



UNIVERSIDADE FEDERAL DO RIO GRANDE - FURG
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

Dissertação de Mestrado

**Modelagem, Simulação e Controle de Uma Plataforma
Robótica Omnidirecional para Aplicação em Treinamento
de Marcha**

Victor Barros Coch

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande - FURG, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação

Orientador: Prof. Dr. Vinícius Menezes de Oliveira

Rio Grande, 2024

Ficha Catalográfica

C661m Coch, Victor Barros.

Modelagem, simulação e controle de uma plataforma robótica omnidirecional para aplicação em treinamento de marcha / Victor Barros Coch. – 2024.

100 f.

Dissertação (mestrado) – Universidade Federal do Rio Grande – FURG, Programa de Pós-Graduação em Computação, Rio Grande/RS, 2024.

Orientador: Dr. Vinícius Menezes de Oliveira.

1. Robótica Assistiva 2. Treinamento de Marcha 3. Robô Omnidirecional I. Oliveira, Vinícius Menezes de II. Título.

CDU 004

Catálogo na Fonte: Bibliotecário José Paulo dos Santos CRB 10/2344

DISSERTAÇÃO DE MESTRADO

**Modelagem, Simulação e Controle de Uma Plataforma Robótica
Omnidirecional para Aplicação em Treinamento de Marcha**

VICTOR BARROS COCH

Banca examinadora:

Prof. Dr. Marcelo Rita Pias (FURG) Documento assinado digitalmente
MARCELO RITA PIAS
Data: 25/10/2024 14:51:39-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Renato Ventura Bayan Henriques (UFRGS)

Prof. Dr. Eduardo Rocon de Lima (UPM)

Prof. Dr. Vinícius Menezes de Oliveira (FURG)
Orientador Documento assinado digitalmente
VINICIUS MENEZES DE OLIVEIRA
Data: 24/10/2024 09:16:05-0300
Verifique em <https://validar.iti.gov.br>

Dedico este trabalho aos amigos, colegas e professores que me acompanharam ao longo desta jornada acadêmica, pelo apoio, incentivo e pelas valiosas lições que contribuíram para esta realização.

AGRADECIMENTOS

Primeiramente, expresso minha profunda gratidão ao meu orientador, Prof. Vinícius Menezes de Oliveira, por despertar e cultivar em mim o interesse pela pesquisa na área de robótica. Sua orientação foi fundamental para o direcionamento e desenvolvimento deste trabalho, fornecendo não apenas conhecimento técnico, mas também apoio e incentivo ao longo de todo o processo. Agradeço também aos meus colegas do grupo REHABOT, cujas contribuições foram essenciais para o aprimoramento da minha pesquisa. Suas sugestões e discussões enriqueceram minha metodologia e ampliaram minhas perspectivas, tornando o ambiente de trabalho mais colaborativo e produtivo. Reservo um agradecimento especial à minha companheira, Iza, cuja paciência, compreensão e apoio incondicional foram fundamentais durante este trajeto. Sua força e presença foram essenciais para que eu superasse os desafios e concretizasse este sonho. Por fim, expresso gratidão à minha família, que sempre esteve ao meu lado, oferecendo apoio e motivação ao longo de toda esta jornada.

Is science of any value?
I think a power to do something is of value.
— RICHARD P. FEYNMAN

RESUMO

COCH, Victor Barros. **Modelagem, Simulação e Controle de Uma Plataforma Robótica Omnidirecional para Aplicação em Treinamento de Marcha**. 2024. 100 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

A robótica assistiva tem progredido rapidamente, impulsionada por avanços em computação, sistemas de controle e instrumentação. Na reabilitação motora, especialmente para crianças com paralisia cerebral (PC), o treinamento de marcha desempenha um papel fundamental na recuperação e no desenvolvimento da independência funcional. A adoção de tecnologias assistivas nesses processos pode acelerar o tratamento, aliviando o esforço físico dos fisioterapeutas, além de aumentar a eficiência das técnicas terapêuticas e garantir maior precisão e repetibilidade nos movimentos reabilitativos. Esta dissertação explora a modelagem, simulação, controle e prototipagem de uma plataforma robótica omnidirecional, com o intuito de ampliar as possibilidades de movimento durante o treinamento de marcha. Conceitos como holonomicidade e seus efeitos na modelagem dinâmica da plataforma são discutidos em detalhes, complementados por simulações numéricas para validar os princípios analisados. Além disso, são utilizadas ferramentas modulares amplamente reconhecidas na pesquisa em robótica móvel, como ROS e Gazebo, para validar os resultados e explorar novas abordagens no campo.

Palavras-chave: Robótica Assistiva, Treinamento de Marcha, Robô Omnidirecional.

ABSTRACT

COCH, Victor Barros. **Modelling, Simulation and Control of an Omnidirectional Platform with Applications in Gait Training**. 2024. 100 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

Assistive robotics has been advancing rapidly, driven by innovations in computing, control systems, and instrumentation. In motor rehabilitation, particularly for children with cerebral palsy (CP), gait training plays an important role in recovery and the development of patients' functional independence. The integration of assistive technologies into these processes can accelerate treatment, reduce the physical strain on physiotherapists, increase the efficiency of therapeutic techniques, and provide greater precision and repeatability in rehabilitative movements. This dissertation explores the modeling, simulation, control, and prototyping of an omnidirectional robotic platform, aimed at expanding movement possibilities during gait training. Concepts such as holonomic behavior and its effects on the platform's dynamic modeling are discussed in detail, supported by numerical simulations to validate the analyzed principles. Furthermore, widely recognized modular tools in mobile robotics research, such as ROS and Gazebo, are incorporated to validate the results and explore new approaches in the field.

Keywords: Assistive Robotics, Cerebral Palsy, Gait Training, Omnidirectional Robot.

LISTA DE FIGURAS

Figura 1	Logo do projeto REHABOT	20
Figura 2	Movimentação lateral do robô diferencial (ADĂSCĂLIȚEI; DO-ROFTEI, 2011)	24
Figura 3	Movimentação lateral do robô omnidirecional	24
Figura 4	Diagrama original da roda <i>Mecanum</i> (ILON, 1975)	25
Figura 5	Modos básicos de movimentação da plataforma	25
Figura 6	Espaços holonômicos e não holonômicos	27
Figura 7	Controle de trajetória, adaptado de (SOLEA; FILIPESCU; NUNES, 2009)	28
Figura 8	Modelo robótico genérico com elos e juntas (JOSEPH; CACACE, 2018)	31
Figura 9	Visualização de um Elo URDF (JOSEPH; CACACE, 2018)	31
Figura 10	Plataforma <i>Discover2Walk</i> (DELGADO-OLEAS et al., 2023)	33
Figura 11	Plataforma <i>CPWalker</i> (BAYON et al., 2017)	35
Figura 12	Plataforma <i>C-ALEX</i> (JIN; CUI; AGRAWAL, 2015)	36
Figura 13	Andador robótico (CHOTIKUNNAN et al., 2017)	37
Figura 14	Protótipo utilizado por Zeidis e Zimmermann (ZIMMERMANN; ZEIDIS; ABDELRAHMAN, 2014)	38
Figura 15	Andador robótico descrito por (JIA; SONG; XU, 2013)	40
Figura 16	Robô Pioneer 3AT (MATRAJI et al., 2018)	41
Figura 17	Modelo CAD da plataforma	46
Figura 18	Diagrama da plataforma omnidirecional	47
Figura 19	Diagrama da Roda Omnidirecional	48
Figura 20	Diagrama de blocos do sistema exato	62
Figura 21	Diagrama de blocos do sistema aproximado	62
Figura 22	Diagrama de blocos do sistema exato com distúrbio	64
Figura 23	Diagrama de blocos do sistema aproximado com distúrbio	64
Figura 24	Estrutura URDF da plataforma	65
Figura 25	Diagrama de nós da simulação dinâmica	66
Figura 26	Diagrama de nós da simulação cinemática	66
Figura 27	Sistema de Controle por Realimentação de Estado	67
Figura 28	Diagrama dos sistemas do protótipo	70
Figura 29	Modelo CAD do protótipo em escala	72
Figura 30	Esquemático da junta com motor CC	73
Figura 31	Diagrama de Blocos da Junta Motriz	74
Figura 32	Identificação do polo mecânico da junta	75

Figura 33	Identificação da indutância de armadura	76
Figura 34	Momento de inércia do protótipo	79
Figura 35	Movimentos básicos de translação	81
Figura 36	Movimentos básicos de rotação	81
Figura 37	Trajatória holonômica com distúrbio de força oposta	83
Figura 38	Trajatória com distúrbio de força perpendicular	83
Figura 39	Trajatória de entrada holonômica com distúrbio de torque	84
Figura 40	Erro de trajetória com distúrbio de torque	84
Figura 41	Trajatória não holonômica	85
Figura 42	Erro de trajetória não holonômica	86
Figura 43	Simulação da plataforma em gazebo e ROS	87
Figura 44	Roda <i>Mecanum</i> no Gazebo	87
Figura 45	Movimentos básicos da plataforma no RViz	88
Figura 46	Movimento complexo com controle cinemático da plataforma	88
Figura 47	Controle cinemático no Gazebo	89
Figura 48	Simulação de controle cinemático	90
Figura 49	Erro e ação de controle	90
Figura 50	Protótipo inicial da plataforma em escala	91
Figura 51	Placa de circuito impresso do protótipo	92
Figura 52	Protótipo completo da plataforma em escala	92
Figura 53	Teste de integração e controle cinemático de deslocamento	93

LISTA DE TABELAS

Tabela 1	Dispositivos robóticos revisados	43
Tabela 2	Parâmetros cinemáticos e dinâmicos do robô	47
Tabela 3	Componentes utilizados no protótipo	71
Tabela 4	Comunicação dos módulos	71
Tabela 5	Parâmetros da junta motriz	76
Tabela 6	Parâmetros cinemáticos e dinâmicos do protótipo	79
Tabela 7	Desvios-padrão e erros de trajetória	82
Tabela 8	Desvios-padrão e erros de trajetória com distúrbios	83

LISTA DE ABREVIATURAS E SIGLAS

AAN	<i>Assist As Needed</i> (Assistência Conforme Necessário)
CAD	<i>Computer Aided Design</i> (Desenho Assistido Por Computador)
CDPR	<i>Cable Driven Parallel Robot</i> (Robô Cabeado de Juntas Paralelas)
D2W	Robô Discover2Walk
DB	Diagrama de Blocos
FT	Função de Transferência
GDL	Grau(s) de Liberdade
MCC	Motor de Corrente Contínua
MIMO	<i>Multiple Input Multiple Output</i> (Múltiplas Entradas e Múltiplas Saídas)
PCI	Placa de Circuito Impresso
PID	Controle Proporcional, Integral e Derivativo
PM	<i>Phase Margin</i> (Margem de Fase)
ROS	<i>Robot Operating System</i> (Sistema Operacional de Robô)
SMC	<i>Sliding Mode Control</i> (Controle de Modo Deslizante)
STSMC	Controle de Modo Deslizante <i>Super-Twisting</i>
URDF	<i>Universal Robot Description Format</i> (Formato Universal de Descrição de Robô)
R	Raio da Roda <i>Mecanum</i>
l_1	Distância Lateral
l_2	Distância Vertical
m_c	Massa da Plataforma (excluindo rodas)
m_k	Massa Individual da Roda
J_0	Momento de Inércia da Plataforma (excluindo rodas)
J_1	Momento de Inércia Individual da Roda
J_2	Momento de Inércia da Roda em Relação a C
J_c	Momento de Inércia da Plataforma e Rodas
μ	Coefficiente de Atrito Linear

C	Centroide do Robô Móvel
C_x, C_y	Sistema de Coordenadas Associado ao Ponto C
K_i	Centroide da Roda
$\hat{\tau}_i$	Versor do Ponto de Contato da Roda
$\hat{\gamma}_i$	Versor do Eixo de Rotação do Rolete
ρ_i	Vetor de Deslocamento do Ponto C à Roda
ρ_i	Vetor de Deslocamento de Centroide da Roda ao Ponto de Contato
v_{K_i}	Vetor de Velocidade do Centroide da Roda
v_{P_i}	Vetor de Velocidade no Ponto de Contato da Roda
φ_i	Posição Angular da Roda
Ω_i	Vetor Axial de Velocidade Rotacional da Roda
$\dot{\varphi}$	Vetor de Velocidades Angulares das Rodas
$\ddot{\varphi}$	Vetor de Acelerações Angulares das Rodas
ψ	Posição Angular do Robô
$\dot{\psi}$	Velocidade Angular do Robô
c_ψ	Cosseno da Posição Angular ψ
s_ψ	Seno da Posição Angular ψ
Θ	Vetor Axial de Velocidade Rotacional do Robô
v_{c_x}	Velocidade Linear do Robô na Direção de C_x
v_{c_y}	Velocidade Linear do Robô na Direção de C_y
$\dot{x}_c$	Velocidade Linear do Robô no Eixo x
$\dot{y}_c$	Velocidade Linear do Robô no Eixo y
q	Vetor de Coordenadas Generalizadas
$\dot{q}$	Vetor de Velocidades Generalizadas
$\ddot{q}$	Vetor de Acelerações Generalizadas
$Q(\tau)$	Vetor de Torques Generalizados
τ	Vetor de Torques de Entrada
f_d	Vetor de forças de Distúrbio
τ_d	Torque de Distúrbio
S	Matriz de Transformação dos Torques de Entrada
M	Matriz Simétrica de Massa e Inércia
F_v	Matriz de Atrito Linear
$\mathcal{J}_1$	Matriz de Cinemática Inversa do Sistema Exato
$\mathcal{J}_2$	Matriz de Cinemática Direta do Sistema Exato

$\mathcal{J}_3$	Matriz de Cinemática Inversa do Sistema Aproximado
$\mathcal{J}_3^+$	Matriz Pseudo-inversa de Cinemática Direta do Sistema Aproximado
$T_1(\psi)$	Matriz de Rotação
K	Energia Cinética
K_p	Energia Cinética Translacional
K_r	Energia Cinética Rotacional
U	Energia Potencial
$\mathcal{L}$	Lagrangiano
λ	Vetor de Multiplicadores de Lagrange
$P(\dot{\phi})$	Vetor de Reações Não Holonômicas
α	Matriz Antissimétrica da Restrição Não Holonômica em $\dot{x}$
β	Matriz Antissimétrica da Restrição Não Holonômica em $\dot{y}$

SUMÁRIO

1	Introdução	17
1.1	Motivação e Contextualização	17
1.1.1	Robótica para Reabilitação	18
1.1.2	Treinamento de Marcha	19
1.2	Projeto REHABOT	20
1.3	Objetivos	20
1.3.1	Objetivo Geral	20
1.3.2	Objetivos Específicos	21
1.4	Nota Sobre a Organização do Texto	22
2	Revisão Bibliográfica	23
2.1	Revisão da Literatura	23
2.1.1	Plataformas Robóticas Omnidirecionais	23
2.1.2	Rodas <i>Mecanum</i>	24
2.1.3	Sistemas Não Holonômicos	26
2.1.4	Controle de Trajetória	27
2.2	ROS e Gazebo	29
3	Trabalhos Relacionados	32
3.1	Discover2Walk (D2W)	32
3.2	CPWalker	34
3.3	C-ALEX	35
3.4	Chotikunnnan et al.	36
3.5	Zeidis e Zimmermann	37
3.6	Jianchen et al.	39
3.7	Matraji et al.	40
3.7.1	Análise comparativa	41
3.7.2	Tabela comparativa	42
4	Desenvolvimento	44
4.1	Descrição da Plataforma	46
4.1.1	Parâmetros	46
4.2	Modelagem Cinemática	47
4.2.1	Cinemática Inversa	49
4.2.2	Cinemática Direta Exata	50
4.2.3	Cinemática Direta Aproximada	52
4.3	Modelagem Dinâmica Exata	52

4.3.1	Derivadas Parciais de Coordenadas Generalizadas	54
4.3.2	Derivadas Parciais de Restrição	55
4.3.3	Forças de Restrição	55
4.3.4	Perdas de Energia por Fricção Viscosa	57
4.3.5	Equações de Movimento	57
4.4	Modelagem Dinâmica Aproximada	58
4.4.1	Derivadas Parciais de Coordenadas Generalizadas	59
4.4.2	Perdas de Energia por Fricção Viscosa	60
4.4.3	Equação de Movimento	60
4.5	Divergências entre os Modelos Exato e Aproximado	61
4.6	Simulação Numérica	62
4.6.1	Adição de Distúrbios	63
4.6.2	Resolução Numérica	64
4.7	Modelagem Utilizando Gazebo e ROS	65
4.8	Proposta de Controle	66
4.8.1	Modelo Cinemático no Espaço de Estados	66
4.8.2	Controle Cinemático por Realimentação de Estado	67
4.9	Plataforma de Baixo Custo para Testes	69
4.9.1	Software Embarcado	72
4.9.2	Modelo de Junta Motriz	73
4.9.3	Parametrização da Junta Motriz	75
4.9.4	Controlador de Velocidade PID	76
4.9.5	Parametrização da Plataforma de Teste	78
5	Resultados	80
5.1	Nota Sobre Representação Gráfica dos Dados	80
5.2	Simulação Numérica	80
5.3	Comparação Entre os Modelos Aproximado e Exato	82
5.4	Simulação Gazebo/ROS	86
5.5	Sistema de Controle Cinemático	89
5.6	Montagem do Protótipo em Escala	91
5.7	Teste do Controle Cinemático	93
6	Conclusão	94
6.1	Considerações Finais	94
	Referências	96

ANEXOS

1 INTRODUÇÃO

Ao início de cada capítulo, será feita uma breve descrição do conteúdo, com o propósito de facilitar a leitura. Este capítulo introdutório apresenta os conceitos necessários para contextualizar o trabalho no tema proposto, bem como delinear os objetivos gerais e específicos da pesquisa. Além disso, será feita uma breve nota sobre a organização do texto.

1.1 Motivação e Contextualização

As incapacidades relacionadas à marcha, causadas por condições como acidente vascular encefálico (AVE), danos ao sistema nervoso central ou paralisia cerebral (PC), representam o maior grupo de deficiências físicas no mundo (MURRAY et al., 2012). Dentre essas, a paralisia cerebral, também conhecida como encefalopatia crônica não progressiva, é a principal causa de deficiência física em crianças (KRIGGER, 2006). A PC é uma condição neurológica caracterizada por transtornos motores que variam de leves a severos, geralmente acompanhados de rigidez muscular, resultado de lesões cerebrais ocorridas durante o desenvolvimento fetal ou nos primeiros anos de vida. Esses transtornos afetam permanentemente o movimento e a postura, comprometendo a funcionalidade motora do paciente (ROSENBAUM et al., 2007). No Brasil, somente cerca de 90% das crianças com PC chegam à idade adulta, com uma leve predominância da condição em meninos (SIMÕES et al., 2013).

Os estímulos recebidos durante os primeiros meses e anos de vida são fundamentais para o desenvolvimento neurológico e funcional, devido à plasticidade cerebral, que permite ao cérebro se adaptar e se reorganizar em resposta a esses estímulos (NOVAK et al., 2017). Com intervenções adequadas, é possível criar condições que ampliam as oportunidades de interação com o ambiente e favorecem o desenvolvimento de habilidades motoras ao longo do tempo. Assim, o desenvolvimento motor não depende exclusivamente de fatores biológicos, mas também de um ambiente que propicie a aquisição de novas capacidades funcionais (ASSIS-MADEIRA; CARVALHO, 2009).

O objetivo principal do tratamento para a PC não é curar a condição ou restaurar

completamente o estado físico. Em vez disso, busca-se promover a saúde em termos de locomoção, desenvolvimento cognitivo, interação social e independência funcional (KRIGGER, 2006). Nesse sentido, a fisioterapia desempenha um papel essencial, ao facilitar o desenvolvimento saudável e melhorar a qualidade de vida dos pacientes com PC. A reabilitação visa otimizar as habilidades motoras e aumentar a independência do indivíduo, promovendo uma maior participação na vida diária e melhorando sua interação com o ambiente.

Adicionalmente, novos avanços tecnológicos, como o uso de sistemas robóticos e dispositivos assistivos, têm oferecido novas formas de suporte para o tratamento de pacientes com PC. Tais tecnologias são capazes de complementar a fisioterapia tradicional, proporcionando maior precisão nos movimentos e melhor controle durante os exercícios, o que pode potencializar os resultados do tratamento. A utilização de plataformas robóticas tem demonstrado grande potencial em melhorar a eficácia do treinamento motor, ao reduzir a carga de trabalho dos terapeutas e fornecer assistência personalizada ao paciente, ajustando o nível de suporte conforme necessário (SHI et al., 2019).

1.1.1 Robótica para Reabilitação

Uma das principais divisões dentro da robótica de reabilitação é entre robôs projetados para assistência e aqueles voltados para a terapia. Robôs assistivos são focados em compensar as deficiências físicas dos usuários, enquanto os robôs terapêuticos oferecem treinamento motor especializado. Além disso, essa área abrange também o desenvolvimento de próteses, estimulação funcional neural e tecnologias de monitoramento e diagnóstico durante as atividades diárias (ADL) (LOOS; REINKENSMEYER; GUGLIELMELLI, 2016).

Além dos avanços tecnológicos, a robótica assistiva vai além do uso de próteses e exoesqueletos. Diferente de dispositivos vestíveis, muitos equipamentos robóticos não precisam ser utilizados diretamente pelo usuário para promover benefícios à reabilitação (DELLON; MATSUOKA, 2007). Em muitas situações, a interação com o robô é mediada por terapeutas, que ajustam as atividades e monitoram o progresso do paciente, e essa tecnologia também está se tornando mais acessível para uso doméstico, com cuidadores e familiares participando do processo de reabilitação (LOOS; REINKENSMEYER; GUGLIELMELLI, 2016).

O desenvolvimento de dispositivos mecânicos e eletromecânicos para auxiliar na recuperação de pacientes com limitações motoras é um exemplo de como a tecnologia pode melhorar a qualidade de vida das pessoas (GONÇALVE; SIQUEIRA, 2013). Dessa forma, o uso de sistemas robóticos na reabilitação oferece uma série de vantagens em comparação às intervenções tradicionais. Esses dispositivos permitem uma fisioterapia mais intensiva e personalizada, com a possibilidade de implementar estratégias interativas avançadas (FONSECA et al., 2018).

Com a evolução da robótica assistiva, espera-se que essas tecnologias continuem a se expandir, proporcionando maior acessibilidade e oportunidades de reabilitação para pessoas com deficiências. O desenvolvimento contínuo de novos equipamentos e soluções tecnológicas visa não apenas melhorar a eficiência do tratamento, mas também proporcionar uma melhor qualidade de vida para os pacientes que dependem de terapias de reabilitação avançadas.

É importante destacar que, ao se tratar de robótica assistiva, todos os equipamentos devem seguir normas específicas que garantem a padronização, segurança e eficácia para os usuários. Os dispositivos são classificados pela ISO 9999:2016 (ISO, 2016), e a norma IEC 80601-2-78:2019 define os requisitos de segurança para robôs médicos usados em processos de reabilitação (ISO, 2006). Essa normatização é essencial para validar a eficácia desses equipamentos, assegurando a proteção dos pacientes que interagem diretamente com os robôs (ISO, 2014b; IEC, 2010; ISO, 2014a).

1.1.2 Treinamento de Marcha

Entre os vários procedimentos fisioterapêuticos, o treinamento de marcha destaca-se por sua importância na reabilitação motora, sendo essencial para fortalecer as capacidades motoras e promover a independência funcional dos pacientes. O treinamento de marcha tem como objetivo inibir padrões anormais de movimento e facilitar padrões motores normais, além de fortalecer a musculatura e melhorar a resiliência do paciente. Esse processo se baseia nos conceitos de treinamento ativo, onde o paciente participa ativamente dos movimentos, e treinamento passivo, onde os movimentos são guiados por equipamentos ou profissionais (COSTA et al., 2014). Na prática tradicional, o método requer a assistência de vários terapeutas e o uso de dispositivos, como sistemas de suporte parcial de peso, para auxiliar no controle dos movimentos do paciente, otimizando o processo de reabilitação.

As abordagens contemporâneas de treinamento de marcha incluem o uso de equipamentos como esteiras, barras paralelas e sistemas elaborados de suporte, que ajudam a reduzir a carga de trabalho dos terapeutas e a facilitar a execução dos exercícios. A intensidade e a natureza dos exercícios variam conforme a estratégia de reabilitação, podendo ser ativa ou passiva. No entanto, mesmo com o uso de tais ferramentas, as sessões prolongadas de fisioterapia continuam sendo exaustivas e fisicamente exigentes tanto para o terapeuta quanto para o paciente (KUBO et al., 2011).

Em resposta a esses desafios, a terapia robótica tem se tornado uma alternativa promissora, oferecendo maior eficiência e precisão no processo de reabilitação. Plataformas robóticas e exoesqueletos são capazes de auxiliar tanto o paciente quanto os profissionais na execução dos movimentos, reduzindo significativamente o esforço físico dos terapeutas. Esses dispositivos fornecem estímulos cuidadosamente calculados para guiar os movimentos de forma precisa, garantindo maior repetibilidade e controle durante os exer-

cícios (SHI et al., 2019). Embora os equipamentos robóticos ainda apresentem um custo elevado, eles proporcionam benefícios consideráveis, especialmente em grandes centros de reabilitação. A terapia robótica não apenas melhora a eficiência dos processos, mas também acelera a recuperação do paciente, permitindo uma reabilitação mais efetiva e personalizada (ALVES, 2014). Assim, a introdução de tecnologia robótica na fisioterapia de marcha é uma tendência crescente que pode transformar a maneira como a reabilitação é conduzida, aumentando a qualidade do atendimento e os resultados terapêuticos.

1.2 Projeto REHABOT

Tratando de robótica assistiva, este trabalho está inserido no Projeto de Robótica Assistiva para Reabilitação da Universidade Federal do Rio Grande (REHABOT), grupo que visa desenvolver ferramentas nas áreas de robótica e computação para auxiliar pacientes que possuam necessidades relacionadas à reabilitação dos membros superiores e inferiores. O projeto trabalha em colaboração com a Universidade Politécnica de Madri (UPM) e a Universidade Carlos III de Madri (UC3M).



Figura 1: Logo do projeto REHABOT

No contexto do projeto, outras pesquisas relacionadas estão nos trabalhos de Sibyla A. V. Silva (SILVA, 2021), que foca no planejamento de trajetórias para terapia robótica em membros superiores, e André L. S. Kelbouscas (KELBOUSCAS, 2022), que foca no rastreamento cinemático do processo de reabilitação com sensores inerciais.

1.3 Objetivos

1.3.1 Objetivo Geral

O movimento de marcha não é linear, envolvendo deslocamentos laterais periódicos do centro de massa (TESIO; ROTA, 2019). Portanto, é importante que uma plataforma de treinamento de marcha tenha a manobrabilidade necessária para realizar trajetórias que maximizem os resultados da terapia, sem riscos de tombamento ou limitações para o exoesqueleto. O objetivo geral deste trabalho é modelar, simular e controlar um robô móvel

para aplicações em treinamento de marcha e robótica assistiva, utilizando uma plataforma omnidirecional. Uma plataforma com tração holonômica é altamente desejável, pois elimina restrições quanto ao tipo de movimento que o paciente pode executar, como marcha normal, lateral, diagonal ou rotação no próprio eixo (AGUIRRE-OLLINGER; YU, 2021).

1.3.2 Objetivos Específicos

Os objetivos específicos deste trabalho são detalhados para garantir uma abordagem abrangente no desenvolvimento da plataforma robótica omnidirecional voltada para a reabilitação de marcha. Cada um dos pontos contribui diretamente para a construção e validação da solução proposta, tanto no nível teórico quanto prático. São eles:

- Modelagem cinemática da plataforma móvel
- Implementação de rotinas básicas de movimentação
- Parametrização e modelagem dinâmica da plataforma
- Realização de simulações numéricas
- Modelagem tridimensional da plataforma em URDF
- Desenvolvimento de um ambiente de simulação em ROS e Gazebo
- Proposta e implementação de controle cinemático para o robô
- Desenho de um protótipo em escala da plataforma
- Elaboração de sistemas eletrônicos embarcados, sensores e atuadores (*Hardware*)
- Desenvolvimento de interfaces de comunicação e *software* embarcado
- Identificação dos parâmetros cinemáticos e dinâmicos do modelo em escala
- Validação dos modelos implementados através da comparação de trajetórias
- Testes práticos da plataforma

1.4 Nota Sobre a Organização do Texto

Esta dissertação de mestrado está organizada em seis capítulos, cada um desempenhando um papel essencial para a compreensão e desenvolvimento do trabalho. O Capítulo 1 consiste na introdução, onde são apresentados o tema, os objetivos do projeto, a motivação e o contexto em que a pesquisa se insere, além de uma visão geral da estrutura do texto. O Capítulo 2 abrange a revisão bibliográfica, na qual são discutidos os principais conceitos teóricos e os avanços tecnológicos que fundamentam este trabalho. Neste capítulo, são apresentados os estudos e pesquisas que forneceram a base teórica necessária para o desenvolvimento do projeto. O Capítulo 3 examina os trabalhos relacionados, detalhando projetos e pesquisas anteriores que influenciaram e inspiraram esta dissertação. Aqui, são abordadas as contribuições da literatura existente, comparando diferentes abordagens e soluções para o problema investigado. O Capítulo 4 descreve os aspectos técnicos do desenvolvimento do projeto, com ênfase na modelagem matemática dos sistemas propostos, no projeto da plataforma robótica, e nos métodos utilizados para simulação e controle. Este capítulo também detalha os desafios enfrentados durante o processo de implementação. O Capítulo 5 apresenta os resultados obtidos a partir das simulações e testes experimentais. Nesta seção, são discutidos os dados colhidos, bem como a análise quantitativa e qualitativa dos resultados, identificando as implicações das descobertas para o campo da robótica assistiva. Por fim, o Capítulo 6 oferece as considerações finais do trabalho, sintetizando as conclusões alcançadas, discutindo as limitações do estudo e propondo direções para trabalhos futuros. Aqui, também são feitas sugestões para a aplicação prática dos resultados obtidos e para possíveis melhorias nas tecnologias assistivas estudadas. Dessa forma, o texto foi estruturado para guiar o leitor desde o embasamento teórico até as conclusões práticas.

2 REVISÃO BIBLIOGRÁFICA

Este capítulo fornece a fundamentação teórica necessária para o desenvolvimento do projeto, abordando dois principais tópicos: (i) Revisão da Literatura, que abrange os conceitos de plataformas omnidirecionais, sistemas não holonômicos e controle de trajetória; e (ii) ROS e Gazebo, que se concentra nas ferramentas utilizadas para o desenvolvimento de *software* e simulação de robôs.

2.1 Revisão da Literatura

2.1.1 Plataformas Robóticas Omnidirecionais

Veículos omnidirecionais possuem a capacidade de se mover simultaneamente em três graus de liberdade (GDL) no plano: transversal, longitudinal e rotacional. Essa característica de tração multidirecional permite a realização de manobras complexas com maior eficiência, otimizando o tempo e o espaço necessários para deslocamentos em ambientes restritos (DICKERSON; LAPIN, 1991). Em contraste, veículos tradicionais, como os equipados com tração diferencial ou geometria Ackerman, estão limitados a apenas dois GDL: longitudinal e rotacional. Essas limitações são impostas por restrições não holonômicas e singularidades cinemáticas. Singularidades ocorrem quando o sistema atinge um estado de bloqueio, onde o controle de movimento se torna indevido ou impossível, e o acionamento dos atuadores pode gerar tensões excessivas e até danos físicos. Devido a essas restrições, veículos convencionais exigem trajetórias mais elaboradas para cobrir todo o espaço de trabalho tangível (ADĂSCĂLIȚEI; DOROFTEI, 2011), conforme ilustrado na Figura 2.

A principal vantagem dos robôs omnidirecionais sobre plataformas convencionais reside na sua mobilidade superior, especialmente em espaços confinados ou ambientes repletos de obstáculos. Esses robôs são particularmente adequados para operar em locais apertados, como hospitais, fábricas e armazéns, onde a capacidade de movimentação em múltiplas direções simultaneamente se traduz em maior agilidade e eficiência (ADĂSCĂLIȚEI; DOROFTEI, 2011). A comparação entre as Figuras 2 e 3 ressalta essa superioridade, evidenciando como a movimentação transversal simplifica manobras, como o

estacionamento paralelo e outros movimentos que exigem mudanças de direção rápidas.

Além disso, a aplicação de veículos omnidirecionais em plataformas móveis voltadas para a reabilitação, como no treinamento de marcha, oferece benefícios significativos. A capacidade de realizar manobras precisas e complexas torna-se essencial para pacientes com dificuldades motoras. Nesses casos, a precisão de movimentação que o sistema omnidirecional proporciona é fundamental para a adaptação das trajetórias ao estado físico do paciente, facilitando exercícios terapêuticos com maior controle e segurança (AGUIRRE-OLLINGER; YU, 2021). Essa versatilidade permite que o robô ofereça suporte em uma ampla gama de atividades terapêuticas, melhorando a qualidade do atendimento e contribuindo para uma recuperação mais eficaz.

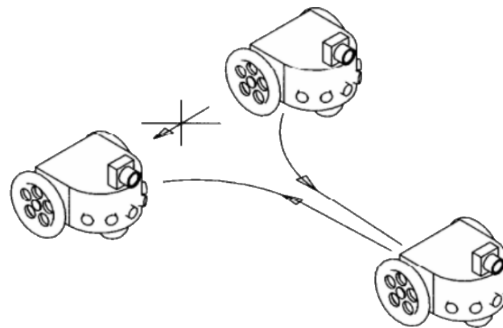


Figura 2: Movimentação lateral do robô diferencial (ADĂSCĂLIȚEI; DOROFTEI, 2011)

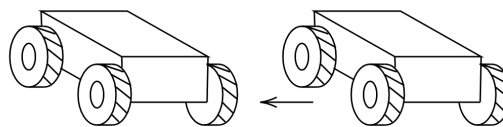


Figura 3: Movimentação lateral do robô omnidirecional

2.1.2 Rodas Mecanum

O desenvolvimento de robôs omnidirecionais frequentemente requer o uso de rodas especiais, como as rodas Mecanum, inventadas por Bengt Erland Ilon em 1975 (ILON, 1975). Essas rodas são caracterizadas por um eixo central com roletes inclinados a 45° , permitindo que o robô se mova em qualquer direção, independentemente da orientação dos eixos principais. Isso facilita a movimentação omnidirecional sem a necessidade de

alterar a posição do corpo do robô. A Figura 4 ilustra o diagrama original da patente dessa tecnologia inovadora.

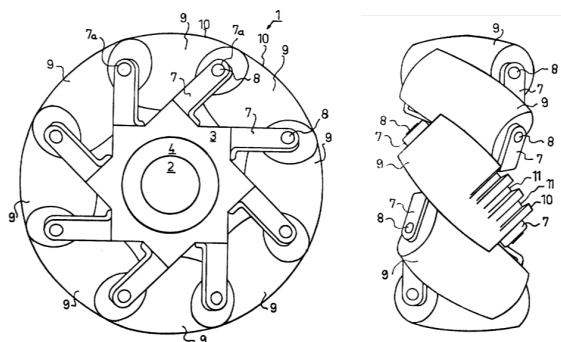


Figura 4: Diagrama original da roda *Mecanum* (ILON, 1975)

O funcionamento das rodas Mecanum baseia-se na divisão da velocidade aplicada em dois componentes: um componente ativo, alinhado ao eixo dos roletes, responsável pela tração, e um componente passivo, perpendicular ao rolete, que permite a movimentação lateral. Cada roda pode ser controlada de forma independente, possibilitando ajustes rápidos e precisos na direção e na velocidade do robô. A Figura 5 mostra alguns dos modos básicos de movimentação permitidos por essa configuração (ADĂSCĂLIȚEI; DOROFTEI, 2011).

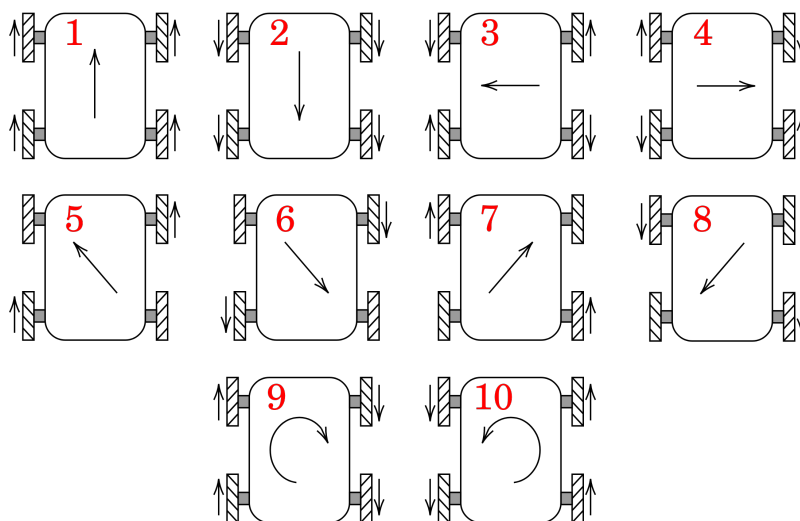


Figura 5: Modos básicos de movimentação da plataforma

Apesar das vantagens em termos de flexibilidade de movimentação, os robôs equipados com rodas Mecanum apresentam algumas limitações significativas. Uma delas é a conversão ineficiente de energia cinética rotacional em movimento translacional, o que pode resultar em perdas de desempenho, especialmente em superfícies irregulares ou com

atrito elevado. Além disso, a suscetibilidade ao deslizamento aumenta a dificuldade de implementar sistemas de odometria precisa, frequentemente exigindo o uso de sensores complementares, como sensores inerciais ou sistemas de visão, para aprimorar a localização e o controle do robô (PENG et al., 2016; AZIZI; RASTEGARPANAH; STOLKIN, 2021).

Outro desafio é a vibração durante a movimentação lateral, um problema comum devido à interação dos roletes com o solo (PENG et al., 2016). Embora ajustes mecânicos nas rodas, como a frenagem independente ou a modificação do ângulo dos roletes, possam mitigar esse problema (ADĂSCĂLIȚEI; DOROFTEI, 2011), essas soluções frequentemente envolvem aumentar a complexidade do sistema, o que pode impactar o custo e a manutenção do robô.

Apesar dessas desvantagens, as rodas Mecanum continuam a oferecer vantagens significativas para aplicações em ambientes planos e controlados, onde a capacidade de movimentação omnidirecional é importante. Quando usadas em condições apropriadas, essas rodas proporcionam alta manobrabilidade e versatilidade, tornando-se uma escolha valiosa para robôs móveis que operam em espaços confinados ou que requerem mudanças rápidas de direção.

2.1.3 Sistemas Não Holonômicos

Em mecânica clássica, os processos dinâmicos podem ser classificados em duas categorias: holonômicos e não holonômicos. Sistemas holonômicos são aqueles que possuem apenas restrições holonômicas, enquanto sistemas não holonômicos possuem ao menos uma restrição não holonômica (ARCZEWSKI; BLAJER, 1996). No contexto da robótica, essas restrições são frequentemente chamadas de "restrições cinemáticas" (SICILIANO et al., 2009), com a seguinte formulação matemática:

$$g_i(q, \dot{q}) = 0 \quad | \quad i = 1, \dots, k < n$$

As restrições cinemáticas limitam a operação do sistema e influenciam diretamente sua dinâmica. Quando uma dessas restrições pode ser expressa em termos de uma relação geométrica — ou seja, depende apenas das coordenadas de posição q e é integrável — ocorre uma redução no número de graus de liberdade do sistema, caracterizando uma restrição holonômica. A expressão para tais restrições é dada por:

$$h_i(q) = 0 \quad | \quad i = 1, \dots, k < n$$

Essas restrições eliminam coordenadas do sistema, simplificando a modelagem e a resolução das equações de movimento. No entanto, quando a restrição não pode ser expressa em termos de uma relação geométrica (ou seja, não é integrável), ela não reduz o número de graus de liberdade, mas restringe o conjunto de velocidades permitidas $\dot{q}$.

Tais restrições são chamadas de não holonômicas. Diferente das restrições holonômicas, elas não eliminam coordenadas, mas afetam o espaço de velocidades do sistema, exigindo técnicas de modelagem mais complexas, como a inclusão de forças de reação. A Figura 6 ilustra a distinção entre os espaços de coordenadas generalizadas e os subespaços de operação impostos por essas restrições cinemáticas.

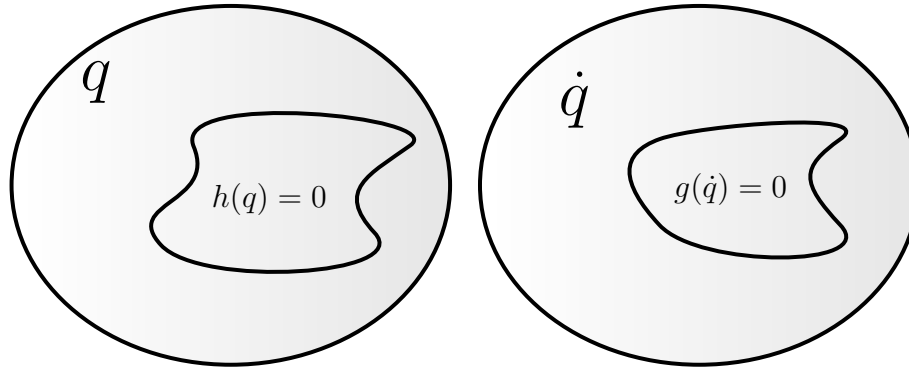


Figura 6: Espaços holonômicos e não holonômicos

Para modelar a dinâmica de sistemas não holonômicos utilizando o formalismo de Lagrange, faz-se uso dos multiplicadores de Lagrange. Esses multiplicadores introduzem restrições adicionais às equações de movimento, confinando as soluções ao espaço permitido pelas restrições não holonômicas. A literatura clássica frequentemente prefere evitar a modelagem de sistemas não holonômicos, quando possível, em função da maior simplicidade dos sistemas holonômicos (ARCZEWSKI; BLAJER, 1996).

No entanto, a pesquisa sobre sistemas não holonômicos tem ganhado grande relevância, especialmente devido à sua conexão com o controle não linear e sua vasta aplicação prática, que inclui desde veículos com rodas, como carros autônomos, até robôs subaquáticos, espaçonaves e sistemas subatuados (JARZĘBOWSKA, 2005). Nesses casos, a modelagem precisa dessas restrições não holonômicas é essencial para o desenvolvimento de algoritmos de controle eficientes, já que esses sistemas frequentemente apresentam complexidades dinâmicas que não podem ser capturadas por abordagens simplificadas. A modelagem correta dessas restrições não apenas melhora a precisão na simulação e controle, mas também é importante para garantir o sucesso em aplicações do mundo real.

Portanto, apesar das dificuldades inerentes à modelagem de sistemas não holonômicos, sua importância em contextos de engenharia moderna e robótica é inegável, exigindo o desenvolvimento de métodos avançados de modelagem e controle para lidar com suas características complexas e dinâmicas.

2.1.4 Controle de Trajetória

O controle de trajetória tem como objetivo principal guiar os atuadores do veículo para que ele siga um caminho de referência que varia ao longo do tempo (MATRAJI et

al., 2018). Esse processo envolve a aplicação de algoritmos que ajustam variáveis como velocidade, direção e aceleração para manter o veículo o mais próximo possível da trajetória planejada, minimizando os erros entre a posição real do veículo e a trajetória desejada. Além disso, considera fatores dinâmicos, como mudanças no terreno e condições ambientais, garantindo uma navegação eficiente e segura.

Para entender a dinâmica do controle de trajetória, é importante notar que tanto a trajetória quanto o robô podem ser descritos por meio de modelos dinâmicos e cinemáticos. O problema de controle, portanto, se transforma na estabilização de uma função de saída que depende dos estados do sistema, de modo a fazer o veículo seguir um alvo virtual. Este alvo assume um caminho e um perfil de velocidade previamente especificados (SOLEA; FILIPESCU; NUNES, 2009), conforme ilustrado na Figura 7.

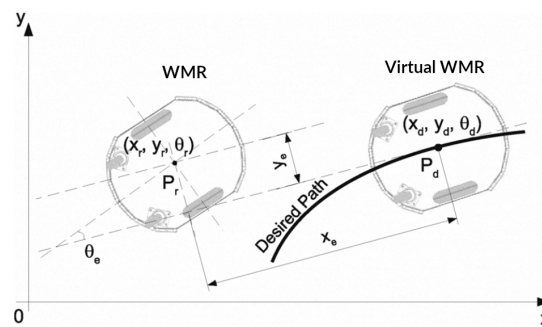


Figura 7: Controle de trajetória, adaptado de (SOLEA; FILIPESCU; NUNES, 2009)

Em sistemas holonômicos, as não-linearidades são geralmente menos acentuadas, o que possibilita a aplicação de técnicas como a realimentação de estado (*full state feedback*) para projetar soluções de controle robustas e eficientes (MORIN; SAMSON, 2004). A realimentação de estado é uma abordagem amplamente utilizada em sistemas dinâmicos para controlar a resposta da planta, ajustando todas as variáveis de estado do sistema com o objetivo de modificar seu comportamento dinâmico.

Esse método envolve o uso de um vetor de estado, que representa as variáveis internas do sistema, e uma matriz de ganhos, projetada para ajustar a influência de cada variável de estado nas entradas de controle (BRUNTON; KUTZ, 2022). Por meio da escolha apropriada dessa matriz de ganhos, é possível posicionar os polos do sistema fechado em locais específicos no plano complexo, moldando a resposta do sistema de acordo com critérios de desempenho como tempo de resposta, amortecimento e estabilidade. Isso oferece grande flexibilidade para o controle, desde que o modelo do sistema seja preciso e todas as variáveis de estado estejam disponíveis para monitoramento e ajuste.

Uma das técnicas mais comuns no projeto de controladores por realimentação de estado é a alocação polos, onde a matriz de ganhos é calculada para assegurar que os polos do sistema fechado atendam aos requisitos de desempenho desejados. Essa metodologia

é eficaz em uma ampla gama de aplicações, especialmente onde a precisão e o controle sobre os parâmetros dinâmicos são essenciais (BRUNTON; KUTZ, 2022).

Entretanto, as leis de controle projetadas para sistemas holonômicos nem sempre podem ser aplicadas diretamente a veículos não holonômicos. Estes últimos exigem abordagens específicas, pois sua natureza introduz restrições adicionais ao movimento, exigindo técnicas de controle adaptadas a essas restrições (KHALAJI; MOOSAVIAN, 2015). Cada técnica de controle tem suas vantagens e desvantagens, sendo necessária uma avaliação cuidadosa para determinar a abordagem mais adequada ao sistema em questão.

Grande parte dos controladores de trajetória propostos na literatura para veículos com rodas são baseados apenas em modelos cinemáticos. No entanto, o uso de modelos dinâmicos é mais adequado para projetar sistemas robustos de rastreamento de trajetória, considerando que os parâmetros do veículo frequentemente apresentam variabilidades e raramente são precisamente conhecidos (MATRAJI et al., 2018). Nesses casos, métodos não lineares, como o controle de modo deslizante, se destacam por sua maior robustez e menor sensibilidade a variações paramétricas, tornando-os uma alternativa vantajosa em comparação aos controladores lineares tradicionais (MATRAJI et al., 2018).

O controle de modo deslizante, em particular, se caracteriza por sua capacidade de lidar com incertezas e perturbações, ajustando dinamicamente as leis de controle para garantir o seguimento da trajetória, mesmo em condições adversas. Isso o torna uma opção atraente para sistemas robóticos móveis, onde a robustez frente a variações de terreno, velocidade e condições ambientais é importante para o desempenho eficiente e seguro da plataforma.

2.2 ROS e Gazebo

Sistemas robóticos, especialmente os robôs móveis, são compostos por uma variedade de peças e dispositivos embarcados, como sensores, atuadores e interfaces, que interagem com sistemas computacionais complexos e, muitas vezes, distribuídos. Esses sistemas devem ser capazes de operar em diversos ambientes e condições para a realização de tarefas específicas (TAKAYA et al., 2016). Uma das principais ferramentas utilizadas para pesquisa e desenvolvimento nessa área é o ROS (*Robot Operating System*) (LINNER et al., 2011), um *framework* amplamente adotado que fornece bibliotecas e ferramentas para aplicações robóticas, tanto em ambientes simulados quanto em implementações no mundo real (JOSEPH; CACACE, 2018).

Uma das características que destacam o ROS na indústria é sua modularidade e compatibilidade multiplataforma. Ele permite a comunicação entre diferentes programas utilizando nodos e mensagens padronizadas, o que facilita a integração de sistemas de *hardware* e *software*. Isso possibilita, por exemplo, que algoritmos de alto desempenho sejam implementados em linguagens de baixo nível, como C e C++, enquanto tarefas menos exi-

gentes sejam gerenciadas em linguagens de alto nível, como Python ou Lisp (JOSEPH; CACACE, 2018). Essa flexibilidade é essencial para a interoperabilidade entre tecnologias que, de outra forma, seriam incompatíveis, removendo barreiras no processo de desenvolvimento e acelerando a criação de novas soluções tecnológicas na robótica.

A principal ferramenta de simulação integrada ao ROS é o Gazebo, um simulador avançado que permite a criação de mundos tridimensionais com física realista, facilitando a interação entre robôs e ambientes simulados, sejam eles estáticos ou dinâmicos (LINNER et al., 2011). Para criar esses mundos e modelos de robôs, é necessário descrever os sistemas no formato URDF (*Universal Robot Description Format*), que define os parâmetros estruturais e funcionais dos componentes do robô. Essa etapa pode ser realizada de forma manual ou através da importação de arquivos CAD já existentes, que são convertidos para o formato URDF por meio de análises procedurais (VYAVAHARE; JAYAPRAKASH; BHARATIA, 2019). Na maioria dos casos, a modelagem manual é preferida, especialmente com o uso de macros através do Xacro, uma extensão que facilita a reutilização de código ao descrever estruturas simétricas ou repetitivas, simplificando a complexidade dos arquivos.

No URDF, os robôs são descritos como uma árvore de elos e juntas interconectados por um conceito de parentesco (Figura 8), o que permite a geração automática dos sistemas de coordenadas, posturas e matrizes de transformação necessárias para o controle do robô. Além disso, a descrição dos parâmetros dinâmicos, como massas, momentos de inércia, malhas visuais e de colisão, pode ser inserida para uma simulação mais realista. A Figura 9 ilustra os elementos fundamentais de um elo de robô, mostrando como suas propriedades dinâmicas e geométricas são representadas no modelo. De forma semelhante, as características das juntas, como limites de movimento e atritos, também podem ser especificadas, permitindo a implementação de algoritmos de controle e cinemática inversa, já disponíveis na plataforma ROS.

Esse processo de modelagem e simulação não apenas proporciona uma representação precisa dos sistemas robóticos, mas também acelera o desenvolvimento e teste de novos algoritmos de controle, sem a necessidade de interagir diretamente com os sistemas físicos desde o início. O ROS, combinado com o Gazebo, oferece uma infraestrutura robusta para pesquisa em robótica, permitindo a experimentação e a validação de conceitos em um ambiente seguro e controlado antes da aplicação em situações reais.

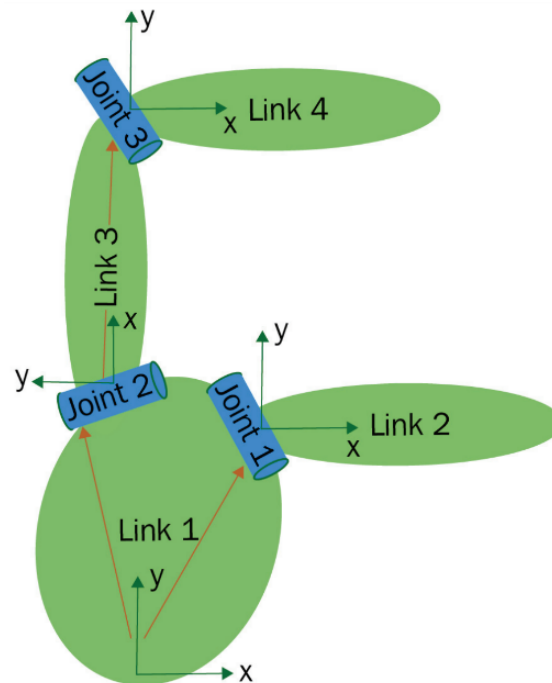


Figura 8: Modelo robótico genérico com elos e juntas (JOSEPH; CACACE, 2018)

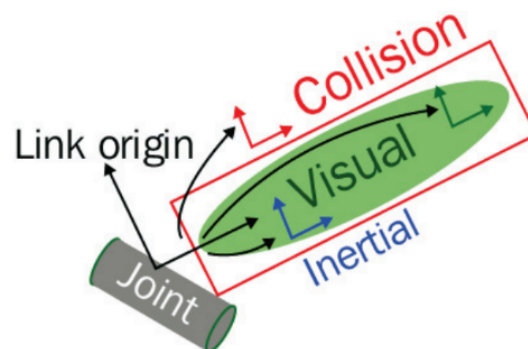


Figura 9: Visualização de um Elo URDF (JOSEPH; CACACE, 2018)

3 TRABALHOS RELACIONADOS

Este capítulo descreve algumas produções encontradas na literatura, que serviram de inspiração e referência para a criação deste trabalho.

3.1 Discover2Walk (D2W)

O sistema D2W (PALOMINO-DÍAZ et al., 2021) é parte integrante deste projeto de pesquisa, desenvolvido em colaboração com o Centro de Automação e Robótica da Universidade Politécnica de Madrid, na Espanha. O objetivo principal desse trabalho é o desenvolvimento de um exoesqueleto com juntas acionadas por cabos, projetado tanto para o treinamento quanto para a assistência à marcha de pacientes com mobilidade reduzida. A inovação deste sistema reside no fato de operar sobre uma plataforma omnidirecional (Figura 10), o que permite uma maior liberdade de movimento em várias direções, proporcionando uma experiência mais versátil durante os exercícios de reabilitação.

Delgado et al. (DELGADO-OLEAS et al., 2023) oferecem uma visão detalhada dos componentes de hardware e software que integram a estrutura do robô. A arquitetura do sistema é baseada no ROS2 (Robot Operating System 2), uma plataforma de código aberto que facilita a implementação de sistemas distribuídos com alta configurabilidade. A modularidade do sistema permite a fácil adaptação e personalização de suas funções, tornando-o adequado para uma ampla gama de aplicações em reabilitação e assistência motora.

A arquitetura do D2W é organizada em três níveis hierárquicos, bioinspirados: o nível alto, o médio e o baixo. O nível alto é responsável pela interface de usuário, percepção do ambiente e planejamento das ações da plataforma, oferecendo uma interface intuitiva que facilita o controle e a interação do terapeuta com o robô. O nível médio é encarregado dos algoritmos que geram os padrões de movimento, processando as informações recebidas para criar trajetórias de marcha otimizadas e personalizadas para cada paciente. O nível baixo, por sua vez, é responsável pela integração dos sensores, atuadores e malhas de controle, coordenando as ações motoras de forma eficiente com base nas instruções do nível médio.

Cada nível opera em frequências de ciclo diferentes para otimizar o desempenho do sistema. O nível alto funciona entre 5 e 7 Hz, o nível médio opera entre 57 e 64 Hz, e o nível baixo alcança uma frequência de 1000 Hz, o que é importante para garantir a precisão e a estabilidade dos movimentos durante o processo de marcha. O nível baixo também gerencia os atuadores e é projetado para minimizar os erros nos movimentos, ajustando os comandos em tempo real. Cada grupo de motores possui um controlador dedicado, responsável por garantir que os requisitos dinâmicos e cinemáticos do sistema sejam atendidos.

A capacidade do D2W de operar em diferentes frequências e a modularidade de sua arquitetura o tornam uma ferramenta poderosa e flexível no campo da reabilitação robótica. A plataforma omnidirecional oferece uma vantagem significativa sobre os sistemas tradicionais de locomoção linear, proporcionando maior liberdade de movimento e, consequentemente, uma reabilitação mais eficaz e adaptativa para os pacientes. Isso, aliado à arquitetura de controle distribuída e a precisão proporcionada pelos sistemas de controle de atuadores, destaca o D2W como uma solução promissora no campo da assistência à marcha e reabilitação motora.

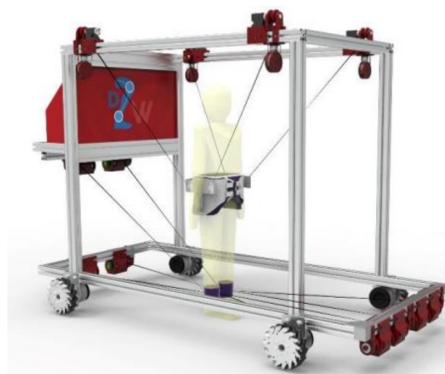


Figura 10: Plataforma *Discover2Walk* (DELGADO-OLEAS et al., 2023)

3.2 CPWalker

Bayón et al. (BAYON et al., 2017) apresentam a plataforma *CPWalker* (Figura 11), que oferece uma combinação inovadora de controle postural e suporte parcial de peso, permitindo que pacientes se movimentem de maneira autônoma com o auxílio de um robô móvel. O sistema integra um exoesqueleto com um andador diferencial inteligente, projetado para reabilitação locomotora supervisionada. Essa abordagem busca aliviar a carga de trabalho dos terapeutas, promovendo a reabilitação precoce e ativa dos pacientes, permitindo sua maior independência durante o tratamento.

O projeto tem como objetivo principal o desenvolvimento de uma plataforma de reabilitação totalmente motorizada, composta por quatro subsistemas fundamentais: (i) controle de locomoção, (ii) suporte parcial do peso corporal, (iii) ajuste de altura do quadril, e (iv) controle das articulações do exoesqueleto. Cada um desses subsistemas é equipado com uma variedade de atuadores e sensores, permitindo um controle preciso dos movimentos, de acordo com os perfis dinâmicos e cinemáticos necessários para cada paciente. O sistema de controle adotado segue a estratégia de assistência conforme necessário (*Assist as Needed* - AAN), que ajusta a assistência fornecida ao paciente com base nas forças detectadas durante a movimentação. Isso garante que o paciente receba apenas o nível mínimo de assistência robótica necessário para completar a tarefa motora, alternando dinamicamente entre reabilitação ativa e passiva, conforme o progresso do tratamento (SHOAIB et al., 2021).

Além da sofisticação técnica, um dos desafios enfrentados pelo *CPWalker* e por outros sistemas de reabilitação motorizada é a dificuldade em mover e atuar nos membros e no corpo dos pacientes, mesmo em indivíduos de pequeno porte, como crianças. Os motores que controlam as articulações do exoesqueleto precisam ser equipados com grandes reduções mecânicas para gerar os altos torques exigidos pelos movimentos de reabilitação, especialmente quando se trata de levantar e sustentar o peso do paciente. Isso aumenta a complexidade do sistema, exigindo um controle de alta precisão.

Outro ponto de limitação do *CPWalker* é o uso de um sistema de locomoção diferencial no andador, o que restringe a liberdade de movimento do paciente. Devido às restrições não holonômicas inerentes a este tipo de locomoção, movimentos como a rotação no local ou a marcha lateral se tornam difíceis de realizar, limitando a capacidade do robô de proporcionar um treinamento de marcha mais versátil e completo. Essa limitação na mobilidade é um desafio comum em sistemas que utilizam locomoção diferencial, impactando a fluidez dos movimentos do paciente durante a terapia.

Apesar desses desafios, a plataforma *CPWalker* representa um avanço significativo na reabilitação robótica, ao proporcionar um ambiente controlado e adaptável para o paciente, integrando os benefícios de um sistema de exoesqueleto motorizado com a assistência proporcionada por um andador robótico inteligente. A flexibilidade proporcionada

pela estratégia *AAN* permite que o sistema ajuste continuamente o nível de assistência, de acordo com a evolução e as necessidades do paciente, promovendo um tratamento mais eficiente e personalizado.

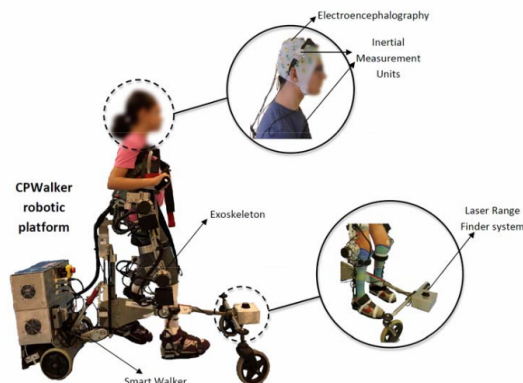


Figura 11: Plataforma *CPWalker* (BAYON et al., 2017)

3.3 C-ALEX

Plataformas recentes no campo da reabilitação robótica exploram o uso de manipuladores paralelos acionados por cabos (CDPRs) como uma alternativa eficiente aos exoesqueletos tradicionais, que exigem alto torque para operação. Esses sistemas apresentam várias vantagens, incluindo baixa inércia, alta capacidade de carga e flexibilidade configuracional. Além disso, oferecem um espaço de trabalho ampliado e maior precisão e rigidez estrutural em comparação com as arquiteturas baseadas em juntas seriais, o que os torna mais adequados para aplicações de reabilitação que exigem ajustes dinâmicos contínuos (EDEN, 2018).

Um exemplo significativo desse tipo de plataforma é o sistema *Cable-drive Active Leg EXoskeleton (C-ALEX)*, descrito por Jin et al. (JIN; CUI; AGRAWAL, 2015). O C-ALEX é um exoesqueleto de membros inferiores composto por três cintas acionadas por cabos, que se conectam à cintura, coxa e canela do usuário. O sistema de controle segue o paradigma de "Assistência conforme necessário" (*Assist as Needed - AAN*), empregando células de carga e unidades de medição inercial (IMUs) para monitorar e ajustar as variáveis-chave, como a posição, orientação e tensão nos cabos não rígidos que formam a estrutura de suporte.

Um dos principais diferenciais do C-ALEX é sua capacidade de estimar e ajustar os estados do sistema em tempo real, o que permite maior precisão na assistência ao movimento. No entanto, este sistema apresenta uma limitação significativa: ele é projetado para funcionar em um ambiente controlado, como uma esteira, restringindo a mobilidade do paciente a um plano fixo. Essa restrição reduz a versatilidade do sistema em contextos de reabilitação mais amplos, onde a movimentação livre em diversas direções seria

benéfica para um treinamento motor mais completo e diversificado.

Apesar dessa limitação, o uso de CDPRs, como no C-ALEX, representa uma inovação importante no desenvolvimento de sistemas de reabilitação robótica. Ao eliminar a necessidade de motores de torque elevado em cada articulação, esses sistemas podem ser mais leves e menos invasivos, ao mesmo tempo em que oferecem suporte dinâmico ajustável de acordo com as necessidades terapêuticas individuais de cada paciente. Isso sugere que, com mais desenvolvimento, essas plataformas poderiam expandir seu escopo de uso para além de ambientes restritos, integrando-se a contextos mais completos de reabilitação.

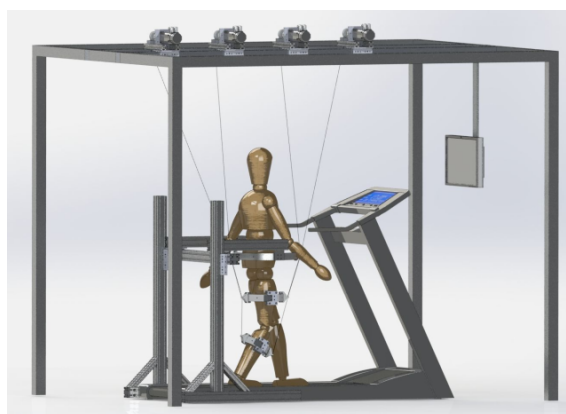


Figura 12: Plataforma C-ALEX (JIN; CUI; AGRAWAL, 2015)

3.4 Chotikunnnan et al.

Chotikunnnan et al. (CHOTIKUNNNAN et al., 2017) propõem um andador assistivo equipado com rodas omnidirecionais *Mecanum*, que oferecem vantagens significativas em comparação aos sistemas de locomoção diferencial tradicionais. A principal vantagem dessas rodas está na sua capacidade de permitir movimentação em qualquer direção sem a necessidade de manobras complexas. Isso resulta em maior agilidade e controle preciso de posição, facilitando a locomoção em ambientes apertados ou com obstáculos.

O design omnidirecional do andador oferece flexibilidade para o usuário, permitindo movimentações laterais, diagonais, e rotacionais sem perda de estabilidade, o que é especialmente útil em situações que exigem ajustes rápidos de posição. Essa capacidade de controle posicional preciso torna o dispositivo ideal para assistência em tarefas cotidianas que exigem mobilidade fluida e sem restrições de direção.

Apesar dessas vantagens, o foco deste trabalho é limitado à assistência de mobilidade convencional e não está diretamente voltado para aplicações em reabilitação, diferentemente de outros projetos que integram tecnologias avançadas para apoio terapêutico ou recuperação motora. No entanto, o potencial de adaptação deste sistema para contextos de reabilitação é notável, considerando sua capacidade de movimentação precisa e ajustável. Com modificações adicionais, como a incorporação de sensores de feedback biomecânico

ou mecanismos de suporte parcial de peso, o andador omnidirecional poderia ser transformado em uma plataforma eficaz para intervenções terapêuticas, especialmente para pacientes com dificuldades de locomoção ou desequilíbrio.

Em suma, o trabalho de Chotikunnan et al. apresenta um avanço importante em dispositivos de mobilidade assistiva, oferecendo uma alternativa mais eficiente aos andadores tradicionais com rodas diferenciais. O uso de rodas *Mecanum* abre novas possibilidades para futuras inovações no campo da robótica assistiva, particularmente em aplicações que exigem alta precisão e flexibilidade de movimento.

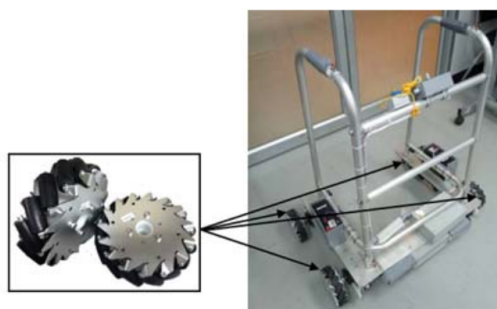


Figura 13: Andador robótico (CHOTIKUNNAN et al., 2017)

3.5 Zeidis e Zimmermann

Zeidis e Zimmermann (ZEIDIS; ZIMMERMANN, 2019; ZIMMERMANN; ZEIDIS; ABDELRAHMAN, 2014) conduzem estudos detalhados sobre a modelagem de robôs omnidirecionais equipados com rodas *Mecanum*, que permitem o movimento em qualquer direção sem a necessidade de manobras complexas. Um dos desafios principais ao modelar esses sistemas está nas restrições não holonômicas impostas pelos roletes das rodas, que evitam o deslizamento lateral, mas ao mesmo tempo criam uma dinâmica complexa. Para lidar com essas restrições, os autores utilizaram a equação de Chaplygin, que elimina a necessidade de utilizar multiplicadores de Lagrange explícitos, simplificando o tratamento matemático do sistema.

Além disso, os autores introduzem o conceito de "holonomização" como uma técnica para aproximar o comportamento do sistema a partir de mínimos quadrados. Essa abordagem permite analisar como as restrições não holonômicas afetam as características cinemáticas do robô, especialmente no que diz respeito à precisão de controle e à eficiência de movimento. O modelo dinâmico derivado da simplificação cinemática, embora mantenha restrições similares e termos antissimétricos, ainda possibilita a aplicação direta das equações de Euler-Lagrange, sem a necessidade dos multiplicadores de Lagrange, o que facilita sua implementação prática.

As simulações realizadas pelos autores foram fundamentais para compreender o com-

portamento do robô sob diferentes condições, mas não incluíram a modelagem de termos de fricção nas juntas, o que resulta em algumas discrepâncias nos resultados teóricos e práticos. No entanto, testes práticos foram realizados utilizando uma plataforma omnidirecional projetada especificamente para auxiliar a mobilidade, demonstrando a viabilidade da abordagem teórica. A plataforma mostrou-se eficiente em termos de controle preciso da posição e orientação, apesar das limitações impostas pela ausência de uma modelagem completa da fricção.

Os estudos de Zeidis e Zimmermann também discutem as implicações práticas dessas modelagens para aplicações em robótica assistiva e de reabilitação. A capacidade de movimentação omnidirecional, possibilitada pelas rodas *Mecanum*, é especialmente útil em ambientes confinados e em situações que exigem alta precisão de movimentação, como no auxílio de pessoas com mobilidade reduzida. A combinação de técnicas de modelagem cinemática avançada e simulações práticas oferece um caminho promissor para o desenvolvimento de robôs mais eficientes e adaptáveis, capazes de operar de forma suave e precisa em diversas condições, o que pode beneficiar tanto a robótica industrial quanto a robótica assistiva.



Figura 14: Protótipo utilizado por Zeidis e Zimmermann (ZIMMERMANN; ZEIDIS; ABDELRAHMAN, 2014)

3.6 Jianchen et al.

Jianchen et al. (JI et al., 2021) desenvolveram um robô móvel omnidirecional integrado a um sistema de suporte parcial de peso, voltado para a assistência de mobilidade e reabilitação de pacientes com doença de Parkinson. O dispositivo, configurado como um andador robótico, é capaz de reconhecer padrões de intenção de movimento dos usuários por meio de sensores inerciais posicionados na região da pélvis. Esses sensores capturam os sinais de movimento, permitindo que o robô se ajuste automaticamente às necessidades individuais do paciente, facilitando tanto o treinamento em marcha quanto as atividades cotidianas. A ênfase do projeto está na segurança e na melhoria da qualidade de vida dos usuários, oferecendo suporte preciso durante o processo de reabilitação.

O trabalho envolve diversas etapas, como a modelagem do sistema, o controle da locomoção, a geração de trajetórias de movimento e a realização de testes clínicos preliminares. Os resultados desses testes indicaram respostas cinemáticas satisfatórias, com o robô ajustando-se adequadamente às variações nos padrões de movimento dos pacientes. Além disso, foi observada uma melhoria nos aspectos terapêuticos, sugerindo que o uso do andador pode contribuir para uma reabilitação mais eficaz e segura.

Dentre os principais benefícios do sistema, destacam-se a capacidade de fornecer suporte dinâmico conforme a necessidade do paciente, ajustando o nível de assistência de acordo com a sua capacidade motora em tempo real. Essa característica reduz o risco de quedas e lesões, um fator crítico no tratamento de pacientes com Parkinson, que frequentemente apresentam instabilidade postural e dificuldades de locomoção. A mobilidade omnidirecional do robô também oferece maior liberdade de movimento, permitindo deslocamentos em várias direções, o que melhora a eficiência do treinamento de marcha e aumenta a flexibilidade das atividades de reabilitação.

Os testes clínicos preliminares demonstraram resultados promissores em termos de aceitação pelos pacientes e melhoria das respostas motoras. Os pesquisadores observaram que o andador não apenas facilitou a reabilitação física, mas também aumentou a confiança dos pacientes ao realizar atividades diárias, devido à percepção de maior segurança e controle sobre seus movimentos. O próximo passo proposto pelos autores é expandir os ensaios clínicos para avaliar os impactos a longo prazo do uso do sistema em diferentes grupos de pacientes, bem como realizar ajustes no design para melhorar a ergonomia e a usabilidade do dispositivo.



Figura 15: Andador robótico descrito por (JIA; SONG; XU, 2013)

3.7 Matraji et al.

Matraji et al. (MATRAJI et al., 2018) conduzem uma investigação detalhada sobre o controle de trajetória utilizando o controle de modo deslizante Super-Twisting Sliding Mode Control (STSMC) adaptativo, aplicado a veículos de quatro rodas convencionais (não omnidirecionais), onde há incertezas significativas nos parâmetros do sistema. Os algoritmos STSMC são amplamente reconhecidos por sua robustez diante de incertezas e perturbações externas, sendo uma escolha eficaz em cenários onde as dinâmicas do sistema são pouco precisas ou sujeitas a variações.

O estudo foi implementado em um robô Pioneer 3AT, equipado com encoders ópticos de quadratura, que fornecem medições precisas das posições e velocidades angulares e lineares do sistema. O controle de junta também foi realizado por meio de controladores PID (Proporcional, Integral e Derivativo), com interrupções de controle ocorrendo a cada 5ms, o que garante uma alta taxa de atualização nas respostas do sistema. Os algoritmos STSMC foram integrados a um ambiente de simulação executado em Simulink, permitindo a análise e validação do desempenho do controle em tempo real.

Os resultados demonstraram que o controle STSMC oferece um tempo de resposta menor em comparação aos controladores PID convencionais, o que resulta em uma maior rapidez na estabilização do sistema frente a distúrbios ou mudanças de trajeto. Contudo, observou-se que, em regime permanente, o comportamento dos controladores STSMC e PID foi similar, com oscilações visíveis no controle PID, que se manifestam como pequenas flutuações na posição final. O aspecto adaptativo do controle STSMC foi empregado para minimizar essas vibrações indesejadas, proporcionando uma suavização do comportamento dinâmico.

Para validar os resultados do estudo, foi implementado um processo de identificação de posição e orientação do veículo utilizando um sistema de rastreamento por câmera com alta resolução, de 2,7k e 30 quadros por segundo, posicionada a uma altura de três metros acima do veículo. O robô foi equipado com um conjunto de LEDs que facilitou a detecção por meio de algoritmos de visão computacional. Essa metodologia permitiu a captura precisa do movimento do robô, fornecendo dados que corroboraram a eficácia do controle adaptativo STSMC em minimizar erros e oscilações.

Esses achados sugerem que o STSMC adaptativo pode ser uma solução eficaz para o controle de robôs móveis em ambientes onde há incertezas e perturbações frequentes, com potencial de aplicação em uma ampla gama de sistemas robóticos que requerem alta precisão e robustez frente a condições dinâmicas variáveis.



Figura 16: Robô Pioneer 3AT (MATRAJI et al., 2018)

3.7.1 Análise comparativa

Os trabalhos apresentados oferecem abordagens diversas para o desenvolvimento de sistemas robóticos voltados à reabilitação e assistência à locomoção, cada um com características e aplicações distintas. O sistema Discover2Walk (D2W) se destaca por sua arquitetura modular bioinspirada, organizada em três níveis hierárquicos (alto, médio e baixo), que proporcionam alta configurabilidade e precisão nos movimentos. A plataforma omnidirecional do D2W permite maior liberdade de locomoção durante o processo de marcha, sendo adequada tanto para o treinamento quanto para a assistência.

Por outro lado, a plataforma CPWalker combina um exoesqueleto com um andador diferencial inteligente, sendo projetada para a reabilitação autônoma supervisionada, com controle postural e suporte parcial de peso. A utilização da estratégia *Assist as Needed* (AAN) diferencia o CPWalker, pois ajusta o nível de assistência ao paciente conforme necessário, reduzindo a carga sobre os terapeutas e permite maior flexibilidade na reabilitação. Essa abordagem oferece um equilíbrio entre locomoção ativa e passiva, promovendo a participação ativa do paciente no processo de recuperação.

O sistema Cable-drive Active Leg Exoskeleton (C-ALEX), por sua vez, utiliza uma

abordagem inovadora com manipuladores paralelos acionados por cabos, eliminando a necessidade de motores de alto torque nas juntas. Essa configuração reduz a inércia e melhora a eficiência energética do sistema, sendo altamente precisa na execução de movimentos. No entanto, o C-ALEX apresenta uma limitação significativa ao restringir a reabilitação a esteiras, reduzindo a flexibilidade do sistema em comparação a plataformas móveis como o D2W e o CPWalker.

Em contrapartida, o andador assistivo desenvolvido por Chotikunnnan et al. se diferencia pela utilização de rodas omnidirecionais Mecanum, o que proporciona liberdade de movimento em qualquer direção. Embora essa configuração seja vantajosa em termos de controle posicional, o foco deste sistema é a assistência à mobilidade convencional, não sendo direcionado à reabilitação motora, como o CPWalker ou o D2W. Além disso, estudos teóricos de Zeidis e Zimmermann sobre a modelagem de robôs omnidirecionais também se baseiam em rodas Mecanum e investigam a dinâmica de sistemas com restrições não holonômicas, porém sem foco direto na reabilitação.

O trabalho de Jianchen et al. desenvolve um robô móvel omnidirecional com suporte parcial de peso, voltado para a reabilitação de pacientes com doença de Parkinson. Diferente dos outros sistemas, este robô emprega sensores inerciais para detectar a intenção de movimento do paciente, facilitando o treinamento de marcha de maneira adaptativa.

Por fim, o estudo de Matraji et al. sobre o controle de trajetória com algoritmos Super-Twisting Sliding Mode Control (STSMC) para veículos de quatro rodas oferece contribuições relevantes para o controle robusto em sistemas robóticos. Embora o foco principal não seja a reabilitação, os métodos desenvolvidos podem ser aplicados para melhorar a precisão no controle de atuadores em plataformas como o D2W e o CPWalker, onde a precisão do movimento é essencial.

De maneira geral, o CPWalker e o D2W se destacam como soluções abrangentes para a reabilitação motora, cada um com seus pontos fortes em termos de controle, flexibilidade e aplicabilidade clínica. O C-ALEX oferece uma alternativa eficiente, mas com limitações em termos de mobilidade, enquanto os trabalhos de Chotikunnnan et al. e Zeidis e Zimmermann trazem inovações na locomoção, sem foco direto em reabilitação. O sistema de Jianchen et al. tem potencial promissor para reabilitação, especialmente no contexto de doenças específicas como o Parkinson, enquanto o estudo de Matraji et al. contribui com avanços no controle de trajetória, relevantes para diversas aplicações robóticas.

3.7.2 Tabela comparativa

A Tabela 1 apresenta uma breve comparação entre os trabalhos relacionados, destacando algumas das plataformas de reabilitação de marcha.

Plataforma	Tipo de Locomoção	Foco Principal	Controle	Movimentação
D2W (PALOMINO-DÍAZ et al., 2021)	Omnidirecional, acionado por cabos	Assistência e treinamento de marcha	Modular (ROS2)	Alta liberdade
CPWalker (BAYON et al., 2017)	Diferencial, juntas seriais	Assistência e treinamento de marcha	AAN	Movimentação restrita
C-ALEX (JIN; CUI; AGRAWAL, 2015)	Acionado por cabos	Reabilitação em esteira	AAN	Restrita a esteira
Chotikunnan et al. (CHOTIKUN-NAN et al., 2017)	Omnidirecional	Assistência à mobilidade	PID	Alta liberdade
Zeidis e Zimmermann (ZEIDIS; ZIMMERMANN, 2019)	Omnidirecional	Modelagem dinâmica e cinemática	Controle não holonômico	Alta liberdade
Jianchen et al. (JIA; SONG; XU, 2013)	Omnidirecional	Reabilitação para Parkinson	Controle inercial	Alta liberdade
textbfMatraji et al. (MATRAJI et al., 2018)	Não omnidirecional	Controle de trajetória	STSMC Adaptativo	Limitada

Tabela 1: Dispositivos robóticos revisados

4 DESENVOLVIMENTO

Ao abordar a robótica móvel aplicada à reabilitação da marcha, o presente trabalho foca no desenvolvimento e estudo de uma plataforma robótica específica: um robô móvel omnidirecional equipado com rodas Mecanum. A modelagem cinemática é o ponto de partida, envolvendo desafios de álgebra vetorial e mecânica clássica. Devido à configuração paralela das juntas do robô, o processo inicia-se com a análise da cinemática inversa, que fornece a relação entre a posição desejada do robô e os ângulos das rodas. A partir daí, o modelo é invertido para derivar a cinemática direta, permitindo calcular o movimento do robô com base nos ângulos das rodas.

Neste estágio, o projeto se desdobra em duas abordagens distintas para tratar a cinemática e a dinâmica do sistema. A primeira abordagem é a aproximação por mínimos quadrados, que simplifica o modelo cinemático, facilitando o desenvolvimento do controle e a formulação das equações dinâmicas. Essa abordagem torna o sistema mais acessível, com uma implementação computacional mais eficiente. A segunda abordagem utiliza uma resolução exata por meio da redução dimensional do espaço de soluções, que oferece uma representação mais precisa e detalhada do comportamento do robô. No entanto, essa técnica introduz complexidades adicionais, como restrições não holonômicas e a necessidade de multiplicadores de Lagrange para resolver as equações de movimento. Ambas as técnicas são implementadas com o formalismo de Lagrange, criando uma base sólida para o controle e simulação dinâmica, ao mesmo tempo em que permite uma validação abrangente do sistema.

Após a formulação das abordagens dinâmicas, o trabalho prossegue com o desenvolvimento de modelos numéricos. Primeiramente, é elaborado um programa em Python para validar os conceitos teóricos, ilustrando, por meio de gráficos e dados, os diferentes estados e modos de movimentação da plataforma robótica. Este programa permite observar, em tempo real, a resposta do sistema a diversas condições iniciais e de controle, fornecendo informações valiosas sobre a performance do robô. Em seguida, o modelo é implementado no ambiente ROS2 (Robot Operating System 2) e simulado no Gazebo, um ambiente de simulação tridimensional amplamente utilizado na robótica. A simulação no Gazebo permite a visualização detalhada do robô em um cenário virtual, além

de facilitar a integração com outros sistemas, como manipuladores paralelos ou robôs de exoesqueleto controlados por cabos. Essa etapa é fundamental para testar o comportamento do robô antes de sua construção física, garantindo que os algoritmos de controle e os modelos dinâmicos funcionem conforme o esperado.

O desenvolvimento do projeto culmina com a implementação de algoritmos de controle adequados ao robô omnidirecional e à criação de um protótipo físico de baixo custo em escala reduzida. Este protótipo permite validar os conceitos desenvolvidos ao longo do trabalho de forma prática, testando o robô em um ambiente real. A construção do protótipo também oferece a oportunidade de realizar ajustes e refinamentos nos algoritmos de controle, além de fornecer dados experimentais para futuras expansões e melhorias no projeto. O protótipo serve como uma ponte entre a simulação e o desenvolvimento de uma plataforma em escala completa, sendo uma etapa importante antes da apresentação dos resultados finais.

4.1 Descrição da Plataforma

O robô é composto de uma estrutura de alumínio extrudado e porcas T, pela facilidade de construção e prototipagem, sendo subdividido em duas partes principais: (i) conjunto de tração omnidirecional e (ii) conjunto de suporte de peso do paciente. A primeira habilita a mobilidade da plataforma, enquanto a segunda irá suportar o CDPR. Estes aspectos podem ser visualizados no modelo CAD (Figura 17), adaptado do sistema D2W (DELGADO-OLEAS et al., 2023). Para os fins deste trabalho, o foco será no aspecto móvel da plataforma.

Visto isso, já que a modelagem completa tridimensional de plataformas robóticas móveis apresenta demasiada complexidade analítica, sendo preferida a simulação em software por motor de física (SEEGMILLER, 2014), a obtenção das equações de movimento é realizada sobre um sistema planar. Posteriormente a modelagem no Gazebo é realizada, para integrar a funcionalidade 3D, possibilitando simulações mais complexas.

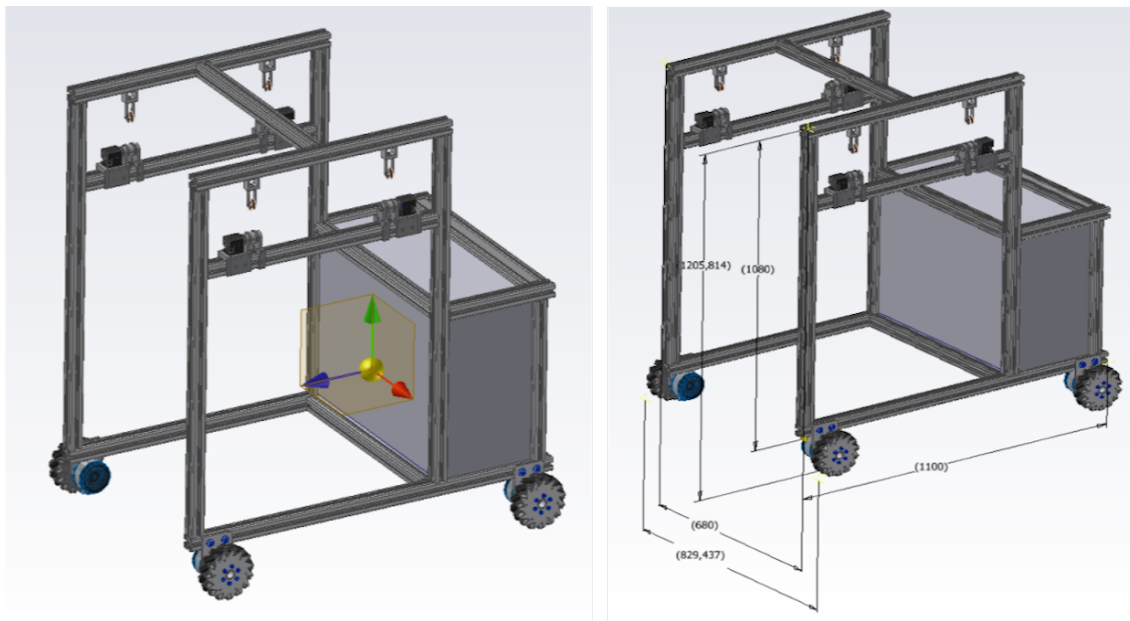


Figura 17: Modelo CAD da plataforma

4.1.1 Parâmetros

Os seguintes parâmetros, essenciais para o desenvolvimento do equacionamento planar da plataforma, foram extraídos diretamente do modelo CAD. Esses dados são cruciais para garantir a precisão na modelagem e na simulação do comportamento dinâmico do robô. Os parâmetros incluem:

Parâmetro	Valor	Unidade	Descrição
R	0.127	m	raio da roda
l_1	0.5975	m	distância lateral
l_2	0.735	m	distância vertical
m_c	15	Kg	massa da plataforma
m_k	6.5	Kg	massa individual da roda
J_0	0.32	$Kg \cdot m^2$	momento de inércia da plataforma
J_1	$6.25e - 4$	$Kg \cdot m^2$	momento de inércia individual da roda
J_2	$3.13e - 4$	$Kg \cdot m^2$	momento de inércia da roda em relação a C
μ	0.02	—	coeficiente de atrito

Tabela 2: Parâmetros cinemáticos e dinâmicos do robô

4.2 Modelagem Cinemática

Após considerar diversos métodos de modelagem, descritos em (AZIZI; RASTEGARPANAH; STOLKIN, 2021), (HAN; ZHU, 2019), (HENDZEL; RYKAŁA, 2017), (JIA; SONG; XU, 2013), (TĂTAR et al., 2014), (TLALE; VILLIERS, 2008), (TSAI; TAI; LEE, 2011), (ZEIDIS; ZIMMERMANN, 2019) e (ZIMMERMANN; ZEIDIS; ABDELRAHMAN, 2014), a técnica apresentada por Zeidis e Zimmerman (ZEIDIS; ZIMMERMANN, 2019), por ter um uso mais elegante e generalizado de álgebra vetorial, foi escolhida e modificada para os propósitos deste projeto. A estrutura mecânica básica do robô móvel pode ser descrita por meio de um diagrama (Figura 18), onde o centroide C e as quatro rodas omnidirecionais estão presentes.

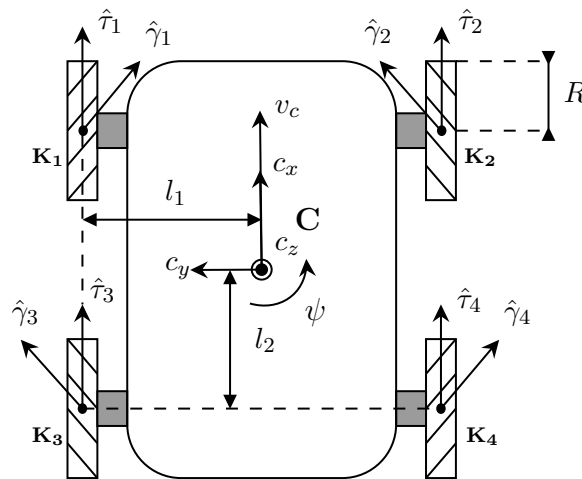


Figura 18: Diagrama da plataforma omnidirecional

Outrossim, é possível descrever um diagrama para a n -ésima roda (Figura 19), onde K_i é o centroide da roda, e P_i o ponto de contato do rolete com a superfície.

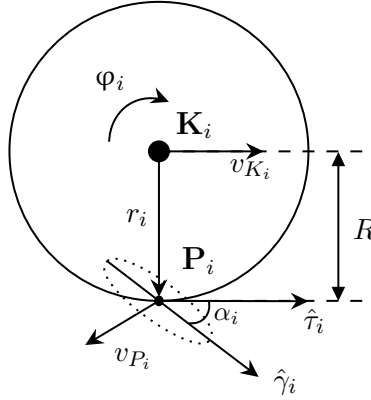


Figura 19: Diagrama da Roda Omnidirecional

Para começar o equacionamento, é definido o vetor de velocidade no ponto de contato v_{P_i} , em função da rotação e da velocidade do centro da roda v_{K_i} . Para representar a velocidade angular, é utilizado o vetor axial Ω_i , tal que $\|\Omega_i\| = \dot{\varphi}_i$.

$$v_{P_i} = v_{K_i} + \Omega_i \times r_i \quad (1)$$

Destaca-se que o eixo do rolete $\hat{\gamma}_i$ é perpendicular ao vetor de velocidade do ponto de contato v_{P_i} .

$$v_{P_i} \cdot \hat{\gamma}_i = 0 \quad (2)$$

Portanto, multiplicando a equação 1 por $\hat{\gamma}_i$ e substituindo a equação 2, obtem a expressão 3.

$$0 = (v_{K_i} + \Omega_i \times r_i) \cdot \hat{\gamma}_i \quad (3)$$

Assim, o versor tangente à roda $\hat{\tau}_i$ é aproveitado para simplificar a equação 3.

$$\Omega_i \times r_i = R\dot{\varphi}_i \hat{\tau}_i \quad (4)$$

$$\begin{aligned} v_{K_i} \cdot \hat{\gamma}_i &= R\dot{\varphi}_i (\hat{\tau}_i \cdot \hat{\gamma}_i) \\ v_{K_i} \cdot \hat{\gamma}_i &= R\dot{\varphi}_i \cos \alpha \end{aligned} \quad (5)$$

Para relacionar a movimentação do centro de massa da roda com o centro de massa do robô, é definido o vetor de deslocamento $\rho_i = \overrightarrow{CK_i}$, em conjunto com o vetor axial Θ , tal que $\|\Theta\| = \dot{\psi}$.

$$v_{K_i} = v_c + \Theta \times \rho_i \quad (6)$$

Então, substituem-se 6 e 5 resultando na equação 7, que representa a condição de não deslizamento dos roletes aplicada à cinemática da plataforma.

$$\begin{aligned} v_c \cdot \hat{\gamma}_i + (\Theta \times \rho_i) \cdot \hat{\gamma}_i &= R\dot{\phi}_i \cos \alpha \\ v_c \cdot \hat{\gamma}_i + (\rho_i \times \hat{\gamma}_i) \cdot \Theta &= R\dot{\phi}_i \cos \alpha \end{aligned} \quad (7)$$

Em seguida, a decomposição de 7 em função dos componentes de cada vetor obtém a relação 8.

$$v_{c_x} \gamma_{i_x} + v_{c_y} \gamma_{i_y} + (\rho_{i_x} \gamma_{i_y} - \rho_{i_y} \gamma_{i_x}) \dot{\psi} = R\dot{\phi}_i \cos \alpha \quad (8)$$

É necessário então definir $\cos \alpha$, ρ_i e $\hat{\gamma}_i$ para $i = 1, 2, 3, 4$.

$$\gamma_{1_x} = \gamma_{4_x} = \frac{1}{\sqrt{2}} \quad \gamma_{1_y} = \gamma_{4_y} = \frac{-1}{\sqrt{2}}$$

$$\gamma_{2_x} = \gamma_{3_x} = \frac{1}{\sqrt{2}} \quad \gamma_{2_y} = \gamma_{3_y} = \frac{1}{\sqrt{2}}$$

$$\rho_{1_x} = \rho_{2_x} = l_2 \quad \rho_{1_y} = \rho_{3_y} = l_1$$

$$\rho_{3_x} = \rho_{4_x} = -l_2 \quad \rho_{2_y} = \rho_{4_y} = -l_1$$

$$\cos \alpha = \cos \left(\frac{\pi}{4} \right) = \frac{1}{\sqrt{2}}$$

Assim é gerado o sistema de equações 9.

$$\begin{cases} v_{c_x} - v_{c_y} - (l_1 + l_2) \dot{\psi} - R\dot{\phi}_1 = 0 \\ v_{c_x} + v_{c_y} + (l_1 + l_2) \dot{\psi} - R\dot{\phi}_2 = 0 \\ v_{c_x} + v_{c_y} - (l_1 + l_2) \dot{\psi} - R\dot{\phi}_3 = 0 \\ v_{c_x} - v_{c_y} + (l_1 + l_2) \dot{\psi} - R\dot{\phi}_4 = 0 \end{cases} \quad (9)$$

4.2.1 Cinemática Inversa

Ao reescrever o sistema 9 na forma vetorial-matricial, desenvolve a seguinte equação de cinemática inversa do robô móvel, que possibilita obter os estados de juntas em função das velocidades lineares e angulares do centroide.

$$\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & -1 & -(l_1 + l_2) \\ 1 & 1 & (l_1 + l_2) \\ 1 & 1 & -(l_1 + l_2) \\ 1 & -1 & (l_1 + l_2) \end{bmatrix} \begin{bmatrix} v_{c_x} \\ v_{c_y} \\ \dot{\psi} \end{bmatrix} \quad (10)$$

4.2.2 Cinemática Direta Exata

O sistema 10 é sobredeterminado, e não possui solução para valores arbitrários de $\dot{\phi}_i$, portanto é necessário restringir o espaço de soluções. Para isto, $\dot{\phi}_4$ deve ser temporariamente removido.

$$\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & -1 & -(l_1 + l_2) \\ 1 & 1 & (l_1 + l_2) \\ 1 & 1 & -(l_1 + l_2) \end{bmatrix} \begin{bmatrix} v_{c_x} \\ v_{c_y} \\ \dot{\psi} \end{bmatrix} = \mathcal{J}_1 \begin{bmatrix} v_{c_x} \\ v_{c_y} \\ \dot{\psi} \end{bmatrix} \quad (11)$$

Pode-se então inverter a matriz $\mathcal{J}_1$, encontrando a seguinte relação 12.

$$\mathcal{J}_1^{-1} = \frac{R}{2} \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & \frac{1}{l_1 + l_2} & \frac{-1}{l_1 + l_2} \end{bmatrix}$$

$$\therefore \begin{bmatrix} v_{c_x} \\ v_{c_y} \\ \dot{\psi} \end{bmatrix} = \frac{R}{2} \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & \frac{1}{l_1 + l_2} & \frac{-1}{l_1 + l_2} \end{bmatrix} \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} \quad (12)$$

Agora, basta reescrever $\dot{\phi}_4$ como uma combinação linear de $\dot{\phi}_1$, $\dot{\phi}_2$ e $\dot{\phi}_3$.

$$\dot{\phi}_4 = \begin{bmatrix} 1 & -1 & (l_1 + l_2) \end{bmatrix} \cdot \mathcal{J}_1^{-1} \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix}$$

$$\therefore \dot{\phi}_4 = \begin{bmatrix} 1 & 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} \quad (13)$$

Ao juntar as equações 12 e 13, o modelo cinemático direto parcial 14 é obtido.

$$\begin{bmatrix} v_{cx} \\ v_{cy} \\ \dot{\psi} \\ \dot{\phi}_4 \end{bmatrix} = \mathcal{J}_2 \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} \quad (14)$$

Onde a matriz $\mathcal{J}_2$ é:

$$\mathcal{J}_2 = \begin{bmatrix} \frac{R}{2} & \frac{R}{2} & 0 \\ -\frac{R}{2} & 0 & \frac{R}{2} \\ 0 & \frac{R}{2(l_1+l_2)} & \frac{-R}{2(l_1+l_2)} \\ 1 & 1 & -1 \end{bmatrix} \quad (15)$$

Assim, é possível isolar o sistema de equações que representa restrições de cinemática direta:

$$\begin{cases} v_{cx} = \frac{R}{2}(\dot{\phi}_1 + \dot{\phi}_2) \\ v_{cy} = \frac{R}{2}(-\dot{\phi}_1 + \dot{\phi}_3) \\ \dot{\psi} = \frac{R}{2(l_1+l_2)}(\dot{\phi}_2 - \dot{\phi}_3) \\ \dot{\phi}_4 = \dot{\phi}_1 + \dot{\phi}_2 - \dot{\phi}_3 \end{cases} \quad (16)$$

Porém, em 16, as velocidades lineares ainda estão em função de projeções no sistema de coordenadas móvel, do ponto C . Para obter as velocidades $\dot{x}_c$ e $\dot{y}_c$, em relação ao sistema de coordenadas global, é requerido considerar a rotação em ψ .

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\psi} \end{bmatrix} = T_1(\psi) \begin{bmatrix} v_{cx} \\ v_{cy} \\ \dot{\psi} \end{bmatrix} \quad (17)$$

$$T_1(\psi) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Pela incorporação de 17 na equação 14, se obtém a relação 18, que constitui o modelo cinemático direto completo da plataforma, representado em vetores e matrizes de blocos.

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\psi} \\ \dot{\phi}_4 \end{bmatrix} = \left[\begin{array}{c|c} T_1(\psi) & \mathbf{0} \\ \hline \mathbf{0} & 1 \end{array} \right] \mathcal{J}_2 \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} \quad (18)$$

4.2.3 Cinemática Direta Aproximada

A cinemática direta também pode ser realizada pela técnica aproximada. Assim, retornando à equação 10 de cinemática inversa, uma forma de obter a cinemática direta é por aproximação de mínimos quadrados, através da matriz pseudo-inversa de Moor-Penrose (AZIZI; RASTEGARPANAH; STOLKIN, 2021).

$$\begin{bmatrix} v_{c_x} \\ v_{c_y} \\ \dot{\psi} \end{bmatrix} = \mathcal{J}_3^+ \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix} \quad (19)$$

Onde:

$$\mathcal{J}_3 = \frac{1}{R} \begin{bmatrix} 1 & -1 & -(l_1 + l_2) \\ 1 & 1 & (l_1 + l_2) \\ 1 & 1 & -(l_1 + l_2) \\ 1 & -1 & (l_1 + l_2) \end{bmatrix} \quad (20)$$

$$\mathcal{J}_3^+ = (\mathcal{J}_3^T \mathcal{J}_3)^{-1} \mathcal{J}_3^T = \frac{R}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ \frac{-1}{(l_1+l_2)} & \frac{1}{(l_1+l_2)} & \frac{-1}{(l_1+l_2)} & \frac{1}{(l_1+l_2)} \end{bmatrix} \quad (21)$$

Pode-se, então, isolar as equações de cinemática direta aproximada no sistema 22

$$\begin{cases} v_{c_x} = \frac{R}{4} (\dot{\phi}_1 + \dot{\phi}_2 + \dot{\phi}_3 + \dot{\phi}_4) \\ v_{c_y} = \frac{R}{4} (-\dot{\phi}_1 + \dot{\phi}_2 + \dot{\phi}_3 - \dot{\phi}_4) \\ \dot{\psi} = \frac{R}{4(l_1+l_2)} (-\dot{\phi}_1 + \dot{\phi}_2 - \dot{\phi}_3 + \dot{\phi}_4) \end{cases} \quad (22)$$

Neste caso, as velocidades lineares continuam sendo projetadas no sistema de coordenadas móvel do robô. Para conseguir as velocidades $\dot{x}_c$ e $\dot{y}_c$, se deve utilizar novamente a equação 17.

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\psi} \end{bmatrix} = T_1(\psi) \mathcal{J}_3^+ \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix} \quad (23)$$

Portanto, a equação 23 estabelece a cinemática direta aproximada da plataforma.

4.3 Modelagem Dinâmica Exata

Ao progredir para a modelagem dinâmica, a partir de (ZEIDIS; ZIMMERMANN, 2019), se observa o sistema 18, de onde são retiradas as equações de restrição.

$$\dot{x}_c = \frac{R}{\sqrt{2}} \cos\left(\psi - \frac{\pi}{4}\right) \dot{\varphi}_1 + \frac{R}{2} (\cos \psi \dot{\varphi}_2 - \sin \psi \dot{\varphi}_3) \quad (24)$$

$$\dot{y}_c = \frac{R}{\sqrt{2}} \sin\left(\psi - \frac{\pi}{4}\right) \dot{\varphi}_1 + \frac{R}{2} (\sin \psi \dot{\varphi}_2 + \cos \psi \dot{\varphi}_3) \quad (25)$$

$$\dot{\psi} = \frac{R}{2(l_1 + l_2)} (\dot{\varphi}_2 - \dot{\varphi}_3) \quad (26)$$

$$\dot{\varphi}_4 = \dot{\varphi}_1 + \dot{\varphi}_2 - \dot{\varphi}_3 \quad (27)$$

As equações 26 e 27 podem ser integradas, formulando restrições holonômicas, onde c_1 e c_2 são constantes.

$$\begin{aligned} \int (\dot{\psi}) dt &= \frac{R}{2(l_1 + l_2)} \int (\dot{\varphi}_2 - \dot{\varphi}_3) dt \\ \therefore \psi &= \frac{R}{2(l_1 + l_2)} (\varphi_2 - \varphi_3) + c_1 \end{aligned} \quad (28)$$

$$\begin{aligned} \int (\dot{\varphi}_4) dt &= \int (\dot{\varphi}_1 + \dot{\varphi}_2 - \dot{\varphi}_3) dt \\ \therefore \varphi_4 &= \varphi_1 + \varphi_2 - \varphi_3 + c_2 \end{aligned} \quad (29)$$

Porém, as expressões 24 e 25 não são integráveis, configurando restrições não holonômicas. Em função disso, o equacionamento dinâmico deverá considerar as forças de restrição em função de multiplicadores de Lagrange. Assim, é definida a equação de Euler-Lagrange, onde $G(q)^T \lambda$ são as forças de restrição, $F_v(\dot{q})$ são forças de atrito linear e Q são forças de entrada generalizadas.

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} + G(q)^T \lambda + F_v(\dot{q}) = Q \quad (30)$$

$$q = [\varphi_1 \quad \varphi_2 \quad \varphi_3]^T \quad (31)$$

Já que o movimento do robô é planar, a energia potencial é nula, portanto o Lagrangiano é:

$$\begin{aligned} \mathcal{L} &= K - U \\ U &= 0 \end{aligned} \quad (32)$$

Definindo a energia cinética, os termos rotacionais e translacionais são separados, obtendo a expressão 36.

$$K = K_r + K_p \quad (33)$$

$$K_p = \frac{m_s}{2} (\dot{x}_c^2 + \dot{y}_c^2) \quad (34)$$

$$K_r = \frac{J_c}{2} \dot{\psi}^2 + \frac{J_1}{2} (\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + \dot{\phi}_4^2) \quad (35)$$

$$\therefore \mathcal{L} = \frac{m_s}{2} (\dot{x}_c^2 + \dot{y}_c^2) + \frac{J_c}{2} \dot{\psi}^2 + \frac{J_1}{2} (\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + \dot{\phi}_4^2) \quad (36)$$

Para obter o Lagrangiano somente em função das coordenadas de junta, o sistema 14 é modificado para obter as seguintes relações:

$$(\dot{x}_c^2 + \dot{y}_c^2) = v_{c_x}^2 + v_{c_y}^2 = \frac{R^2}{4} (2\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + 2\dot{\phi}_1\dot{\phi}_2 - 2\dot{\phi}_1\dot{\phi}_3) \quad (37)$$

$$\dot{\psi}^2 = \frac{R^2}{4(l_1 + l_2)^2} (\dot{\phi}_2^2 + \dot{\phi}_3^2 - 2\dot{\phi}_2\dot{\phi}_3) \quad (38)$$

$$(\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + \dot{\phi}_4^2) = 2\dot{\phi}_1^2 + 2\dot{\phi}_2^2 + 2\dot{\phi}_3^2 + 2\dot{\phi}_1\dot{\phi}_2 - 2\dot{\phi}_1\dot{\phi}_3 - 2\dot{\phi}_2\dot{\phi}_3 \quad (39)$$

Assim, substituindo 37, 38 e 39 na equação 36, é obtida a expressão final do Lagrangiano 40.

$$\begin{aligned} \mathcal{L} = & \frac{A}{2} (2\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + 2\dot{\phi}_1\dot{\phi}_2 - 2\dot{\phi}_1\dot{\phi}_3) + \frac{B}{2} (\dot{\phi}_2^2 + \dot{\phi}_3^2 - 2\dot{\phi}_2\dot{\phi}_3) \\ & + \frac{C}{2} (2\dot{\phi}_1^2 + 2\dot{\phi}_2^2 + 2\dot{\phi}_3^2 + 2\dot{\phi}_1\dot{\phi}_2 - 2\dot{\phi}_1\dot{\phi}_3 - 2\dot{\phi}_2\dot{\phi}_3) \end{aligned} \quad (40)$$

Onde:

$$A = \frac{m_s R^2}{4}, \quad B = \frac{J_c R^2}{4(l_1 + l_2)^2}, \quad C = J_1$$

$$m_s = m_c + 4m_k, \quad J_c = J_0 + 4(J_2 + m_k(l_1^2 + l_2^2))$$

4.3.1 Derivadas Parciais de Coordenadas Generalizadas

Para continuar a modelagem dinâmica, é necessário obter as derivadas parciais da equação 40 em relação às variáveis relevantes.

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_1} &= \frac{1}{2} [A(4\dot{\varphi}_1 + 2\dot{\varphi}_2 - 2\dot{\varphi}_3) + B(0) + C(4\dot{\varphi}_1 + 2\dot{\varphi}_2 - 2\dot{\varphi}_3)] \\
&= 2(A + C)\dot{\varphi}_1 + (A + C)\dot{\varphi}_2 - (A + C)\dot{\varphi}_3 \\
\therefore \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_1} \right) &= 2(A + C)\ddot{\varphi}_1 + (A + C)\ddot{\varphi}_2 - (A + C)\ddot{\varphi}_3 \quad (41)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_2} &= \frac{1}{2} [A(2\dot{\varphi}_2 + 2\dot{\varphi}_1) + B(2\dot{\varphi}_2 - 2\dot{\varphi}_3) + C(4\dot{\varphi}_2 + 2\dot{\varphi}_1 - 2\dot{\varphi}_3)] \\
&= (A + C)\dot{\varphi}_1 + (A + B + 2C)\dot{\varphi}_2 - (B + C)\dot{\varphi}_3 \\
\therefore \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_2} \right) &= (A + C)\ddot{\varphi}_1 + (A + B + 2C)\ddot{\varphi}_2 - (B + C)\ddot{\varphi}_3 \quad (42)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_3} &= \frac{1}{2} [A(2\dot{\varphi}_3 - 2\dot{\varphi}_1) + B(2\dot{\varphi}_3 - 2\dot{\varphi}_2) + C(4\dot{\varphi}_3 - 2\dot{\varphi}_1 - 2\dot{\varphi}_2)] \\
&= -(A + C)\dot{\varphi}_1 - (B + C)\dot{\varphi}_2 + (A + B + 2C)\dot{\varphi}_3 \\
\therefore \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_3} \right) &= -(A + C)\ddot{\varphi}_1 - (B + C)\ddot{\varphi}_2 + (A + B + 2C)\ddot{\varphi}_3 \quad (43)
\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial \varphi_1} = \frac{\partial \mathcal{L}}{\partial \varphi_2} = \frac{\partial \mathcal{L}}{\partial \varphi_3} = 0 \quad (44)$$

4.3.2 Derivadas Parciais de Restrição

Em função das restrições não holonômicas, também é necessário obter as parciais em relação às coordenadas de velocidade $\dot{x}_c$ e $\dot{y}_c$. Para isto é aplicado o Lagrangiano, definido na equação 36.

$$\frac{\partial \mathcal{L}}{\partial \dot{x}_c} = m_s \dot{x}_c = \frac{m_s R}{2} \left[\sqrt{2} \sin \left(\psi + \frac{\pi}{4} \right) \dot{\varphi}_1 + \cos(\psi) \dot{\varphi}_2 - \sin(\psi) \dot{\varphi}_3 \right] \quad (45)$$

$$\frac{\partial \mathcal{L}}{\partial \dot{y}_c} = m_s \dot{y}_c = \frac{m_s R}{2} \left[\sqrt{2} \sin \left(\psi - \frac{\pi}{4} \right) \dot{\varphi}_1 + \sin(\psi) \dot{\varphi}_2 + \cos(\psi) \dot{\varphi}_3 \right] \quad (46)$$

4.3.3 Forças de Restrição

Já que o modelo em questão pode ser interpretado como um sistema de Chaplygin (ZEIDIS; ZIMMERMANN, 2019), as forças não holonômicas de restrição, referentes aos multiplicadores de Lagrange, possuem o formato de 47.

$$G(q)^T \lambda = P(\dot{\varphi})$$

$$P_i = \sum_{k=1}^3 \left(\frac{\partial \mathcal{L}}{\partial \dot{x}_c} \alpha_{ki} + \frac{\partial \mathcal{L}}{\partial \dot{y}_c} \beta_{ki} \right) \dot{\varphi}_k \quad (47)$$

Onde:

$$\alpha = \mathcal{J}a(\varphi) - \mathcal{J}a(\varphi)^T \quad (48)$$

$$\beta = \mathcal{J}b(\varphi) - \mathcal{J}b(\varphi)^T \quad (49)$$

Para calcular as matrizes antissimétricas α e β , primeiramente a relação 28 é substituída em 24 e 25, isolando as variáveis de junta, para obter os coeficientes a_i e b_i .

$$\dot{x}_c = a_1 \dot{\varphi}_1 + a_2 \dot{\varphi}_2 + a_3 \dot{\varphi}_3 \quad (50)$$

$$\dot{y}_c = b_1 \dot{\varphi}_1 + b_2 \dot{\varphi}_2 + b_3 \dot{\varphi}_3 \quad (51)$$

Onde:

$$\begin{aligned} a_1 &= \frac{R}{\sqrt{2}} \cos\left(\psi - \frac{\pi}{4}\right), \quad a_2 = \frac{R}{2} \cos \psi, \quad a_3 = -\frac{R}{2} \sin \psi \\ b_1 &= \frac{R}{\sqrt{2}} \sin\left(\psi - \frac{\pi}{4}\right), \quad b_2 = \frac{R}{2} \sin \psi, \quad b_3 = \frac{R}{2} \cos \psi \end{aligned}$$

Deve-se então desenvolver os jacobianos:

$$\begin{aligned} \mathcal{J}a(\varphi) &= \begin{bmatrix} \frac{\partial a_1}{\partial \varphi_1} & \frac{\partial a_1}{\partial \varphi_2} & \frac{\partial a_1}{\partial \varphi_3} \\ \frac{\partial a_2}{\partial \varphi_1} & \frac{\partial a_2}{\partial \varphi_2} & \frac{\partial a_2}{\partial \varphi_3} \\ \frac{\partial a_3}{\partial \varphi_1} & \frac{\partial a_3}{\partial \varphi_2} & \frac{\partial a_3}{\partial \varphi_3} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{R^2 \sqrt{2} \sin(\psi - \frac{\pi}{4})}{4(l_1 + l_2)} & \frac{R^2 \sqrt{2} \sin(\psi - \frac{\pi}{4})}{4(l_1 + l_2)} \\ 0 & -\frac{R^2 \sin(\psi)}{4(l_1 + l_2)} & \frac{R^2 \sin(\psi)}{4(l_1 + l_2)} \\ 0 & -\frac{R^2 \cos(\psi)}{4(l_1 + l_2)} & \frac{R^2 \cos(\psi)}{4(l_1 + l_2)} \end{bmatrix} \\ \therefore \alpha &= \frac{R^2}{2\sqrt{2}} \begin{bmatrix} 0 & -\sin(\psi - \frac{\pi}{4}) & \sin(\psi - \frac{\pi}{4}) \\ \sin(\psi - \frac{\pi}{4}) & 0 & \sin(\psi + \frac{\pi}{4}) \\ -\sin(\psi - \frac{\pi}{4}) & -\sin(\psi + \frac{\pi}{4}) & 0 \end{bmatrix} \end{aligned} \quad (52)$$

$$\begin{aligned} \mathcal{J}b(\varphi) &= \begin{bmatrix} \frac{\partial b_1}{\partial \varphi_1} & \frac{\partial b_1}{\partial \varphi_2} & \frac{\partial b_1}{\partial \varphi_3} \\ \frac{\partial b_2}{\partial \varphi_1} & \frac{\partial b_2}{\partial \varphi_2} & \frac{\partial b_2}{\partial \varphi_3} \\ \frac{\partial b_3}{\partial \varphi_1} & \frac{\partial b_3}{\partial \varphi_2} & \frac{\partial b_3}{\partial \varphi_3} \end{bmatrix} = \begin{bmatrix} 0 & \frac{R^2 \sqrt{2} \cos(\psi - \frac{\pi}{4})}{4(l_1 + l_2)} & -\frac{R^2 \sqrt{2} \cos(\psi - \frac{\pi}{4})}{4(l_1 + l_2)} \\ 0 & \frac{R^2 \cos(\psi)}{4(l_1 + l_2)} & -\frac{R^2 \cos(\psi)}{4(l_1 + l_2)} \\ 0 & -\frac{R^2 \sin(\psi)}{4(l_1 + l_2)} & \frac{R^2 \sin(\psi)}{4(l_1 + l_2)} \end{bmatrix} \\ \therefore \beta &= \frac{R^2}{2\sqrt{2}} \begin{bmatrix} 0 & \sin(\psi + \frac{\pi}{4}) & -\sin(\psi + \frac{\pi}{4}) \\ -\sin(\psi + \frac{\pi}{4}) & 0 & \sin(\psi - \frac{\pi}{4}) \\ \sin(\psi + \frac{\pi}{4}) & -\sin(\psi - \frac{\pi}{4}) & 0 \end{bmatrix} \end{aligned} \quad (53)$$

Finalmente, ao utilizar as equações 45, 46, 47, 52 e 53, são obtidas as forças de restrição.

$$\begin{aligned}
P_1 &= -\frac{m_s R^3}{8(l_1 + l_2)}(\dot{\phi}_2 + \dot{\phi}_3)(\dot{\phi}_2 - \dot{\phi}_3) \\
P_2 &= \frac{m_s R^3}{8(l_1 + l_2)}(\dot{\phi}_1 - \dot{\phi}_3)(\dot{\phi}_2 - \dot{\phi}_3) \\
P_3 &= \frac{m_s R^3}{8(l_1 + l_2)}(\dot{\phi}_1 + \dot{\phi}_2)(\dot{\phi}_2 - \dot{\phi}_3) \\
\therefore P(\dot{\phi}) &= \frac{m_s R^3}{8(l_1 + l_2)} \begin{bmatrix} (\dot{\phi}_2 + \dot{\phi}_3)(\dot{\phi}_2 - \dot{\phi}_3) \\ (\dot{\phi}_1 - \dot{\phi}_3)(\dot{\phi}_2 - \dot{\phi}_3) \\ (\dot{\phi}_1 + \dot{\phi}_2)(\dot{\phi}_2 - \dot{\phi}_3) \end{bmatrix} \quad (54)
\end{aligned}$$

4.3.4 Perdas de Energia por Fricção Viscosa

O equacionamento apresentado não inclui perdas de energia por fricção no Lagrangiano, portanto os termos relevantes são desenvolvidos separadamente.

$$D_v = \frac{\mu}{2} (2\dot{\phi}_1^2 + 2\dot{\phi}_2^2 + 2\dot{\phi}_3^2 + 2\dot{\phi}_1\dot{\phi}_2 - 2\dot{\phi}_1\dot{\phi}_3 - 2\dot{\phi}_2\dot{\phi}_3) \quad (55)$$

$$\frac{\partial D_v}{\partial \dot{\phi}_1} = 2\mu\dot{\phi}_1 + \mu\dot{\phi}_2 - \mu\dot{\phi}_3 \quad (56)$$

$$\frac{\partial D_v}{\partial \dot{\phi}_2} = \mu\dot{\phi}_1 + 2\mu\dot{\phi}_2 - \mu\dot{\phi}_3 \quad (57)$$

$$\frac{\partial D_v}{\partial \dot{\phi}_3} = -\mu\dot{\phi}_1 - \mu\dot{\phi}_2 + 2\mu\dot{\phi}_3 \quad (58)$$

$$\therefore F_v(\dot{\phi}) = \begin{bmatrix} 2\mu & \mu & -\mu \\ \mu & 2\mu & -\mu \\ -\mu & -\mu & 2\mu \end{bmatrix} \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} \quad (59)$$

4.3.5 Equações de Movimento

Ao reescrever a equação de Euler-Lagrange na forma vetorial-matricial, se obtém a expressão 60.

$$M \begin{bmatrix} \ddot{\phi}_1 \\ \ddot{\phi}_2 \\ \ddot{\phi}_3 \end{bmatrix} + F_v(\dot{\phi}) + P(\dot{\phi}) = Q(\tau) \quad (60)$$

Onde:

$$M = \begin{bmatrix} 2(A+C) & (A+C) & -(A+C) \\ (A+C) & (A+B+2C) & -(B+C) \\ -(A+C) & -(B+C) & (A+B+2C) \end{bmatrix}$$

$$Q(\tau) = S\tau = \begin{bmatrix} \tau_1 + \tau_4 \\ \tau_2 + \tau_4 \\ \tau_3 - \tau_4 \end{bmatrix}$$

$$S = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Finalmente, é possível isolar as acelerações angulares de junta, obtendo o modelo dinâmico exato 61.

$$\begin{bmatrix} \ddot{\varphi}_1 \\ \ddot{\varphi}_2 \\ \ddot{\varphi}_3 \end{bmatrix} = M^{-1}Q(\tau) - M^{-1}F_v(\dot{\varphi}) - M^{-1}P(\dot{\varphi}) \quad (61)$$

4.4 Modelagem Dinâmica Aproximada

No sistema aproximado, os termos antissimétricos de restrições não holonômicas são cancelados, portanto, é empregada a equação de Euler-Lagrange sem multiplicadores.

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} + F_v(\dot{q}) = Q \quad (62)$$

$$q = [\varphi_1 \quad \varphi_2 \quad \varphi_3 \quad \varphi_4]^T \quad (63)$$

Neste caso, o Lagrangiano tem o mesmo formato da equação 36. Para descrevê-lo somente em função das coordenadas de junta, o sistema 22 é modificado para obter as seguintes relações:

$$(\dot{x}_c^2 + \dot{y}_c^2) = v_{c_x}^2 + v_{c_y}^2 = \frac{R^2}{8} (\dot{\varphi}_1^2 + \dot{\varphi}_2^2 + \dot{\varphi}_3^2 + \dot{\varphi}_4^2 + 2\dot{\varphi}_1\dot{\varphi}_4 + 2\dot{\varphi}_2\dot{\varphi}_3) \quad (64)$$

$$\begin{aligned} \dot{\psi}^2 = & \frac{R^2}{16(l_1 + l_2)^2} (\dot{\varphi}_1^2 + \dot{\varphi}_2^2 + \dot{\varphi}_3^2 + \dot{\varphi}_4^2 - 2\dot{\varphi}_1\dot{\varphi}_2 + 2\dot{\varphi}_1\dot{\varphi}_3 - 2\dot{\varphi}_1\dot{\varphi}_4 \\ & - 2\dot{\varphi}_2\dot{\varphi}_3 + 2\dot{\varphi}_2\dot{\varphi}_4 - 2\dot{\varphi}_3\dot{\varphi}_4) \end{aligned} \quad (65)$$

Portanto, o Lagrangiano é:

$$\begin{aligned}
\mathcal{L} = & \frac{A_1}{2} (\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + \dot{\phi}_4^2 + 2\dot{\phi}_1\dot{\phi}_4 + 2\dot{\phi}_2\dot{\phi}_3) + \frac{B_1}{2} (\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + \dot{\phi}_4^2 \\
& - 2\dot{\phi}_1\dot{\phi}_2 + 2\dot{\phi}_1\dot{\phi}_3 - 2\dot{\phi}_1\dot{\phi}_4 - 2\dot{\phi}_2\dot{\phi}_3 + 2\dot{\phi}_2\dot{\phi}_4 - 2\dot{\phi}_3\dot{\phi}_4) \\
& + \frac{C_1}{2} (\dot{\phi}_1^2 + \dot{\phi}_2^2 + \dot{\phi}_3^2 + \dot{\phi}_4^2)
\end{aligned} \tag{66}$$

Onde:

$$A_1 = \frac{m_s R^2}{8}, \quad B_1 = \frac{J_c R^2}{16(l_1 + l_2)^2}, \quad C_1 = J_1$$

4.4.1 Derivadas Parciais de Coordenadas Generalizadas

É necessário então calcular as derivadas parciais do Lagrangeano em relação às coordenadas generalizadas e suas derivadas.

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \dot{\phi}_1} &= \frac{1}{2} [A_1(2\dot{\phi}_1 + 2\dot{\phi}_4) + B_1(2\dot{\phi}_1 - 2\dot{\phi}_2 + 2\dot{\phi}_3 - 2\dot{\phi}_4) + C_1(2\dot{\phi}_1)] \\
&= (A_1 + B_1 + C_1) \dot{\phi}_1 - (B_1) \dot{\phi}_2 + (B_1) \dot{\phi}_3 + (A_1 - B_1) \dot{\phi}_4
\end{aligned}$$

$$\therefore \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\phi}_1} \right) = (A_1 + B_1 + C_1) \ddot{\phi}_1 - (B_1) \ddot{\phi}_2 + (B_1) \ddot{\phi}_3 + (A_1 - B_1) \ddot{\phi}_4 \tag{67}$$

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \dot{\phi}_2} &= \frac{1}{2} [A_1(2\dot{\phi}_2 + 2\dot{\phi}_3) + B_1(2\dot{\phi}_2 - 2\dot{\phi}_1 - 2\dot{\phi}_3 + 2\dot{\phi}_4) + C_1(2\dot{\phi}_2)] \\
&= - (B_1) \dot{\phi}_1 + (A_1 + B_1 + C_1) \dot{\phi}_2 + (A_1 - B_2) \dot{\phi}_3 + (B_1) \dot{\phi}_4
\end{aligned}$$

$$\therefore \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\phi}_2} \right) = - (B_1) \ddot{\phi}_1 + (A_1 + B_1 + C_1) \ddot{\phi}_2 + (A_1 - B_2) \ddot{\phi}_3 + (B_1) \ddot{\phi}_4 \tag{68}$$

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \dot{\phi}_3} &= \frac{1}{2} [A_1(2\dot{\phi}_3 + 2\dot{\phi}_2) + B_1(2\dot{\phi}_3 + 2\dot{\phi}_1 - 2\dot{\phi}_2 - 2\dot{\phi}_4) + C_1(2\dot{\phi}_3)] \\
&= (B_1) \dot{\phi}_1 + (A_1 - B_1) \dot{\phi}_2 + (A_1 + B_1 + C_1) \dot{\phi}_3 - (B_1) \dot{\phi}_4
\end{aligned}$$

$$\therefore \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\phi}_3} \right) = (B_1) \ddot{\phi}_1 + (A_1 - B_1) \ddot{\phi}_2 + (A_1 + B_1 + C_1) \ddot{\phi}_3 - (B_1) \ddot{\phi}_4 \tag{69}$$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_4} &= \frac{1}{2} [A_1(2\dot{\varphi}_4 + 2\dot{\varphi}_1) + B_1(2\dot{\varphi}_4 - 2\dot{\varphi}_1 + 2\dot{\varphi}_2 - 2\dot{\varphi}_3) + C_1(2\dot{\varphi}_4)] \\ &= (A_1 - B_1) \dot{\varphi}_1 + (B_1) \dot{\varphi}_2 - (B_1) \dot{\varphi}_3 + (A_1 + B_1 + C_1) \dot{\varphi}_4\end{aligned}$$

$$\therefore \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\varphi}_4} \right) = (A_1 - B_1) \ddot{\varphi}_1 + (B_1) \ddot{\varphi}_2 - (B_1) \ddot{\varphi}_3 + (A_1 + B_1 + C_1) \ddot{\varphi}_4 \quad (70)$$

$$\frac{\partial \mathcal{L}}{\partial \varphi_1} = \frac{\partial \mathcal{L}}{\partial \varphi_2} = \frac{\partial \mathcal{L}}{\partial \varphi_3} = \frac{\partial \mathcal{L}}{\partial \varphi_4} = 0 \quad (71)$$

4.4.2 Perdas de Energia por Fricção Viscosa

Neste caso, já que φ_4 faz parte das coordenadas, os termos de fricção linear são idênticos.

$$D_v = \frac{\mu}{2} (\dot{\varphi}_1^2 + \dot{\varphi}_2^2 + \dot{\varphi}_3^2 + \dot{\varphi}_4^2) \quad (72)$$

$$\frac{\partial D_v}{\partial \dot{\varphi}_1} = \mu \dot{\varphi}_1 \quad (73)$$

$$\frac{\partial D_v}{\partial \dot{\varphi}_2} = \mu \dot{\varphi}_2 \quad (74)$$

$$\frac{\partial D_v}{\partial \dot{\varphi}_3} = \mu \dot{\varphi}_3 \quad (75)$$

$$\frac{\partial D_v}{\partial \dot{\varphi}_4} = \mu \dot{\varphi}_4 \quad (76)$$

$$F_v(\dot{\varphi}) = \mu \begin{bmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \\ \dot{\varphi}_4 \end{bmatrix} \quad (77)$$

4.4.3 Equação de Movimento

Reescrevendo a equação de Newton-Euler na forma vetorial-matricial, é revelada a equação 78

$$M_1 \begin{bmatrix} \ddot{\varphi}_1 \\ \ddot{\varphi}_2 \\ \ddot{\varphi}_3 \\ \ddot{\varphi}_4 \end{bmatrix} + F_v(\dot{\varphi}) = Q(\tau) \quad (78)$$

Onde:

$$M_1 = \begin{bmatrix} A_1 + B_1 + C_1 & -B_1 & B_1 & A_1 - B_1 \\ -B_1 & A_1 + B_1 + C_1 & A_1 - B_1 & B_1 \\ B_1 & A_1 - B_1 & A_1 + B_1 + C_1 & -B_1 \\ A_1 - B_1 & B_1 & -B_1 & A_1 + B_1 + C_1 \end{bmatrix} \quad (79)$$

$$Q(\tau) = [\tau_1 \quad \tau_2 \quad \tau_3 \quad \tau_4]^T \quad (80)$$

Portanto, isolando as acelerações angulares de juntas, o modelo dinâmico aproximado é descrito pela equação 81.

$$\begin{bmatrix} \ddot{\varphi}_1 \\ \ddot{\varphi}_2 \\ \ddot{\varphi}_3 \\ \ddot{\varphi}_4 \end{bmatrix} = M_1^{-1}Q(\tau) - M_1^{-1}F_v(\dot{\varphi}) \quad (81)$$

4.5 Divergências entre os Modelos Exato e Aproximado

Para manter ambas as modelagens ancoradas no mesmo contexto, é importante entender suas convergências e divergências. O equacionamento aproximado, já que não apresenta forças provenientes de restrições não holonômicas, é atualmente o mais utilizado na literatura, pela facilidade da implementação. Isso pode ser visto em (AZIZI; RASTEGARPANAH; STOLKIN, 2021), (HAN; ZHU, 2019), (HENDZEL; RYKAŁA, 2017), (JIA; SONG; XU, 2013), (TĂȚAR et al., 2014) e (TSAI; TAI; LEE, 2011). Observando os sistemas de equações 16 e 22, a grande diferença ocorre pelo tratamento dos estados angulares das rodas como independentes, configurando uma violação da restrição holonômica descrita na equação 27, causada pela inversão aproximada da cinemática por mínimos quadrados. Portanto, o modelo dinâmico aproximado apresenta um número superior de GDL e não é equivalente ao modelo exato.

Entretanto, se a trajetória do robô respeitar artificialmente a equação 27, ambos os sistemas convergem na mesma solução. Mesmo que diferentes, os modelos se comportam similarmente a uma determinada classe de trajetórias no espaço de juntas. Chamaremos esta classe de “trajetórias holonômicas” por respeitarem a restrição holonômica de soma

das velocidades angulares, assim anulando as reações de restrições não holonômicas no equacionamento exato. Da mesma forma, no espaço de trabalho cartesiano do robô, a imagem desta classe holonômica é definida pela disjunção entre a translação no plano xy e a rotação no eixo z . Outrossim, se a plataforma realiza somente trajetórias que não modificam sua orientação de forma contínua, os dois modelos são matematicamente equivalentes. Isto pode ser verificado se o ângulo ψ for constante, tornando as restrições cinemáticas representadas pelas equações 24 e 25 integráveis.

Uma forma de implementar estas trajetórias em malha aberta, com torques constantes de entrada, é pela aplicação da restrição de torque 82, derivada da equação 27.

$$\begin{aligned}\frac{d}{dt}\dot{\phi}_4 &= \frac{d}{dt}(\dot{\phi}_1 + \dot{\phi}_2 - \dot{\phi}_3) \\ \ddot{\phi}_4 &= \ddot{\phi}_1 + \ddot{\phi}_2 - \ddot{\phi}_3\end{aligned}$$

$$\therefore \tau_4 = \tau_1 + \tau_2 - \tau_3 \quad (82)$$

4.6 Simulação Numérica

Para realizar testes numéricos no sistema, são primeiramente construídos os diagramas de blocos, que permitem visualizar o fluxo de sinais e estados em regime contínuo. Os DBs dos modelos exato e aproximado podem ser observados nas Figuras 20 e 21, respectivamente.

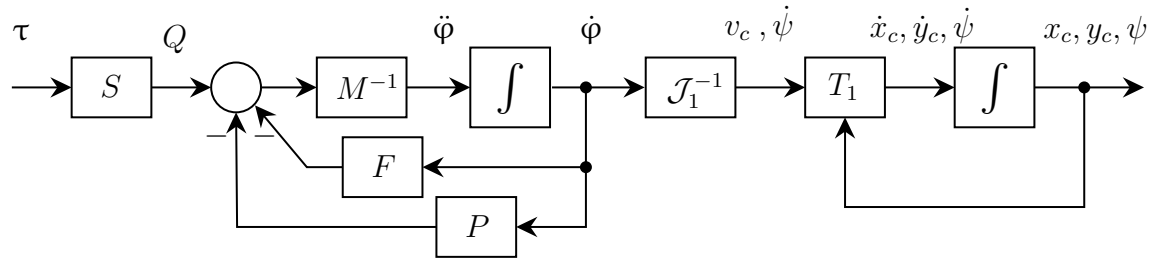


Figura 20: Diagrama de blocos do sistema exato

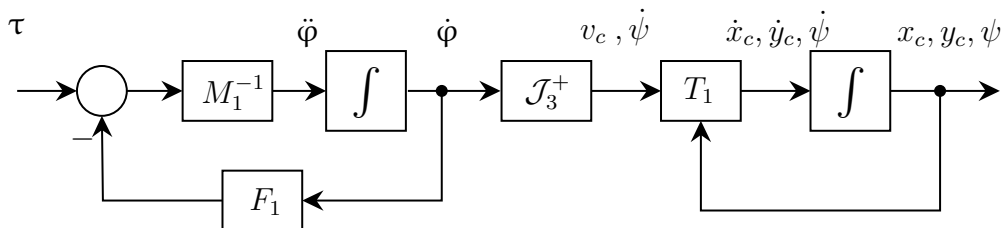


Figura 21: Diagrama de blocos do sistema aproximado

4.6.1 Adição de Distúrbios

Visando medir as reações dos modelos, não somente aos torques de junta, mas às possíveis forças e torques de interação com o paciente, foi decidido adicionar aos DBs os distúrbios f_d e τ_d .

$$f_d = \begin{bmatrix} m_s & 0 \\ 0 & m_s \end{bmatrix} \begin{bmatrix} \dot{v}_{c_x} \\ \dot{v}_{c_y} \end{bmatrix} \quad (83)$$

$$\tau_d = J_c \ddot{\psi} \quad (84)$$

Assim, é necessário criar uma matriz que transforma os distúrbios em ações sobre as acelerações angulares, utilizando a cinemática inversa. Para o sistema exato, a expressão 11 deve ser diferenciada em relação ao tempo.

$$\begin{bmatrix} \ddot{\phi}_1 \\ \ddot{\phi}_2 \\ \ddot{\phi}_3 \end{bmatrix} = \frac{d}{dt} \left\{ \mathcal{J}_1 \begin{bmatrix} v_{c_x} \\ v_{c_y} \\ \dot{\psi} \end{bmatrix} \right\} = \mathcal{J}_1 \begin{bmatrix} \dot{v}_{c_x} \\ \dot{v}_{c_y} \\ \ddot{\psi} \end{bmatrix} \quad (85)$$

Ao juntar as equações 83, 84 e 85, é obtida a expressão matricial 86.

$$\ddot{\phi}_d = \mathcal{D} \begin{bmatrix} f_{c_x} \\ f_{c_y} \\ \tau_d \end{bmatrix} \quad (86)$$

Onde:

$$\mathcal{D} = \mathcal{J}_1 \begin{bmatrix} \frac{1}{m_s} & 0 & 0 \\ 0 & \frac{1}{m_s} & 0 \\ 0 & 0 & \frac{1}{J_c} \end{bmatrix} \quad (87)$$

Traduzindo isto no diagrama de blocos, a 22 é obtida.

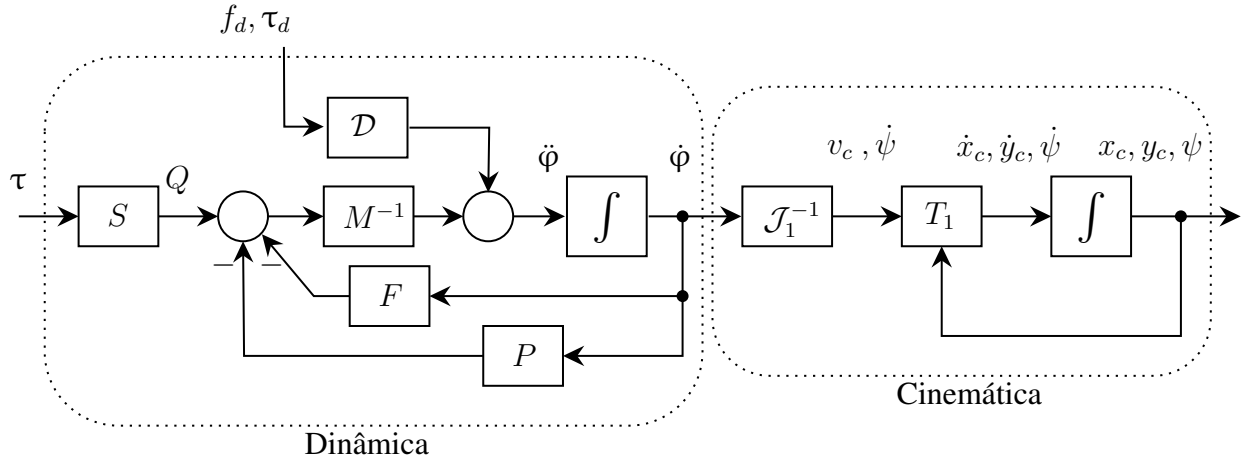


Figura 22: Diagrama de blocos do sistema exato com distúrbio

Similarmente, utilizando a matriz de cinemática inversa do modelo aproximado (equação 20) e as equações 83 e 84, obtém-se:

$$\mathcal{D}_1 = \mathcal{J}_3 \begin{bmatrix} \frac{1}{m_s} & 0 & 0 \\ 0 & \frac{1}{m_s} & 0 \\ 0 & 0 & \frac{1}{J_c} \end{bmatrix} \quad (88)$$

Portanto, o DB do sistema aproximado com distúrbio pode ser observado na Figura 23.

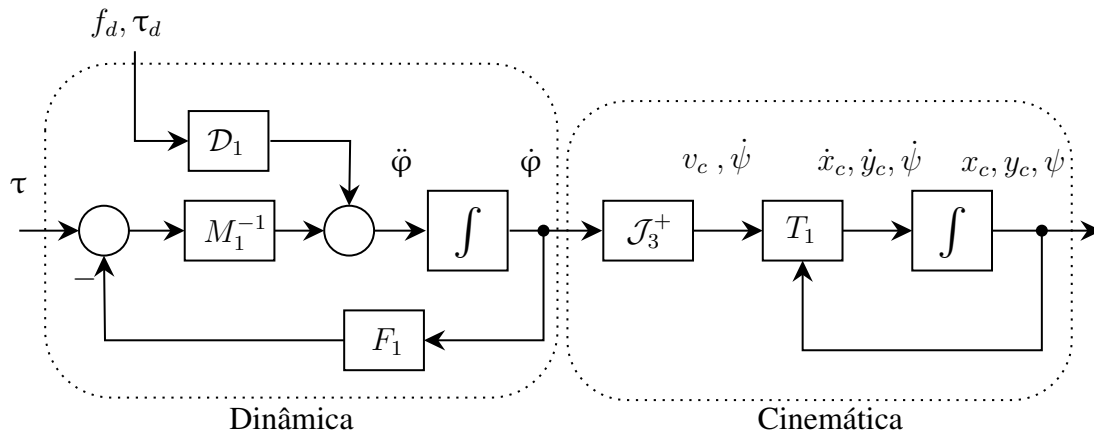


Figura 23: Diagrama de blocos do sistema aproximado com distúrbio

4.6.2 Resolução Numérica

Tendo completado a adição dos distúrbios, os sistemas foram implementados em código *Python*. Para realizar esta tarefa, foi primeiramente criado o vetor de estado, contendo as saídas dos integradores ($\dot{\phi}$, x_c , y_c e ψ). Em seguida, foi construída uma função que implementa as equações diferenciais da dinâmica e da cinemática, retornando

a derivada do vetor de estado ($\ddot{\phi}$, $\dot{x}_c$, $\dot{y}_c$ e $\dot{\psi}$). Então, a função *Scipy.integrate.odeint()* (PETZOLD, 1983), em conjunto de um laço para iterar os vetores de entrada, integra as equações diferenciais, fechando a malha de simulação. Finalmente, os resultados são plotados em gráficos através da biblioteca *Matplotlib*, ou exportados para arquivamento ou utilização em outro programa.

4.7 Modelagem Utilizando Gazebo e ROS

Para realizar a simulação no Gazebo, foi necessário converter o modelo CAD (Figura 17) em malhas visuais e de colisão compatíveis com o programa. Para isto, o arquivo foi separado em três malhas principais: (i) estrutura principal, (ii) roda *Mecanum* e (iii) roda *Mecanum* espelhada. Então, os arquivos foram exportados e convertidos através do software *Blender* para o formato compatível (.dae).

Para a modelagem mecânica, diversos arquivos URDF foram desenvolvidos utilizando macros *Xacro* para repetir sistematicamente as construções simétricas de elos e juntas das rodas, com seus parâmetros cinemáticos e dinâmicos (Figura 24). Outros arquivos de descrição foram criados, incluindo os *plugins* necessários para realizar o controle de juntas e de movimentação do robô. Em função da alta complexidade da roda *Mecanum*, com diversos roletes e juntas não ortogonais, a simulação completa pelo modelo URDF no Gazebo seria muito custosa para processamento em tempo real. Portanto, foi escolhida a opção híbrida de modelar as rodas como cilindros e implementar diretamente as equações de movimento, para que a simulação *on-line* se comporte como o sistema equacionado, em adição à dinâmica tridimensional do motor de física.

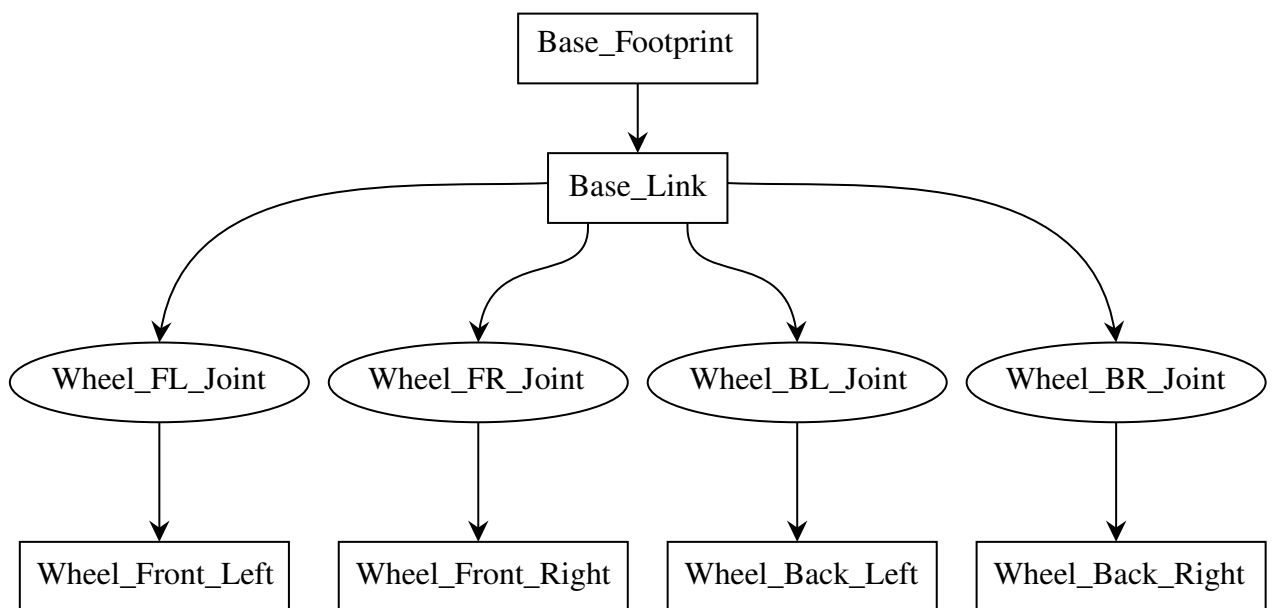


Figura 24: Estrutura URDF da plataforma

Posteriormente, a estrutura de nodos foi criada para comportar a simulação dinâmica (Figura 25). Nesse sentido, as mensagens de juntas e de movimentação do centroide são publicadas separadamente, possibilitando o funcionamento síncrono ou assíncrono da plataforma real e da simulação, para fins de comparação e validação de modelos e controles realizados na prática.

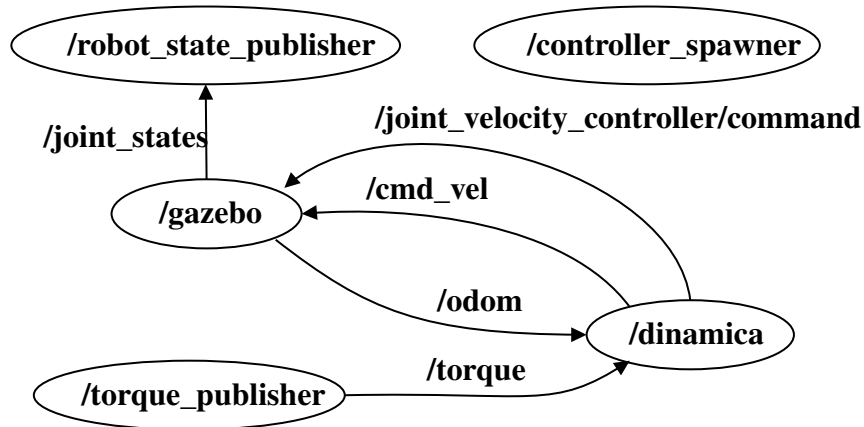


Figura 25: Diagrama de nós da simulação dinâmica

Alternativamente, uma estrutura foi criada somente com entrada de velocidades, para possibilitar os testes de controle puramente cinemático (Figura 26).

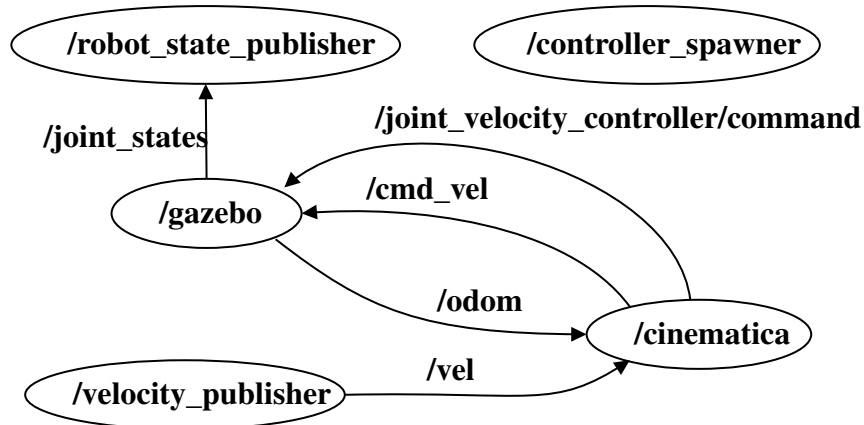


Figura 26: Diagrama de nós da simulação cinemática

4.8 Proposta de Controle

Após completar a modelagem e encaminhar algumas simulações numéricas, é desenvolvida uma proposta de controle cinemático no espaço de estados.

4.8.1 Modelo Cinemático no Espaço de Estados

Para iniciar o equacionamento do controle, é necessário reescrever o sistema no formato característico do espaço de estados. Neste caso é utilizado o modelo cinemático

exato, por resultar em menos variáveis e respeitar a condição holonômica, evitando demasiada derrapamento dos roletes. Para garantir a invertibilidade da matriz do sistema, dimensões idênticas são escolhidas para as variáveis de entrada e saída.

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \\ \mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}u \end{cases}$$

Onde:

$$\mathbf{x} = \begin{bmatrix} x_c \\ y_c \\ \psi \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} x_c \\ y_c \\ \psi \end{bmatrix}, \quad u = \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} T_1 \mathcal{J}_1^{-1} \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{I}_{3 \times 3} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} \mathbf{0} \end{bmatrix}$$

4.8.2 Controle Cinemático por Realimentação de Estado

Resolvendo utilizar a arquitetura de controle por realimentação de estado (BRUNTON; KUTZ, 2022), foi elaborado o diagrama de blocos representado na Figura 27.

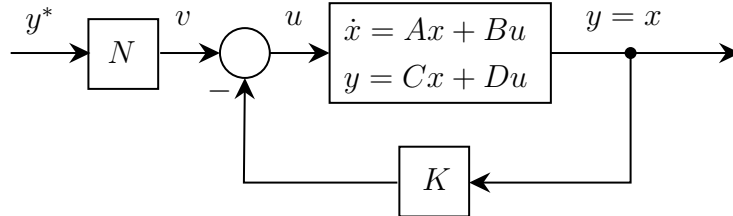


Figura 27: Sistema de Controle por Realimentação de Estado

O primeiro passo a ser realizado é a alocação dos polos do sistema através da ação de controle retroativa. Assim, a lei de controle que estabiliza o sistema é:

$$u = -Kx$$

Onde $K_{3 \times 3}$ é a matriz de ganhos da realimentação de estado.

$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}$$

Para isso, são definidos os três polos desejados do sistema: dois posicionais a_p iguais e um rotacional a_r .

$$\lambda_d(s) = (s - a_p)^2 + (s - a_r) \quad (89)$$

Pela simplicidade do modelo, pode-se realizar a atribuição dos autovalores de polos referentes à equação 89, conforme a expressão 90, adaptada de (BARSAYAN; PURWAR, 2010).

$$K = -(CB)^{-1}CA + (CB)^{-1}J_pC \quad (90)$$

Onde J_p é a matriz diagonal de autovalores desejados.

$$J_p = \begin{bmatrix} -a_p & 0 & 0 \\ 0 & -a_p & 0 \\ 0 & 0 & -a_r \end{bmatrix}$$

Já que a matriz A é nula, em função da dinâmica de integração, e a matriz C é a identidade, os ganhos de realimentação de estado são obtidos em 91.

$$K = B^{-1}J_p$$

$$\therefore K = \begin{bmatrix} \frac{-a_p(c_\psi + s_\psi)}{R} & \frac{a_p(c_\psi - s_\psi)}{R} & \frac{a_r(l_1 + l_2)}{R} \\ \frac{a_p(-c_\psi + s_\psi)}{R} & \frac{-a_p(c_\psi + s_\psi)}{R} & \frac{-a_r(l_1 + l_2)}{R} \\ \frac{a_p(-c_\psi + s_\psi)}{R} & \frac{a_p(c_\psi + s_\psi)}{R} & \frac{a_r(l_1 + l_2)}{R} \end{bmatrix} \quad (91)$$

Observando o equacionamento e as dimensões dos vetores e matrizes envolvidos no sistema, podemos concluir que estamos lidando com um sistema de controle de múltiplas entradas e múltiplas saídas (MIMO, do inglês Multiple Input Multiple Output). Em um sistema MIMO, tanto o número de variáveis de entrada quanto o número de variáveis de saída são maiores que um, implicando em uma interação mais complexa entre os estados do sistema e os sinais de controle.

Nesse tipo de sistema, as entradas (vetor de controle) afetam diretamente múltiplas saídas (vetor de estados ou de variáveis controladas), tornando o projeto de controle mais desafiador, pois as variáveis de entrada não influenciam apenas uma saída isoladamente, mas sim um conjunto de saídas interdependentes. Isso requer o uso de técnicas avançadas para garantir que o comportamento desejado seja alcançado em todo o sistema de forma coordenada, evitando a introdução de instabilidades ou a degradação do desempenho em qualquer das saídas.

Assim, para realizar o seguimento de referência, é necessário calcular o valor da matriz de entrada N , que transforma o regulador desenvolvido em um sistema de controle capaz de rastrear os sinais de referência de múltiplos estados. Assim, já que o sistema é inversível (a matriz 92 não é singular), com procedimento similar a (BARSAYAN; PURWAR, 2010), a lei de controle para seguimento de referência é obtida da seguinte

forma:

$$\mathcal{H} = \left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array} \right] \quad (92)$$

$$u = Ny^* - Ky$$

$$N = K$$

$$\therefore u = K(y^* - y) \quad (93)$$

Assim, a equação 93 permite o controle de posição do sistema, tendo como saída as velocidades angulares de cada junta. Para obter uma realização prática deste sistema, é necessário elaborar controladores de velocidade independentes para cada roda. Já que o conjunto de ganhos sempre define uma matriz A de com autovalores negativos (Hurwitz), a estabilidade do sistema é garantida, desde que não haja grande variação de parâmetros (OGATA, 2020).

4.9 Plataforma de Baixo Custo para Testes

Muitos projetos de sistemas robóticos enfrentam limitações significativas quando se trata da transição da simulação para a implementação prática. Isso ocorre principalmente devido ao alto custo de componentes como sensores, atuadores e estruturas mecânicas, que restringem o progresso da pesquisa. Quando os sistemas ficam restritos ao ambiente de simulação, perde-se a oportunidade de lidar com dinâmicas não modeladas, desafios de implementação e integração que surgem apenas na prática. Para superar essas barreiras e validar os modelos desenvolvidos, além de avaliar o desempenho prático dos controladores propostos, foi elaborado um protótipo miniaturizado de uma plataforma omnidirecional equipada com rodas Mecanum.

Este protótipo foi desenvolvido em uma escala reduzida de 1:3 em relação às dimensões da plataforma original, utilizando componentes mecânicos e eletrônicos populares para reduzir custos. A Tabela 3 detalha os principais componentes utilizados, destacando itens como o microcontrolador ESP32, sensores inerciais, motores de corrente contínua e rodas Mecanum de 90mm. Embora o protótipo seja menor, ele permite a implementação de controladores de juntas independentes, além de controle de velocidade e torque, e integração total com o sistema ROS. O diagrama ilustrado na Figura 28 fornece uma visão geral do sistema, destacando a arquitetura distribuída do software entre o cliente, embarcado no robô, e o servidor, rodando em um computador externo. Esta abordagem permite a operação remota via comunicação sem fio (UDP) e o uso de baterias recarregáveis,

proporcionando flexibilidade e autonomia ao sistema.

No protótipo, o cliente ROS embarcado no microcontrolador ESP32 é responsável por coletar dados dos sensores e executar loops de controle em nível médio, como rastreamento e estimação de posição, operando a uma frequência de 50Hz. Já os controladores de junta, baseados no microcontrolador ATmega328p, lidam com o controle de velocidade, torque e odometria, com frequência de 1000Hz, garantindo precisão na execução dos movimentos. O protocolo de comunicação entre os diferentes módulos está resumido na Tabela 4, com o ESP32 se comunicando com os controladores e sensores via I2C.

A arquitetura do protótipo não apenas suporta a operação autônoma do robô, como também facilita a integração de dados para análise posterior. O servidor, rodando em um computador desktop, executa algoritmos de nível médio e alto, como a geração de trajetórias, além de armazenar os dados coletados para validação futura. A integração com o ambiente de simulação Gazebo também permite uma comparação detalhada entre a trajetória simulada e a trajetória real executada pelo robô. Para aprimorar a precisão dos resultados, um sistema de referência visual com câmeras pode ser utilizado, validando a odometria e os modelos de controle implementados.

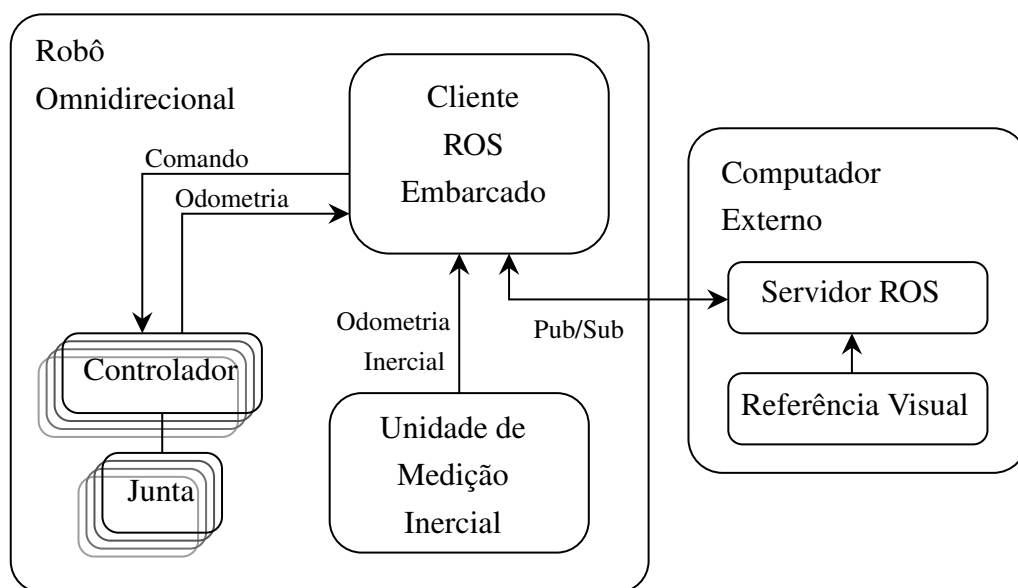


Figura 28: Diagrama dos sistemas do protótipo

Componente	Valor Unitário	Quantidade
Microcontrolador ESP32	R\$60,00	1
Microcontrolador ATmega328p	R\$22,00	4
Unidade Inercial MPU9250	R\$82,00	1
Acionamento Ponte H L298N	R\$10,00	2
Conjunto Motor CC JGB37-520	R\$75,00	4
Conjunto Roda Mecanum 90mm	R\$30,00	4
Bateria de Lítio-íon 18650	R\$40,00	6
Gerenciador de Carga BMS 3S/20A	R\$15,00	1
Placa de Acrílico Cristal 20mm	R\$50,00	1
Valor Total: R\$975,00		

Tabela 3: Componentes utilizados no protótipo

Dispositivo	Sistema	Protocolos de Comunicação
Computador Desktop	Ubuntu 22.04 LTS	TCP/UDP
Microcontrolador ESP32	ESP-IDF	TCP/UDP, I2C
Microcontrolador ATmega328p	ATmega-AVR	I2C
Unidade Inercial MPU9250	-	I2C

Tabela 4: Comunicação dos módulos

O protótipo foi projetado com uma estrutura composta por uma base de acrílico e componentes plásticos impressos em 3D para acomodar a eletrônica. A Figura 29 apresenta o modelo CAD do protótipo em escala, ilustrando a construção modular do sistema. Este protótipo não só oferece uma plataforma viável para a validação prática dos conceitos desenvolvidos, mas também serve como base para futuras expansões e refinamentos no projeto. Com um custo total relativamente baixo (conforme descrito na Tabela 3), o protótipo permite testar e ajustar algoritmos de controle e de movimentação antes da aplicação em uma versão em escala real da plataforma.

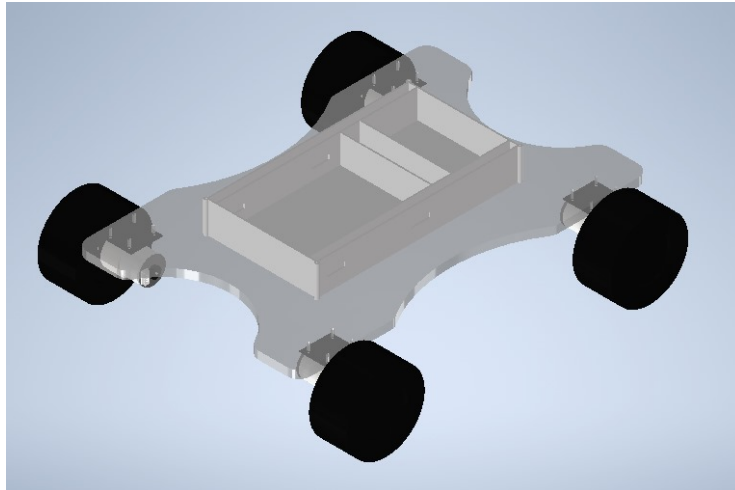


Figura 29: Modelo CAD do protótipo em escala

4.9.1 Software Embarcado

A arquitetura do sistema foi concebida com o objetivo de facilitar sua implementação em ambientes educacionais, como laboratórios acadêmicos, promovendo a integração entre teoria e prática de maneira acessível. Para garantir simplicidade de uso e adaptação, o software foi desenvolvido com foco na transparência, acessibilidade e portabilidade, especialmente em relação a futuras atualizações de hardware. O gerenciamento de versões foi realizado utilizando o Git, que oferece rastreamento eficiente das modificações e assegura que o histórico de desenvolvimento seja mantido de forma clara e organizada, conforme destacado por Loeliger (LOELIGER; MCCULLOUGH, 2012). Isso permite que equipes de desenvolvimento possam colaborar de forma eficaz, evitando conflitos de código e facilitando o processo de revisão.

Adicionalmente, o uso de contêineres Docker foi implementado para garantir replicação fácil do ambiente de desenvolvimento em diferentes plataformas, promovendo uma consistência que elimina a necessidade de configurações complexas em novos dispositivos. Estudos apontam que essa abordagem é especialmente útil em contextos educacionais, onde a diversidade de hardware e software pode se tornar um desafio. Segundo Boettiger (BOETTIGER, 2015), a utilização de Docker não apenas simplifica a configuração de ambientes, como também promove uniformidade entre diferentes plataformas, criando uma experiência de uso homogênea tanto para alunos quanto para profissionais. Isso facilita o onboarding de novos usuários, ao mesmo tempo que mantém a integridade do ambiente de desenvolvimento original.

Essa estratégia de implementação garante que o projeto tenha uma vida útil prolongada, permitindo que futuros alunos e pesquisadores possam facilmente adaptar o sistema a novos requisitos ou tecnologias sem a necessidade de grandes refatorações. Além disso, o uso de ferramentas de automação e versionamento permite que o projeto continue a evoluir de maneira modular e eficiente, promovendo a sustentabilidade do software em

diferentes versões e contextos educacionais. Dessa forma, a arquitetura não só atende às necessidades imediatas, mas também prepara o sistema para expansões futuras, garantindo sua flexibilidade e longevidade no uso acadêmico e profissional.

4.9.2 Modelo de Junta Motriz

Visando realizar o controle das juntas, é necessário modelar a estrutura interna do sistema motriz, que converte energia elétrica em acionamento mecânico rotacional (SICILIANO et al., 2009). Para isso, foi elaborado o esquemático presente na Figura 30.

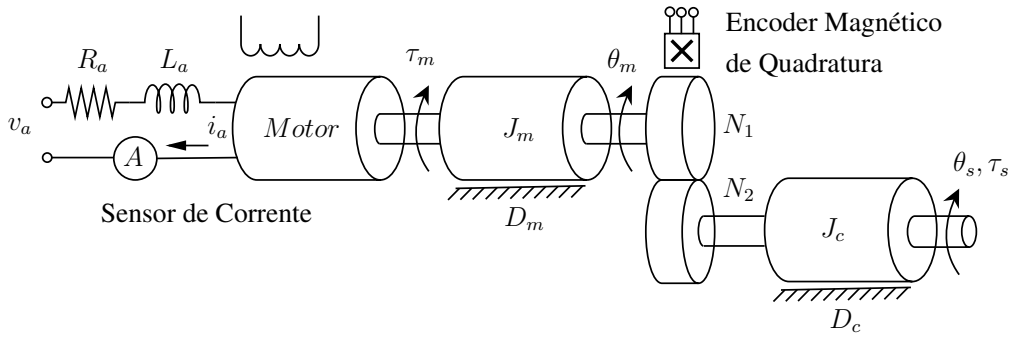


Figura 30: Esquemático da junta com motor CC

Este diagrama descreve o circuito de armadura de um motor de corrente contínua (MCC), junto à caixa de transmissão com redução mecânica e suas cargas dinâmicas associadas. O sistema também inclui um sensor de corrente e um *encoder* de quadratura, que permitem estimar as variáveis de posição, velocidade e torque, necessárias para o controle de realimentação. Iniciando o equacionamento, o sistema mecânico que está após a caixa de redução deve ser referido para o sistema mecânico do motor (NISE, 2020), e por conseguinte resulta em um equivalente mecânico com o coeficiente de atrito D_e e momento de inércia J_e . A partir disso, a planta do sistema é posta em um DB, representado pela Figura 31.

$$N = \frac{N_1}{N_2}$$

$$J_e = J_m + N^2 \cdot J_c$$

$$D_e = D_m + N^2 \cdot D_c$$

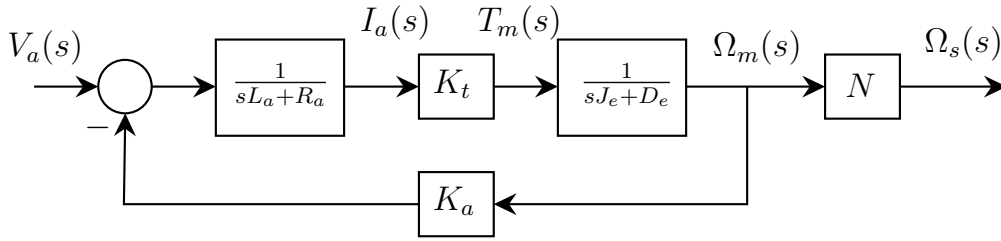


Figura 31: Diagrama de Blocos da Junta Motriz

Do diagrama de blocos, são retiradas as seguintes equações:

$$\begin{aligned}
 I_a(s) &= \frac{1}{sL_a + R_a} (V_a(s) - K_a \Omega_m(s)) \\
 sL_a I_a(s) + R_a I_a(s) &= V_a(s) - K_a \Omega_m(s) \\
 sL_a I_a(s) &= V_a(s) - K_a \Omega_m(s) - R_a I_a(s) \\
 sI_a(s) &= \frac{1}{L_a} V_a(s) - \frac{K_a}{L_a} \Omega_m(s) - \frac{R_a}{L_a} I_a(s) \\
 \dot{i}_a &= \frac{1}{L_a} v_a - \frac{K_a}{L_a} \omega_m - \frac{R_a}{L_a} i_a
 \end{aligned} \tag{94}$$

$$\begin{aligned}
 \Omega_m(s) &= \frac{1}{sJ_e + D_e} K_t I_a(s) \\
 sJ_e \Omega_m(s) + D_e \Omega_m(s) &= K_t I_a(s) \\
 sJ_e \Omega_m(s) &= K_t I_a(s) - D_e \Omega_m(s) \\
 s\Omega_m(s) &= \frac{K_t}{J_e} I_a(s) - \frac{D_e}{J_e} \Omega_m(s) \\
 \dot{\omega}_m &= \frac{K_t}{J_e} i_a - \frac{D_e}{J_e} \omega_m
 \end{aligned} \tag{95}$$

$$\Omega_s(s) = N \Omega_m(s)$$

$$\omega_s = N \omega_m \tag{96}$$

Como próximo passo, as expressões 94, 95 e 96 devem ser traduzidas na forma vetorial-matricial, resultando no modelo de espaço de estados que pode ser utilizado para simular a controlar a junta.

$$\begin{aligned}
\mathbf{x} &= \begin{bmatrix} i_a \\ \omega_m \end{bmatrix}, & \mathbf{y} &= \begin{bmatrix} \tau_s \\ \omega_s \end{bmatrix}, & u &= v_a \\
\dot{\mathbf{x}} &= \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_a}{L_a} \\ \frac{K_t}{J_e} & -\frac{D_e}{J_e} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix} u \\
y &= \begin{bmatrix} \frac{K_t}{N} & 0 \\ 0 & N \end{bmatrix} \mathbf{x} + 0u
\end{aligned} \tag{97}$$

4.9.3 Parametrização da Junta Motriz

Para um controlador poder ser elaborado e sintonizado apropriadamente, é necessário obter todos os parâmetros relevantes, mediante diversos ensaios e procedimentos de teste. Estes processos foram baseados na metodologia de Mór (MÓR, 2017). A primeira etapa foi o levantamento do polo mecânico do sistema, através da resposta ao degrau em ensaio de rotor livre, resultando na obtenção da Figura 32. O degrau de tensão de 10,63V foi aplicado com uma fonte de laboratório, enquanto a velocidade foi medida pelo próprio *encoder* de quadratura da junta. Os dados foram alimentados em um modelo de regressão exponencial de primeira ordem, obtendo os valores de ganho e constante de tempo. Assim, foram retirados o ganho K_a e a constante de tempo mecânica dependente dos coeficientes de atrito e momento de inércia equivalentes D_e e J_e .

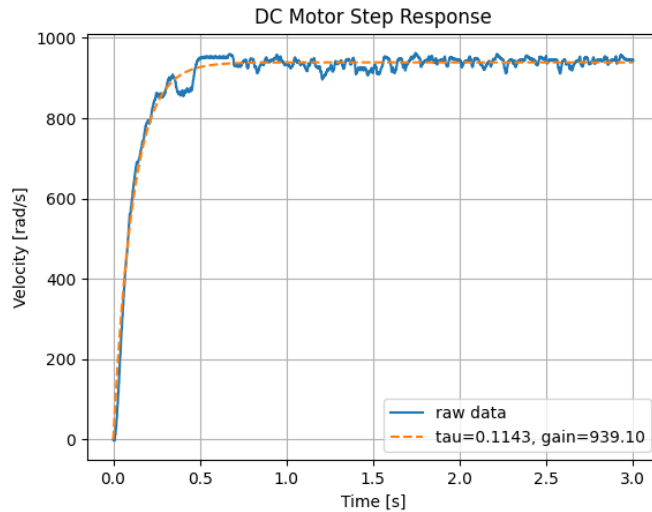


Figura 32: Identificação do polo mecânico da junta

Posteriormente, os coeficientes do polo elétrico foram reconhecidos via ensaios de rotor bloqueado, sendo a resistência de armadura R_a identificada pela relação entre a tensão e a corrente de parada, enquanto a indutância de armadura L_a foi levantada pela constante de tempo do sinal de corrente, obtida com um osciloscópio (Figura 33).

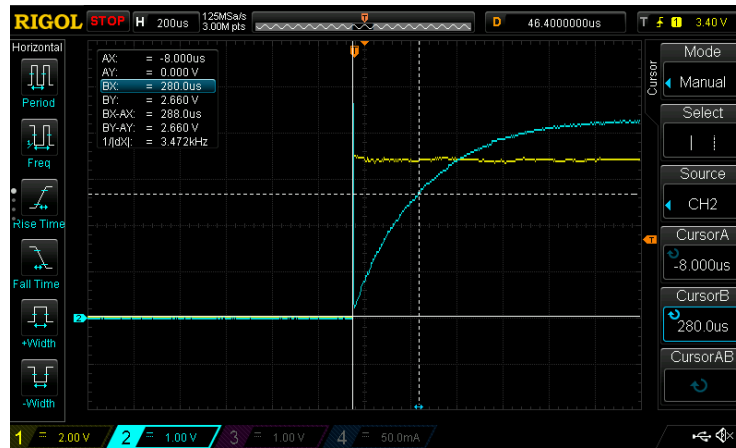


Figura 33: Identificação da indutância de armadura

Por conseguinte, através do levantamento de dados e da manipulação algébrica das equações do sistema, os parâmetros encontrados e suas unidades relevantes podem ser observados na tabela 5.

Parâmetro	Valor	Unidade
R_a	5,2	Ω
L_a	4,44	mH
J_e	$1,02e - 4$	$kg \cdot m^2$
D_e	$8,88e - 4$	$N \cdot m \cdot s \cdot rad^{-1}$
K_t	0,5	$N \cdot m \cdot A^{-1}$
K_a	0,5	$V \cdot s \cdot rad^{-1}$
N	30	—

Tabela 5: Parâmetros da junta motriz

4.9.4 Controlador de Velocidade PID

Atualmente, com o avanço acelerado dos processos de fabricação, o mundo tecnológico tem se deparado com uma crescente complexidade. A complexidade computacional, os desafios de confiabilidade e a execução em tempo real nos sistemas de controle industrial tornam os controladores que utilizam um único controle para todas as juntas de manipuladores caros e difíceis de implementar (FAREH et al., 2016). Assim, o controle descentralizado, independente por junta, foi selecionado para ser implementado nos microcontroladores ATmega. Tendo identificado os parâmetros do sistema, foi decidido implementar um controlador PID, visto que há necessidade de resposta rápida a distúrbios (OGATA, 2020). A função de transferência prática do controlador é descrita pela seguinte equação (FT):

$$C(s) = \frac{K_p + K_d s}{\frac{1}{p_1} s + 1} \times \frac{s + z_1}{s}$$

Pode-se identificar um polo extra p_1 , cujo motivo é tornar a FT própria. Caso contrário, o cálculo da ação de controle no sistema real não é possível (OGATA, 2020). Vistas as características do modelo, foi decidida a aplicação de uma técnica clássica de determinação dos ganhos, a partir da resposta em frequência (NISE, 2020). O primeiro passo é a definição dos requisitos de desempenho. Foi escolhido arbitrariamente um instante de pico menor ou igual a $200ms$ e ultrapassagem percentual menor ou igual a 5%.

$$t_p = \frac{\pi}{\omega_n \sqrt{1 - \zeta^2}} \leq 0.2$$

$$M_p = e^{\frac{-\zeta\pi}{\sqrt{1 - \zeta^2}}} \leq 0.05$$

Para isso, são calculados os parâmetros de amortecimento ζ , frequência natural ω_n e a margem de fase PM desejada.

$$\zeta = 0.69$$

$$\omega_n = 21.7rad/s$$

$$PM = 100 \times \zeta + 15^\circ = 84^\circ$$

Dessa forma, o procedimento de sintonização consiste em encontrar os ganhos para obter uma margem de fase maior ou igual a 84° , em uma frequência de corte de $21.7rad/s$. Para isso, a fase e o ajuste na frequência de corte desejada são:

$$\angle(G_p(21.7j)) = -160^\circ$$

$$\phi = -180 + 160 + 84 = 64^\circ$$

Assim, obtemos os ganhos:

$$K_p = \frac{\cos(64^\circ)}{|G_p(21.7j)|} \approx 394.52$$

$$K_d = \frac{\sin(64^\circ)}{21.7 \times |G_p(21.7j)|} \approx 37.28$$

Posteriormente, é definido um valor pequeno de z_1 , para não influenciar na margem de fase, e um valor de p_1 suficientemente grande para não afetar a dinâmica do sistema.

$$z_1 = 0.01 \times \omega_c = 0.217$$

$$p_1 = 100 \times \omega_c = 2170$$

Portanto, a função de transferência resultante do controlador é:

$$C(s) = \frac{(394.52 + 37.28s)(s + 0.217)}{s(\frac{1}{2170}s + 1)} \quad (98)$$

Este sistema de controle pode ser implementado diretamente em *hardware* analógico, ou discretizado para uso em controladores digitais. Como se trata de um controlador digital, deve-se transformar a FT 98 do domínio contínuo para o discreto, através da aproximação de Tustin (SANTINA; STUBBERUD; HOSTETTER, 2019), utilizando o período T de amostragem de $1ms$.

$$\mathcal{C}(z) \approx \mathcal{C}(s) \Big|_s = \frac{2}{T} \times \frac{z+1}{z-1}$$

$$\therefore \mathcal{C}(z) = \frac{z(39009z - 77599) + 38590}{z(z - 0,959) - 0.041} \quad (99)$$

Dessa forma, é obtida a realização no espaço de estados discreto, que pode ser implementada diretamente nos microcontroladores encarregados do controle de juntas:

$$\mathbf{x}(k+1) = \begin{bmatrix} 0 & 0,041 \\ 1 & 0,959 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 40180 \\ -40180 \end{bmatrix} u(k) \quad (100)$$

$$y(k) = \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{x}(k) + 39009 \times u(k)$$

4.9.5 Parametrização da Plataforma de Teste

Já realizada a parametrização e controle de velocidade da junta motriz, também é necessário identificar os parâmetros do protótipo em escala, espelhando os valores da tabela 2. Dessa forma, os pesos e dimensões equivalentes foram medidos diretamente do protótipo, enquanto o momento de inércia da plataforma foi calculado através do modelo CAD (Figura 34). Além disso, o momento da roda foi calculado previamente, como a inércia equivalente da junta, referenciada na tabela 5.

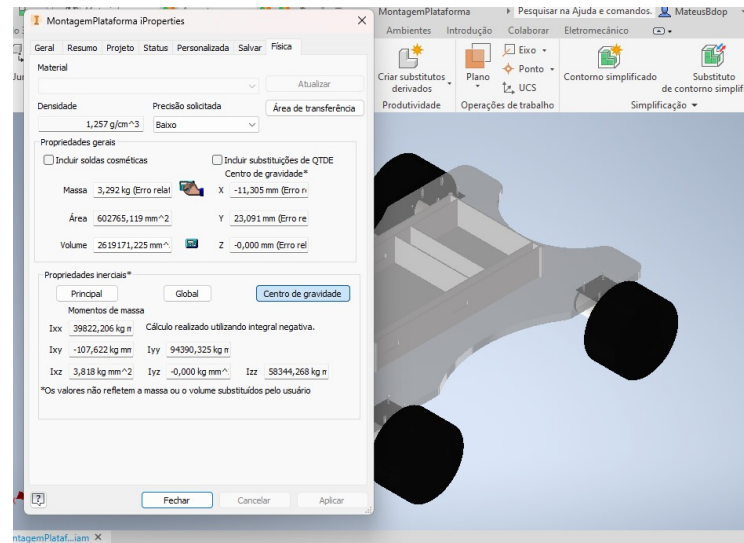


Figura 34: Momento de inércia do protótipo

Assim, todos os parâmetros cinemáticos e dinâmicos do protótipo estão resumidos na tabela 6.

Parâmetro	Valor	Unidade	Descrição
R	0.097	m	raio da roda
l_1	0.32	m	distância lateral
l_2	0.39	m	distância vertical
m_c	3,28	Kg	massa da plataforma
m_k	0,155	Kg	massa individual da roda
J_0	$5,8344e - 4$	$Kg \cdot m^2$	momento de inércia da plataforma
J_1	$1,02e - 4$	$Kg \cdot m^2$	momento de inércia individual da roda
J_2	$1,5312e - 2$	$Kg \cdot m^2$	momento de inércia da roda em relação a C
μ	$8,88e - 4$	—	coeficiente de atrito

Tabela 6: Parâmetros cinemáticos e dinâmicos do protótipo

5 RESULTADOS

Neste capítulo são apresentadas as soluções, discussões e simulações oriundas dos modelos sintetizados anteriormente, começando pela verificação das rotinas básicas de movimentação observadas na revisão da literatura. Em seguida, é realizado um estudo prático sobre as divergências entre os equacionamentos exato e aproximado, mediante testes de entradas constantes e distúrbios de forças e torques aplicados ao corpo do robô. Por fim, são apresentados os resultados de simulação no ROS e Gazebo, além da implementação prática do protótipo.

5.1 Nota Sobre Representação Gráfica dos Dados

Muitos dos resultados e simulações nesta seção estão constituídos na forma de gráficos. Nestes, os traços simbolizam caminhos de translação e setas representam a orientação do robô, amostradas em período constante, selecionado para maior legibilidade. Também, há momentos nas trajetórias onde distúrbios estão agindo, este fato é registrado pela mudança de cor do traço, do tempo inicial ao final da ação do distúrbio, como indicado na legenda.

5.2 Simulação Numérica

Com a simulação numérica implementada no capítulo prévio, o primeiro passo foi a verificação de funcionamento da cinemática, acompanhando as rotinas básicas de movimentação, apresentadas na Figura 5. Feito isto, os gráficos 35 e 36 demonstram que a plataforma sujeita a entradas constantes de junta age como esperado, realizando todos os dez movimentos de translação e rotação idealizados.

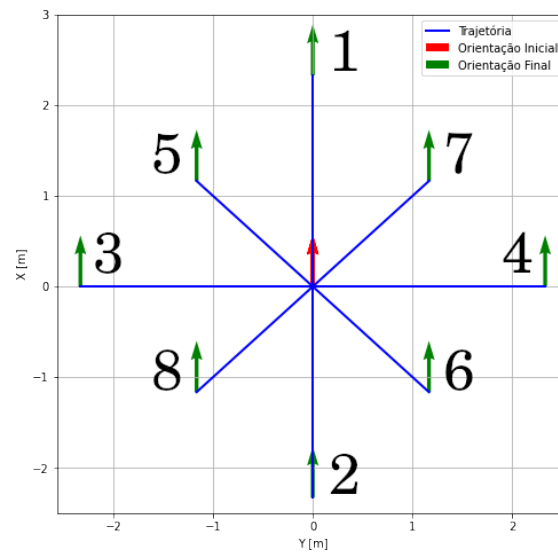


Figura 35: Movimentos básicos de translação

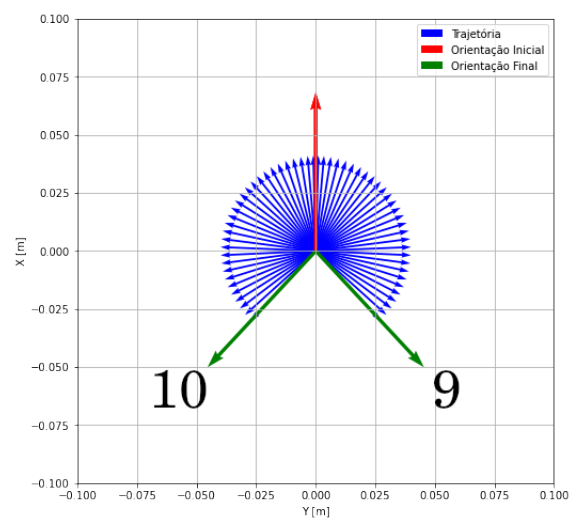


Figura 36: Movimentos básicos de rotação

5.3 Comparação Entre os Modelos Aproximado e Exato

Para verificar as afirmações realizadas durante a modelagem, são examinados os modos de movimentação básicos. Já que estas trajetórias não apresentam variações simultâneas de posição e orientação, são holonômicas no sentido do robô, e, portanto, não devem causar divergência entre os modelos. Ao realizar testes de torque constante, com um tempo de 10s para cada simulação, foram registrados os resultados na tabela 7, onde os desvios-padrão e erros médios de trajetória indicam fortes evidências de que os modelos são similares para esta classe movimento, apresentando somente pequenas divergências entre as trajetórias, causadas por erros de discretização.

Movimento	Posição		Rotação	
	σ	ε	σ	ε
1	6.18e-12	1.71e-12	2.69e-13	2.99e-15
2	6.17e-12	1.73e-12	2.74e-13	-2.24e-15
3	6.21e-12	1.72e-12	2.69e-13	3.42e-15
4	6.22e-12	1.73e-12	2.75e-13	1.53e-15
5	3.79e-12	9.97e-13	2.13e-14	4.35e-16
6	3.91e-12	1.04e-12	2.17e-14	-3.77e-16
7	4.00e-12	1.07e-12	2.26e-14	2.77e-16
8	4.00e-12	1.07e-12	4.03e-14	-4.22e-16
9	1.15e-12	1.56e-12	8.63e-13	1.06e-14
10	8.15e-13	1.12e-12	8.27e-13	-3.69e-14

Tabela 7: Desvios-padrão e erros de trajetória

Diante disso, para testar a reação dos sistemas a distúrbios, outra simulação foi realizada, com o movimento lateral. Em ambos os modelos foram aplicadas forças descontínuas, uma oposta ao movimento (Figura 37) e outra perpendicular ao movimento (Figura 38).

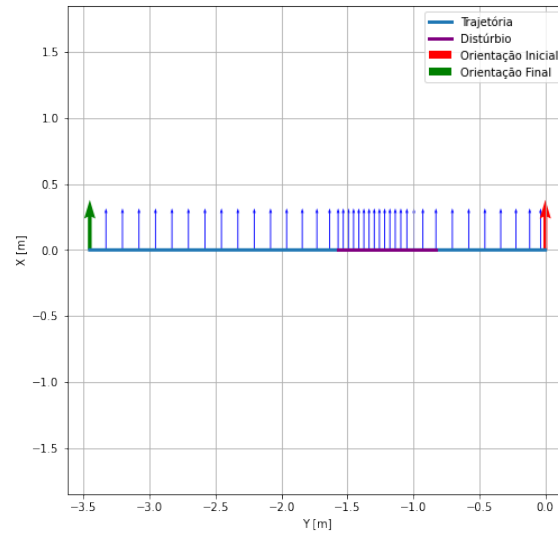


Figura 37: Trajetória holonômica com distúrbio de força oposta

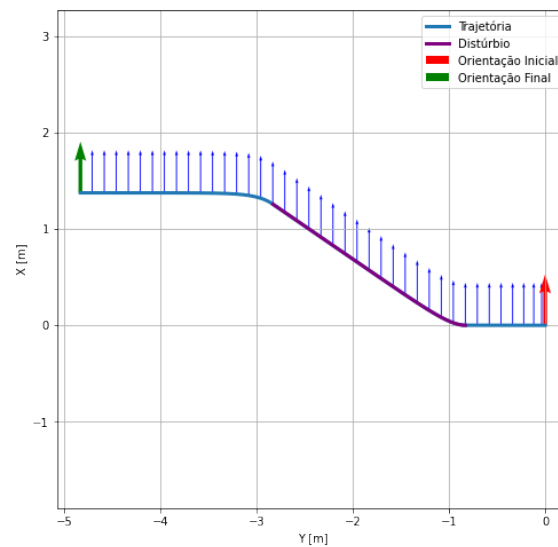


Figura 38: Trajetória com distúrbio de força perpendicular

Deve-se observar na tabela 8 o resultado dos testes, onde novamente as divergências entre trajetórias são extremamente pequenas. Mesmo com distúrbios de força aplicados, não ocorre mudança de orientação da plataforma, validando a similaridade do sistema aproximado em relação ao exato para esta classe de trajetórias.

Distúrbio	Posição		Rotação	
	σ	ε	σ	ε
Oposto	7.96e-15	9.06e-15	2.19e-16	2.00e-16
Perpendicular	2.99e-15	7.83e-15	2.60e-16	2.58e-16

Tabela 8: Desvios-padrão e erros de trajetória com distúrbios

Agora, para explicitar as diferenças entre os modelos, foi introduzido um distúrbio de torque, visando modificar a orientação da plataforma durante o movimento translacional. Pela não integrabilidade das restrições cinemáticas, forças de reação são geradas, resultando em divergências na simulação (Figura 39).

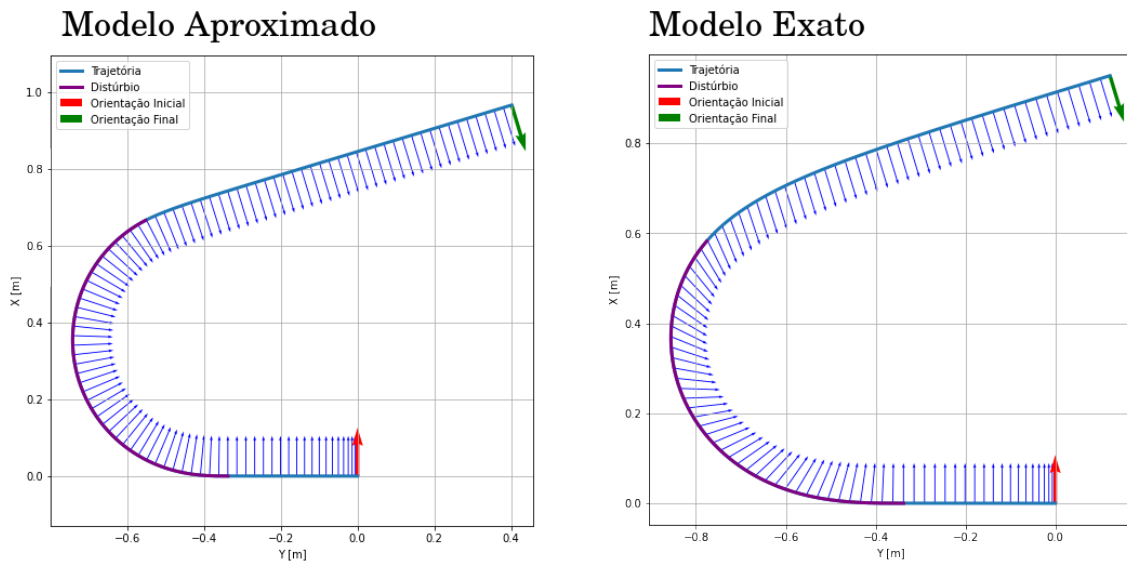


Figura 39: Trajetória de entrada holonômica com distúrbio de torque

Interpretando o resultado do teste na Figura 40, inicialmente a divergência entre os modelos é nula. Porém, aplicando o distúrbio, é causado um erro de posição. Todavia, após a remoção do torque, este erro torna-se estacionário. Dessa forma, a simulação indica novamente uma conexão entre a mudança de posição rotacional e a divergência posicional entre os modelos.

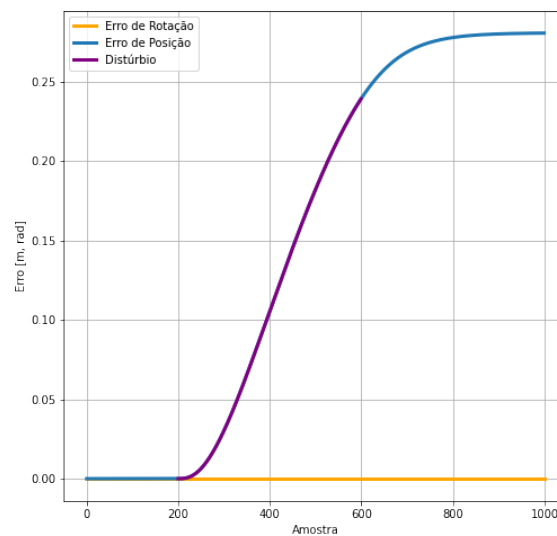


Figura 40: Erro de trajetória com distúrbio de torque

Além do distúrbio, foi implementada uma simulação onde a restrição holonômica 82 não é seguida nos torques de entrada, resultando em forças de reação provenientes de restrições não holonômicas e, conseqüentemente, divergência entre os modelos (Figura 41).

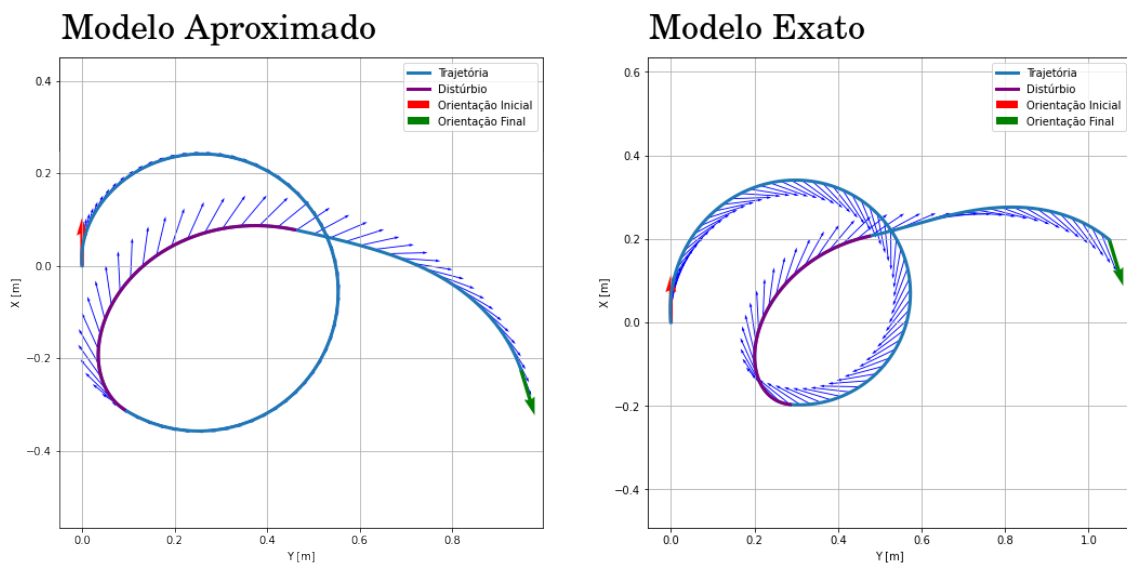


Figura 41: Trajetória não holonômica

Os erros de posição e rotação estão representados graficamente na Figura 42, onde é possível observar que há divergência antes e depois da aplicação do distúrbio, principalmente em relação às coordenadas de posição. Isto ocorre, pois a trajetória no espaço de juntas não respeita a condição 27, resultando em forças de restrição de multiplicadores de Lagrange, conseqüentemente afetando o caminho percorrido pelo modelo exato, mesmo sem interação com forças externas.

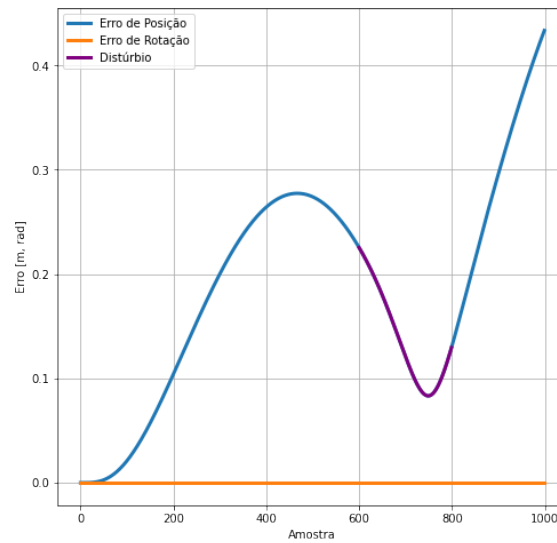


Figura 42: Erro de trajetória não holonômica

Dessa forma, o modelo aproximado não será preciso caso os torques de entrada sejam linearmente independentes, ou caso exista qualquer interação com o paciente que resulte em mudança de orientação da plataforma durante o movimento. Para fins de controle dinâmico, porém, é possível que as margens do controlador sejam suficientes para compensar estas divergências. Isto será investigado em maior profundidade através da realização do controle dinâmico da plataforma.

5.4 Simulação Gazebo/ROS

Com o modelo implementado em ROS, foi possível realizar a simulação tridimensional da plataforma em um mapa configurável no Gazebo (Figuras 43 e 44). Assim, os nodos desenvolvidos foram utilizados para a interação da plataforma com rodas *Mecanum*.

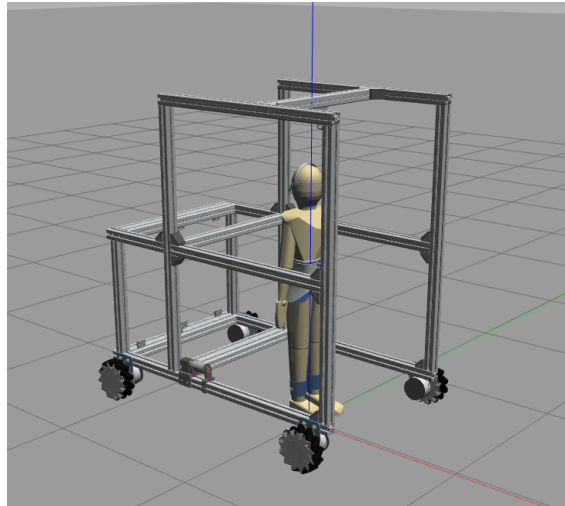


Figura 43: Simulação da plataforma em gazebo e ROS

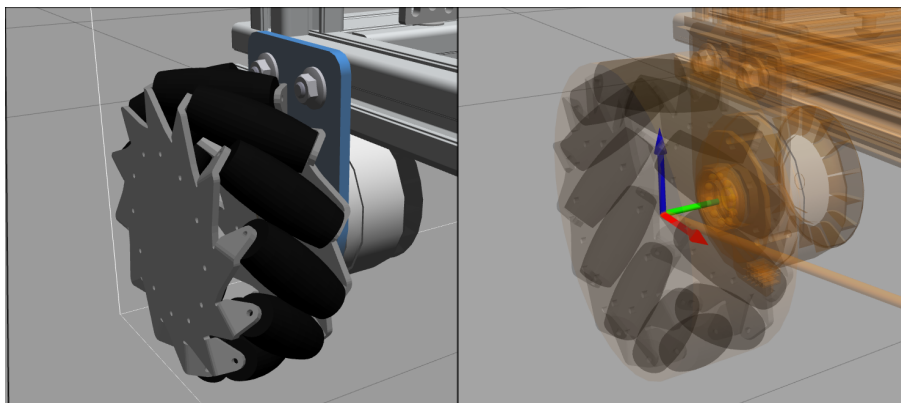


Figura 44: Roda *Mecanum* no Gazebo

Ao comparar a movimentação no Gazebo com as rotinas já executadas na simulação numérica, o módulo RViz foi utilizado para obter os registros temporais de odometria, resultando na Figura 45, onde os movimentos básicos holonômicos da plataforma podem ser identificados no ROS. A diferença de deslocamento entre os modos de movimentação diagonais dos gráficos 35 e 45 ocorre pelo fato de que a simulação em *Python* foi realizada em tempo regrado, enquanto a simulação no Gazebo teve duração arbitrária.

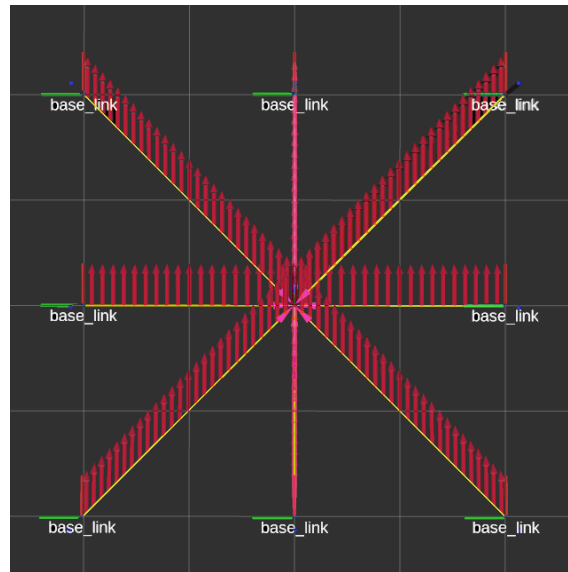


Figura 45: Movimentos básicos da plataforma no RViz

Também, foram executados testes do algoritmo de controle cinemático desenvolvido no capítulo anterior, implementados no ROS. A Figura 46 representa o resultado do controle, levando o veículo ao objetivo com ajuste simultâneo de posição e orientação, resultando em uma trajetória complexa.

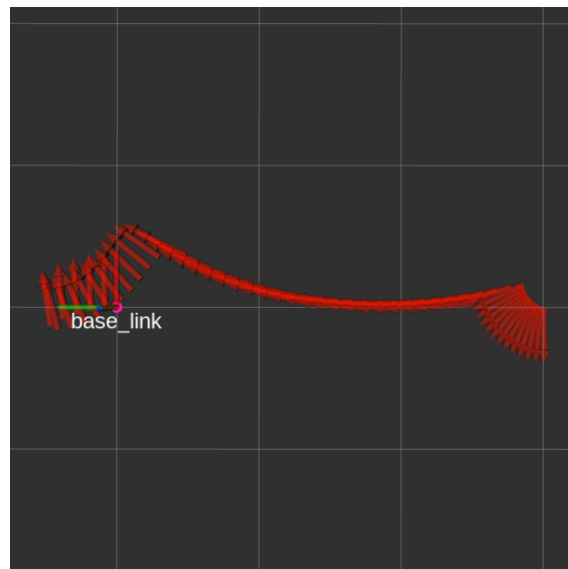


Figura 46: Movimento complexo com controle cinemático da plataforma

Da mesma forma, pode-se observar diretamente no Gazebo a movimentação do robô e a convergência dos sinais de posição e velocidade (Figura 47).

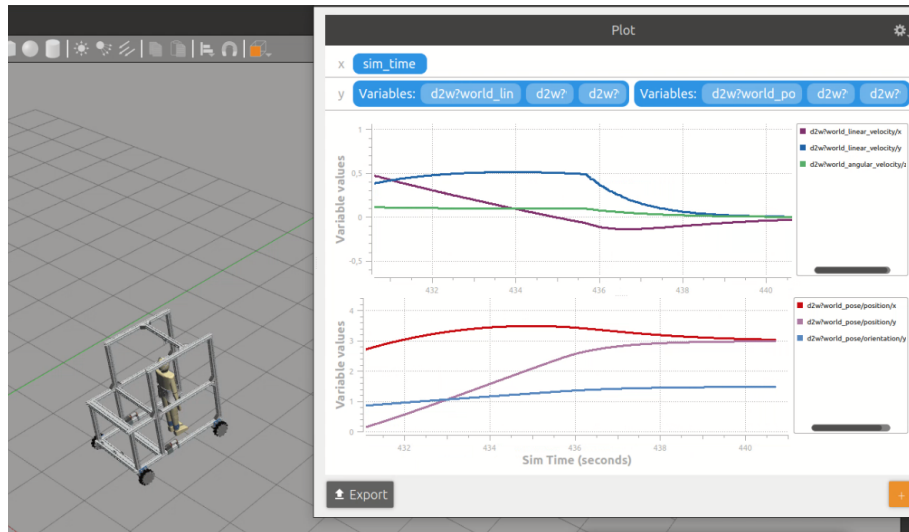


Figura 47: Controle cinemático no Gazebo

Neste caso, uma mensagem de referência é publicada com a posição e orientação desejadas. Assim, um nó controlador lê a odometria e gera a ação de controle apropriada. Isso possibilita a implementação de um sistema de controle de trajetória, pela publicação de uma sequência temporal de parâmetros de referência, permitindo a execução de caminhos arbitrários durante o processo de treinamento de marcha.

5.5 Sistema de Controle Cinemático

Para elaborar na proposta de controle cinemático MIMO, foram realizadas simulações básicas para verificar o desempenho do algoritmo. Uma destas simulações pode ser visualizada na Figura 48, onde o controle ajusta a posição e orientação do robô, seguindo a referência em degrau dos parâmetros de posição e orientação, simultaneamente. Para limitar a ação de controle, foi aplicada uma saturação de $\pm 10 \text{ rad/s}$ nas velocidades angulares das rodas.

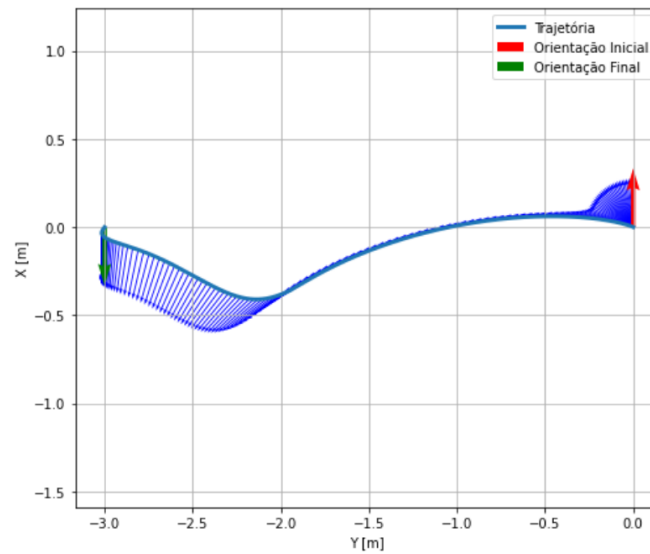


Figura 48: Simulação de controle cinemático

Para melhor analisar o resultado, foram plotados os sinais de erro de referência e as velocidades angulares das rodas (Figura 49). Observa-se que os erros e as velocidades (ação de controle) convergem em zero, atingindo as condições estabelecidas no projeto do controlador, em tempo finito. Entretanto, a saturação da ação de controle causa um comportamento dinâmico diferente do desejado, ao haver sobrepasso no erro de posição em x . Isto pode ser corrigido pela redução da magnitude dos polos desejados, porém com penalidade no tempo de convergência.

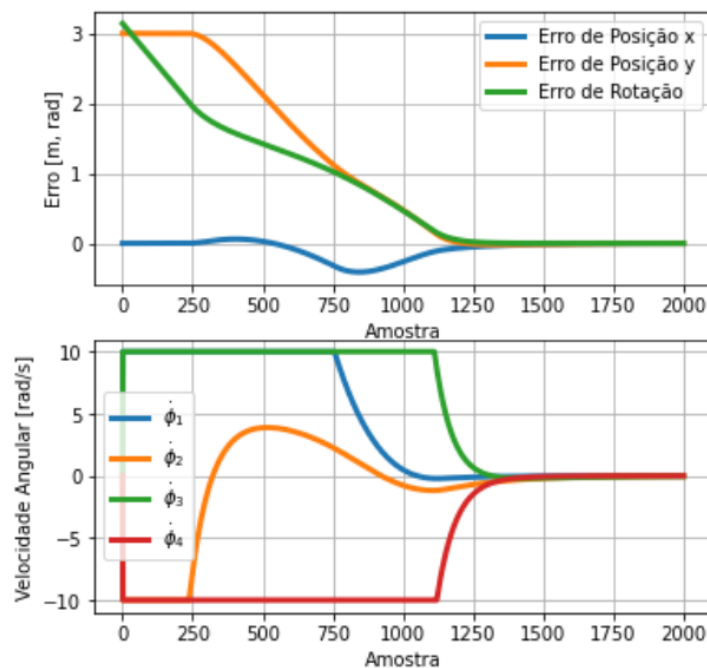


Figura 49: Erro e ação de controle

Adicionalmente, esta simulação foi realizada a fim de exemplo, pois a aplicação posterior será em controle de trajetória, onde o caminho tomado pelo robô pode ser melhor definido.

5.6 Montagem do Protótipo em Escala

Conforme o modelo esquematizado no capítulo anterior, foi realizada a montagem física e a integração eletrônica do protótipo em escala, seguindo o planejamento detalhado. O protótipo é composto por quatro motores independentes, cada um acoplado a uma roda Mecanum, o que confere à plataforma uma capacidade de movimentação omnidirecional. Cada uma das rodas é controlada por um atuador de junta com controlador integrado, que executa laços de realimentação para garantir precisão. Esses controladores recebem os parâmetros de referência e retornam dados de odometria, comunicando-se digitalmente com o microcontrolador ESP32, o qual possui uma instância embarcada do ROS (Robot Operating System).

A comunicação entre o sistema embarcado e o servidor externo, que roda em um sistema Linux, é realizada via rede Wi-Fi, permitindo o envio e recebimento de dados em tempo real. O servidor captura dados relevantes para monitoramento e análise dos movimentos do protótipo. A Figura 50 ilustra o estágio inicial de montagem, onde as conexões elétricas e de comunicação foram estabelecidas utilizando conectores cabeados para assegurar a robustez das interligações.

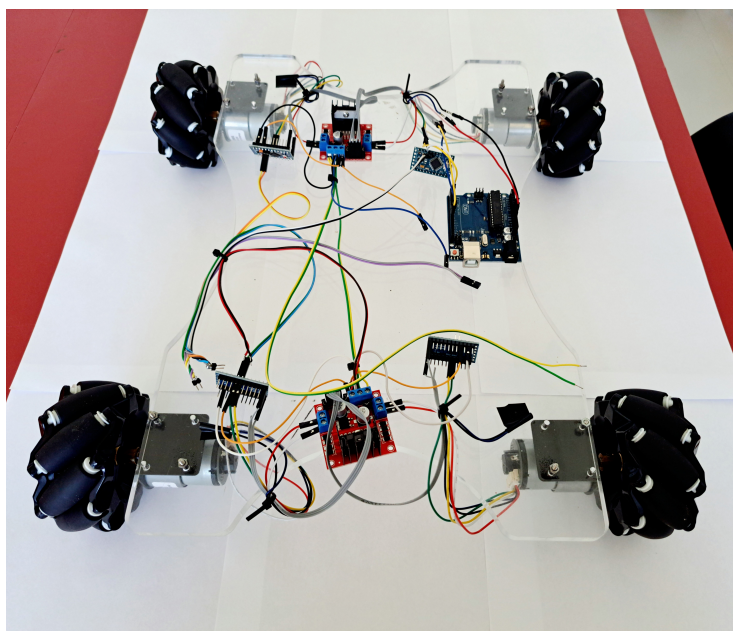


Figura 50: Protótipo inicial da plataforma em escala

Após a verificação básica de funcionamento da plataforma, o processo de prototipagem foi avançado com a criação de uma placa de circuito impresso (PCI). Esta placa

integra todas as conexões entre os módulos eletrônicos, simplificando o cabeamento e melhorando a eficiência do sistema. A Figura 51 mostra os detalhes do circuito desenvolvido, otimizando a organização e minimizando os erros de interconexão entre os componentes.

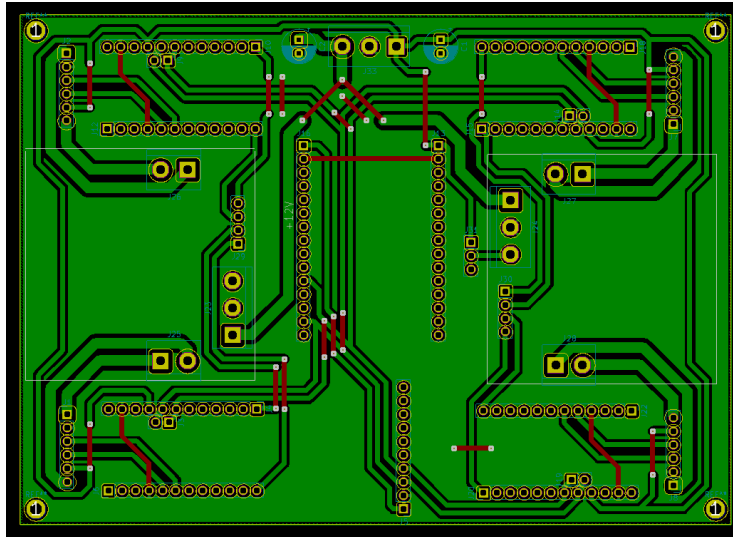


Figura 51: Placa de circuito impresso do protótipo

A partir dessa etapa, aprimoramentos foram feitos na estrutura física do protótipo. Foi impressa uma caixa para abrigar os dispositivos eletrônicos, baterias, conversores de potência e o cabeamento, proporcionando uma solução mais compacta e segura para a movimentação do robô (Figura 52).

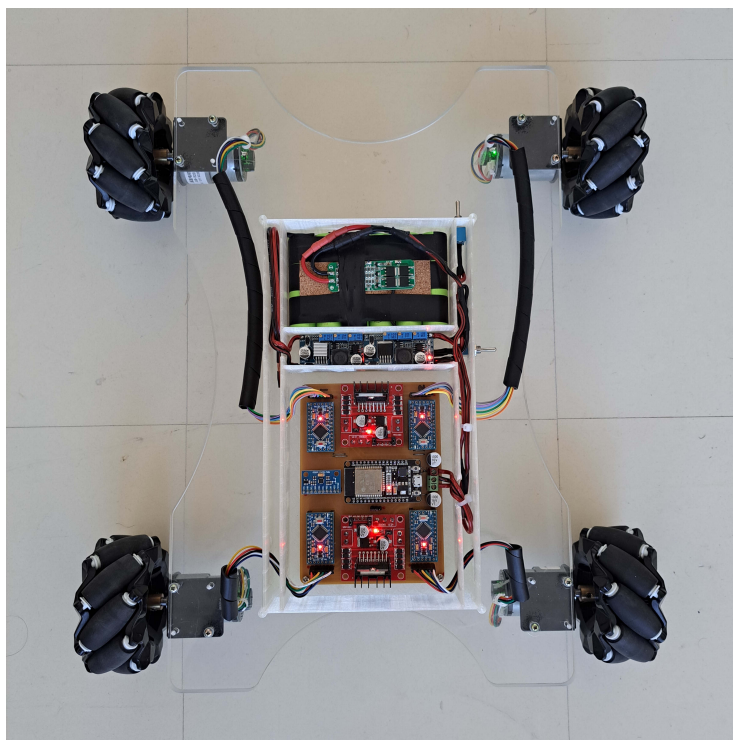


Figura 52: Protótipo completo da plataforma em escala

5.7 Teste do Controle Cinemático

Com a montagem completa e todos os subsistemas implementados, foi realizado um teste de validação da interação entre os componentes de hardware e software, incluindo o controle de trajetória e das juntas independentes. O controle cinemático foi testado em um ambiente controlado para verificar o desempenho da plataforma durante o deslocamento. A Figura 53 ilustra o teste de integração, onde a plataforma executa o controle de movimento conforme os parâmetros predefinidos.

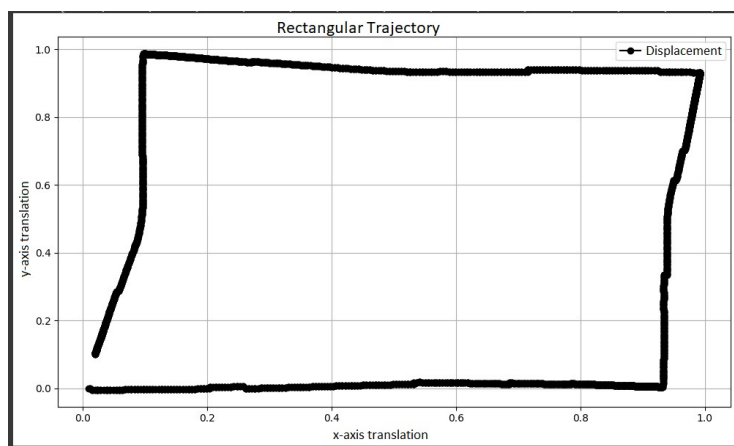


Figura 53: Teste de integração e controle cinemático de deslocamento

Este teste demonstrou a eficiência do sistema em integrar as diversas camadas de controle e sensores, garantindo que a plataforma atenda às especificações de movimentação omnidirecional, além de validar a implementação prática dos controladores propostos.

6 CONCLUSÃO

Este capítulo apresenta as considerações finais deste trabalho, destacando os resultados alcançados até o momento, bem como uma visão geral dos trabalhos futuros planejados ou relacionados a este projeto.

6.1 Considerações Finais

Em síntese, o desenvolvimento de diversos modelos e simulações, a proposta de controle cinemático e a construção de um protótipo de baixo custo para o robô omnidirecional com rodas Mecanum permitiram alcançar os objetivos traçados. Os resultados obtidos e as análises realizadas não apenas forneceram uma compreensão aprofundada da plataforma omnidirecional D2W, mas também enriqueceram o campo de estudo dos robôs móveis multidirecionais em geral, complementando a revisão bibliográfica e elucidando conceitos previamente pouco explorados ou mal definidos.

Uma das principais questões levantadas ao longo deste trabalho foi: o paradigma omnidirecional oferece vantagens significativas de mobilidade no contexto da reabilitação de marcha? Para investigar essa questão, foram discutidos aspectos críticos como a holonomicidade e seus impactos na modelagem dinâmica e no desempenho da plataforma. O robô, no caso estudado, comporta-se como um sistema holonômico em um espaço limitado de coordenadas, o que confere alta manobrabilidade, um atributo essencial no contexto da reabilitação. Tal característica facilita a adaptação às diferentes necessidades de pacientes, permitindo ajustes precisos e rápidos durante as sessões terapêuticas.

Embora o sistema apresente algumas restrições não holonômicas, geralmente descon sideradas na literatura, os experimentos indicam que essas limitações têm impacto mínimo na operação prática. A capacidade do robô de ajustar-se dinamicamente em tempo real, centralizando ativamente a massa na plataforma, reduz o risco de instabilidade ou tombamento, um fator importante para a segurança do paciente durante o uso em ambientes clínicos.

Outro ponto de destaque é a flexibilidade da plataforma ao permitir a execução de trajetórias variadas, como marcha lateral, diagonal e rotação estacionária. Esses movimen-

tos, muitas vezes inviáveis em sistemas de reabilitação tradicionais, oferecem um grau de liberdade ampliado, o que pode contribuir significativamente para melhorar a eficácia dos tratamentos de marcha. Essa mobilidade aprimorada possibilita que o terapeuta personalize o treinamento de acordo com as necessidades específicas do paciente, tornando a reabilitação mais dinâmica e eficiente.

Portanto, os resultados preliminares sugerem que o uso de uma plataforma omnidirecional pode proporcionar uma vantagem considerável na reabilitação de marcha, principalmente no que tange à capacidade de adaptar-se às particularidades de cada paciente e às demandas clínicas. Contudo, estudos futuros ainda precisam validar essa hipótese em maior escala e explorar mais a fundo as interações entre o controle dinâmico e o ambiente de reabilitação.

REFERÊNCIAS

- ADĂSCĂLIȚEI, F.; DOROFTEI, I. Practical applications for mobile robots based on mecanum wheels-a systematic survey. *The Romanian Review Precision Mechanics, Optics and Mechatronics*, v. 40, p. 21–29, 2011.
- AGUIRRE-OLLINGER, G.; YU, H. Omnidirectional platforms for gait training: Admittance-shaping control for enhanced mobility. *Journal of Intelligent & Robotic Systems*, Springer, v. 101, p. 1–17, 2021.
- ALVES, P. O. P. T. *Projeto e síntese de um mecanismo de quatro barras com vista ao desenvolvimento de um dispositivo de reabilitação da marcha humana*. Dissertação (Mestrado) — Universidade do Minho, 2014.
- ARCZEWSKI, K.; BLAJER, W. A unified approach to the modelling of holonomic and nonholonomic mechanical systems. *mathematical modelling of systems*, Taylor & Francis, v. 2, n. 3, p. 157–174, 1996.
- ASSIS-MADEIRA, E. A.; CARVALHO, S. G. de. Paralisia cerebral e fatores de risco ao desenvolvimento motor: uma revisão teórica. *Cadernos de Pós-Graduação em distúrbios do Desenvolvimento*, v. 9, n. 1, 2009.
- AZIZI, M. R.; RASTEGARPANAH, A.; STOLKIN, R. Motion planning and control of an omnidirectional mobile robot in dynamic environments. *Robotics*, Multidisciplinary Digital Publishing Institute, v. 10, n. 1, p. 48, 2021.
- BARSAIYAN, P.; PURWAR, S. Comparison of state feedback controller design methods for mimo systems. In: IEEE. *2010 International Conference on Power, Control and Embedded Systems*. [S.l.], 2010. p. 1–6.
- BAYON, C. et al. Development and evaluation of a novel robotic platform for gait rehabilitation in patients with cerebral palsy: Cpwalker. *Robotics and autonomous systems*, Elsevier, v. 91, p. 101–114, 2017.
- BOETTIGER, C. An introduction to docker for reproducible research. *ACM SIGOPS Operating Systems Review*, ACM New York, NY, USA, v. 49, n. 1, p. 71–79, 2015.
- BRUNTON, S. L.; KUTZ, J. N. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. [S.l.]: Cambridge University Press, 2022.
- CHOTIKUNNAN, P. et al. The design and construction of surrounding control system for the rehabilitative walker using mecanum wheel. In: IEEE. *2017 10th Biomedical Engineering International Conference (BMEiCON)*. [S.l.], 2017. p. 1–4.

COSTA, V. d. S. et al. Efeitos da aplicação do método bobath e dotreino em esteira com suporte parcial de peso na reabilitação da marcha pós-avc: Uma revisão sistemática. *Rev. bras. ciênc. saúde*, p. 161–166, 2014.

DELGADO-OLEAS, G. et al. Diseño y desarrollo de una arquitectura electrónica bioinspirada para el control de sistemas de asistencia a la locomoción. *Revista Iberoamericana de Automática e Informática industrial*, 2023.

DELLON, B.; MATSUOKA, Y. Prosthetics, exoskeletons, and rehabilitation [grand challenges of robotics]. *IEEE robotics & automation magazine*, IEEE, v. 14, n. 1, p. 30–34, 2007.

DICKERSON, S. L.; LAPIN, B. D. Control of an omni-directional robotic vehicle with mecanum wheels. In: IEEE. *NTC'91-National Telesystems Conference Proceedings*. [S.l.], 1991. p. 323–328.

EDEN, J. P. *Modelling, analysis and control of multi-link cable-driven manipulators*. Dissertação (Mestrado) — University of Melbourne, 2018.

FAREH, R. et al. Tracking trajectory in the workspace of rigid manipulators using distributed adaptive control strategy. *Advanced Robotics*, Taylor & Francis, v. 30, n. 6, p. 361–373, 2016.

FONSECA, L. et al. Investigating upper limb movement classification on users with tetraplegia as a possible neuroprosthesis interface. In: *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. [S.l.: s.n.], 2018. p. 5053–5056. ISSN 1558-4615.

GONÇALVE, A. C. B. F.; SIQUEIRA, A. A. G. Estado da arte em reabilitação robótica de membros inferiores de pessoas com ave. *Ensaio e Ciência: Ciências Biológicas, Agrárias e da Saúde*, Universidade Anhanguera, v. 17, n. 5, 2013.

HAN, Y.; ZHU, Q. Robust optimal control of omni-directional mobile robot using model predictive control method. In: IEEE. *2019 Chinese Control Conference (CCC)*. [S.l.], 2019. p. 4679–4684.

HENDZEL, Z.; RYKAŁA, Ł. Modelling of dynamics of a wheeled mobile robot with mecanum wheels with the use of lagrange equations of the second kind. *International Journal of Applied Mechanics and Engineering*, v. 22, n. 1, 2017.

IEC. *IEC 60601-2-33:2010. Medical electrical equipment - Part 2-33: Particular requirements for the basic safety and essential performance of magnetic resonance equipment for medical diagnosis*. 2010. [Online; acessado 18-Julho-2020]. Disponível em: <<https://webstore.iec.ch/publication/2647>>.

ILON, B. E. *Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base*. [S.l.]: Google Patents, 1975. US Patent 3,876,255.

ISO. *ISO 22523:2006. External limb prostheses and external orthoses — Requirements and test methods*. 2006. [Online; acessado 18-Julho-2020]. Disponível em: <<https://www.iso.org/standard/37546.html>>.

- ISO. *ISO 13482:2014. Robots and robotic devices — Safety requirements for personal care robots*. 2014. [Online; acessado 18-Julho-2020]. Disponível em: <<https://www.iso.org/standard/53820.html>>.
- ISO. *ISO 7176-1:2014. Wheelchairs — Part 1: Determination of static stability*. 2014. [Online; acessado 18-Julho-2020]. Disponível em: <<https://www.iso.org/standard/56817.html>>.
- ISO. *ISO 9999:2016 Assistive products for persons with disability - Classification and terminology*. 2016. [Online; acessado 18-Julho-2020]. Disponível em: <<https://www.iso.org/standard/60547.html>>.
- JARZĘBOWSKA, E. Dynamics modeling of nonholonomic mechanical systems: theory and applications. *Nonlinear Analysis: Theory, Methods & Applications*, Elsevier, v. 63, n. 5-7, p. e185–e197, 2005.
- Jl, J. et al. Design and control of an omni-directional robotic walker based on human–machine interaction. *IEEE Access*, IEEE, v. 9, p. 111358–111367, 2021.
- JIA, Y.; SONG, X.; XU, S. S.-D. Modeling and motion analysis of four-mecanum wheel omni-directional mobile platform. In: IEEE. *2013 CACS International Automatic Control Conference (CACS)*. [S.l.], 2013. p. 328–333.
- JIN, X.; CUI, X.; AGRAWAL, S. K. Design of a cable-driven active leg exoskeleton (c-alex) and gait training experiments with human subjects. In: IEEE. *2015 IEEE International Conference on Robotics and Automation (ICRA)*. [S.l.], 2015. p. 5578–5583.
- JOSEPH, L.; CACACE, J. *Mastering ROS for Robotics Programming: Design, build, and simulate complex robots using the Robot Operating System*. [S.l.]: Packt Publishing Ltd, 2018.
- KELBOUSCAS, A. L. da S. *Rastreo cinemático de membros superiores com Aplicações em Robótica Assistiva*. Dissertação (Mestrado) — Universidade Federal do Rio Grande, 2022.
- KHALAJI, A. K.; MOOSAVIAN, S. A. A. Adaptive sliding mode control of a wheeled mobile robot towing a trailer. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, SAGE Publications Sage UK: London, England, v. 229, n. 2, p. 169–183, 2015.
- KRIGGER, K. W. Cerebral palsy: an overview. *American family physician*, v. 73, n. 1, p. 91–100, 2006.
- KUBO, K. et al. Gait rehabilitation device in central nervous system disease: a review. *Journal of Robotics*, Hindawi, v. 2011, 2011.
- LINNER, T. et al. Modeling and operating robotic environments using gazebo/ros. In: *Proceedings of the 28th international symposium on automation and robotics in construction (ISARC2011)*. [S.l.: s.n.], 2011. p. 957–962.
- LOELIGER, J.; MCCULLOUGH, M. *Version Control with Git: Powerful tools and techniques for collaborative software development*. [S.l.]: "O'Reilly Media, Inc.", 2012.

LOOS, H. M. Van der; REINKENSMEYER, D. J.; GUGLIELMELLI, E. Rehabilitation and health care robotics. In: *Springer handbook of robotics*. [S.l.]: Springer, 2016. p. 1685–1728.

MATRAJI, I. et al. Trajectory tracking control of skid-steered mobile robot based on adaptive second order sliding mode control. *Control Engineering Practice*, Elsevier, v. 72, p. 167–176, 2018.

MÓR, J. L. *Modelagem, identificação de parâmetros e controle de velocidade de um robô soldador*. Dissertação (Mestrado) — Universidade Federal do Rio Grande, 2017.

MORIN, P.; SAMSON, C. Trajectory tracking for non-holonomic vehicles: overview and case study. In: IEEE. *Proceedings of the Fourth International Workshop on Robot Motion and Control (IEEE Cat. No. 04EX891)*. [S.l.], 2004. p. 139–153.

MURRAY, C. J. et al. Disability-adjusted life years (dalys) for 291 diseases and injuries in 21 regions, 1990–2010: a systematic analysis for the global burden of disease study 2010. *The lancet*, Elsevier, v. 380, n. 9859, p. 2197–2223, 2012.

NISE, N. S. *Control systems engineering*. [S.l.]: John Wiley & Sons, 2020.

NOVAK, I. et al. Early, accurate diagnosis and early intervention in cerebral palsy: advances in diagnosis and treatment. *JAMA pediatrics*, American Medical Association, v. 171, n. 9, p. 897–907, 2017.

OGATA, K. *Modern control engineering*. 2020.

PALOMINO-DÍAZ, V. et al. Diseño conceptual de una plataforma robótica para ayudar a que los niños con parálisis cerebral descubran cómo caminar. In: *XII Simposio CEA de Bioingeniería*. [S.l.: s.n.], 2021. p. 83.

PENG, T. et al. Mechanical design and control system of an omni-directional mobile robot for material conveying. *Procedia Cirp*, Elsevier, v. 56, p. 412–415, 2016.

PETZOLD, L. Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations. *SIAM journal on scientific and statistical computing*, SIAM, v. 4, n. 1, p. 136–148, 1983.

ROSENBAUM, P. et al. A report: the definition and classification of cerebral palsy april 2006. *Dev Med Child Neurol Suppl*, v. 109, n. suppl 109, p. 8–14, 2007.

SANTINA, M. S.; STUBBERUD, A. R.; HOSTETTER, G. H. Discrete-time equivalents to continuous-time systems. In: *Control System Fundamentals*. [S.l.]: CRC press, 2019. p. 267–281.

SEEGMILLER, N. A. *Dynamic model formulation and calibration for wheeled mobile robots*. Tese (Doutorado) — Carnegie Mellon University - Pittsburgh, 2014.

SHI, D. et al. A review on lower limb rehabilitation exoskeleton robots. *Chinese Journal of Mechanical Engineering*, Springer, v. 32, n. 1, p. 1–11, 2019.

SHOAIB, M. et al. Cable driven rehabilitation robots: Comparison of applications and control strategies. *IEEE Access*, IEEE, 2021.

SICILIANO, B. et al. *Robotics*. [S.l.]: Springer London, 2009.

SILVA, S. A. V. e. *Planejamento de Trajetoria para Terapia Assistida por Robô para Membros Superiores*. Dissertação (Mestrado) — Universidade Federal do Rio Grande, 2021.

SIMÕES, C. C. et al. A experiência dos pais no cuidado dos filhos com paralisia cerebral. *Revista Eletrônica de Enfermagem*, v. 15, n. 1, p. 138–45, 2013.

SOLEA, R.; FILIPESCU, A.; NUNES, U. Sliding-mode control for trajectory-tracking of a wheeled mobile robot in presence of uncertainties. In: IEEE. *2009 7th Asian control conference*. [S.l.], 2009. p. 1701–1706.

TAKAYA, K. et al. Simulation environment for mobile robots testing using ros and gazebo. In: IEEE. *2016 20th International Conference on System Theory, Control and Computing (ICSTCC)*. [S.l.], 2016. p. 96–101.

TĂȚAR, M. O. et al. Design and development of an autonomous omni-directional mobile robot with mecanum wheels. In: IEEE. *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*. [S.l.], 2014. p. 1–6.

TESIO, L.; ROTA, V. The motion of body center of mass during walking: a review oriented to clinical applications. *Frontiers in neurology*, Frontiers, p. 999, 2019.

TLALE, N.; VILLIERS, M. de. Kinematics and dynamics modelling of a mecanum wheeled mobile platform. In: IEEE. *2008 15th International Conference on Mechatronics and Machine Vision in Practice*. [S.l.], 2008. p. 657–662.

TSAI, C.-C.; TAI, F.-C.; LEE, Y.-R. Motion controller design and embedded realization for mecanum wheeled omnidirectional robots. In: IEEE. *2011 9th World Congress on intelligent control and automation*. [S.l.], 2011. p. 546–551.

VYAVAHARE, P.; JAYAPRAKASH, S.; BHARATIA, K. Construction of urdf model based on open source robot dog using gazebo and ros. In: IEEE. *2019 Advances in Science and Engineering Technology International Conferences (ASET)*. [S.l.], 2019. p. 1–5.

ZEIDIS, I.; ZIMMERMANN, K. Dynamics of a four-wheeled mobile robot with mecanum wheels. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, Wiley Online Library, v. 99, n. 12, p. e201900173, 2019.

ZIMMERMANN, K.; ZEIDIS, I.; ABDELRAHMAN, M. Dynamics of mechanical systems with mecanum wheels. In: *Applied Non-Linear Dynamical Systems*. [S.l.]: Springer, 2014. p. 269–279.