

**MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
CURSO DE GRADUAÇÃO EM ENGENHARIA MECÂNICA E DE
AUTOMÓVEL**

VINÍCIUS RIBEIRO CAVALEIRO DE MACEDO

SIMULAÇÃO E CONTROLE DE VEÍCULOS ROBÓTICOS TERRESTRES

**RIO DE JANEIRO
2022**

VINÍCIUS RIBEIRO CAVALEIRO DE MACEDO

SIMULAÇÃO E CONTROLE DE VEÍCULOS ROBÓTICOS TERRESTRES

Projeto de Final de Curso apresentado ao Curso de Graduação em Engenharia Mecânica e de Automóvel do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Bacharel em Engenharia Mecânica e de Automóvel.

Orientadores: Maj Gustavo Simão Rodrigues, D.Sc.
 Cap Elias Dias Rossi Lopes, D.Sc.

Rio de Janeiro

2022

CIP - Catalogação na Publicação

Macedo, Vinícius Ribeiro Cavaleiro de
Simulação e Controle de Veículos Robóticos
Terrestres / Vinícius Ribeiro Cavaleiro de Macedo.
– Rio de Janeiro, 2022.
57 f.

Orientadores: Gustavo Simão Rodrigues; Elias Dias
Rossi Lopes.

Projeto Final de Curso (graduação) – Instituto
Militar de Engenharia, Bacharel em Engenharia
Mecânica e de Automóveis, 2022.

1. controle preditivo baseado em modelo. 2.
esterçamento diferencial. 3. seguimento de
trajetória. 4. veículo robótico. 5. modelo dinâmico.
I. Rodrigues, Gustavo Simão, orient. II. Lopes,
Elias Dias Rossi, orient. III. Título

VINÍCIUS RIBEIRO CAVALEIRO DE MACEDO

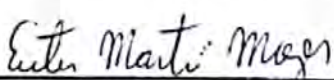
Simulação e Controle de Veículos Robóticos Terrestres

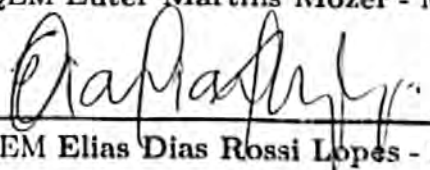
Projeto de Final de Curso apresentado ao Curso de Graduação em Engenharia Mecânica e de Automóvel do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Bacharel em Engenharia Mecânica e de Automóvel.

Orientadores: Maj Gustavo Simão Rodrigues e Cap Elias Dias Rossi Lopes.

Aprovado em Rio de Janeiro, 11 de outubro de 2022, pela seguinte banca examinadora:


Maj QEM Gustavo Simão Rodrigues - D.Sc. do IME - Presidente


Maj QEM Euter Martins Mozer - M.Sc. do IME


Cap QEM Elias Dias Rossi Lopes - D.Sc. do IME.

Rio de Janeiro
2022

AGRADECIMENTOS

Após esta árdua jornada, não posso me privar de agradecer à minha família por ter me feito chegar até aqui. Minha mãe Sabrina, meu pai Júlio César e meu irmão Rodrigo foram e são essenciais em toda a minha construção como profissional e, sem dúvidas, como pessoa, sempre me motivando e sendo o pilar emocional em que pude me amparar nos momentos mais difíceis.

Agradeço imensamente também aos meus companheiros Gabriel Fernandez, Tales Rocha, Vilson Wenis, Rogério Souza, Lucas Frota, Eduardo Willrich e Lauro Schramm, que, como companheiros de engenharia, ombrearam comigo durante o curso e, em meio às adversidades e grandes cargas de trabalhos e provas, mantiveram meu foco, disciplina e sempre me ajudaram no que fosse necessário.

Gostaria de fazer agradecimentos especiais aos agentes diretos do meu processo de aprendizado, meus professores e orientadores. Inicialmente agradecer aos professores Félix e Márcio Pinheiro, que não só me proporcionaram uma base muito boa em física e matemática, como me apontaram os caminhos a serem trilhados ainda no pré-vestibular. Agradeço também aos professores Jayme Mendes, Egydio Carvalho e Ricardo Teixeira, os quais, com seus ensinamentos de excelência, me proporcionaram uma experiência de aprendizado motivadora, buscando sempre aprimoramento e o melhor do conhecimento. E por fim, gostaria de agradecer imensamente aos meus orientadores Gustavo Simão e Elias Rossi, os quais me acompanharam durante grande parte de minha trajetória no Instituto, sempre se colocando à disposição, com paciência, presteza, preocupação com meu desenvolvimento e aprimoramento contínuo.

RESUMO

Um dos grandes desafios, na atualidade, para tornar sistemas robóticos, sobre rodas ou lagartas, plenamente funcionais é a otimização do sistema de controle embarcado. Desta forma, o presente trabalho estuda, implementa computacionalmente e analisa a aplicação de Controle Preditivo Baseado em Modelo neste tipo de veículo robótico. Para isso, foi utilizado o modelo dinâmico de veículo com esterçamento diferencial, em que não há esterçamento das rodas e a mudança de direção é dada pela diferença de forças de tração em cada roda. Com isso, três estudos de caso são simulados e analisados: controle PID, MPC e controle em cascata de MPC combinado com PID. Analisa-se a estratégia de controle de cada caso, os controladores são calibrados e os resultados são apresentados em figuras e tabelas comparativas. A precisão e o tempo de processamento são discutidos e considerações finais demonstrando uma maior eficiência do último caso, que alia vantagens dos dois controladores associados.

Palavras-chave: controle preditivo baseado em modelo. esterçamento diferencial. seguimento de trajetória. veículo robótico. modelo dinâmico.

ABSTRACT

One of the great challenges, nowadays, to make robotic systems, wheeled or tracked, fully functional is the optimization of the on-board control systems. Thus, the present work aims to study, computationally implement, and analyze the application of Model-Based Predictive Control in this type of robotic vehicle. So far, therefore the dynamic model of a vehicle with differential steering has been used, in which there is no steering of the wheels, and the direction is given by the difference of traction forces in each wheel. With this, three case studies are simulated and analyzed: PID control, MPC, and MPC cascade control combined with PID. The control strategy of each case is analyzed, the controllers are calibrated, and the results are presented in figures and comparative tables. The precision and the processing time are discussed, and final considerations demonstrate a greater efficiency of the last case, which combines the advantages of the two associated controllers.

Keywords: model-based predictive control. skid-steering. path tracking. robot vehicle. dynamic model.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama de blocos do PID	21
Figura 2 – Estratégia do MPC	23
Figura 3 – Diagrama de blocos do MPC	24
Figura 4 – Esquema de robô com esterçamento diferencial	26
Figura 5 – Diagrama de corpo livre de uma roda	27
Figura 6 – Atrito de Coulomb	28
Figura 7 – Atrito de Coulomb com aproximação linear	29
Figura 8 – Rover Lynxmotion Aluminum 4WD1	30
Figura 9 – Trajetórias de referência	31
Figura 10 – Iterações da posição do robô	32
Figura 11 – Composição de posições	33
Figura 12 – EC1 - curvas circulares: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T . . .	37
Figura 13 – EC1 - curvas abruptas: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T . . .	38
Figura 14 – (a) e (b) dados para s. (c) e (d) dados para q. (e) e (f) dados para N. .	40
Figura 15 – EC2 - curvas circulares: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T . . .	41
Figura 16 – EC2 - curvas abruptas: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T . . .	42
Figura 17 – Diagrama de blocos do controle em cascata	43
Figura 18 – (a) e (b) dados para s. (c) e (d) dados para q. (e) e (f) dados para N. .	45
Figura 19 – EC3 - curvas circulares: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T . . .	47
Figura 20 – EC3 - curvas abruptas: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T . . .	48

LISTA DE TABELAS

Tabela 1 – Parâmetros físicos	30
Tabela 2 – Limites físicos para as simulações	31
Tabela 3 – Valores nominais das saídas para as trajetórias	32
Tabela 4 – Ganhos do PID para o EC1	35
Tabela 5 – EC1: parâmetros de desempenho	36
Tabela 6 – Parâmetros do MPC para o EC2	39
Tabela 7 – EC2: parâmetros de desempenho	40
Tabela 8 – Parâmetros do MPC para o EC3	44
Tabela 9 – Ganhos do PID para o EC3	46
Tabela 10 – EC3: parâmetros de desempenho	46
Tabela 11 – Erros percentuais para simulações das curvas circulares	49
Tabela 12 – Erros percentuais para simulações das curvas abruptas	49
Tabela 13 – Tempo médio de cada iteração	51

LISTA DE ABREVIATURAS E SIGLAS

MPC	<i>Model-based Predictive Control</i>
UGV	<i>Unmanned Ground Vehicle</i>
FIS	<i>Fuzzy inference system</i>
GPS	<i>Global Positioning System</i>
UKF	<i>Unscented Kalman Filter</i>
PID	Proporcional-Integral-Derivativo
RMSE	<i>Root mean square error</i>
EC1	Estudo de caso 1
EC2	Estudo de caso 2
EC3	Estudo de caso 3

LISTA DE SÍMBOLOS

u	Sinal de controle
x	Estados
z	Saídas do sistema
r	Sinal de referência
e	Erro entre a referência e a saída
o	Estados secundários no controle em cascata
t	Tempo
k_P	Ganho proporcional
k_I	Ganho integral
k_D	Ganho derivativo
N	Janela de predição
CF	Função custo
Q	Matriz de peso das saídas
q	Potência de 10 que multiplica a matriz identidade em Q
S	Matriz de variação de sinais de controle
s	Potência de 10 que multiplica a matriz identidade em S
R	Matriz de peso das entradas de referência
Δt	Intervalo de tempo por iteração
m_r	Massa do robô
J_w	Momento de inércia polar das rodas do robô
I_z	Momento de inércia polar do robô em relação ao eixo ortogonal ao solo
y_r	Ordenada da posição do centro de massa do robô em referencial absoluto
x_r	Abscissa da posição do centro de massa do robô em referencial absoluto
ϕ	Ângulo de esterçamento do robô

ω_1	Velocidade angular da roda direita
ω_2	Velocidade angular da roda esquerda
v_x	Velocidade linear longitudinal do robô
$v_{x,roda}$	Velocidade linear do centro da roda
ω_r	Velocidade angular do robô
$F_{x,1}$	Força de tração na roda direita
$F_{x,2}$	Força de tração na roda esquerda
L	Distância entre os centros das rodas
r_w	Raio das rodas
c_v	Coefficiente de atrito viscoso
g	Gravidade
T_1	Torque na roda direita
T_2	Torque na roda esquerda
μ	Coefficiente de atrito roda solo
F_z	Força vertical sobre cada roda
F_μ	Força de atrito
F_{ap}	Força aplicada a um corpo sob atrito em uma condição estática
v_d	Velocidade de deslizamento
$v_{d,max}$	Velocidade máxima de deslizamento

SUMÁRIO

1	INTRODUÇÃO	14
1.1	MOTIVAÇÃO	14
1.2	OBJETIVOS	14
2	REVISÃO BIBLIOGRÁFICA	16
2.1	VEÍCULOS TERRESTRES NÃO-TRIPULADOS (UGVS)	16
2.2	<i>SKID-STEERING</i>	17
2.3	SEGUIMENTO DE TRAJETÓRIA	18
2.4	CONTROLE PREDITIVO BASEADO EM MODELO (MPC)	19
3	FUNDAMENTAÇÃO TEÓRICA	21
3.1	CONTROLE PID	21
3.2	CONTROLE PREDITIVO BASEADO EM MODELO (MPC)	22
3.3	MODELO PLANO DE VEÍCULO ROBÓTICO COM ESTERÇAMENTO DIFERENCIAL	25
3.4	MODELO DINÂMICO DAS RODAS	27
3.5	MODELO DE ATRITO	28
4	METODOLOGIA	30
4.1	TRAJETÓRIAS E AVALIAÇÃO	31
4.2	GERAÇÃO DOS DADOS DE REFERÊNCIA	32
4.3	CALIBRAÇÃO DO PID	33
4.4	CALIBRAÇÃO DO MPC	33
5	ESTUDO DE CASO 1: CONTROLE PID	35
5.1	ESTRATÉGIA DE CONTROLE	35
5.2	CALIBRAÇÃO DO PID	35
5.3	RESULTADOS	36
6	ESTUDO DE CASO 2: CONTROLE PREDITIVO	39
6.1	ESTRATÉGIA DE CONTROLE	39
6.2	CALIBRAÇÃO DO MPC	39
6.3	RESULTADOS	39
7	ESTUDO DE CASO 3: CONTROLE EM CASCATA	43
7.1	ESTRATÉGIA DE CONTROLE	43
7.2	CALIBRAÇÃO	44

7.3	RESULTADOS	46
8	ANÁLISE COMPARATIVA DE RESULTADOS	49
8.1	PRECISÃO	49
8.2	TEMPO DE PROCESSAMENTO	50
9	CONCLUSÃO	52
	REFERÊNCIAS	54

1 INTRODUÇÃO

Com o rápido desenvolvimento da inteligência artificial, da tecnologia automotiva e dos sistemas de sensoriamento, a utilização de veículos terrestres não-tripulados tem sido cada vez mais presente e importante, com aplicações tanto militares quanto civis (1). Esse tipo de tecnologia tem por funções primordiais substituir a mão-de-obra humana em situações indesejáveis ou tarefas perigosas, além de reduzir os custos e aumentar a eficiência da tarefa em que está sendo empregado (2). Para isso, muitas vezes são utilizadas plataformas veiculares robóticas autônomas. Segundo Murtaza et al.(3), máquinas autônomas são máquinas inteligentes que fazem o sensoriamento, planejam e então agem de forma racional para realizar uma determinada tarefa, ou alcançar determinado objetivo, sem nenhuma interação com qualquer humano.

Dessa forma, existem diversas aplicações para este tipo de tecnologia, tais como *rovers* de exploração espacial, identificação de potenciais áreas de risco na mineração, operação ininterrupta na agricultura, transporte de cargas pesadas e volumosas em ambiente fabril, entre outras (4). Aplicações militares muito se destacam como no manuseio e desativação de bombas e minas, operações de manutenção de paz, vigilância de terreno, missões de resgate, recuperação, patrulha e até de combate (1).

1.1 Motivação

Nesse contexto, um dos maiores desafios para tornar funcional sistemas veiculares robóticos, sobre rodas ou lagartas, é a eficiência e precisão do sistema de controle embarcado. Atualmente há uma grande gama de sistemas e algoritmos de controle, e dentre eles um se destaca, o Controle Preditivo Baseado em Modelo (MPC). Muito já foi desenvolvido sobre esta estratégia de controle, porém ainda há a necessidade de melhorias em sua implementação para torná-lo cada vez mais funcional, em especial no que tange ao tempo de processamento, que é relativamente alto para emprego online em hardwares embarcados.

1.2 Objetivos

O presente trabalho tem por objetivo o estudo, implementação computacional e análise de controle de trajetória de veículos robóticos utilizando o MPC. Para isso, estuda-se o modelo dinâmico de veículos com esterçamento diferencial, onde não há esterçamento das rodas e a rotação do robô é dada pela diferença de forças de tração em cada roda. Deseja-se comparar três diferentes estratégias de controle: controle PID, Controle MPC e controle em cascata MPC combinado com PID. Para isso três estudos de caso são feitos,

sendo analisadas duas trajetórias, trajetória com curvas circulares e trajetória com curvas abruptas.

2 REVISÃO BIBLIOGRÁFICA

2.1 Veículos Terrestres Não-tripulados (UGVs)

Muito se desenvolveu acerca de veículos terrestres não-tripulados (UGV) nas últimas décadas, principalmente pela grande gama de aplicações desse tipo de tecnologia, desde veículos remotamente controlados até *rovers* espaciais. Almayyahi et al.(5) propõem um sistema de inferência *fuzzy* (FIS) baseado em sensoriamento para resolução de mudança na navegação de UGVs em ambientes caóticos e dinâmicos. O FIS consiste de dois controladores, em que o primeiro se baseia na distância ao obstáculo pela direita, pela frente e pela esquerda, e o segundo emprega a diferença de ângulo entre a frente do veículo e o ângulo do alvo para obter a melhor rota com o menor esforço e atraso. Três cenários são montados e testados, variando-se os tamanhos e velocidades dos obstáculos, provando-se a capacidade de evitar obstáculos e se mover em direção ao seu destino.

Ni, Hu e Xiang(6) descrevem diversos mecanismos e processos dos principais subsistemas de UGVs, com especial ênfase nos subsistemas de direção acionados por motores independentes em cada roda. Dessa forma, pôde-se analisar o modo de direcionamento diagonal, que melhora grandemente a manobrabilidade do veículo. Em seguida, propõe-se um controlador robusto para atingir o *sideslip* requerido no movimento diagonal e evitar taxa de *yaw*, conseguindo melhorar significativamente o desempenho do movimento. Por fim, testes e experimentos foram feitos para validar o controlador.

Hu et al.(7) projetam uma estratégia de controle de esterçamento para se evitar obstáculos em veículos terrestres não-tripulados. Nele divide-se os obstáculos em estáticos e dinâmicos, e, usando o modelo bicicleta para os veículos, propõe-se a estratégia de controle preditivo baseada em modelo contínuo no tempo. Por fim, compara-se os resultados de simulações feitas no Matlab® com o software CarSim®.

Ni, Hu e Xiang(1) fazem uma revisão sobre veículos terrestres não tripulados, focando no projeto e controle dinâmico. Faz-se um apanhado geral sobre as diversas aplicações e categorias deste tipo de veículo, salientando o uso militar, bem como explicitando o papel chave da percepção de ambiente, planejamento de movimento, controle dinâmico do chassi e controle baseado na nuvem, sendo este último tido pelos autores como tendência futura.

2.2 *Skid-steering*

Há diversos tipos de veículos terrestres não-tripulados, como mostrado, porém um tipo é especialmente salientável, o veículo robótico com esterçamento diferencial, ou *skid-steering*. Neste tipo de veículo o esterçamento é dado pela diferença de torques nas rodas. Dessa forma, muito foi estudado e desenvolvido sobre estes.

Mazur e Cholewiński(8) apresentam um modelo de controle para um manipulador sobre um robô com esterçamento diferencial. Foi utilizada uma plataforma REX que se move apenas em um plano com vista superior. Neste modelo tem-se cinco graus de liberdade: movimento em x , y , yaw e rotações das 4 rodas, porém as rodas de um mesmo lado estão acopladas por correia, e deriva-se o modelo dinâmico a partir da fórmula de Lagrange. Por fim, o modelo dinâmico da plataforma e do braço são unidos e simulações são feitas.

Barrero, Tilaguy e Nova(9) desenvolvem um sistema de navegação de alta precisão para robôs usados na agricultura, logo usados em exteriores, dessa forma utilizou-se um modelo de robô de quatro rodas com esterçamento diferencial. Para este modelo tem-se 5 graus de liberdade e de entrada são dadas as velocidades angulares das rodas em cada lado, e de saída a velocidade longitudinal do robô e sua taxa de yaw . Como sensoriamento utiliza-se GPS, magnetômetro e *encoders*.

Lv, Kang e Qin(10) usam o modelo com 5 graus de liberdade para robô com esterçamento diferencial, como diversos anteriores, porém agora apresenta-se um novo modelo de localização para estes robôs em ambiente interno. Neste modelo faz-se o sensoriamento com *encoders*, giroscópio e magnetômetro, garantindo significativa melhoria na precisão dos resultados.

Ibrahim et al.(11) utilizam coordenadas polares em vez de cartesianas para localizar o robô no espaço, porém ainda se trabalha em um plano e com um modelo de 3 graus de liberdade, retirando-se os das rodas, pois o foco é o controle a partir das coordenadas polares. Com isso, fez-se um algoritmo de seguidor de trajetória baseado em controle postural descontínuo adaptativo.

Fernandez, Herrera e Cerrada(12) usam, novamente, o modelo de 5 graus de liberdade para fazer um controle simplificado de um robô com esterçamento diferencial seguidor de trajetória com aplicação na agricultura. Neste modelo, desejou-se controlar apenas a distância do centro de massa do robô à linha central da trajetória. Da mesma forma, as estratégias de controle mais utilizadas também são expostas e explicadas: geométrica & cinemática, dinâmica, ótima, adaptativa, baseada em modelo e a clássica. Vantagens e desvantagens de cada uma são mostradas de forma crítica e algumas possibilidades futuras de melhoria são elencadas.

2.3 Seguimento de Trajetória

Salienta-se uma aplicação notória para este tipo de veículo que é o seguimento de trajetória pré-definida ou estipulada. Para isto a automação de robôs foi muito desenvolvida, em especial os algoritmos de controle.

Em (13) apresenta-se um algoritmo de repetição de trajetória usando controle preditivo baseado em modelo não linear, porém com base em aprendizado, que visa ser utilizado em grandes ambientes externos que não se pode usar GPS. A localização do controlador é fornecida por um sistema de navegação e mapeamento baseado em visão integrado. O cerne do algoritmo é reduzir o erro do rastreamento usando a experiência real do robô, em vez de complexos algoritmos e modelos analíticos de interação solo-pneu, topologia do terreno e dinâmica do robô. O algoritmo foi testado em três diferentes robôs, sendo dois com esterçamento diferencial e um com tipo Ackermann.

Ostafew, Schoellig e Barfoot(14) apresentam um algoritmo de controle preditivo baseado em modelo não linear para seguidor de trajetória em terrenos *off-road*, semelhante ao trabalho anterior. Porém, neste adicionam-se restrições robustas ao modelo, que podem representar obstáculos sólidos ou a localização de limites. Assim como no anterior a localização do controlador é fornecida por um sistema de navegação e mapeamento baseado em visão integrado. Utilizou-se dois robôs consideravelmente diferentes para os testes, obtendo-se bons resultados.

Laurense, Goh e Gerdes(15) discutem a necessidade de em algumas aplicações de veículos autônomos trabalhar-se no limite de fricção pneu-solo, porém não se pode perder a manobrabilidade, logo a capacidade de esterçamento. Para isso, desenvolveu-se uma estratégia de controle baseada no ângulo de *slip* lateral para uma máxima força de tração, utilizando-se um modelo bicicleta para dinâmica lateral, com esterçamento Ackermann.

Amer et al.(16) fazem uma revisão do estado da arte e dos desafios na área de modelagem e estratégias de controle para veículos autônomos terrestres seguidores de trajetória. Os modelos dinâmicos de veículos mais usados são explicados e sumarizados, como: geométrico, cinemático, dinâmico linear e não-linear, porém todos para veículos com esterçamento Ackermann.

Sun et al.(17) apresentam um método de linearização para o modelo do veículo e do pneu, vistas as não-linearidades presentes nestes modelos, sobre o algoritmo do controle preditivo baseado em modelo, e propõe um controle de esterçamento para veículos autônomos. Utiliza-se o modelo bicicleta para dinâmica lateral. Projeta-se o controlador para minimizar o desvio lateral à altas velocidades. O ângulo entre a velocidade vetorial do veículo e o caminho é escolhido de estado de referência para o controle. O controlador é testado em Simulink® e CarSim®, mostrando a melhora no desempenho alcançada pelo modelo.

Cui et al.(18) unem um estimador de estados e parâmetros veiculares a um método de controle para melhoria de desempenho e estabilidade veicular a alta velocidade em veículos autônomos seguidores de trajetória. Para o estimador utilizou-se do UKF, modelando-se o veículo com o modelo bicicleta juntamente com a fórmula mágica para modelagem dos pneus. Além disso, foi proposto o controle preditivo baseado em modelo de múltiplas restrições e foi então utilizado para calcular o ângulo de esterçamento para se seguir a trajetória planejada. Ao final, avaliou-se o desempenho do modelo proposto com simulações no CarSim[®] e no Matlab/Simulink[®].

Wang et al.(19) propõem uma forma melhorada do controle preditivo baseado em modelo aplicada em veículos autônomos seguidores de trajetória. Adicionou-se um sistema de controle adaptativo fuzzy para o peso da função custo ao modelo clássico de MPC. Utilizou-se o modelo bicicleta para modelo dinâmico do veículo. com isso atingiu-se alta precisão no controle, demonstrada por simulações em Matlab/Simulink[®] e CarSim[®].

Bai et al.(20) descobriram que trocando entre o controle preditivo baseado em modelo linear, não linear e outros, pode-se operar a diferentes velocidades longitudinais, mas não se pode ativamente ajustar a velocidade longitudinal, em robôs autônomos seguidores de trajetória. Dessa forma, propuseram uma estrutura de controle MPC que pode ativamente ajustar a velocidade longitudinal enquanto seguindo, chamada de MPC de multicamadas. Para isso, usou-se novamente o modelo bicicleta para modelagem dinâmica do veículo.

Em (21), em vez de usar um controlador baseado em modelo cinemático (restrições não-holonômicas), faz-se um controlador baseado na mecânica dos pneus para um robô seguidor de caminho e sobre rodas. Para isso, utiliza-se o modelo de robô com esterçamento diferencial. Simulações mostraram que este método é mais preciso que o convencional.

2.4 Controle Preditivo Baseado em Modelo (MPC)

Uma das formas que ganhou grande atenção para se abordar problemas de veículos seguidores de trajetória é o controle preditivo baseado em modelo (MPC). Em (22) é apresentado o MPC contínuo para seguidor de trajetória. O autor defende que em muitas situações não se consegue uma amostragem uniforme para o controle, graças a imperfeição nos sensores, relógios mal sincronizados, atrasos não determinísticos no controle e por conta do tempo de pré-processamento desconhecido. Dessa forma, devido à abordagem em tempo contínuo, a aplicação do MPC proposto não é limitada à uniformidade de amostragem. Com isso, é demonstrado que em situações gerais envolvendo amostras uniformes no tempo, o MPC contínuo tem desempenho similar ao MPC discreto, porém aquele tem resultados melhores quando não, além de mostrar melhor robustez nos parâmetros de projeto das leis de controle.

Negri, Cavalca e Celiberto(23) apresentam um MPC não linear com características de tolerância a falhas a ser usado em manipulador robótico de dois graus de liberdade. O MPC utilizado é obtido pelo treinamento de rede neural artificial em cascata totalmente conectada. Foram utilizadas duas diferentes abordagens de tratamento de falhas: a estratégia robusta e a estratégia adaptativa. Na primeira tem-se uma implementação mais simples em relação a adaptativa, e apresenta resultados mais rápidos e estáveis em relação ao MPC sem tratamento de falhas. No segundo, tem-se uma maior melhoria no desempenho do controlador, porém com a desvantagem de ter mais parâmetros para se calibrar.

Song, Shao e Qu(24) propõem um método de controle para seguidores de trajetória com tempo variante. Faz-se para um veículo de duas dimensões e é usado o modelo bicicleta para modelagem dinâmica. Este método considera a influência da velocidade longitudinal do robô e a estabilidade do robô na curvatura da trajetória sob complexas condições de direção em baixas velocidades.

Farag(25) desenvolve um controlador de MPC que permite manobrabilidade em trajetórias complexas para veículos autônomos, dissertando sobre todos os detalhes do projeto e estágios de implementação do algoritmo proposto. Para isso, como sinal de entrada tem-se a posição precisa do veículo, sua velocidade instantânea e a trajetória de referência a ser seguida, e como saídas tem-se o comando de válvula borboleta e o comando de esterçamento. Utilizou-se o modelo bicicleta para modelagem dinâmica do veículo autônomo. Segundo o autor, uma das principais contribuições do artigo é a forma da função objetiva usada no controlador, a qual tem 9 diferentes tipos de termo, cada um com um peso, levando em consideração diversos aspectos. Com isso, conseguiu-se simular o controle de veículos autônomos em complexas trajetórias e com precisão.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Controle PID

Sistemas de controle são sistemas que visam gerenciar o comportamento de alguma entidade física, por meio de um conjunto de equipamentos e dispositivos. Deseja-se então a partir de um sinal de controle (u), controlar as saídas (z) de um sistema, ou uma planta. Dessa forma, o controle Proporcional-Integral-Derivativo, ou controle PID, é uma estratégia de controle clássica desenvolvida por meio da experiência, tentativa e erro, e que utilizando o erro entre as saídas e suas referências (e), a integral deste erro e a derivada deste, pode-se gerar uma entrada de controle que consiga atender aos critérios.

Dessa forma, três constantes são definidas: ganho proporcional (k_P), ganho integral (k_I) e ganho derivativo (k_D), de forma que:

$$u(t) = k_P \cdot e(t) + k_I \cdot \int_0^t e(\tau) \cdot d\tau + k_D \cdot \frac{d}{dt}e(t) \quad (3.1)$$

O ganho proporcional tem por objetivo aumentar a ação de controle em um sistema com um erro grande, de forma a acelerá-lo e fazê-lo alcançar o valor desejado rapidamente. Já o ganho integral tem por objetivo minimizar o erro estacionário no seguimento da referência e a resposta a perturbações. O ganho derivativo tem por objetivo melhorar a resposta transiente e diminuir o *overshoot*. A Figura 1 mostra o diagrama de blocos do controle PID.

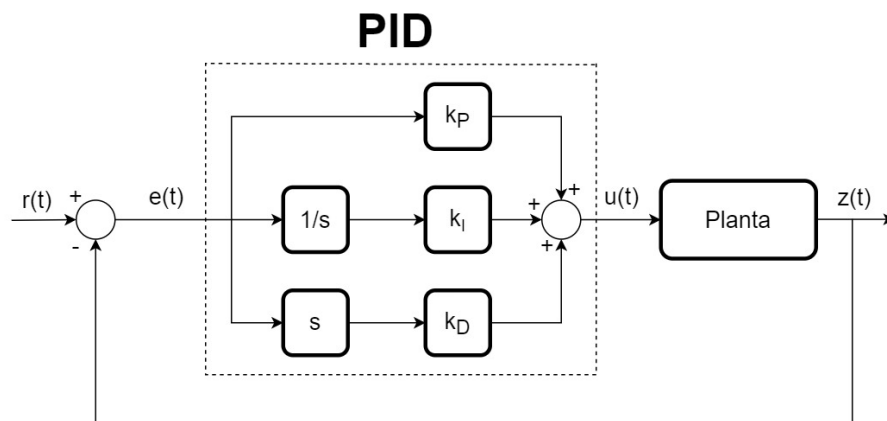


Figura 1 – Diagrama de blocos do PID

3.2 Controle Preditivo Baseado em Modelo (MPC)

O MPC é uma estratégia de controle baseada em otimização. Diferentemente dos controles convencionais que analisam os estados e saídas medidas, juntamente com a referência dada, para definir o próximo sinal de controle, o MPC utiliza a predição de estados futuros, em uma janela (ou horizonte) de predição, para definir o próximo sinal a ser enviado para os atuadores. Dessa forma, consegue-se sinais de controle mais suaves, mais eficientes e observando limitações físicas dos estados e dos atuadores.

Uma boa analogia para entender esta estratégia de otimização é pensar que uma estratégia de controle convencional é como um motorista de um carro que dirige seu veículo olhando para o chão, ou para trás, para seguir uma trajetória, enquanto no MPC o motorista está olhando para frente, vendo os próximos pontos que deve passar e baseando sua direção nestas informações. Este tipo de estratégia de controle tem se mostrado muito poderosa nos últimos anos, e isso se deve a alguns fatores como:

1. No MPC consegue-se impor restrições ao modelo, o que impede soluções inviáveis fisicamente, como acelerações inalcançáveis para veículos, por exemplo;
2. Funcional para sistemas não-lineares e fortemente não-lineares;
3. Flexível, por sua característica de recálculo a cada iteração;
4. *On-line*, pois utiliza-se de medições em tempo real para adequação à tarefa e para recálculo dos sinais de controle.

Apesar disso, grande parte dos algoritmos de MPC ainda são computacionalmente custosos, o que ainda dificulta sua utilização em aplicações *on-line*, logo ainda há a necessidade de *hardwares* que proporcionem rápida capacidade de processamento. Entretanto, com o avanço tecnológico dos computadores, esta tarefa vem se tornando cada vez mais fácil e acessível.

Adentrando esta estratégia de controle mais profundamente, deseja-se, a partir do sinal de controle u e do sinal de referência r , e com base em um modelo matemático, fazer com que a saída z alcance os valores desejados. Para isso, a partir do estado atual do sistema em t , utiliza-se o modelo matemático para prever os valores de entrada u , em uma janela de predição N , que geram os melhores valores de saída ao longo de toda a janela. Dessa forma, é feita a predição dos valores de entrada que otimizam as saídas, de uma amostragem $t + 1$ até $t + N$.

Vale ressaltar que estas amostragens são as iterações nas quais acontece o envio do sinal de controle. Em aplicações *on-line* de controle, tem-se janelas de amostragem, ou seja, uma janela de tempo entre uma coleta de dados por sensores e a seguinte, e consequente

janela de tempo entre o envio de sinais consecutivos para os atuadores. Dessa forma, no MPC é feita a predição dos sinais de controle u para N coletas a frente. A Figura 2 elucida o funcionamento da estratégia de otimização do MPC. Nela deseja-se alcançar o valor da referência $r(t)$, e para isso é definida em um tempo t uma sequência de sinais de controle $u(t)$ a partir do modelo. Com isso, calcula-se as saídas preditas \hat{z} para uma dada janela N .

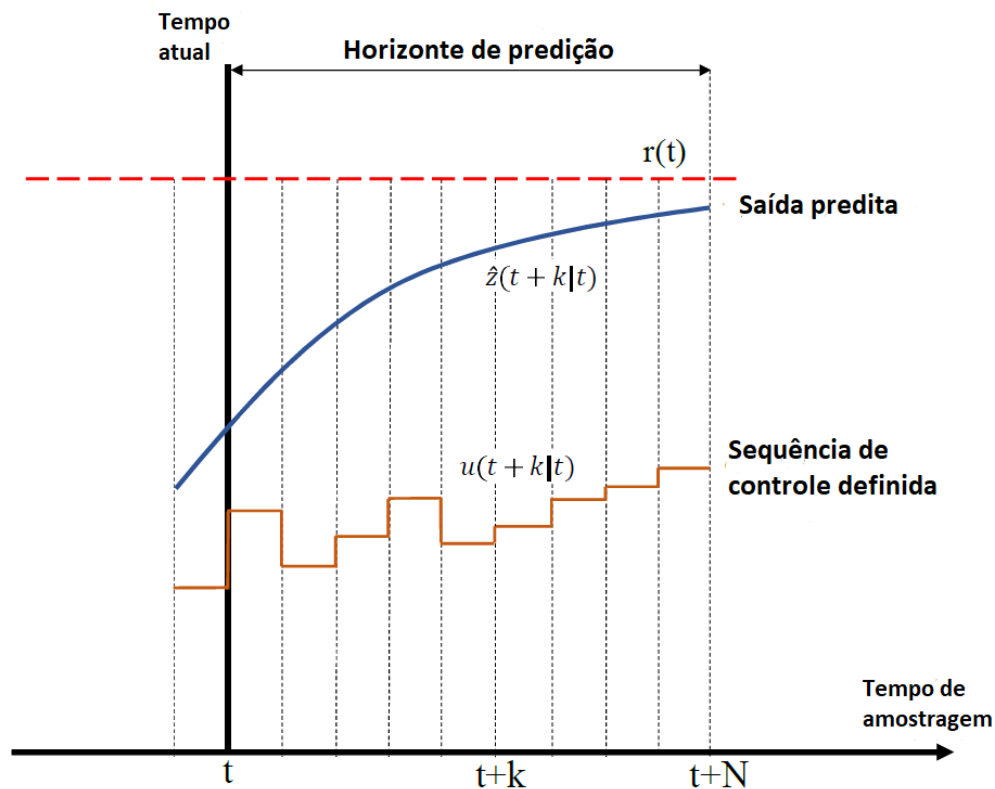


Figura 2 – Estratégia do MPC (26)

Após a realização da predição dos N valores de saída, tem-se os valores de entrada (sinal de controle) que os geraram, então descarta-se os $N - 1$ últimos valores de u e utiliza-se apenas o primeiro para ser utilizado no controle. Em seguida, é dado um *shift* em toda a janela de predição, passa-se para a próxima iteração e a janela é movida juntamente com a iteração, passando do que seria de $t + 1$ a $t + N$ para $t + 2$ a $t + N + 1$, e então repete-se o processo de otimização. A Figura 3 mostra o diagrama de blocos de um sistema de controle preditivo.

Para que esta estratégia funcione é necessário um critério de otimização dos valores u na janela, e para isso utiliza-se uma função custo CF , ou função objetivo, cuja minimização permite a obtenção dos valores ótimos de u . Existem diversas formulações para o equacionamento desta função, e o que será aqui utilizado é o seguinte:

$$CF(u(t:t+N), x(t:t+N)) = \sum_{i=t}^{t+N} \|z(i) - r(i)\|_Q^2 + \|u(i) - u_r(i)\|_R^2 + \|\Delta u(i)\|_S^2 \quad (3.2)$$

Em que $u_{t:t+N}$ e $x_{t:t+N}$ são os valores das entradas a serem calculadas e estados gerados por elas, indo de t até $t+N$. Nesta função custo tem-se um somatório de 3 parcelas ao longo da janela de predição. A primeira parcela é referente à diferença entre as saídas z geradas por u e a referência imposta, valor que se deseja alcançar, objetivando-se obter a menor diferença entre eles. A segunda parcela é referente à diferença entre o sinal de controle u e a referência dada a ele u_r . A última parcela é referente a diferença entre o sinal de controle em um tempo $t+k+1$ e um tempo $t+k$, pois dessa forma evita-se variações muito bruscas no sinal, o que na prática é inviável. É importante salientar que no processo de otimização o primeiro estado tem que ser o estado inicial, logo $x_t = x_0$.

Vale ressaltar que a notação $\|r\|_H^2$ quer dizer $r^T H r$, ou seja, a norma ao quadrado de um vetor ponderado por uma matriz H diagonal, que funciona como peso para este valor. Desta forma, consegue-se contabilizar maior peso para parcelas consideradas com maior relevância.

Assim sendo, pode-se dizer que o MPC tem quatro principais parâmetros de projeto: modelo de predição, função custo, horizonte de predição e restrições. Observa-se que para a definição do projeto em um *trade-off* processamento-acurácia, quanto mais complexo e rico for o modelo de predição e função custo, e maior a janela de predição, melhor será a acurácia dos resultados, porém maior o tempo de processamento será necessário, tornando muitas vezes inviável de utilização.

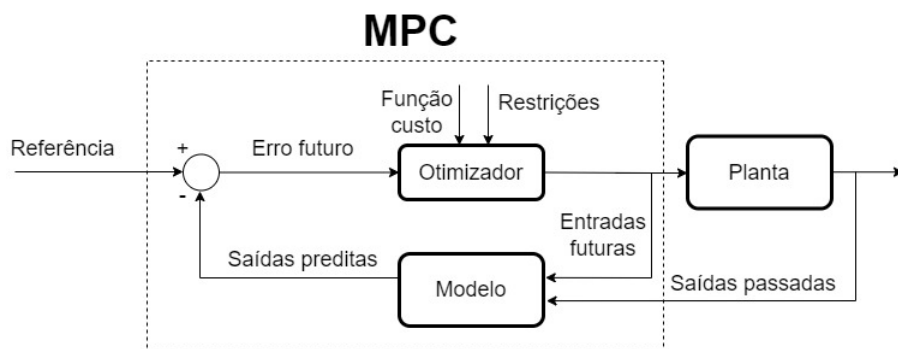


Figura 3 – Diagrama de blocos do MPC

De forma matemática, pode-se resumir o MPC ao seguinte modelo:

$$\begin{aligned}
u_{t:t+N} = \quad & \operatorname{argmin} \quad CF(u, x) \\
\text{sujeitos a} \quad & x(i+1) = f(u(i); x(i)) \\
& z(i+1) = h(x(i+1)) \\
& x(t) = x_0 \\
& u_{min} \leq u(i) \leq u_{max} \\
\text{onde} \quad & u_{t:t+N} = u(t : t + N)
\end{aligned} \tag{3.3}$$

onde CF é a função custo, f o modelo matemático, N a janela de predição, e $u_{min} \leq u(i) \leq u_{max}$ as restrições.

Existe ainda uma segunda abordagem, chamada solução por *multiple shooting* que utiliza como argumento a composição dos sinais de controle juntamente com os estados. Apesar de mais custoso computacionalmente, esta abordagem tem maior precisão e consegue levar em consideração os limites dos valores dos estados, e dessa forma será a utilizada no presente estudo. Resume-se esta abordagem no seguinte modelo matemático:

$$\begin{aligned}
[u_{t:t+N}, x_{t:t+N}] = \quad & \operatorname{argmin} \quad CF(u, x) \\
\text{sujeitos a} \quad & x(i+1) = f(u(i); x(i)) \\
& z(i+1) = h(x(i+1)) \\
& x(t) = x_0 \\
& u_{min} \leq u(i) \leq u_{max} \\
& x_{min} \leq x(i) \leq x_{max} \\
\text{onde} \quad & u_{t:t+N} = u(t : t + N) \\
& x_{t:t+N} = x(t : t + N)
\end{aligned} \tag{3.4}$$

onde $x_{min} \leq x(i) \leq x_{max}$ são as restrições dos estados.

3.3 Modelo Plano de Veículo Robótico com Esterçamento Diferencial

Define-se esterçamento diferencial como o esterçamento veicular proporcionado pela diferença de forças de tração produzidas no lado esquerdo e direito do veículo, podendo até ser alcançada a rotação deste sobre um eixo estacionário. Este tipo de esterçamento é classicamente usado em carros de combate sobre lagartas. Para o equacionamento de um robô com esterçamento diferencial, modelado como um veículo com um eixo único, para fins de simplificação, tem-se: o centro de massa no ponto médio do eixo e massa total igual a m_r , rodas com momento de inércia J_w , e robô com momento de inércia polar em relação ao eixo ortogonal ao solo I_z . Para o modelo plano, não será considerada a dinâmica vertical, e se tem por hipótese que não movimento transversal ao eixo do robô. A Figura 4 mostra o esquema do modelo.

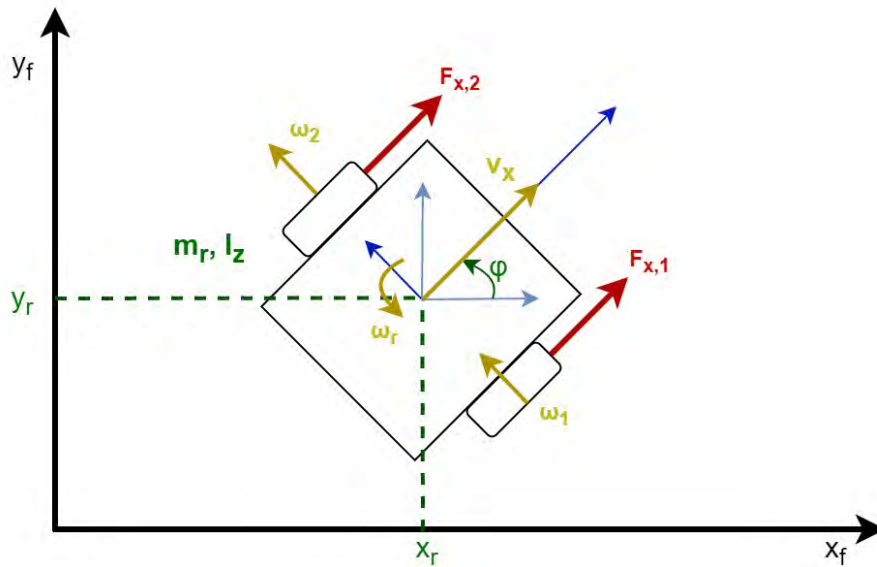


Figura 4 – Esquema de robô com esterçamento diferencial

O diagrama do corpo livre é mostrado na Figura 4. Como as forças desenvolvidas nos pneus são mais facilmente descritas no referencial do veículo, as equações de movimento do robô são definidas por meio das equações de Newton-Euler. No referencial do robô, para o centro de massa tem-se:

$$m_r \frac{d}{dt} v = \sum F \quad (3.5)$$

Daí, tem-se em um modelo plano de dinâmica longitudinal:

$$m_r \cdot \dot{v}_x = F_{x,1} + F_{x,2} \quad (3.6)$$

com v_x sendo a velocidade do centro de massa na direção longitudinal do robô, no referencial deste, e $F_{x,1}$ e $F_{x,2}$ as forças de tração nas rodas direita e esquerda, respectivamente.

Pela equação de Newton-Euler para o robô, tem-se para o movimento plano de rotação:

$$I_z \frac{d}{dt} \omega_r = \sum M_o \quad (3.7)$$

E então:

$$I_z \cdot \dot{\omega}_r = F_{x,1} \cdot \frac{L}{2} - F_{x,2} \cdot \frac{L}{2} \quad (3.8)$$

com L sendo a distância entre os centros das rodas.

Dessa forma, chega-se nas equações:

$$\dot{v}_x = \frac{F_{x,1}}{m_r} + \frac{F_{x,2}}{m_r} \quad (3.9)$$

$$\dot{\omega}_r = F_{x,1} \cdot \frac{L}{2I_z} - F_{x,2} \cdot \frac{L}{2I_z} \quad (3.10)$$

Além disso, pela própria cinemática do movimento mostrada na Figura 4 podemos dizer que:

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} v_x \cdot \cos(\phi) \\ v_x \cdot \sin(\phi) \\ \omega_r \end{bmatrix} \quad (3.11)$$

3.4 Modelo dinâmico das rodas

Pela lei de Euler para as rodas e segundo a Figura 5:

$$J_w \cdot \dot{\omega}_1 = T_1 - F_{x,1} \cdot r_w - M_{res,1} \quad (3.12)$$

$$J_w \cdot \dot{\omega}_2 = T_2 - F_{x,2} \cdot r_w - M_{res,2} \quad (3.13)$$

com r_w o raio das rodas, $M_{res} = c_v \cdot \omega_w$, e c_v o coeficiente de atrito viscoso nos eixos destas (atribuindo uma dissipação de energia no eixo, comum em motores elétricos), T_1 e T_2 os torques dos motores aplicados nas rodas 1 e 2, respectivamente.

$$\dot{\omega}_1 = \frac{T_1}{J_w} - \frac{F_{x,1} \cdot r_w}{J_w} - \frac{c_v \cdot \omega_1}{J_w} \quad (3.14)$$

$$\dot{\omega}_2 = \frac{T_2}{J_w} - \frac{F_{x,2} \cdot r_w}{J_w} - \frac{c_v \cdot \omega_2}{J_w} \quad (3.15)$$

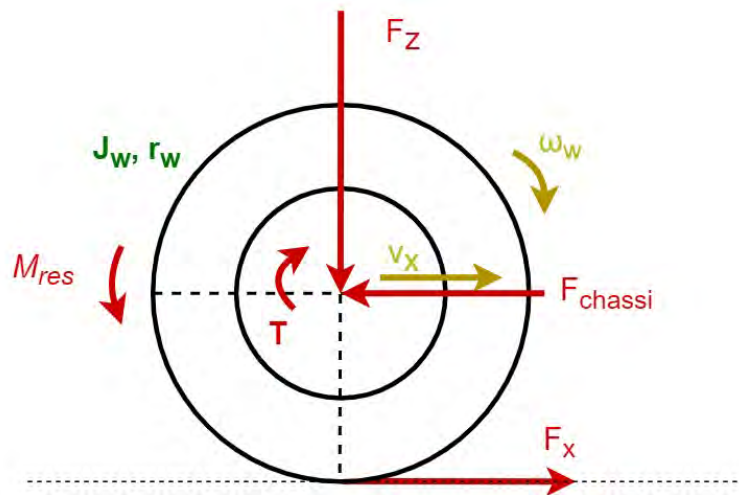


Figura 5 – Diagrama de corpo livre de uma roda

3.5 Modelo de atrito

Alguns modelos de atrito mais simples geralmente são utilizados para pneus mais rígidos. Um dos modelos de atrito mais conhecidos é o modelo de Coulomb. Neste modelo, a força de atrito é constante, oposta ao movimento, proporcional a força normal de contato e com constante de proporcionalidade μ . A Figura 6 mostra o comportamento do atrito de Coulomb, que é equacionada da seguinte forma:

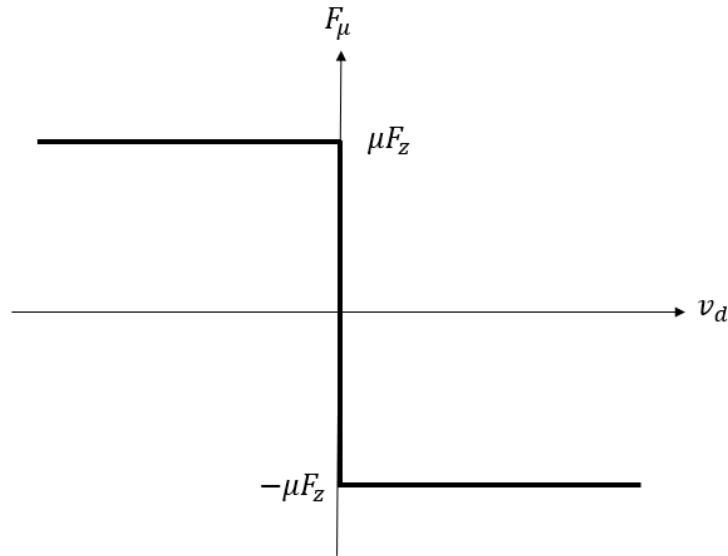


Figura 6 – Atrito de Coulomb

$$F_{\mu} = \begin{cases} -\mu F_z \cdot \text{sign}(v_d) & , \text{ para } v_d \neq 0 \\ -F_{ap} & , \text{ para } v_d = 0 \end{cases} \quad (3.16)$$

onde F_z representa o valor da força normal de contato na superfície, F_{ap} a força aplicada na condição estática, com $-\mu \cdot F_z \leq F_{ap} \leq \mu \cdot F_z$, v_d a velocidade de deslizamento entre as superfícies e $\text{sign}(v_d)$ o sinal desta velocidade.

Neste trabalho, para evitar descontinuidades no modelo e amortecer oscilações, utiliza-se uma aproximação linear em torno de $v_d = 0$. Dessa forma, na simulação estipula-se um pequeno intervalo de velocidades relativas, $-v_{d,max} < v_d < +v_{d,max}$, onde é realizada esta aproximação linear da força de atrito conforme a Figura 7 mostra.

Desse modo, com a aproximação tem a força de atrito dada:

$$F_{\mu} = \begin{cases} -\mu F_z \cdot v_d / v_{d,max} & , \text{ para } v_d \in (-v_{d,max}, +v_{d,max}) \\ -\mu F_z \cdot \text{sign}(v_d) & , \text{ para } v_d \notin (-v_{d,max}, +v_{d,max}) \end{cases} \quad (3.17)$$

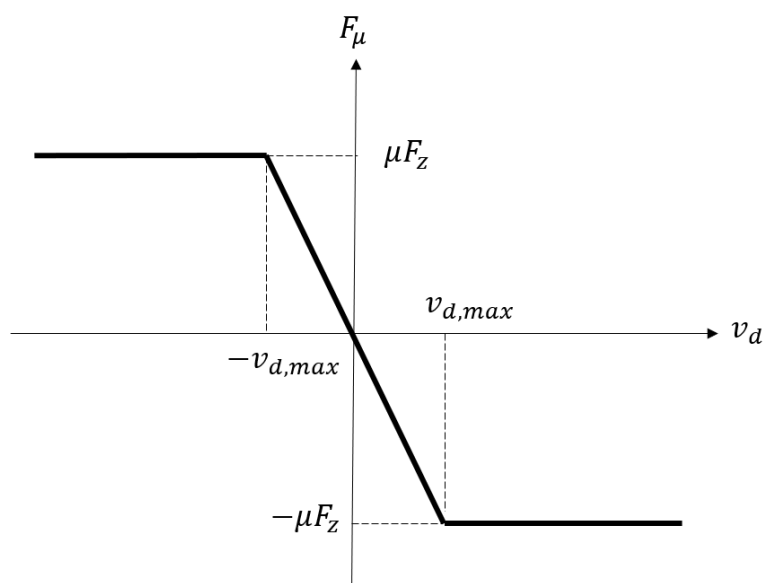


Figura 7 – Atrito de Coulomb com aproximação linear

4 METODOLOGIA

Foram feitas simulações do algoritmo de MPC utilizando o software MatLab[®] onde o algoritmo foi implementado a partir da *toolbox* CasADi. Modelou-se matematicamente a dinâmica do *rover* mostrado na Figura 8, usando-o como objeto de análise.



Figura 8 – Rover Lynxmotion Aluminum 4WD1

Para isso, confrontando os dados do robô com o modelo de veículo com esterçamento diferencial mostrado na fundamentação teórica, tem-se os seguintes parâmetros físicos, mostrados na Tabela 1:

Tabela 1 – Parâmetros físicos (27)

Parâmetro	Símbolo	Valor	Unidade
Massa	m_r	4,0	kg
Gravidade	g	9,81	N/kg
Momento de inercia do robô	I_z	0,0338	$kg \cdot m^2$
Distância entre as rodas	L	0,2735	m
Coefficiente de atrito pneu solo	μ	0,7	–
Raio do pneu	r_w	0,06	m
Momento de inercia polar da roda	J_w	$3,31 \cdot 10^{-5}$	$kg \cdot m^2$
Coefficiente de atrito visco	c_v	0,02	Nms/rad

Além disso, levando em consideração as limitações físicas do *rover*, tem-se os seguintes limites para os estados, mostrados na Tabela 2:

Para analisar o desempenho do MPC dividiu-se o trabalho em 3 estudos de caso: no primeiro fez-se o controle do *rover* usando um PID simples; no segundo usando apenas o MPC; e no terceiro fez-se um controle em cascata, onde o controle do chassi era feito pelo MPC e o das rodas por um PID.

Tabela 2 – Limites físicos para as simulações

Entidade	Valor mínimo	Valor máximo
v_x	-0,9	0,9
ω_r	-9,1	9,1
ω_1 e ω_2	-20,9	20,9
T_1 e T_2	-0,45	0,45

4.1 Trajetórias e avaliação

Para os três estudos de caso simulou-se duas trajetórias: na primeira há duas curvas circulares, simulando uma troca de faixa; na segunda a troca de faixa é feita de forma abrupta, de modo que a trajetória é composta por três segmentos de reta. As duas trajetórias são mostradas na Figura 9.

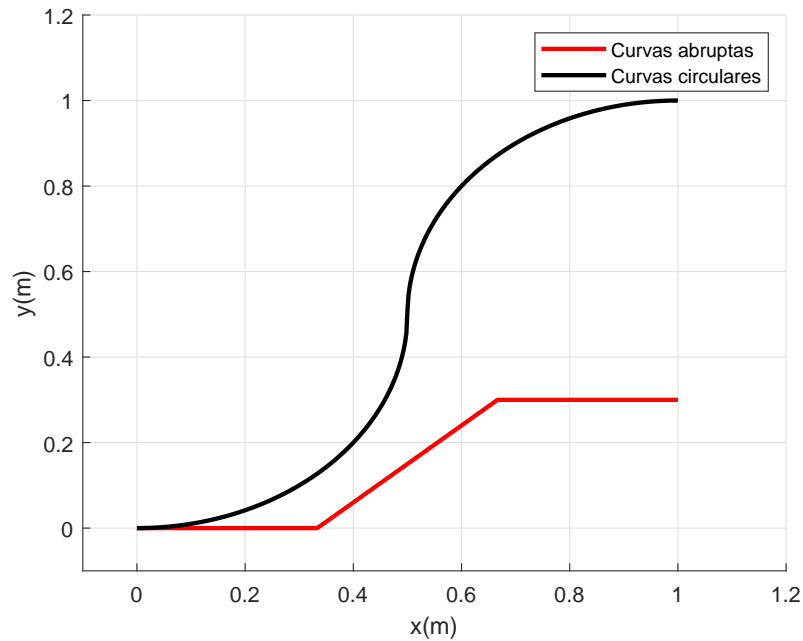


Figura 9 – Trajetórias de referência

Para fazer as análises de precisão alcançada pelo controle, ou seja, quão bem o controlador conseguiu fazer as saídas se aproximarem das referências, usou-se a raiz do erro médio quadrático (RMSE), dada pela Equação 4.1.

$$RMSEz = \sqrt{\frac{\sum_{i=1}^n (r(i) - z(i))^2}{n}} \quad (4.1)$$

onde o subíndice i refere-se à iteração i , e n é o número total de iterações que foram feitas na simulação. Após gerar os RMSE das saídas, divide-se este valor por um valor nominal da referência daquela saída para se ter uma noção percentual do erro. Dessa forma:

$$erro(\%)z = 100 \frac{RMSEz}{z_{nominal}} \quad (4.2)$$

Para a escolha dos valores nominais utilizou-se os valores máximos alcançados para cada uma das saídas, para as duas trajetórias, com exceção do valor de para ω_r na trajetória com curvas abruptas, onde considerou-se o valor físico máximo de 9,1 como nominal. A Tabela 3 mostra os valores escolhidos.

Tabela 3 – Valores nominais das saídas para as trajetórias

Valores nominais	x_r	y_r	ϕ	v_x	ω_r
Circular	1,0	1,0	1,571	0,524	1,0
Abrupto	1,0	0,3	0,983	0,45	9,1

4.2 Geração dos dados de referência

Para a geração dos dados de referência para o controlador define-se inicialmente um vetor de tempo, a partir do tempo final para simulação e do intervalo de tempo entre cada iteração. A partir disso, para cada iteração é definida uma posição (x_{ref} e y_{ref}). Como deve-se definir as referências para velocidade longitudinal e velocidade angular do robô para cada iteração, usando a Figura 10, que mostra 3 iterações consecutivas, e a Figura 11, que mostra o triângulo de composição de posições, são geradas as referências de ângulo de esterçamento, taxa de *yaw* e velocidade longitudinal, pelas Equações 4.3, 4.4 e 4.5 respectivamente.

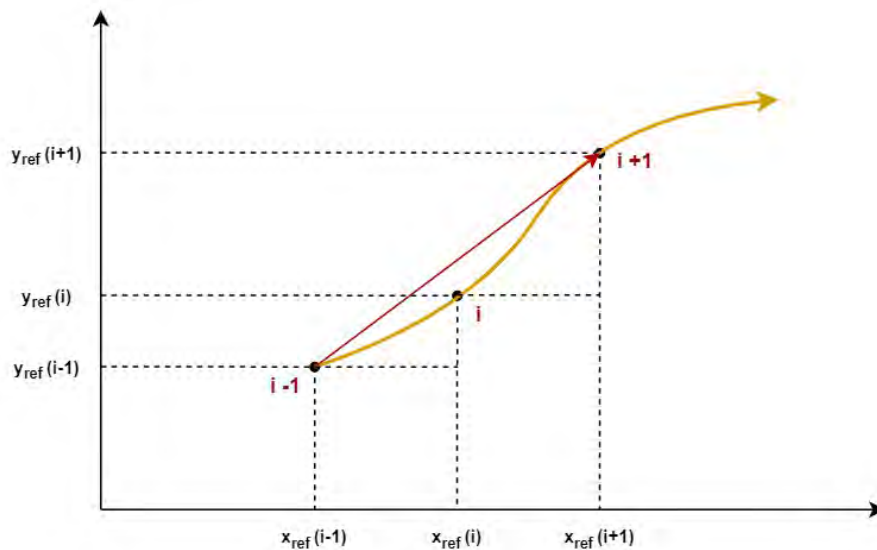


Figura 10 – Iterações da posição do robô

$$\phi_{ref}(i) = \tan^{-1} \left(\frac{\Delta y(i+1, i-1)}{\Delta x(i+1, i-1)} \right) \quad (4.3)$$

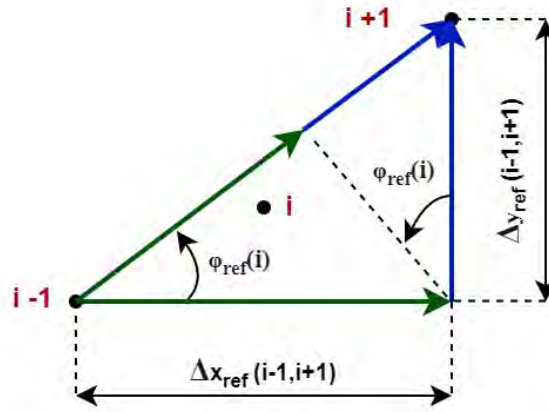


Figura 11 – Composição de posições

$$\omega_{r,ref}(i) = \frac{\phi_{ref}(i) - \phi_{ref}(i-1)}{dt} \quad (4.4)$$

$$v_{x,ref}(i) = \frac{\Delta y(i+1, i-1) \cdot \sin(\phi_{ref}(i)) + \Delta x(i+1, i-1) \cdot \cos(\phi_{ref}(i))}{2 \cdot dt} \quad (4.5)$$

com,

$$\Delta y(i+1, i-1) = y_{ref}(i+1) - y_{ref}(i-1) \quad (4.6)$$

$$\Delta x(i+1, i-1) = x_{ref}(i+1) - x_{ref}(i-1) \quad (4.7)$$

4.3 Calibração do PID

A calibração do PID é feita por tentativa e erro, observando-se os resultados, RMSE e gráficos, enquanto se varia o k_P , k_I e k_D . Para calibrar os controles PID o seguinte procedimento foi adotado:

1. inicialmente faz-se $k_I = 0$ e $k_D = 0$ para analisar os efeitos da variação de k_P na resposta do sistema;
2. define-se o k_P a partir das análises;
3. varia-se o valor de k_I enquanto k_D permanece nulo;
4. define-se um k_I que elimine o erro em regime permanente.
5. varia-se o k_D definindo um que melhore a resposta transiente e reduza o *overshoot* da resposta.

4.4 Calibração do MPC

Para se obter bons resultados no MPC deve-se calibrar três parâmetros da estratégia de controle: a matriz de peso da variação de sinais de entrada (S), a matriz de peso das

saídas (Q) e o tamanho da janela de amostragem (N). Para avaliá-los utiliza-se o RMSE de cinco estados (x_r, y_r, ϕ, v_x e ω_r) e o tempo de processamento.

Define-se inicialmente as matrizes Q e S como uma matriz identidade vezes uma potência de 10, ou seja, $Q = 10^q \cdot I$ e $S = 10^s \cdot I$, dessa forma a calibração é feita nos valores q, s e N . Escolhe-se um conjunto de valores de cada parâmetro a serem testados e seus resultados analisados. Os valores definidos para teste foram:

$$q = [-2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8] \quad (4.8)$$

$$s = [-2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8] \quad (4.9)$$

$$N = [3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad 9 \quad 10 \quad 12 \quad 14 \quad 16 \quad 18 \quad 20 \quad 25 \quad 30 \quad 35] \quad (4.10)$$

Para cada valor de parâmetro calcula-se os valores de RMSE ditos e o tempo de simulação. Depois disso os valores para um mesmo critério de análise são normalizados pelo maior valor, então os resultados são analisados em valores de 0 a 1. Em seguida analisa-se as tendências dos resultados e define-se os valores dos parâmetros de forma a minimizar os erros e o tempo de simulação.

Deve-se lembrar que o valor q é o valor da potência de 10 que multiplica a matriz identidade para os pesos das saídas, ou seja, q reflete a importância que está sendo dada para as saídas (z) na hora de computar a função custo. Como deseja-se levar mais em consideração as saídas do que a variação do sinal de entrada, que reflete a "inércia" do sinal de controle impedindo mudanças grandes e bruscas deste, deseja-se sempre $s \leq q$.

Além disso, para a escolha de N deve-se levar em conta que, como uma N maior quer dizer uma janela de predição maior, naturalmente o controle levará em conta mais dados da curva olhando para o futuro, dessa forma em casos de mudança de direção o erro tende a aumentar suavemente, já que o controlador enxergará a curva antes e aplicará a ação de controle correspondente. Isso, porém, não quer dizer um pior sistema de controle, mas sim consequência natural da estratégia de controle. Com o aumento do N o tempo de processamento tende a aumentar, pois mais colunas estão sendo adicionadas para o cálculo da função custo. Desta forma, deve-se procurar valores médios de N que não aumentem muito o tempo de processamento e o erro, mas que não sejam tão pequenos que o sistema perca a capacidade preditiva.

5 ESTUDO DE CASO 1: CONTROLE PID

5.1 Estratégia de controle

No primeiro estudo de caso, fez-se a simulação do *rover* completo utilizando um controle PID. Dessa forma, define-se os seguintes estados e entradas do controlador:

$$x = [x_r \quad y_r \quad \phi \quad v_x \quad \omega_r \quad \omega_1 \quad \omega_2]^T \quad (5.1)$$

$$u = [T_1 \quad T_2]^T \quad (5.2)$$

Para as saídas, precisa-se de um conjunto de saídas que relacionem os estados e consigam gerar referências para as entradas de torque em cada uma das rodas. Dessa forma, as saídas do sistema são definidas como as velocidades dos centros das rodas do robô, pois se esta velocidade está menor que a referência, mais torque é requerido. Então:

$$z = [v_{x,roda1} \quad v_{x,roda2}]^T \quad (5.3)$$

pela cinemática do robô, tem-se:

$$v_{x,roda1} = v_x + \omega_r \frac{L}{2} \quad (5.4)$$

$$v_{x,roda2} = v_x - \omega_r \frac{L}{2} \quad (5.5)$$

5.2 Calibração do PID

Seguindo a metodologia de calibração proposta, os valores dos ganhos foram definidos e são mostrados na Tabela 4.

Tabela 4 – Ganhos do PID para o EC1

Ganho	Valor
k_P	10
k_I	20
k_D	0,05

5.3 Resultados

Feita a calibração, seguiu-se com as simulações das duas trajetórias propostas na metodologia. Os resultados das simulações para a posição ($x_r \times y_r$), velocidade longitudinal (v_x), velocidade angular (ω_r), forças de tração (F_x), velocidades angulares das rodas (ω_1 e ω_2) e torques nas rodas (T) são mostrados para as curvas circulares, Figura 12, e curvas abruptas, Figura 13. A Tabela 5 mostra os parâmetros de desempenho de cada uma das simulações para o estudo de caso 1.

Tabela 5 – EC1: parâmetros de desempenho

Curva	RMSE x_r	RMSE y_r	RMSE ϕ	RMSE v_x	RMSE ω_r	$\Delta t_m [s]$
Circular	0,0091	0,0170	0,0361	0,0541	0,1957	3,87e-5
Brusca	0,0395	0,0667	0,2850	0,0286	42,8121	4,48e-5

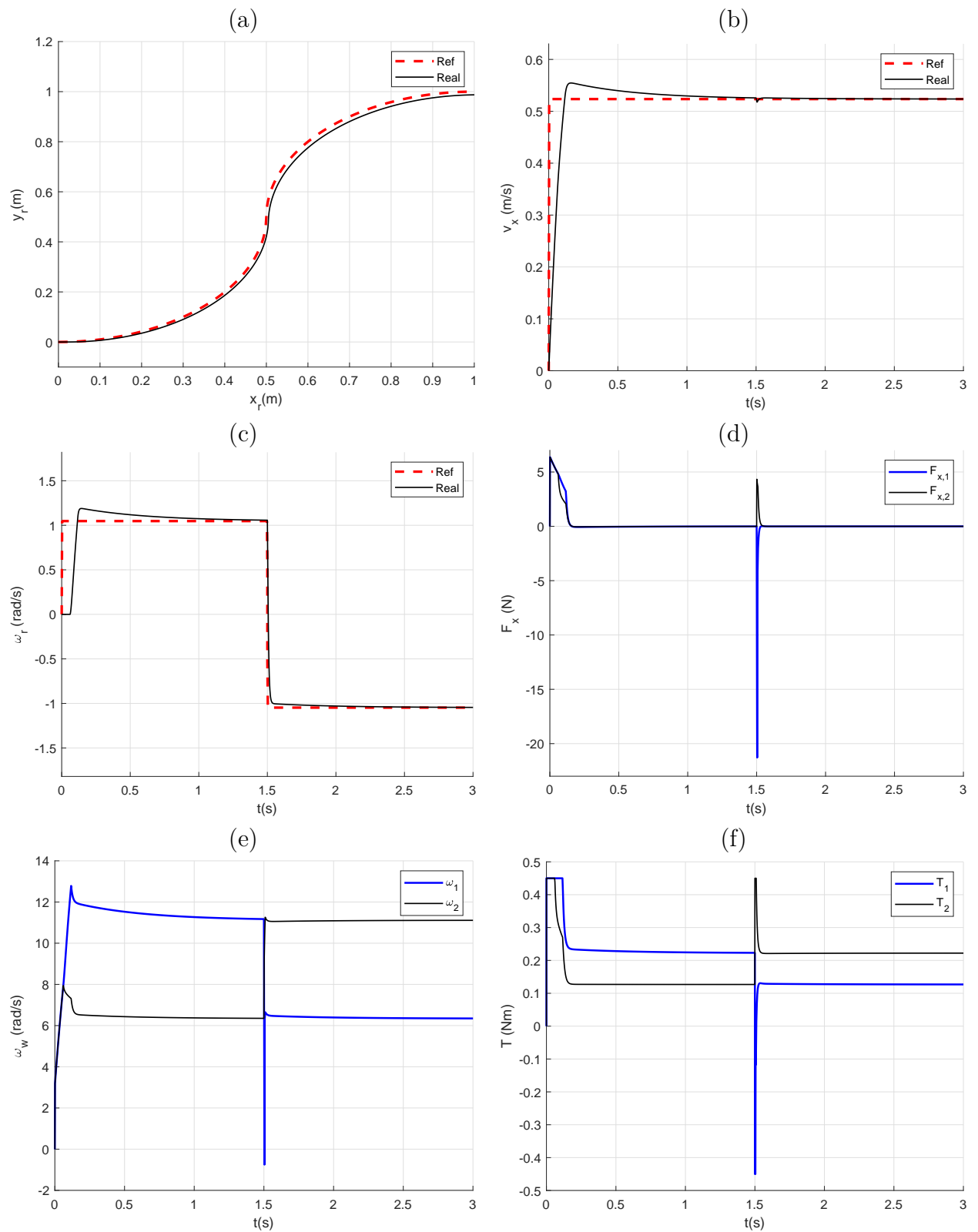


Figura 12 – EC1 - curvas circulares: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_w (f) T .

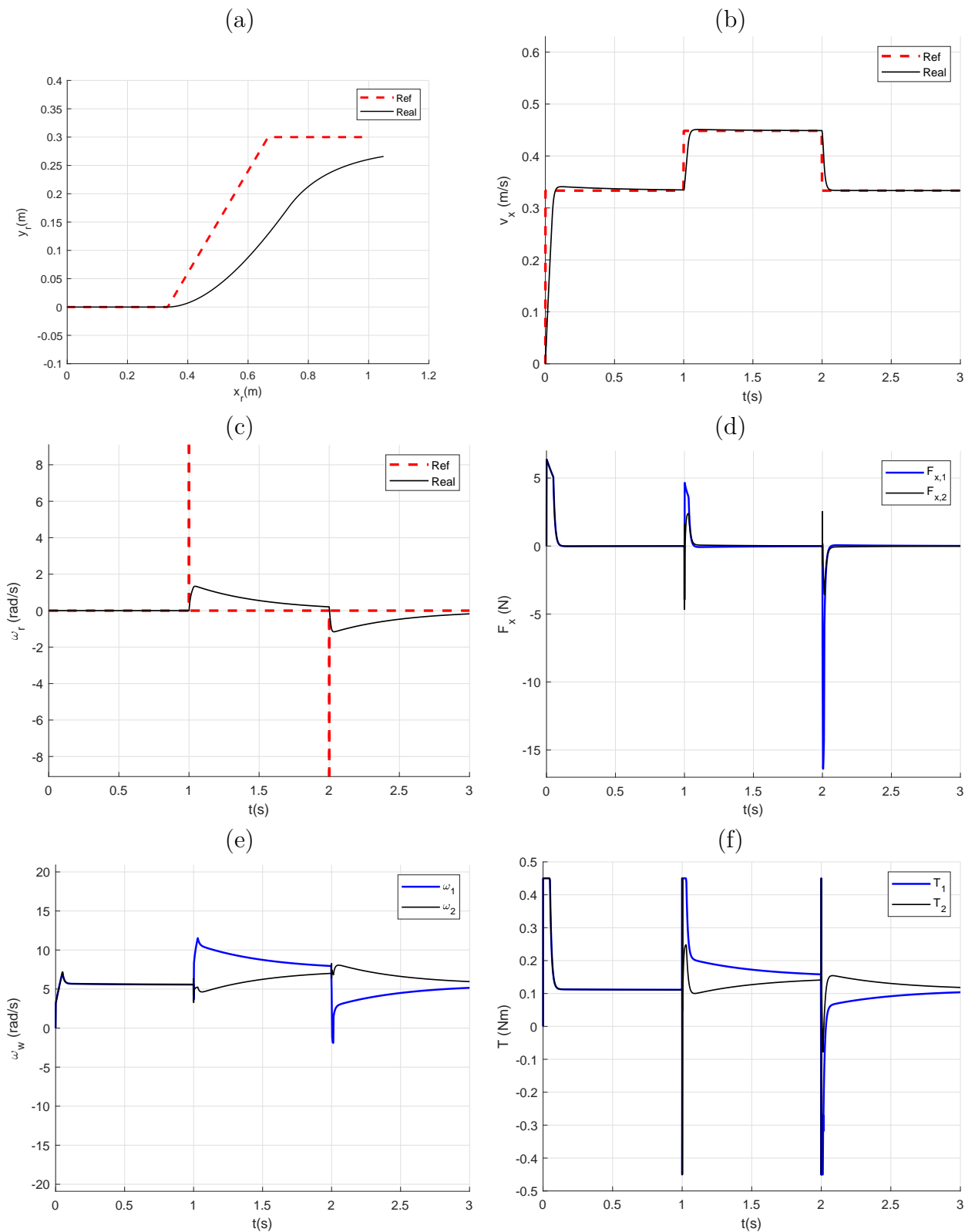


Figura 13 – EC1 - curvas abruptas: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_w (f) T .

6 ESTUDO DE CASO 2: CONTROLE PREDITIVO

6.1 Estratégia de controle

Para o segundo estudo, fez-se a simulação do *rover* completo utilizando o MPC. Dessa forma, define-se os seguintes estados, saídas e entradas do controlador:

$$x = [x_r \quad y_r \quad \phi \quad v_x \quad \omega_r \quad \omega_1 \quad \omega_2]^T \quad (6.1)$$

$$z = [x_r \quad y_r \quad \phi \quad v_x \quad \omega_r]^T \quad (6.2)$$

$$u = [T_1 \quad T_2]^T \quad (6.3)$$

Diferentemente o estudo de caso 1, aqui considera-se que todos os estados, menos as velocidades angulares das rodas, são saídas mensuráveis ou estimáveis.

6.2 Calibração do MPC

A calibração para o MPC foi feita seguindo o procedimento proposto, com especial atenção às observações finais descritas. Os resultados para cada parâmetro são mostrados na Figura 14. Dessa forma, os valores de parâmetros escolhidos são:

Tabela 6 – Parâmetros do MPC para o EC2

Parâmetro	Valor
q	5
s	3
N	12

6.3 Resultados

Após calibrado, as duas trajetórias propostas foram simuladas e analisadas. As Figuras 15 e 16 mostram os resultados das simulações para curvas circulares e abruptas, respectivamente, bem como a Tabela 7 mostra os parâmetros de desempenho analisados.

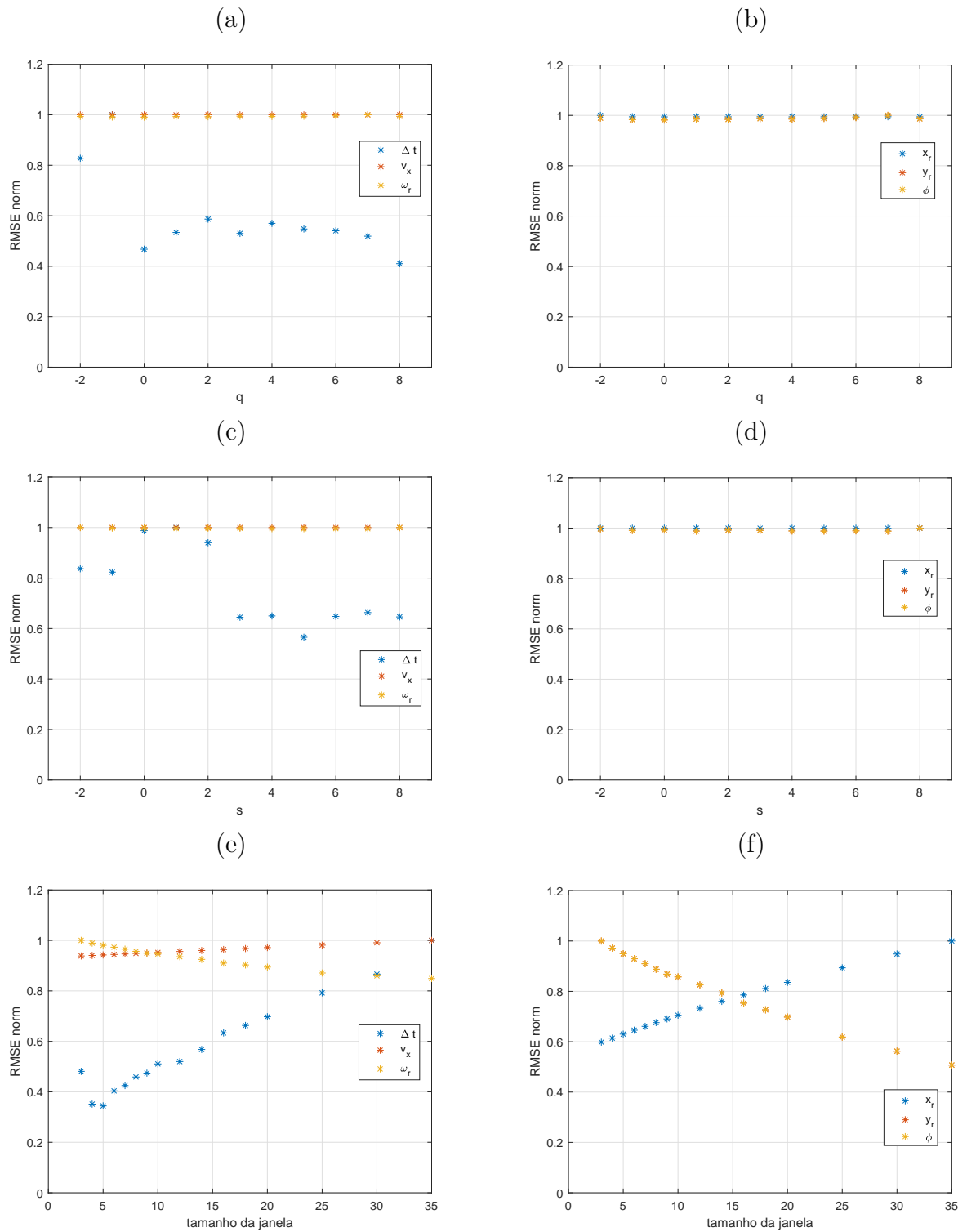


Figura 14 – (a) e (b) dados para s . (c) e (d) dados para q . (e) e (f) dados para N .

Tabela 7 – EC2: parâmetros de desempenho

Curva	RMSE x_r	RMSE y_r	RMSE ϕ	RMSE v_x	RMSE ω_r	$\Delta t_m [s]$
Circular	0,0220	0,0019	0,0056	0,0464	0,0718	0,0193
Brusca	0,0724	0,1997	0,4228	0,0181	42,81	0,0207

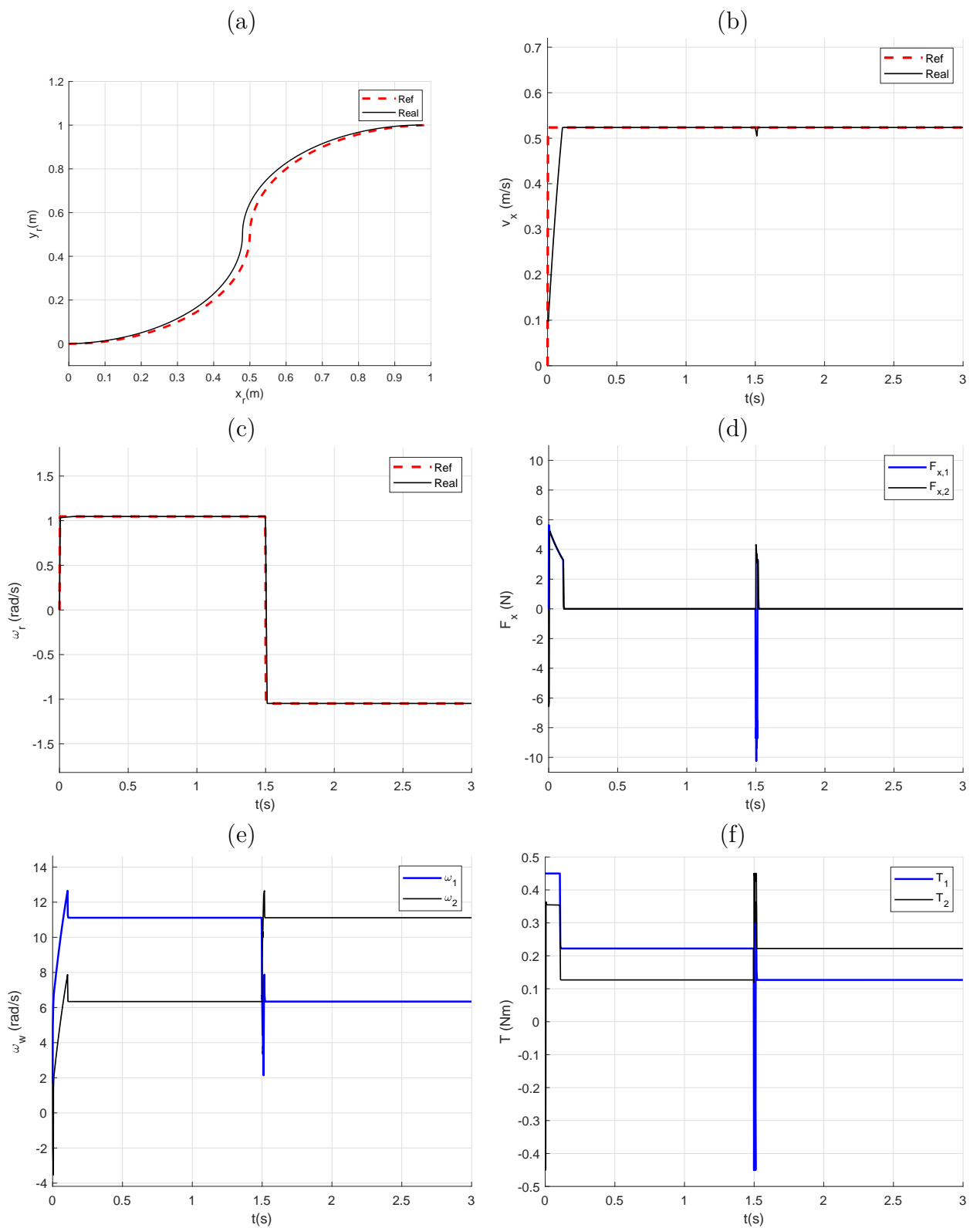


Figura 15 – EC2 - curvas circulares: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T .

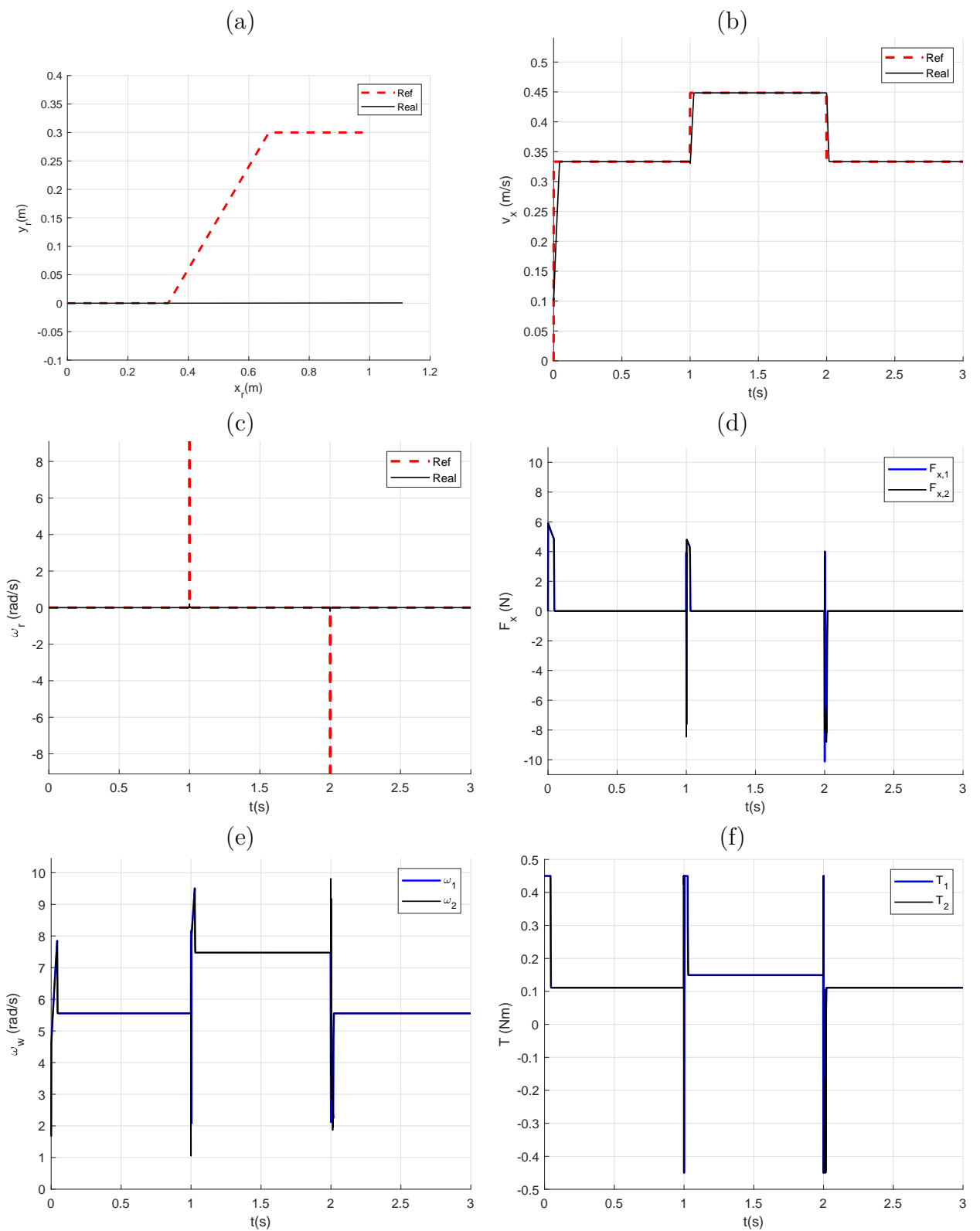


Figura 16 – EC2 - curvas abruptas: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_w (f) T .

7 ESTUDO DE CASO 3: CONTROLE EM CASCATA

7.1 Estratégia de controle

Para o terceiro estudo de caso, fez-se um controle em cascata. Para este caso as referências ($r(t)$) são as entradas do MPC, juntamente com a saída da planta ($z(t)$). A saída do MPC é uma entrada de referência ($u_r(t)$) a qual, sob uma relação estática, gera estados secundários de referência ($o_r(t)$) que, subtraídos dos estados secundários reais ($o(t)$) calculados na saída da planta, dão o sinal de entrada da planta. O diagrama de blocos do controle em cascata é mostrado na Figura 17.

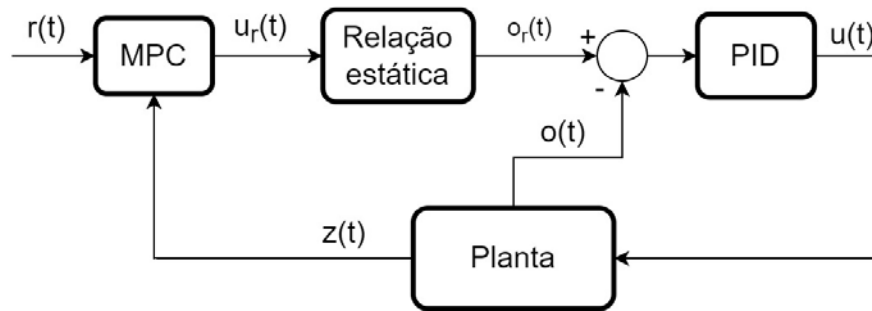


Figura 17 – Diagrama de blocos do controle em cascata

Dessa forma, define-se os estados para o controle do MPC pela Equação 7.1, representando a dinâmica do chassi do robô, devolvendo como entrada de referência as forças no chassi, como mostrado na Equação 7.2.

$$x_{mpc} = [x_r \quad y_r \quad \phi \quad v_x \quad \omega_r]^T \quad (7.1)$$

$$u_r = [F_{x,1}^r \quad F_{x,2}^r]^T \quad (7.2)$$

Em seguida utiliza-se a relação estática entre a força de tração e o deslizamento da roda, mostrada na Equação 3.17, para gerar os valores estados secundários de referência, que são os deslizamentos nas rodas, como mostra a Equação 7.3. Dessa forma a relação estática é dada pela Equação 7.4.

$$o_r = [v_{d,1}^r \quad v_{d,2}^r]^T \quad (7.3)$$

$$v_d^r = \begin{cases} -v_{d,max} \cdot F_x^r / (\mu F_z) & , \text{ para } F_x^r \in (-\mu F_z, +\mu F_z) \\ -v_{d,max} \cdot \text{sign}(F_x^r) & , \text{ para } F_x^r \notin (-\mu F_z, +\mu F_z) \end{cases} \quad (7.4)$$

Em seguida, os deslizamentos de referência são comparados com os deslizamentos reais, mostrados na Equação 7.5, dados pela saída da planta. Estes estados secundários são calculados a partir dos estados do PID, mostrados na Equação 7.6, e são dados pelas Equações 7.7 e 7.8.

$$o = [v_{d,1} \quad v_{d,2}]^T \quad (7.5)$$

$$x_{pid} = [x_r \quad y_r \quad \phi \quad v_x \quad \omega_r \quad \omega_1 \quad \omega_2]^T \quad (7.6)$$

$$v_{d,1} = \left(v_x + \omega_r \frac{L}{2} \right) - \omega_1 \cdot r_w \quad (7.7)$$

$$v_{d,2} = \left(v_x - \omega_r \frac{L}{2} \right) - \omega_2 \cdot r_w \quad (7.8)$$

Por fim, o PID calcula o sinal de controle da planta, dado pela Equação 7.9, e aplica nesta, resultando nas saídas da Equação 7.10.

$$u = [T_1 \quad T_2]^T \quad (7.9)$$

$$z = [x_r \quad y_r \quad \phi \quad v_x \quad \omega_r]^T \quad (7.10)$$

7.2 Calibração

A calibração para o controle em cascata foi dividida em duas partes, primeiramente fez-se a calibração do MPC referente à dinâmica do chassi, e depois foi feita a calibração do PID referente a dinâmica do robô completo.

- Calibração do MPC: dinâmica do chassi

O procedimento de calibração do MPC foi seguido como descrito e os resultados são mostrados na Figura 18. Dessa forma, a Tabela 8 mostra os valores de parâmetros escolhidos.

Tabela 8 – Parâmetros do MPC para o EC3

Parâmetro	Valor
q	6
s	3
N	10

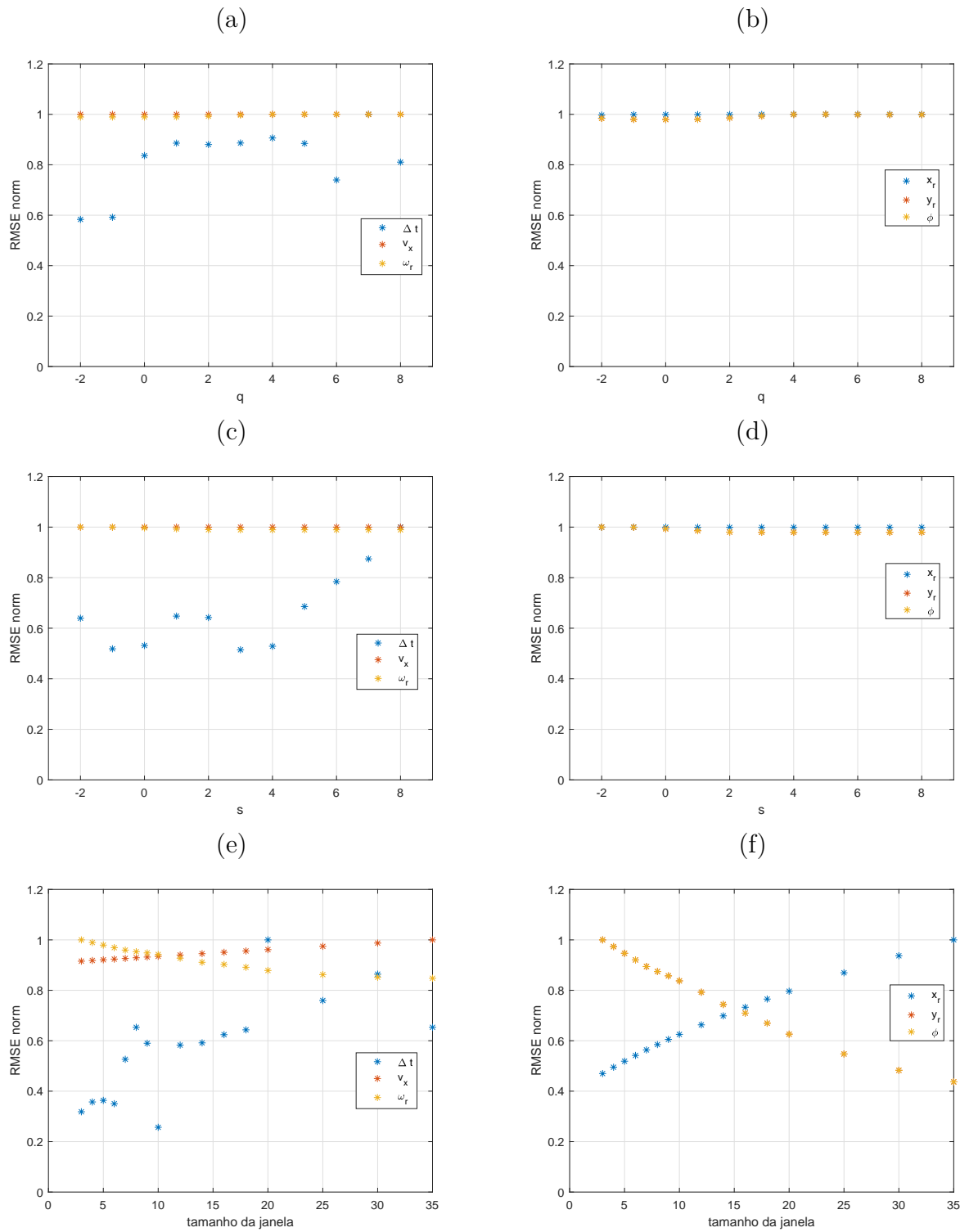


Figura 18 – (a) e (b) dados para s . (c) e (d) dados para q . (e) e (f) dados para N .

- Calibração do PID: dinâmica do robô completo

Em seguida a calibração do PID foi feita também como mostrada na metodologia. Os valores escolhidos para os ganhos são mostrados na Tabela 9.

Tabela 9 – Ganhos do PID para o EC3

Ganho	Valor
k_P	5
k_I	5
k_D	$3 \cdot 10^{-4}$

7.3 Resultados

Calibrados os parâmetros de projeto para o controle, as trajetórias foram simuladas em ambiente computacional. Os resultados são apresentados nas Figuras 19 e 20, bem como os parâmetros de desempenho são destacados na Tabela 10.

Tabela 10 – EC3: parâmetros de desempenho

Curva	RMSE x_r	RMSE y_r	RMSE ϕ	RMSE v_x	RMSE ω_r	$\Delta t_m [s]$
Circular	0,0239	0,0032	0,0087	0,0498	0,0830	0,0067
Brusca	0,0714	0,1998	0,4229	0,0202	42,81	0,0066

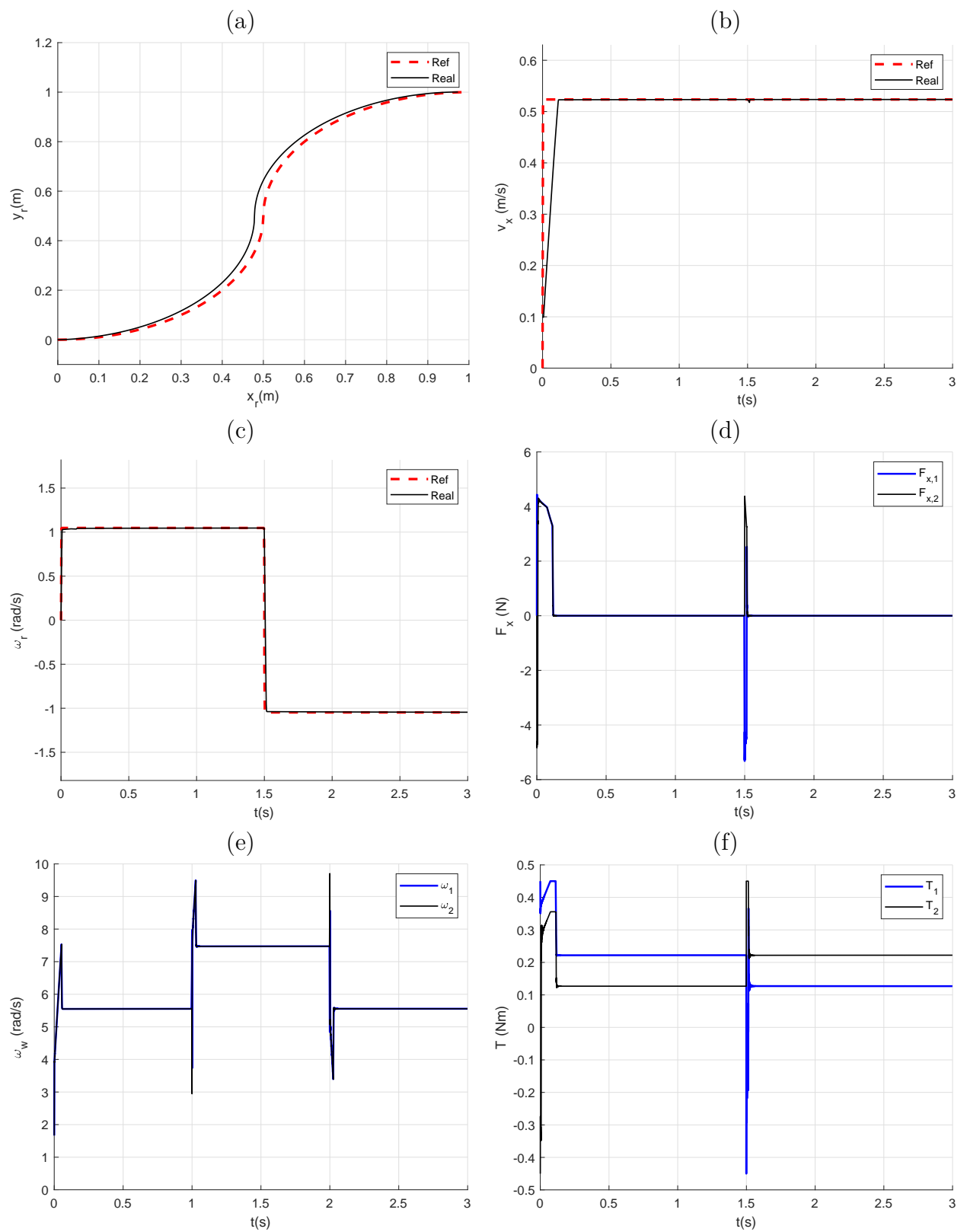


Figura 19 – EC3 - curvas circulares: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_r (f) T .

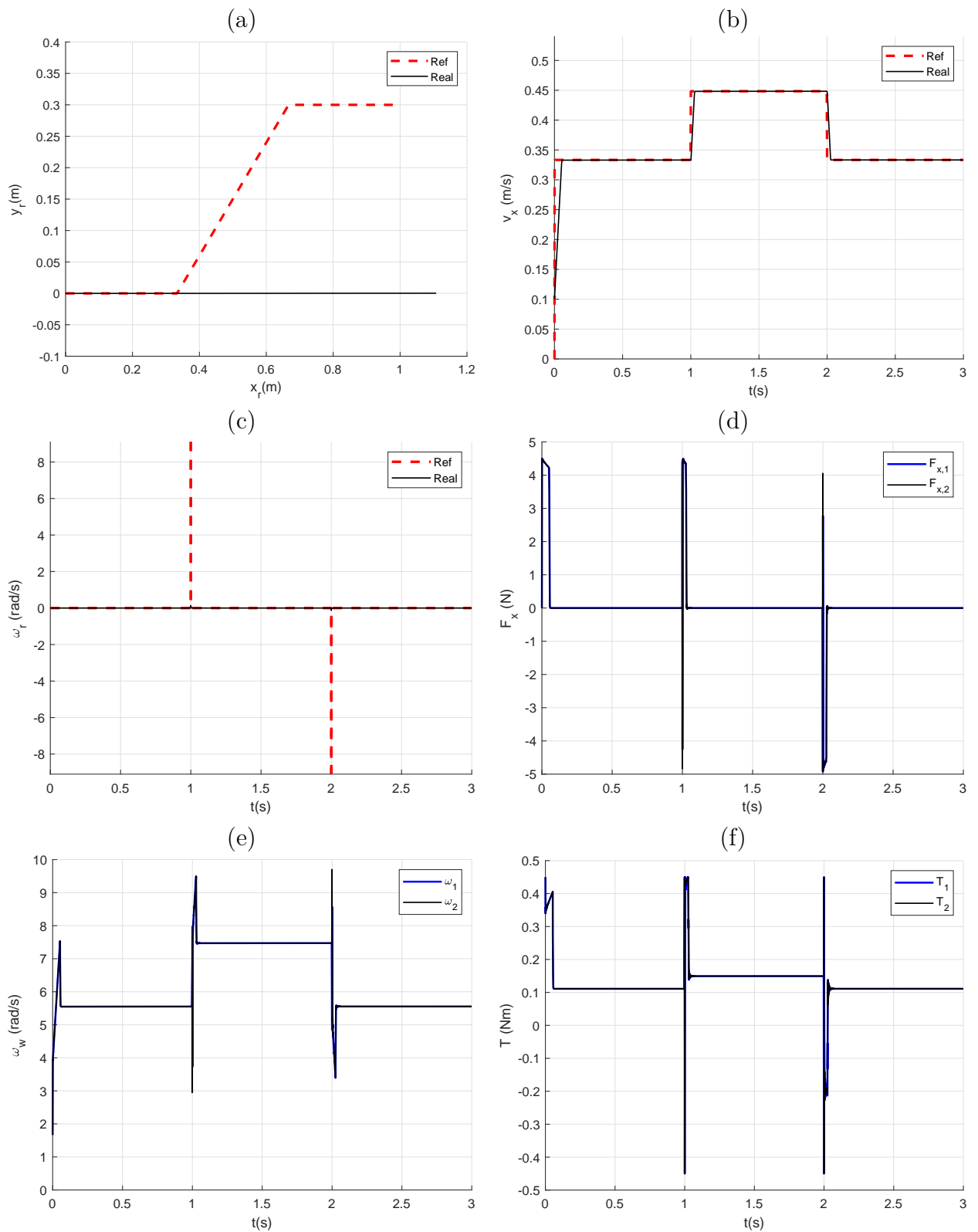


Figura 20 – EC3 - curvas abruptas: (a) x_r e y_r (b) v_x (c) ω_r (d) F_x (e) ω_w (f) T .

8 ANÁLISE COMPARATIVA DE RESULTADOS

A precisão e o tempo de simulação foram os dois principais aspectos analisados para comparar os resultados dos três estudos de caso.

8.1 Precisão

As Tabelas 11 e 12 mostram os erros percentuais calculados a partir das Tabelas 5, 7 e 10 pela Equação 4.2 com os valores nominais da Tabela 3.

Tabela 11 – Erros percentuais para simulações das curvas circulares

Erro(%)	x_r	y_r	ϕ	v_x	ω_r
EC1	0,91	1,70	2,30	10,3	19,6
EC2	2,20	0,19	0,36	8,85	7,18
EC3	2,39	0,32	0,55	9,50	8,30

Tabela 12 – Erros percentuais para simulações das curvas abruptas

Erro(%)	x_r	y_r	ϕ	v_x	ω_r
EC1	3,95	22,2	29,0	6,36	470
EC2	7,24	66,6	43,0	4,02	470
EC3	7,14	66,6	43,0	4,49	470

Primeiramente percebe-se, pelas Figuras 13 (a), 16 (a) e 20 (a), referentes as curvas abruptas, que nenhum dos controladores conseguiu fazer o seguimento de trajetória. Para os controladores que tinham MPC, isso indica que na matriz de pesos das saídas (Q), o peso referente as posições x_r , y_r e ϕ não está grande o suficiente para fazer o controlador aplicar uma ação de controle visando retornar o robô para a sua trajetória. Além disso, para o PID isso deve-se ao fato de a entrada de controle depender das saídas em apenas uma iteração, sendo que na trajetória com curva abrupta o ω_r variou muito em um intervalo extremamente pequeno, logo o PID não conseguiu captar a mudança. Isso fica claro quando se analisa o erro percentual de y_r e ϕ na Tabela 12, onde aqueles alcançam valores da ordem das dezenas na porcentagem. Desse modo, as análises seguintes se baseiam nos valores dos erros referentes a v_x e a ω_r .

Em termos práticos é mais coerente pensar no controle baseado em referências em v_x e ω_r apenas, pois microcontroladores de *rovers* como o aqui estudado costumam ter estes dois parâmetros como entrada de referência. Para adicionar-se a posição, deve-se ter algum tipo de sensoriamento que em tempo real consiga fazer esta medição, como um GPS. Isto, porém, dificulta a implementação e encarece o sensoriamento.

Na Tabela 12 desconsidera-se os valores de erro percentual para ω_r . Os valores alcançados por este erro estão na casa das centenas, ou seja, o erro ultrapassa em algumas vezes o valor máximo possível para a saída. Isso deve-se à descontinuidade imposta para a velocidade angular em um intervalo de tempo extremamente curto, o que é fisicamente impossível. Desta forma esses valores não fornecem informações coerentes, sendo assim, desconsiderados.

Para os valores de erro percentual de v_x e ω_r mostrados na Tabela 11, e v_x mostrado na Tabela 12, percebe-se uma tendência já esperada: a precisão do controlador MPC completo (EC2) é a maior de todas, sendo seguida pela do controle em cascata (EC3), e terminado com a menor precisão que é a do controle PID clássico (EC1).

Para a v_x nas curvas circulares o controle em cascata teve uma precisão $9,50 - 8,85 = 0,65\%$ menor que o MPC completo, e teve $10,3 - 9,50 = 0,8\%$ a mais de precisão que o PID. Isto mostra um ganho sensível, se for considerada a diferença de precisão do PID clássico para o MPC completo. Analisando ω_r , isto fica ainda mais evidente, já que houve um ganho de $19,6 - 8,3 = 11,3\%$ em precisão do controle em cascata em relação ao PID, e aquele ficou apenas $8,3 - 7,18 = 1,12\%$ menos preciso que o MPC completo. Este padrão também pode ser visto em v_x nas curvas abruptas, onde tem-se $6,36 - 4,49 = 1,87\%$ a mais de precisão do controle em cascata em relação ao PID, e apenas $4,49 - 4,02 = 0,47\%$ a menos que o MPC completo. Portanto, percebe-se que houve um ganho sensível na precisão do controle utilizando um controle em cascata MPC combinado com PID em relação ao PID, e este ganho foi ainda maior se considerar o MPC completo em relação ao PID.

Ressalta-se aqui a característica de suavização proporcionada pelo MPC. Analisando as Figuras 13 (c), (e) e (f), percebe-se que após a descontinuidade dada pelas referências, um tempo considerável foi necessário para voltar a situação de estabilidade dos valores, o que fica claro pelos *overshoots* e pela forma, a princípio, caótica de como os valores se acomodam. Já nas Figuras 16 e 20 (c), (e) e (f), este processo acontece bem mais rapidamente e sem grandes perturbações. Entretanto, da forma como o PID do EC1 foi calibrado, conseguiu-se um tempo de subida razoavelmente pequeno, fazendo com que o *rover* tivesse um menor erro percentual na posição x_r , como fica elucidado nas Tabelas 11 e 12, onde o menor valor deste erro é para o EC1.

8.2 Tempo de processamento

Como esperado e como mostrado na Tabela 8.2, o tempo médio de cada iteração no PID foi extremamente pequeno, sendo da casa de 10^{-5} segundos, ao passo que os casos que usaram MPC tiveram significativo aumento no tempo de processamento. Ressalta-se, no entanto, que houve sensível melhora no tempo de processamento utilizando o controle

em cascata. Este fato evidencia-se nas razões de tempo médio das iterações para o MPC completo e o em cascata:

- Curvas circulares: $\frac{0,0193}{0,0067} = 2,88$
- Curvas abruptas: $\frac{0,0207}{0,0066} = 3,13$

Dessa forma, utilizando os dois controles em série conseguiu-se reduzir em cerca de um terço o tempo de processamento. Este parâmetro é especialmente importante para aplicações *on-line*, onde há a necessidade de rápido processamento para que a central eletrônica descida a próxima ação de controle

Tabela 13 – Tempo médio de cada iteração

Tempo [s]	EC1	EC2	EC3
Circular	$3,87 \cdot 10^{-5}$	0,0193	0,0067
Abrupto	$4,48 \cdot 10^{-5}$	0,0207	0,0066

9 CONCLUSÃO

Com avanço acelerado das tecnologias automotiva e de inteligência artificial, os sistemas de controle se tornaram proeminentes nesse cenário, especialmente no que tange veículos robóticos não-tripulados e autônomos. As aplicações destes são inúmeras, indo desde exploração espacial, passando por desativação de bombas e operações militares, até simples transportadores em linhas de fabricação. Dessa forma, o objetivo do presente trabalho foi estudar, implementar computacionalmente e analisar um algoritmo de controle chamado MPC.

Estudou-se a estratégia do MPC e a do controle PID, com especial ênfase na capacidade de otimização daquele. Utilizando um *rover* como modelo, a dinâmica de *skid-steering* foi modelada juntamente com o atrito de Coulomb linearizado. Após o robô ser devidamente modelado, três estudos de caso foram feitos: utilização do PID para o controle do robô completo (EC1), utilização do MPC para o controle do robô completo (EC2), e utilização de controle em cascata MPC combinado com PID para controle do robô (EC3).

Percebe-se que há um *trade-off* da precisão e do tempo a ser considerado na escolha dos sistemas de controle. Para a decisão do melhor controlador para um determinado equipamento ou processo, deve-se analisar qual o maior requisito, precisão ou tempo, ou o quanto que um pode ser negligenciado em favor do outro, dentro dos requisitos de projeto, tarefa a ser executada pelo engenheiro responsável. analisando o MPC e o PID, aquele mostrou considerável ganho em precisão proporcionado, e aquele grande redução de processamento. Mesmo assim, conseguiu-se juntar ambos os controladores e características intermediárias foram atingidas.

Analisadas a precisão e o tempo de processamento, percebe-se que, pelo executado até então, mesmo o MPC levando em consideração as posições para achar a melhor ação de controle, ele não lida bem com descontinuidades e impulsos fortes na referência. Em uma aplicação real de um veículo robótico autônomo, existe uma central eletrônica que analisa o ambiente no qual o sistema robótico está submetido, a situação em que este se encontra, e assim calcula a melhor rota a ser seguida. Uma forma de obter resultados otimizados para aplicações reais é levar em consideração possíveis descontinuidades e variações abruptas na geração das posições e entradas de referência. Ou seja, para os dados de referência, tentar gerar curvas mais suaves de forma a facilitar o processo de seguimento de referência pelo controlador.

Além disso, vale ressaltar a importância do processo de calibração. Cada controlador tem seus parâmetros primordiais que o tornam utilizável e, muitas vezes, ótimo. O estudo

destes parâmetros e análise dos valores que geram melhor desempenho é fundamental para que a simulação computacional ou aplicação real torne-se eficaz e produza resultados coerentes e relevantes.

Para trabalhos futuros sugere-se a implementação do controlador em um *rover* real. Sugere-se também a junção do controlador MPC com um estimador para o atrito pneu solo, ou outro parâmetro tomado como conhecido neste trabalho. Além disso, pode-se estudar e implementar o uso de dois MPCs em um controle em cascata, ou utilizar-se do menor tempo de processamento do PID e dar a este uma menor janela de controle, atualizando os dados vindos do MPC após um número fixo de iterações daquele. Por fim, sugere-se o estudo de otimização de processos de calibração dos parâmetros do MPC, e a suavização de trajetórias de referência.

REFERÊNCIAS

- 1 NI, J.; HU, J.; XIANG, C. A review for design and dynamics control of unmanned ground vehicle. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, SAGE Publications Sage UK: London, England, v. 235, n. 4, p. 1084–1100, 2021.
- 2 ALLARD, J.; BARRETT, D. S.; FILIPPOV, M.; PACK, R. T.; SVENDSEN, S. *Systems and methods for control of an unmanned ground vehicle*. [S.l.]: Google Patents, 2009. US Patent 7,499,776.
- 3 MURTAZA, Z.; MEHMOOD, N.; JAMIL, M.; AYAZ, Y. Design and implementation of low cost remote-operated unmanned ground vehicle (ugv). In: IEEE. *2014 International Conference on Robotics and Emerging Allied Technologies in Engineering (iCREATE)*. [S.l.], 2014. p. 37–41.
- 4 ALATISE, M. B.; HANCKE, G. P. A review on challenges of autonomous mobile robot and sensor fusion methods. *IEEE Access*, IEEE, v. 8, p. 39830–39846, 2020.
- 5 ALMAYYAH, A.; WANG, W.; HUSSEIN, A. A.; BIRCH, P. Motion control design for unmanned ground vehicle in dynamic environment using intelligent controller. *International Journal of Intelligent Computing and Cybernetics*, Emerald Publishing Limited, 2017.
- 6 NI, J.; HU, J.; XIANG, C. Robust control in diagonal move steer mode and experiment on an x-by-wire ugv. *IEEE/ASME Transactions on Mechatronics*, IEEE, v. 24, n. 2, p. 572–584, 2019.
- 7 HU, C.; ZHAO, L.; CAO, L.; TJAN, P.; WANG, N. Steering control based on model predictive control for obstacle avoidance of unmanned ground vehicle. *Measurement and Control*, SAGE Publications Sage UK: London, England, v. 53, n. 3-4, p. 501–518, 2020.
- 8 MAZUR, A.; CHOLEWIŃSKI, M. Implementation of factitious force method for control of 5r manipulator with skid-steering platform rex. *Bulletin of the Polish Academy of Sciences: Technical Sciences*, p. 71–80, 2016.
- 9 BARRERO, O.; TILAGUY, S.; NOVA, Y. M. Outdoors trajectory tracking control for a four wheel skid-steering vehicle. In: IEEE. *2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA)*. [S.l.], 2018. p. 1–6.
- 10 LV, W.; KANG, Y.; QIN, J. Indoor localization for skid-steering mobile robot by fusing encoder, gyroscope, and magnetometer. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, IEEE, v. 49, n. 6, p. 1241–1253, 2017.
- 11 IBRAHIM, F.; ABOUELSOUD, A.; ELBAB, A. M. F.; OGATA, T. Path following algorithm for skid-steering mobile robot based on adaptive discontinuous posture control. *Advanced Robotics*, Taylor & Francis, v. 33, n. 9, p. 439–453, 2019.
- 12 FERNANDEZ, B.; HERRERA, P. J.; CERRADA, J. A. A simplified optimal path following controller for an agricultural skid-steering robot. *IEEE Access*, IEEE, v. 7, p. 95932–95940, 2019.

- 13 OSTAFEW, C. J.; SCHOELLIG, A. P.; BARFOOT, T. D.; COLLIER, J. Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking. *Journal of Field Robotics*, Wiley Online Library, v. 33, n. 1, p. 133–152, 2016.
- 14 OSTAFEW, C. J.; SCHOELLIG, A. P.; BARFOOT, T. D. Robust constrained learning-based nmmpc enabling reliable mobile robot path tracking. *The International Journal of Robotics Research*, SAGE Publications Sage UK: London, England, v. 35, n. 13, p. 1547–1563, 2016.
- 15 LAURENSE, V. A.; GOH, J. Y.; GERDES, J. C. Path-tracking for autonomous vehicles at the limit of friction. In: IEEE. *2017 American control conference (ACC)*. [S.l.], 2017. p. 5586–5591.
- 16 AMER, N. H.; ZAMZURI, H.; HUDHA, K.; KADIR, Z. A. Modelling and control strategies in path tracking control for autonomous ground vehicles: a review of state of the art and challenges. *Journal of intelligent & robotic systems*, Springer, v. 86, n. 2, p. 225–254, 2017.
- 17 SUN, C.; ZHANG, X.; XI, L.; TIAN, Y. Design of a path-tracking steering controller for autonomous vehicles. *Energies*, Multidisciplinary Digital Publishing Institute, v. 11, n. 6, p. 1451, 2018.
- 18 CUI, Q.; DING, R.; ZHOU, B.; WU, X. Path-tracking of an autonomous vehicle via model predictive control and nonlinear filtering. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, SAGE Publications Sage UK: London, England, v. 232, n. 9, p. 1237–1252, 2018.
- 19 WANG, H.; LIU, B.; PING, X.; AN, Q. Path tracking control for autonomous vehicles based on an improved mpc. *IEEE Access*, IEEE, v. 7, p. 161064–161073, 2019.
- 20 BAI, G.; MENG, Y.; LIU, L.; LUO, W.; GU, Q.; LI, K. A new path tracking method based on multilayer model predictive control. *Applied sciences*, Multidisciplinary Digital Publishing Institute, v. 9, n. 13, p. 2649, 2019.
- 21 BAI, G.; LIU, L.; MENG, Y.; LUO, W.; GU, Q.; WANG, J. Path tracking of wheeled mobile robots based on dynamic prediction model. *IEEE Access*, IEEE, v. 7, p. 39690–39701, 2019.
- 22 ŠKRJANC, I.; KLANČAR, G. A comparison of continuous and discrete tracking-error model-based predictive control for mobile robots. *Robotics and autonomous systems*, Elsevier, v. 87, p. 177–187, 2017.
- 23 NEGRI, G. H.; CAVALCA, M. S. M.; CELIBERTO, L. Neural nonlinear model-based predictive control with fault tolerance characteristics applied to a robot manipulator. *Int. J. Innov. Comput. Inf. Control*, v. 13, n. 6, p. 1981–1992, 2017.
- 24 SONG, X.; SHAO, Y.; QU, Z. A vehicle trajectory tracking method with a time-varying model based on the model predictive control. *IEEE Access*, IEEE, v. 8, p. 16573–16583, 2019.
- 25 FARAG, W. Complex-track following in real-time using model-based predictive control. *International Journal of Intelligent Transportation Systems Research*, Springer, v. 19, n. 1, p. 112–127, 2021.

-
- 26 LOPES, E. D. R. *Estimação e Controle Avançados Aplicados a Sistemas Dinâmicos Veiculares*. Tese (Doutorado).
- 27 MOURA, M.; GUIMARAES, M.; CARVALHO, M. *Modelagem e Simulação de Veículo Terrestre Não-Tripulado*. 140 p. Monografia (Graduação em Engenharia Mecânica) — Instituto Militar de Engenharia, Rio de Janeiro, 2017.