fazer (adaptado de Camargo e Boulos (2005)):

Figura 11 – Translação de sistema de coordenadas



Fonte: Próprio autor

$$\begin{cases} x_1 = \Delta x + x_2 \\ y_1 = \Delta y + y_2 \end{cases}, \quad (11)$$

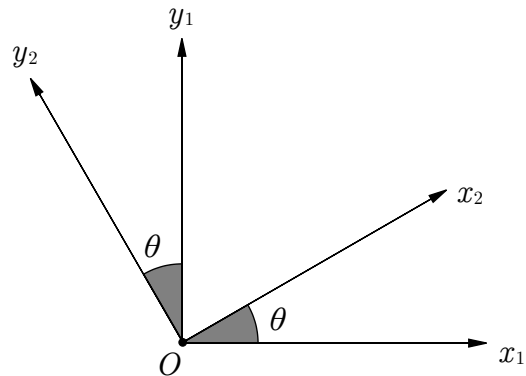
em que (x_1, y_1) e (x_2, y_2) são pontos nos sistemas de coordenadas $x_1O_1y_1$ e $x_2O_2y_2$, respectivamente.

Já a rotação tem equações ligeiramente mais complexas, conforme Sistema 12 (adaptado de Camargo e Boulos (2005)).

$$\begin{cases} x_1 = x_2 \cos \theta - y_2 \sin \theta \\ y_1 = x_2 \sin \theta + y_2 \cos \theta \end{cases}, \quad (12)$$

em que, conforme Figura 12, θ é o ângulo de rotação de um sistema de referência em relação ao outro.

Figura 12 – Rotação de sistema de coordenadas



Fonte: Próprio autor

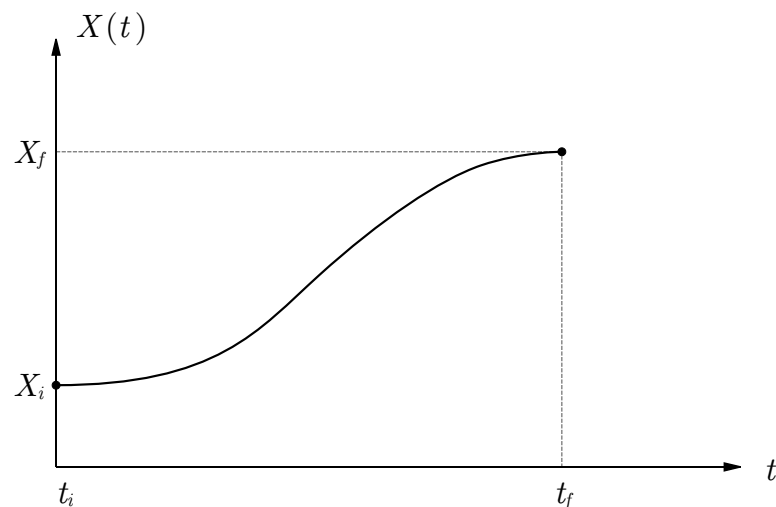
2.4 PLANEJAMENTO DE TRAJETÓRIA

Além de o robô se movimentar de um ponto a outro no espaço, é desejado que isso seja feito de maneira “suave”. Acompanhando Craig (2013), suave significa que a função que descreve a trajetória no tempo, assim como sua primeira derivada, é contínua. Considerando que a função é um polinômio, isso significa que trata-se de um polinômio do terceiro grau.

Duas outras características desejadas são que a velocidade do robô no início e fim da trajetória seja zero.

Na Figura 13 é mostrado o gráfico de uma possível função trajetória.

Figura 13 – Gráfico de uma função de trajetória no tempo



Fonte: Próprio autor

Conforme a descrição feita até aqui, quatro condições precisam ser satisfeitas em $X(t)$. São elas:

$$\begin{cases} X(t_i) = X_i \\ X(t_f) = X_f \\ \dot{X}(t_i) = 0 \\ \dot{X}(t_f) = 0 \end{cases} . \quad (13)$$

Considerando

$$X(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3, \quad (14)$$

o Sistema 13 fica da forma

$$\begin{cases} X_i = a_0 + a_1 t_i + a_2 t_i^2 + a_3 t_i^3 \\ X_f = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \\ 0 = a_1 + 2 a_2 t_i + 3 a_3 t_i^2 \\ 0 = a_1 + 2 a_2 t_f + 3 a_3 t_f^2 \end{cases} . \quad (15)$$

A solução de 15 é:

$$\begin{cases} a_0 = \frac{X_f(t_i^3 - 3t_i^2 t_f) + X_i(3t_i t_f^2 - t_f^3)}{(t_i - t_f)^3} \\ a_1 = \frac{6(X_f - X_i)(t_i t_f)}{(t_i - t_f)^3} \\ a_2 = \frac{3(X_i - X_f)(t_i + t_f)}{(t_i - t_f)^3} \\ a_3 = \frac{2(X_f - X_i)}{(t_i - t_f)^3} \end{cases} . \quad (16)$$

3 METODOLOGIA

As principais etapas deste trabalho podem ser resumidas e listadas, conforme ordem de execução real, da seguinte maneira:

1. Projeto e construção do robô móvel;
2. Fechamento de malha com controle PID dos motores;
3. Geração das trajetórias e cálculo das posições desejadas dos motores ao longo do tempo;
4. Elaboração de um algoritmo para execução da trajetória;
5. Testes e coleta de dados para avaliação dos resultados.

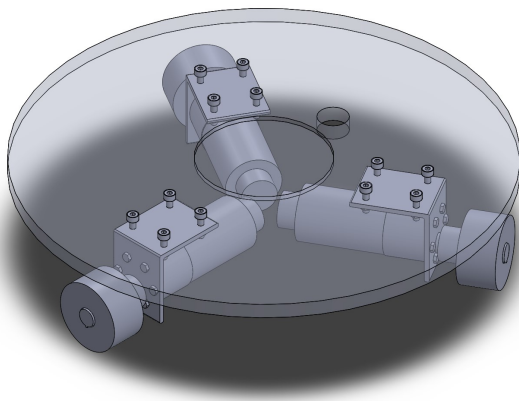
A seguir são apresentadas com mais detalhes o que foi feito em cada parte.

3.1 CONSTRUÇÃO DO ROBÔ MÓVEL

Na primeira etapa foi feito o modelo em software CAD do robô desejado, já levando em conta os materiais disponíveis para sua construção, e os desenhos técnicos das peças a serem usinadas.

Na Figura 14 é mostrado o visual do robô nas versões 3D em CAD e real. O diâmetro do disco de acrílico (“base” do robô) foi escolhido de forma que as rodas não sejam danificadas caso haja uma colisão. Um pequeno furo deslocado do centro da base foi feito para passar os cabos elétricos aos motores e *encoders*. O rebaixo no centro foi feito para centralizar os componentes colocados sobre a base, mas não foi utilizado aqui.

Figura 14 – Visual do robô



(a) Em software CAD

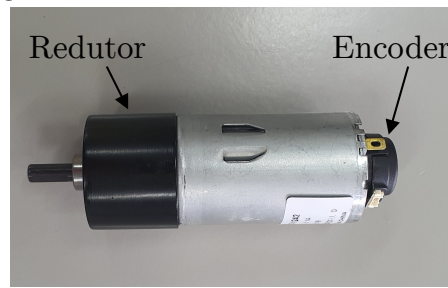


(b) Real

Os motores e rodas foram disponibilizados pelo professor e orientador deste projeto, assim como a placa de acrílico utilizada para fabricação da base do robô.

O motor utilizado é de corrente contínua (CC) com escovas. O fabricante é MS MOTOR e o modelo 37JB6K18G/3657-1242. Valores nominais de tensão, potência e rotação são 12 V, 7 W e 3600 rpm, respectivamente. Na sua parte de trás há um *encoder*, já acoplado ao eixo do motor, do fabricante US DIGITAL, modelo E4P. Este *encoder* gera 360 pulsos a cada rotação do motor, tornando conveniente a medida dos ângulos em graus. No entanto, deve-se levar em conta o redutor planetário, com relação de transmissão 1/18. Assim, cada pulso é equivalente a 18° de volta do eixo do redutor (onde vai acoplada a roda) e 1 volta completa é igual a $360 \cdot 18 = 6480$ pulsos. O motoredutor é mostrado na Figura 15.

Figura 15 – Motoredutor CC utilizado



Fonte: Próprio autor

Para visualização dos roletes responsáveis pelo movimento na direção axial do motor, são mostradas algumas fotos da roda utilizada na Figura 16.

Figura 16 – Roda utilizada no robô



Fonte: Próprio autor

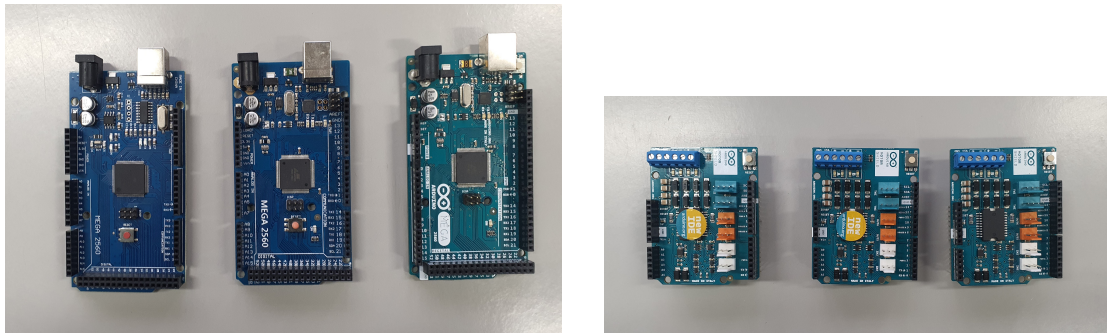
Os suportes que fixam os motores na base de acrílico foram usinados a partir de cantoneiras de alumínio, adquiridas no comércio local. A base e os suportes foram usinados no laboratório de fabricação do departamento de Engenharia Mecânica da UFSCar, por um dos técnicos que trabalham lá.

Por fim, os acoplamentos para as rodas/motores foram fabricados em impressora 3D, disponibilizada pelo professor orientador, que instruiu o aluno quanto ao processo.

Para o controle, foram utilizados três Arduinos MEGA e três módulos *drivers* de motor CC, um para cada Arduino. Na Figura 17 são mostrados os Arduinos e módulos.

O uso de três Arduinos, e não apenas um, é justificado pela necessidade de processamento simultâneo dos sinais recebidos pelos encoders. Um único Arduino (ou mesmo dois) se mostrou incapaz de lidar com a tarefa sozinho.

Figura 17 – Componentes do controlador



(a) Placas de Arduino MEGA

(b) Módulos *drivers* de motor

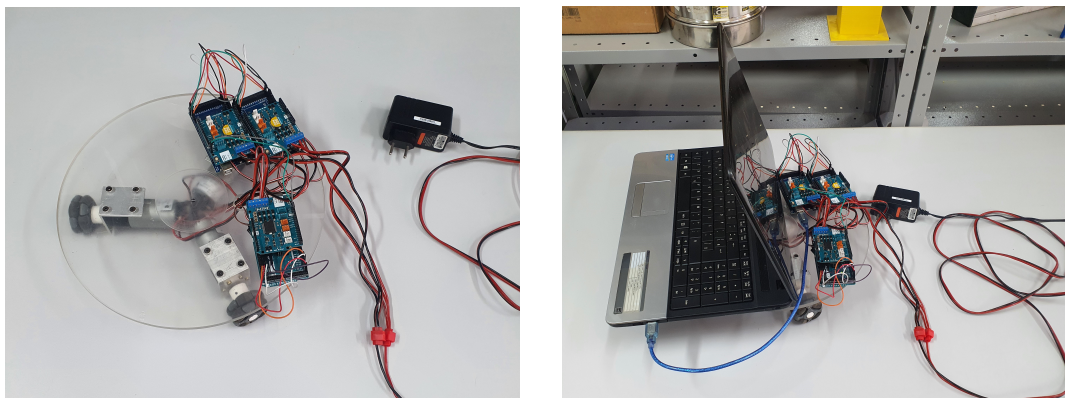
Fonte: Próprio autor

Durante o funcionamento do robô, os Arduinos, com os *drivers* acoplados, ficam sobre a base de acrílico e uma fonte CC 12 Volts alimenta o circuito.

Para coleta de dados, um dos Arduinos é conectado a um notebook. Os dados são recebidos via Serial no próprio Arduino IDE, programa onde são compilados os códigos, para depois serem salvos e tratados com o software Matlab. O notebook não participa do processamento da rotina programada, no entanto, é importante ressaltar que a rota não é recalculada em tempo real, o que está além do escopo deste trabalho.

Na Figura 18 é mostrado o robô montado com os componentes do controlador. Os Arduinos e *drivers* estão dispostos de forma que o espaço restando permita sustentar o notebook sobre o robô.

Figura 18 – Robô montado com o controlador



(a) Sem notebook

(b) Com notebook

Fonte: Próprio autor

3.2 AJUSTE DO CONTROLE PID

O método de ajuste utilizado é o Ziegler-Nichols, conforme apresentado na Seção 2.1. Neste caso, foi escolhido encontrar K_U diretamente e não a partir da curva de reação do processo.

Na Figura 19 são mostrados os gráficos do sinal de resposta no momento que $K_P = K_U$. O ângulo programado para os motores atingirem foi 90° .

Os valores de K_U e P_U são mostrados na Tabela 2. A unidade do ganho proporcional (e de K_U) é $[V/^\circ]$, mas a tensão se refere ao valor de entrada na função “analogWrite” do Arduino, que aceita valores de 0 até 255.

Tabela 2 – Valores dos ganhos e períodos críticos

| Motor | K_U | $P_U[s]$ |
|-------|-------|----------|
| 1 | 48 | 0,109 |
| 2 | 56 | 0,098 |
| 3 | 65 | 0,097 |

Fonte: Próprio autor

A identificação dos motores é mostrada na Figura 20.

Com os valores encontrados foram calculados os ganhos para controle, conforme fórmulas da Tabela 1. Na Tabela 3 a seguir estão os ganhos encontrados para cada motor do robô.

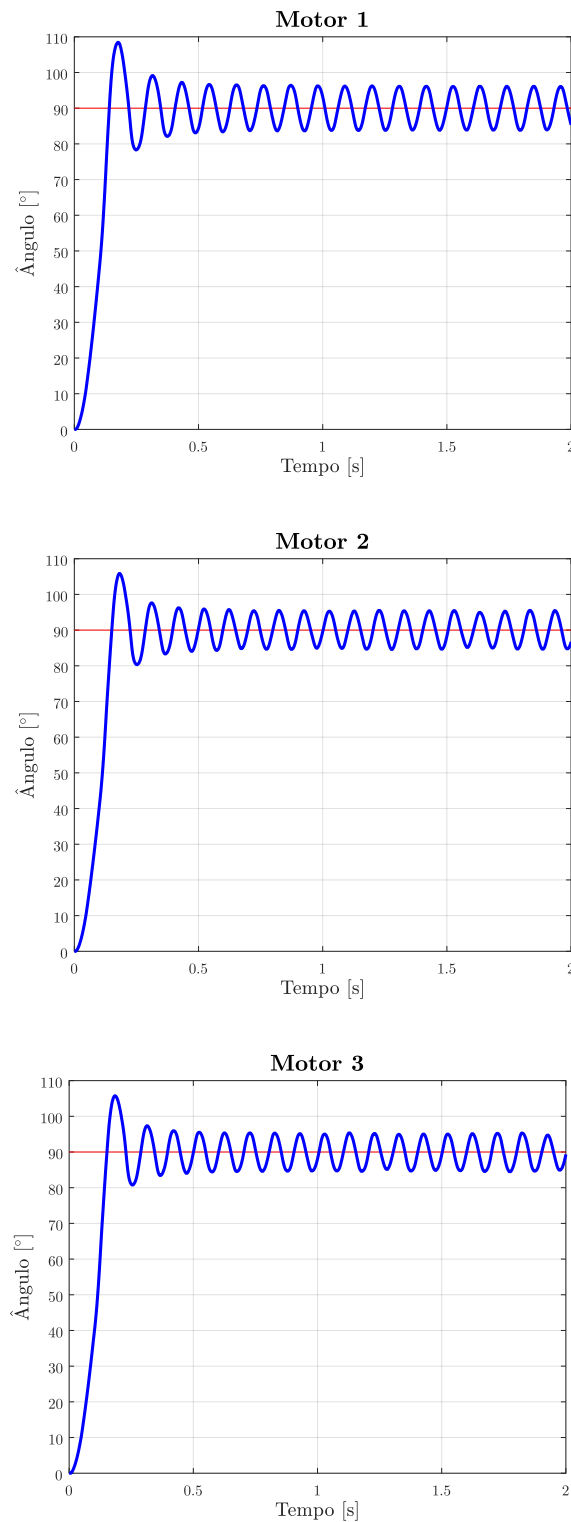
Tabela 3 – Ganhos dos motores pelo método Ziegler-Nichols

| Motor | Tipo de controle | K_P | K_I | K_D |
|-------|------------------|-------|-------|-------|
| 1 | P | 24 | 0 | 0 |
| | PI | 21,6 | 237 | 0 |
| | PID | 28,8 | 526,5 | 0,4 |
| 2 | P | 28 | 0 | 0 |
| | PI | 25,2 | 308,6 | 0 |
| | PID | 33,6 | 685,7 | 0,41 |
| 3 | P | 32,5 | 0 | 0 |
| | PI | 29,25 | 361,9 | 0 |
| | PID | 39 | 804,1 | 0,47 |

Fonte: Próprio autor

Como T_I e T_D são tempos, as unidades de $K_I = K_P/T_I$ e $K_D = K_P T_D$ são $[V/^\circ s]$ e $[V s/^\circ]$, respectivamente. Na Figura 21, estão os gráficos com os sinais obtidos utilizando os ganhos da Tabela 3.

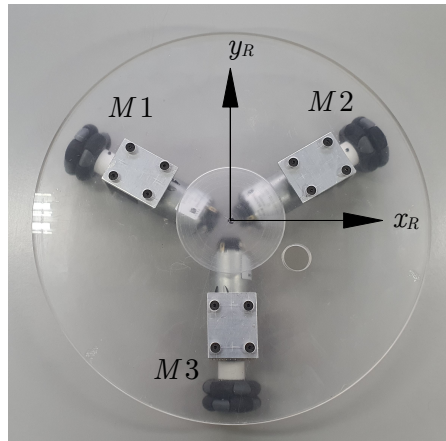
Figura 19 – Respostas com ganho crítico



Fonte: Próprio autor

Foi necessário, no entanto, para obter um resultado satisfatório, adicionar um novo tipo de controle no qual são usados os valores de ganhos PID, mas com $K_I = 0$, chamado de controle PD. Este foi o controle utilizado, pois estabilizou os motores mais

Figura 20 – Identificação dos motores do robô



Fonte: Próprio autor

rapidamente e com quase nenhum sobressinal.

É importante ressaltar que as derivadas e integrais dos erros foram discretizadas para que fosse possível o cálculo, isto é:

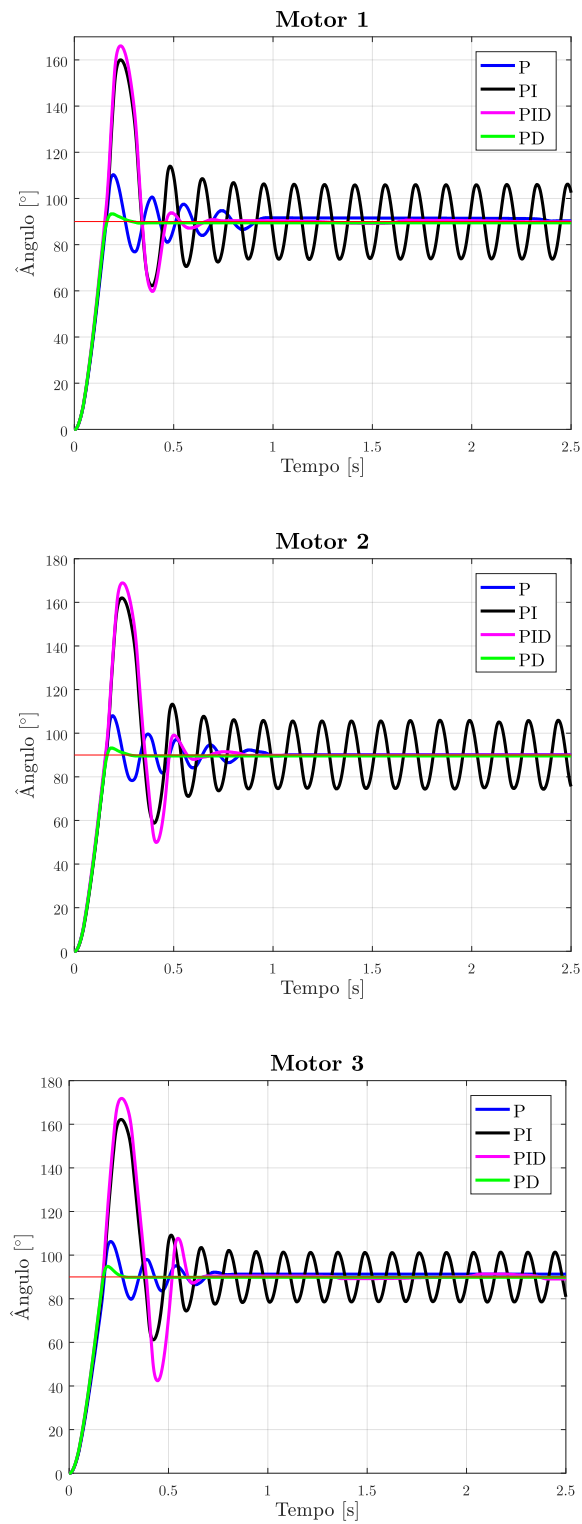
$$\frac{de(t)}{dt} \approx \frac{e(t_i) - e(t_{i-1})}{t_i - t_{i-1}} \quad (17)$$

e

$$\int e(t) dt \approx \frac{[e(t_i) + e(t_{i-1})] (t_i - t_{i-1})}{2}, \quad (18)$$

em que, o índice i se refere à iteração atual e $i - 1$ à iteração anterior.

Figura 21 – Respostas com vários ajustes do controlador



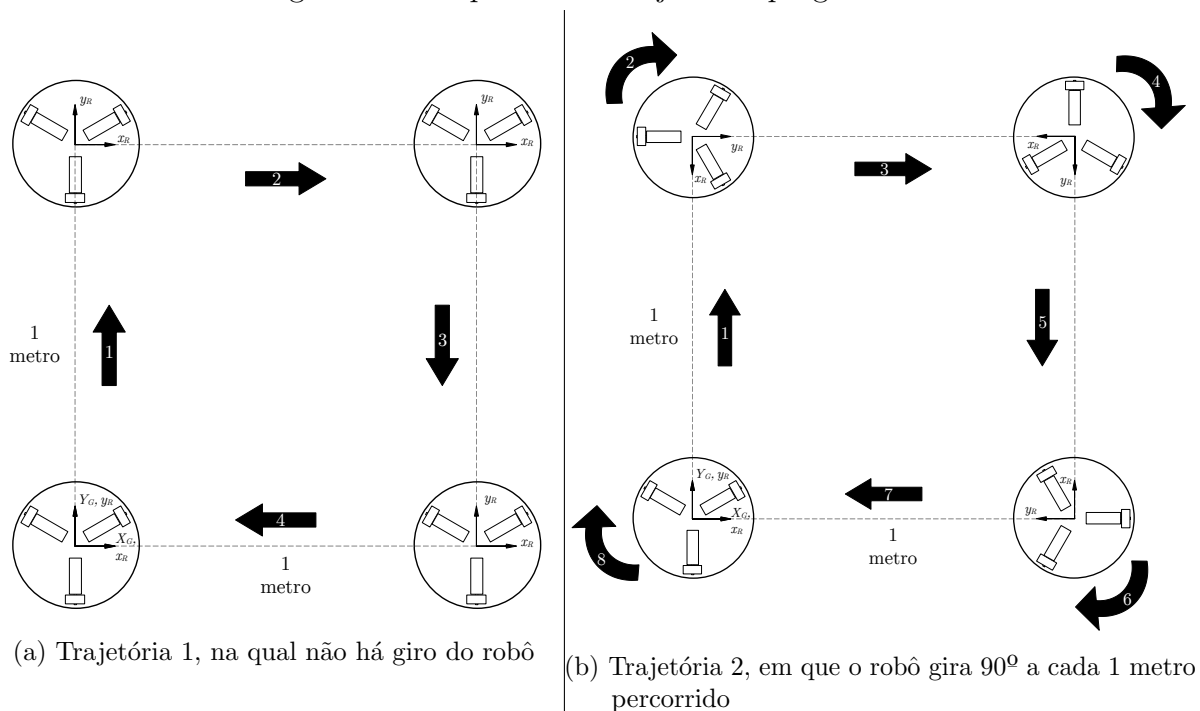
Fonte: Próprio autor

3.3 DETERMINAÇÃO DAS TRAJETÓRIAS E POSIÇÕES A SEREM PERCORRIDAS

Neste trabalho são consideradas duas trajetórias. Na primeira, o robô não rotaciona ao redor do próprio eixo. Já na segunda, ocorre a rotação após um trecho em linha reta. Em ambas as rotas o robô percorre o perímetro de um quadrado de lado unitário, em metros.

Na Figura 22 é mostrado um desenho esquemático do robô percorrendo cada rota. O sistema de coordenadas globais que, do ponto de vista do observador, está fixo no lugar onde se inicia (e finaliza) a trajetória, é denominado por (X_G, Y_G) .

Figura 22 – Esquema das trajetórias programadas



Fonte: Próprio autor

Nas Tabelas 4 e 5 estão as posições conhecidas em função do tempo para as trajetórias 1 e 2, respectivamente.

Tabela 4 – Valores inicialmente definidos para a trajetória 1

| Tempo [s] | X_G [m] | Y_G [m] | θ [rad] |
|-----------|-----------|-----------|----------------|
| 0 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 |
| 8 | 1 | 1 | 0 |
| 12 | 1 | 0 | 0 |
| 16 | 0 | 0 | 0 |

Fonte: Próprio autor

Tabela 5 – Valores inicialmente definidos para a trajetória 2

| Tempo [s] | X_G [m] | Y_G [m] | θ [rad] |
|-----------|-----------|-----------|----------------|
| 0 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 |
| 6 | 0 | 1 | $0,5\pi$ |
| 10 | 1 | 1 | $0,5\pi$ |
| 12 | 1 | 1 | π |
| 16 | 1 | 0 | π |
| 18 | 1 | 0 | $1,5\pi$ |
| 22 | 0 | 0 | $1,5\pi$ |
| 24 | 0 | 0 | 2π |

Fonte: Próprio autor

Utilizando as fórmulas da Seção 2.4, foram calculados os coeficientes dos polinômios de cada trecho das trajetórias ao longo do tempo. Nas Tabelas 6 e 7 são mostrados os coeficientes das trajetórias 1 e 2, respectivamente. Na trajetória 1 é omitido $\theta(t)$, pois este é sempre igual a zero.

Tabela 6 – Coeficientes da trajetória 1

| Função | Coeficiente | Tempo [s] | | | |
|----------|-------------|-----------|---------|---------|---------|
| | | 0 a 4 | 4 a 8 | 8 a 12 | 12 a 16 |
| $X^G(t)$ | a_0 | 0 | 5 | 1 | -80 |
| | a_1 | 0 | -3 | 0 | 18 |
| | a_2 | 0 | 0,5625 | 0 | -1,3125 |
| | a_3 | 0 | -0.0313 | 0 | 0.0313 |
| $Y^G(t)$ | a_0 | 0 | 1 | -27 | 0 |
| | a_1 | 0 | 0 | 9 | 0 |
| | a_2 | 0,1875 | 0 | -0,9375 | 0 |
| | a_3 | -0,0313 | 0 | 0,0313 | 0 |

Fonte: Próprio autor

Nas Figuras 23 e 24 são mostradas as funções ao longo do tempo das trajetórias 1 e 2, respectivamente.

Utilizando as equações de mudança de coordenadas da Seção 2.3, foram calculadas as coordenadas do robô no sistema móvel.

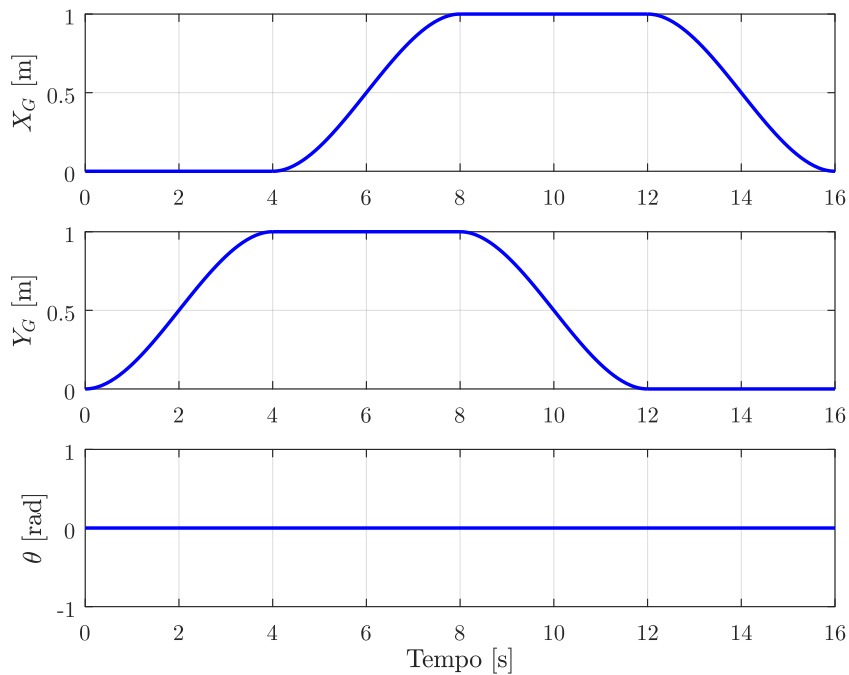
Um detalhe importante é que, da forma como foram calculados os ângulos de rotação dos motores, não é necessário fazer essa mudança no caso da trajetória 1. Seria necessário caso os ângulos fossem zerados em cada iteração do controlador. Em vez disso, optou-se por considerar as iterações passadas acrescentando a posição que deseja-se alcançar no momento. Conseqüentemente, quando o robô não gira em torno do próprio eixo, as coordenadas no início e no fim da trajetória para um mesmo ponto no plano são iguais, da perspectiva do controlador. Naturalmente, o ângulo de rotação do robô, θ , também não

Tabela 7 – Coeficientes da trajetória 2

| $f(t)$ | Coef. | Tempo [s] | | | | | | | |
|-------------|-------|-----------|--------|--------|---------|---------|---------|---------|---------|
| | | 0 a 4 | 4 a 6 | 6 a 10 | 10 a 12 | 12 a 16 | 16 a 18 | 18 a 22 | 22 a 24 |
| $X^G(t)$ | a_0 | 0 | 0 | 13,5 | 1 | 1 | 1 | -242 | 0 |
| | a_1 | 0 | 0 | -5,625 | 0 | 0 | 0 | 37,13 | 0 |
| | a_2 | 0 | 0 | 0,75 | 0 | 0 | 0 | -1,875 | 0 |
| | a_3 | 0 | 0 | -0,031 | 0 | 0 | 0 | 0,031 | 0 |
| $Y^G(t)$ | a_0 | 0 | 1 | 1 | 1 | -80 | 0 | 0 | 0 |
| | a_1 | 0 | 0 | 0 | 0 | 18 | 0 | 0 | 0 |
| | a_2 | 0,188 | 0 | 0 | 0 | -1,313 | 0 | 0 | 0 |
| | a_3 | -0,031 | 0 | 0 | 0 | 0,031 | 0 | 0 | 0 |
| $\theta(t)$ | a_0 | 0 | -43,98 | -1,571 | -512,08 | -3,142 | -1913,2 | -4,712 | -4756,4 |
| | a_1 | 0 | 28,27 | 0 | 141,4 | 0 | 339,292 | 0 | 622,03 |
| | a_2 | 0 | -5,891 | 0 | -12,96 | 0 | -20,03 | 0 | -27,1 |
| | a_3 | 0 | 0,393 | 0 | 0,393 | 0 | 0,393 | 0 | 0,393 |

Fonte: Próprio autor

Figura 23 – Posições do robô para a trajetória 1



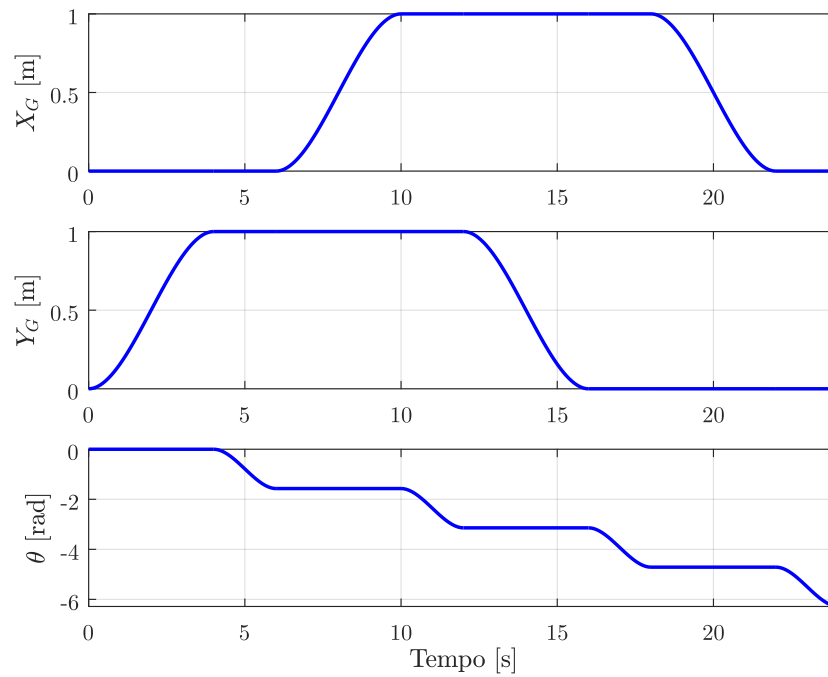
Fonte: Próprio autor

muda com o sistema de coordenadas.

Isso simplifica o controle de posição, pois não é necessário se preocupar com o acúmulo de erros a cada iteração para saber o erro da iteração atual.

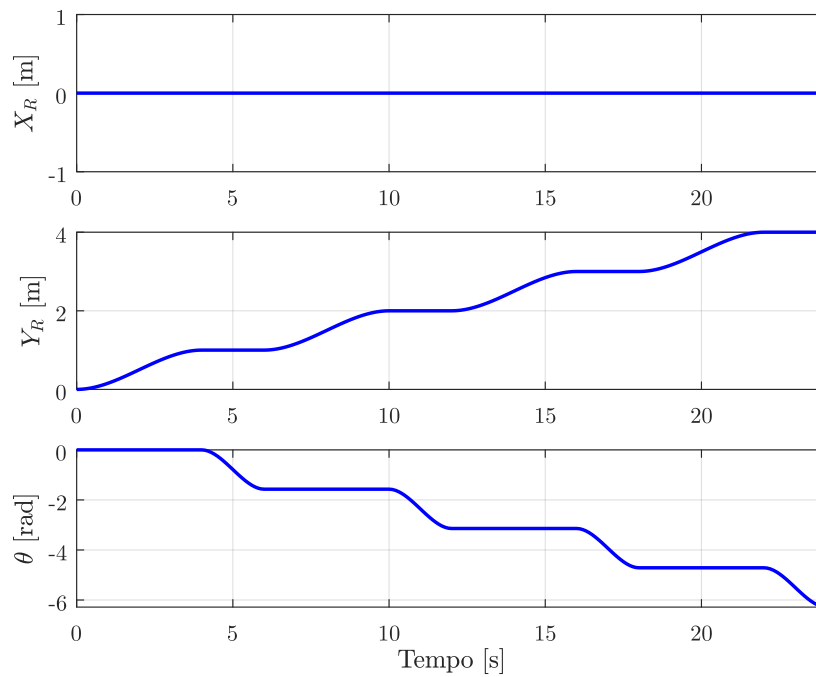
Dessa forma, as coordenadas na Figura 25 indicam as posições atuais mais as que já foram percorridas.

Figura 24 – Posições do robô para a trajetória 2 no sistema de coordenadas global



Fonte: Próprio autor

Figura 25 – Posições do robô para a trajetória 2 no sistema de coordenadas local do robô



Fonte: Próprio autor

Esse sistema é denominado (X_R, Y_R) para diferenciar do (x_R, y_R) , mostrado nas

figuras anteriores. Matematicamente

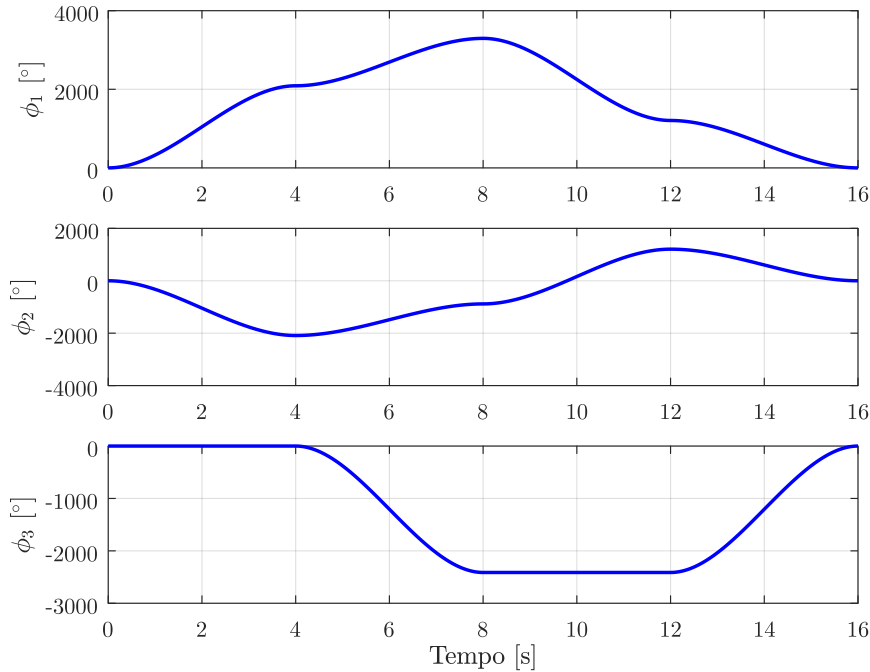
$$\begin{cases} X_i^R = X_{i-1}^R + x_i^R = X_{i-1}^R + (X_i^G - X_{i-1}^G) \cos \theta_i + (Y_i^G - Y_{i-1}^G) \sin \theta_i \\ Y_i^R = Y_{i-1}^R + y_i^R = Y_{i-1}^R - (X_i^G - X_{i-1}^G) \sin \theta_i + (Y_i^G - Y_{i-1}^G) \cos \theta_i \end{cases}, \quad (19)$$

em que os índices i e $i - 1$ indicam as iterações atual e anterior, respectivamente. Como já explicado, na trajetória 1 tem-se $(x_R, y_R) = (X_R, Y_R)$, pois $\theta = 0$, sendo desnecessária a conversão com estas equações.

Após definir os caminhos que o robô deve percorrer, foram calculados os ângulos que os motores precisam rotacionar em cada instante de tempo. As equações utilizadas foram as de cinemática inversa da Seção 2.2, exceto que no lugar de (x^R, y^R) foi utilizado (X^R, Y^R) . Foi considerado que o comprimento do centro do robô até o meio das rodas é $R = 138$ mm. Esta medida também não é necessária para o cálculo da trajetória 1.

Nas Figuras 26 e 27 são mostrados os gráficos com os ângulos dos motores, já convertidos em graus, para as trajetórias 1 e 2, respectivamente.

Figura 26 – Posições angulares dos motores na trajetória 1

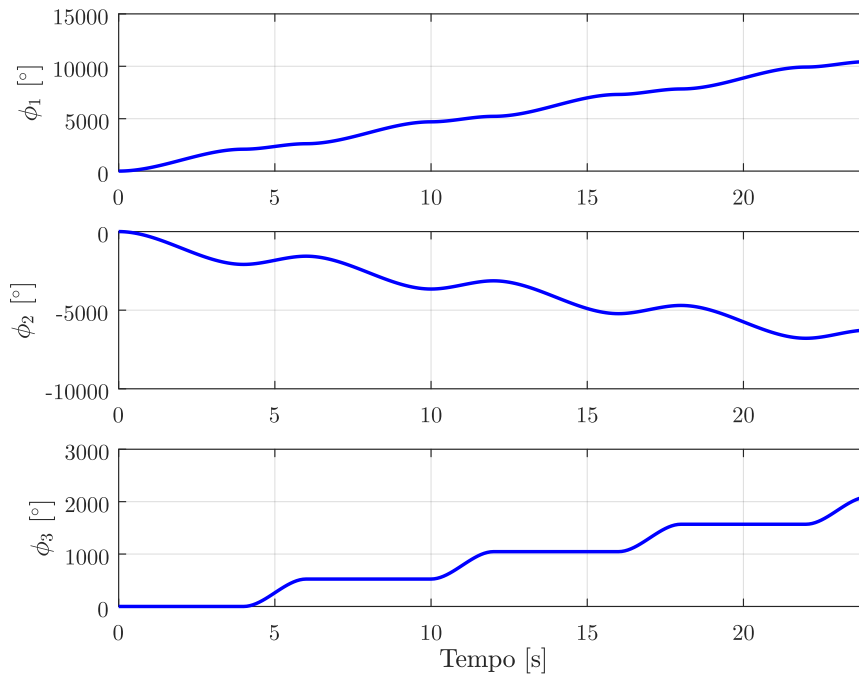


Fonte: Próprio autor

3.4 PROGRAMAÇÃO DO CONTROLADOR

Como explicado na Seção 3.1, foram utilizados três Arduinos para controlar o robô. Em cada um deles foram salvos, em forma de *arrays*, os ângulos mostrados nas Figuras 26 e 27. Para economizar memória, o intervalo de tempo entre cada instância dos vetores

Figura 27 – Posições angulares dos motores na trajetória 2

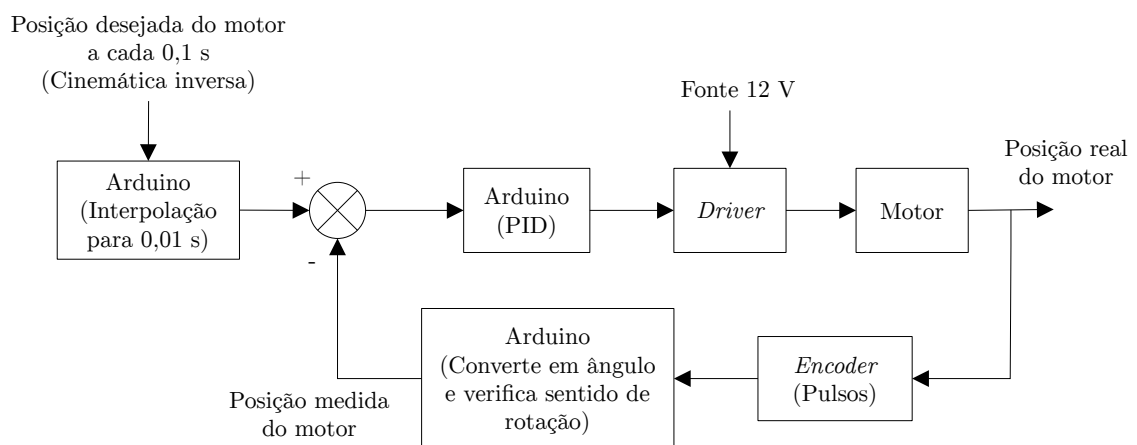


Fonte: Próprio autor

foi definido como 0,1 s, num primeiro momento insuficiente para garantir o movimento contínuo dos motores. A solução escolhida foi interpolar os valores salvos a cada iteração, de forma que, na prática a posição angular dos motores foi definida para cada 0,01 s.

Na Figura 28 é mostrado o diagrama de blocos do controle utilizado em cada motor.

Figura 28 – Diagrama de blocos do controle do motor



Fonte: Próprio autor

Outra consideração importante em relação a programação é como os Arduinos se comunicam. Um deles é responsável por enviar os dados coletados para o computador e

também de receber o comando para início ou fim da rotina. Este Arduino é considerado o “produtor”; os outros dois são os “clientes”. Apesar do nome, as três placas controlam, cada uma um motor. A diferença é que os escravos não se comunicam diretamente, apenas com o mestre. A comunicação é serial via protocolo I2C.

3.5 TESTES E COLETA DE DADOS

Programadas as duas trajetórias, o robô foi posto em teste. Para cada rota foram feitas 10 execuções e calculada a média. Com este resultado foram calculadas as posições do robô usando as equações de cinemática direta e de mudança de coordenadas.

A cinemática direta é resolvida com o Sistema 10, apenas substituindo (x^R, y^R) por (X^R, Y^R) . Já a mudança de coordenadas é igual a do Sistema 19, mas com outras variáveis em evidência, isto é:

$$\begin{cases} X_i^G = X_{i-1}^G + (X_i^R - X_{i-1}^R) \cos \theta_i - (Y_i^R - Y_{i-1}^R) \sin \theta_i \\ Y_i^G = Y_{i-1}^G + (X_i^R - X_{i-1}^R) \sin \theta_i + (Y_i^R - Y_{i-1}^R) \cos \theta_i \end{cases} . \quad (20)$$

Assim é possível comparar a trajetória planejada e a executada, o que é feito no próximo capítulo.

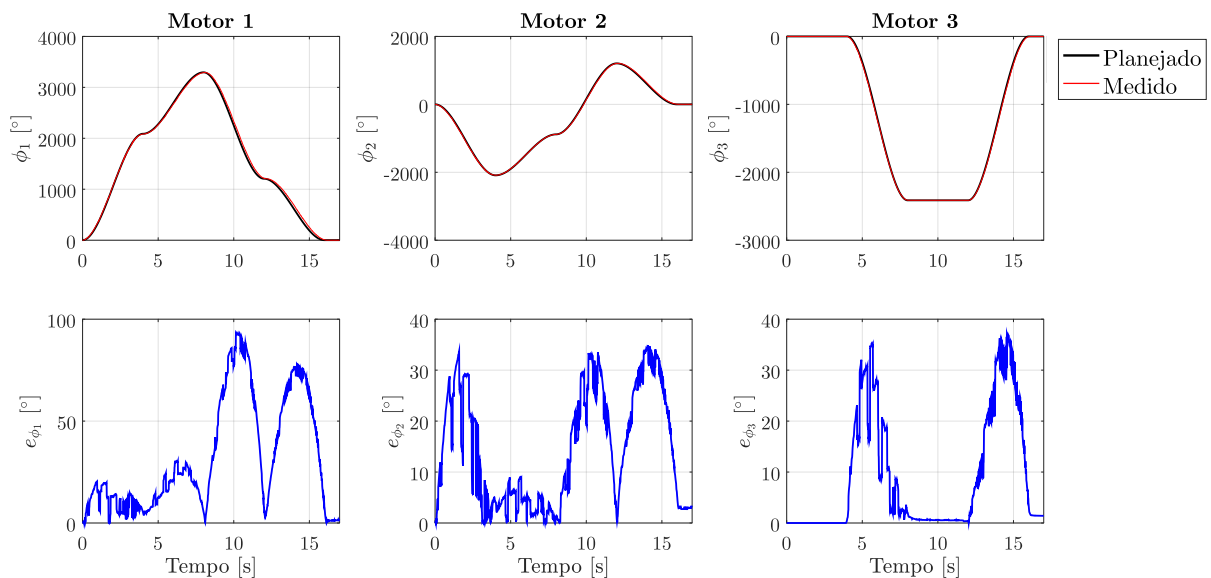
4 RESULTADOS

Os resultados dos testes de execução de trajetória estão apresentados em gráficos a seguir. Para facilitar a leitura, cada trajetória está em uma seção diferente.

4.1 TRAJETÓRIA 1

Na Figura 29 são mostrados os gráficos dos ângulos dos motores e o respectivo erro absoluto, isto é, $e_{\phi_i} = |\phi_i^d - \phi_i^m|$, sendo ϕ_i^d o valor desejado e ϕ_i^m o valor medido, todos em graus.

Figura 29 – Posição dos motores durante trajetória 1



Fonte: Próprio autor

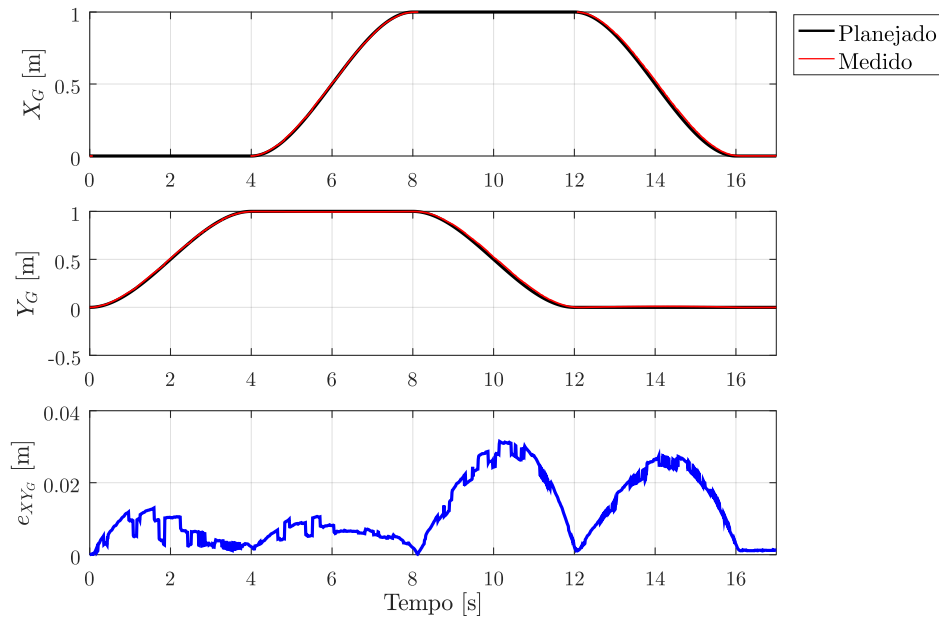
Três observações podem ser feitas sobre os gráficos. A primeira é que os resultados obtidos são quase idênticos aos desejados. Apesar disso (segunda observação), diferenças relevantes são encontradas, especialmente no Motor 1, que chega a ter um erro de mais de 90° . Além disso, todos os motores estabilizam após o tempo final planejado para a rota, 16 s, motivo pelo qual os gráficos foram estendidos até os 17 s, embora não haja no final da trajetória um erro acumulado alto.

A terceira observação é que os erros aumentam nos momentos que os motores aceleram e diminuem conforme a velocidade estabiliza, como é de se esperar. Por conta de as trajetórias serem polinômios de terceiro grau, as acelerações não são contínuas, embora limitadas. Isso, certamente, contribui para que ocorram picos nos gráficos de erro.

Nas Figuras 30, e 31 são mostrados os gráficos de posição do robô com seus respectivos erros absolutos (análogos aos dos motores), sendo $e_{XY_G} = \sqrt{e_{X_G}^2 + e_{Y_G}^2}$. Aqui podem ser notados alguns padrões. primeiro, os erros diminuem (praticamente zeram) nas

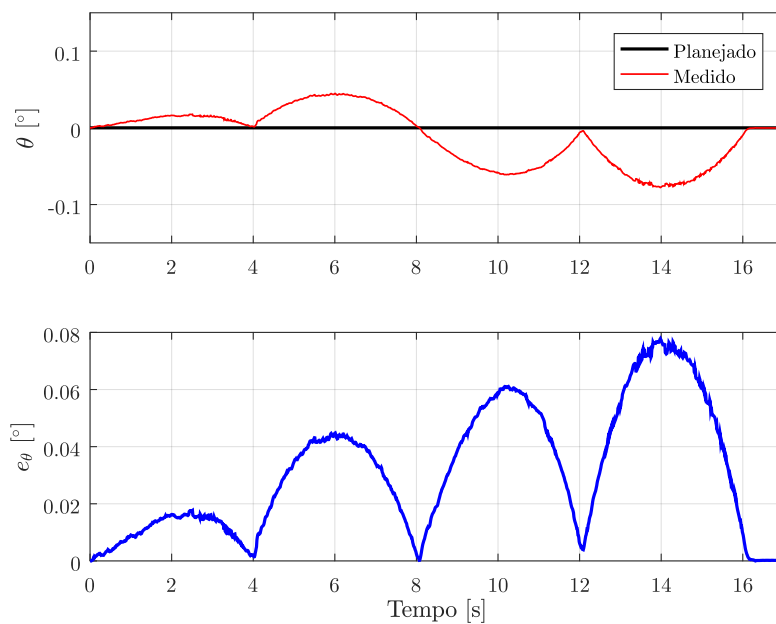
paradas para mudança de direção da trajetória. Além disso, nos momentos que o robô é programado para se movimentar em uma direção, é, naturalmente, nessa direção que ocorrem os maiores erros. Conseqüentemente, e_θ é bem menos expressivo que os demais.

Figura 30 – Posição cartesiana do robô durante trajetória 1



Fonte: Próprio autor

Figura 31 – Posição angular do robô durante trajetória 1



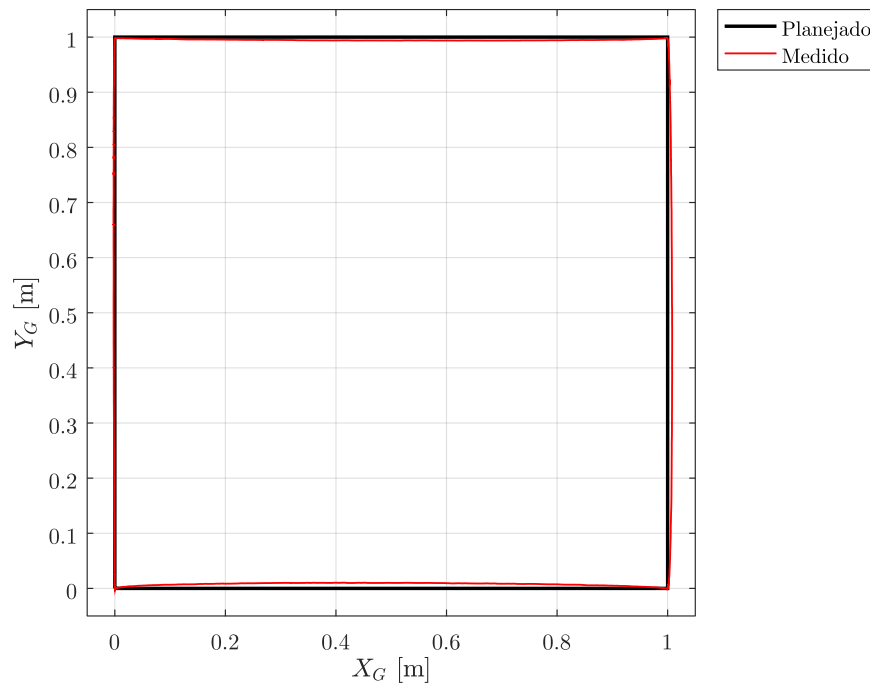
Fonte: Próprio autor

Por fim, os erros chegam, em coordenadas cartesianas, a cerca de 3 cm; valor

relevante, considerando as dimensões do robô.

Para finalizar a apresentação dos resultados relativos a trajetória 1, é mostrado na Figura 32 o gráfico de Y_G versus X_G , possibilitando visualizar o caminho que o robô executou e compará-lo ao desejado.

Figura 32 – Y_G versus X_G na trajetória 1



Fonte: Próprio autor

4.2 TRAJETÓRIA 2

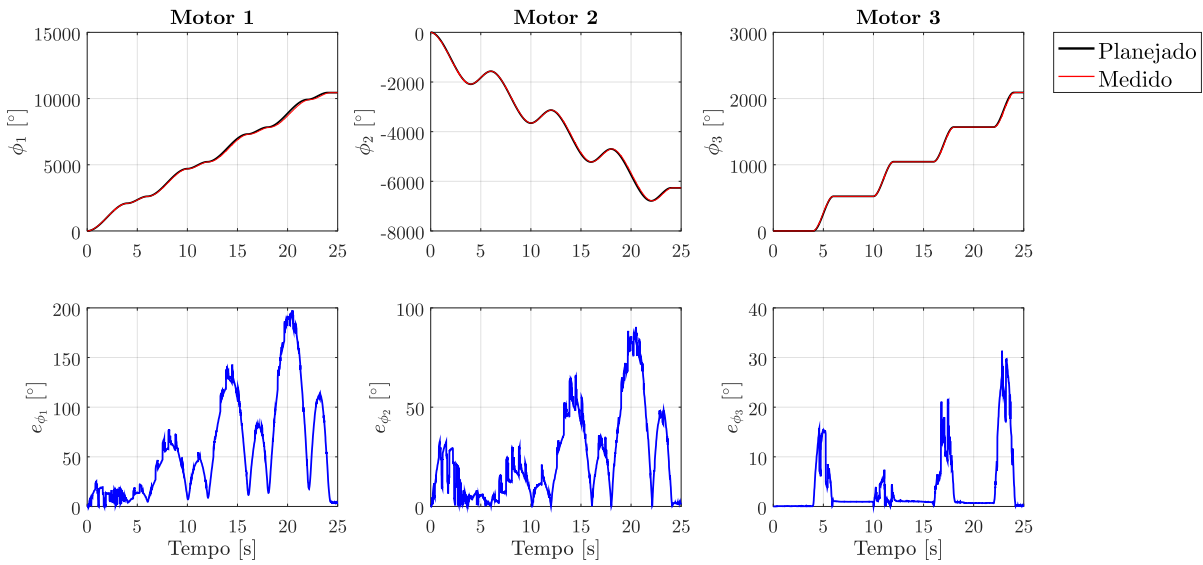
Nas Figura 33 são mostrados os gráficos das posições angulares dos motores em função do tempo na trajetória 2.

Pode ser observado o mesmo comportamento que o visto na trajetória 1: os erros tendem a aumentar nos momentos de aceleração, no entanto, os valores máximos aqui são maiores, chegando a 200° no Motor 1, novamente apresentando os maiores erros.

Nas Figuras 34 e 35 são mostrados os gráficos de posição do robô nos sistemas de coordenadas local (lembrando que as posições são acumuladas a cada iteração) e global. No sistema local os erros são bastante parecidos com os vistos no caso da trajetória 1. Entretanto, há uma situação diferente no global: os erros parecem de fato se acumular conforme o tempo passa, sendo possível notar um erro considerável após o tempo de finalização da rota, 24 s. Na Figura 36 é possível ver que o robô termina fora da posição em que começou a rota, mais precisamente, a 7,25 cm de distância.

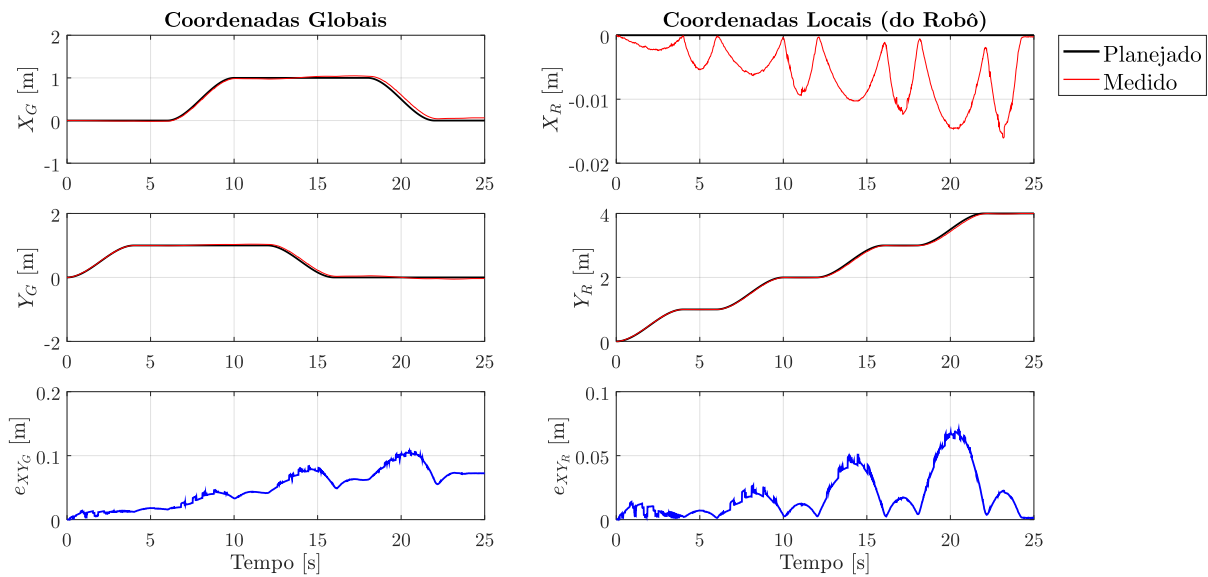
Uma explicação para esse acontecimento pode ser a seguinte: em cada iteração do controle, o robô fica ligeiramente fora da posição desejada, no entanto, como já foi

Figura 33 – Posição dos motores durante trajetória 2



Fonte: Próprio autor

Figura 34 – Posição cartesiana do robô durante trajetória 2

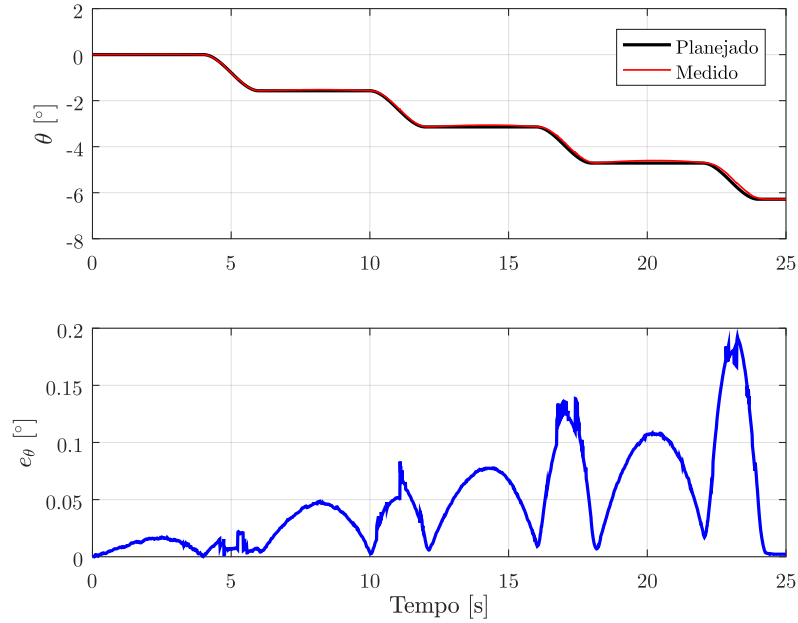


Fonte: Próprio autor

percebido nos gráficos dos ϕ_i , conforme os motores desaceleram, a tendência é zerar os erros: o controlador recupera o atraso e corrige a posição no momento. Posto de outra forma, o robô atinge as posições desejadas, mas não exatamente nos momentos desejados; há um atraso. Este atraso causa o desvio da trajetória quando vista em coordenadas globais, mas não no sistema local, pois, conforme equações do sistema 19, X_R e Y_R levam em consideração as iterações passadas. Isso também explica porque na trajetória 1, que tem movimentos mais difíceis de executar comparada com a trajetória 2, em que o robô sempre anda para frente, os erros se aproximam de zero no final da rotina: na trajetória 1,

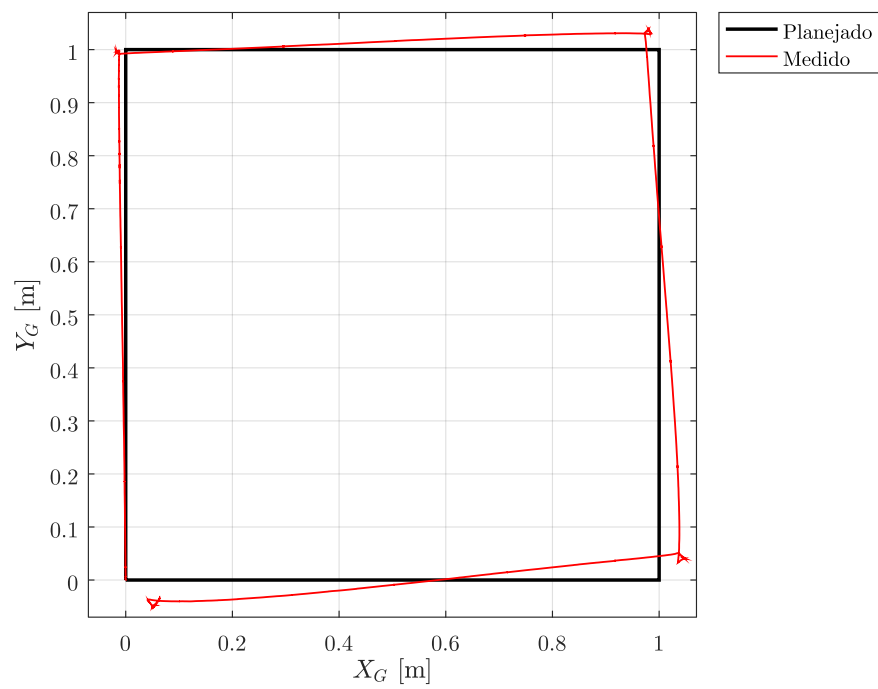
ao contrário da 2, $(X_R, Y_R) = (X_G, Y_G)$.

Figura 35 – Posição angular do robô durante trajetória 2



Fonte: Próprio autor

Figura 36 – Y_G versus X_G na trajetória 2



Fonte: Próprio autor

5 CONCLUSÃO

Neste trabalho foi construído um robô móvel omnidirecional com três graus de liberdade. Foi desenvolvido o controle de posição PID além das equações de cinemática e duas rotas predefinidas para teste prático. Os resultados mostram alguns problemas que não foram abordados, como a dinâmica, isto é, as forças atuantes no robô, em que poderia-se estudar o deslizamento das rodas omnidirecionais, e , possivelmente até mais relevante em termos de resultados finais, não foi desenvolvido nenhuma forma de correção direta de posição do robô no sistema global de coordenadas.

Apesar disso, os objetivos iniciais do trabalho foram alcançados. O controle de posição dos motores foi suficiente bom para executar as rotas de maneira quase perfeita a olho nu, especialmente no caso em que o robô não gira em torno do próprio eixo (trajetória 1).

Pode-se ainda, comparar os erros encontrados com o de um trabalho bastante semelhante, como o de Sofwan et al. (2019): as diferenças de posição foram de dezenas de centímetros. Curiosamente, os autores utilizaram uma malha “externa” para a posição do robô, o que deveria corrigir o problema que este projeto apresentou, no entanto, provavelmente, por terem feito o controle dos motores pela velocidade, tiveram resultados significativamente piores que os apresentados aqui.

Caso decida-se continuar o projeto, o primeiro passo é considerar no controle esta situação de giro do robô e, depois, fazer a integração com algum sensor para que o robô possa perceber o ambiente. A partir disso, será possível a adaptação das rotas para obstáculos que aparecerem.

Outro ponto que pode ser melhorado é a função que descreve as trajetórias. Usando-se polinômios de graus mais altos, por exemplo, pode-se conseguir movimentos ainda mais suaves.

Com todas essas implementações, mas não se limitando nelas, se terá um robô móvel que poderia ser utilizado em aplicações de transporte de peças na indústria ou em atividades domésticas de limpeza, por exemplo.

REFERÊNCIAS

- CAMARGO, I. de; BOULOS, P. Mudança de sistema de coordenadas. In: _____. **Geometria Analítica: um tratamento vetorial**. São Paulo: Pearson, 2005. cap. 21, p. 273–285.
- CANTÚ, E. D. **Desenvolvimento de plataforma robótica omnidirecional**. Monografia (Trabalho de Conclusão de Curso) — Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017. Disponível em: <<http://hdl.handle.net/11624/2151>>. Acesso em: 30 jul. 2023.
- CRAIG, J. J. Geração de trajetórias. In: _____. **Robótica**. São Paulo: Pearson, 2013. cap. 7, p. 192–220.
- DORF, R. C.; BISHOP, R. H. **Sistema de Controle Modernos**. Rio de Janeiro: LTC, 2001.
- FRANKLIN, G. F.; POWELL, J. D.; EMAMI-NAEINI, A. A First Analysis of Feedback. In: _____. **Feedback Control of Dynamic Systems**. New York: Pearson, 2018. cap. 4, p. 208–270.
- MARKETSANDMARKETS. **Mobile Robots Market**. 2023. <<https://www.marketsandmarkets.com/Market-Reports/mobile-robots-market-43703276.html>>. Acesso em: 29 jul. 2023.
- RIJALUSALAM, D. U.; ISWANTO, I. Implementation kinematics modeling and odometry of four omni wheel mobile robot on the trajectory planning and motion control based microcontroller. **Journal of Robotics and Control (JRC)**, v. 2, n. 5, p. 448–455, 2021. ISSN 2715-5072. Disponível em: <<https://doi.org/10.18196/jrc.25121>>. Acesso em: 30 jul. 2023.
- RITTER, G. A. **Modelo genérico de plataforma robótica omnidirecional em código aberto**. Monografia (Trabalho de Conclusão de Curso) — Universidade de Santa Cruz do Sul, Santa Cruz do Sul, 2016. Disponível em: <<https://hdl.handle.net/1822/41924>>. Acesso em: 30 jul. 2023.
- SARMENTO, L. et al. Remote control system for a mobile platform with four mecanum wheels. **International Journal of Mechatronics and Applied Mechanics**, v. 1, p. 274–281, 2017. Disponível em: <<https://hdl.handle.net/1822/53113>>. Acesso em: 30 jul. 2023.
- SHABALINA, K.; SAGITOV, A.; MAGID, E. Comparative analysis of mobile robot wheels design. In: **2018 11th International Conference on Developments in eSystems Engineering (DeSE)**. IEEE, 2018. p. 175–179. Disponível em: <<https://doi.org/10.1109/DeSE.2018.00041>>. Acesso em: 30 jul. 2023.
- SIEGWART, R.; NOURBAKHSI, I. R.; SCARAMUZZA, D. **Introduction to Autonomous Mobile Robots**. London: The MIT Press, 2011.
- SOFWAN, A. et al. Development of omni-wheeled mobile robot based-on inverse kinematics and odometry. In: **2019 6th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE)**. IEEE, 2019. p. 175–179. Disponível em: <<https://doi.org/10.1109/ICITACEE.2019.8904418>>. Acesso em: 30 jul. 2023.

TAGLIAVINI, L. et al. Wheeled mobile robots: State of the art overview and kinematic comparison among three omnidirectional locomotion strategies. **Journal of Intelligent Robotic Systems**, v. 106, n. 57, 2022. Disponível em: <<https://doi.org/10.1007/s10846-022-01745-7>>. Acesso em: 30 jul. 2023.

TAHERI, H.; ZHAO, C. X. Omnidirectional mobile robots, mechanisms and navigation approaches. **Mechanism and Machine Theory**, v. 153, p. 1–28, nov. 2020. Disponível em: <<https://doi.org/10.1016/j.mechmachtheory.2020.103958>>. Acesso em: 30 jul. 2023.

ZIEGLER, J.; NICHOLS, N. B. Optimum settings for automatic controllers. **Transactions of the ASME**, v. 64, p. 759–768, 1942. Disponível em: <<https://doi.org/10.1115/1.4019264>>. Acesso em: 18 jun. 2023.