

UNIVERSIDADE FEDERAL DO MARANHÃO

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

**PROJETO DE CONTROLE ADAPTATIVO E ÓTIMO
APLICADO A CADEIRA DE RODAS MOTORIZADA**

EVANDRO MARTINS ARAÚJO FILHO

ORIENTADOR: PROF. DR. JOÃO VIANA DA FONSECA NETO

São Luís
2018

UNIVERSIDADE FEDERAL DO MARANHÃO

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

**PROJETO DE CONTROLE ADAPTATIVO E
ÓTIMO**

APLICADO A CADEIRA DE RODAS MOTORIZADA

EVANDRO MARTINS ARAÚJO FILHO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão, como parte dos requisitos para a obtenção do título de Mestre em Engenharia de Eletricidade, área de concentração: Automação e Controle de Processos Industriais.
Orientador: Dr. João Viana da Fonseca Neto

São Luís
2018

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Araújo Filho, Evandro Martins.

Projeto de Controle Adaptativo e Ótimo Aplicado a Cadeira de Rodas Motorizada / Evandro Martins Araújo Filho. - 2018. 127 f.

Orientador(a): João Viana da Fonseca Neto.

Dissertação (Mestrado) - Programa de Pós-graduação em Engenharia de Eletricidade /ccet, Universidade Federal do Maranhão, São Luis - MA, 2018.

1. Cadeira de Rodas Motorizada. 2. Controlador PID. 3. Controle Adaptativo. 4. Redes Neurais Artificiais. 5. Sintonia Ótima. I. Fonseca Neto, João Viana da. II. Título .

UNIVERSIDADE FEDERAL DO MARANHÃO

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

PROJETO DE CONTROLE ADAPTATIVO E ÓTIMO APLICADO A CADEIRA DE RODAS

EVANDRO MARTINS ARAÚJO FILHO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão, como parte dos requisitos para a obtenção do título de Mestre em Engenharia de Eletricidade, área de concentração: Automação e Controle de Processos Industriais.

Aprovado em ____ de _____ de _____.

Membros da Banca:

Prof. Dr. João Viana da Fonseca Neto
(Orientador – UFMA)

Prof. Dr. Patrícia Helena Moraes Rego
(UFMA)

Prof. Dr. Raimundo Carlos Silvério Freire
(UFCG)

São Luís
2018

*A meus pais, Evandro Martins e Undine de Fátima, e minha irmã Sybelle Assunção,
exemplos de vida e caráter, pelo apoio incondicional em todas as decisões de minha vida.*

AGRADECIMENTO

Agradeço primeiramente a Deus pelo dom da vida, da sabedoria e da saúde, sem Ele nada seria possível.

Ao meu professor e orientador João Viana pela orientação e dedicação aplicados neste trabalho. Meu agradecimento sincero pelo acompanhamento, pela compreensão, auxílio e tempo despendido durante todo curso, sempre me motivando em todos os momentos em que precisei e que me mostrou os caminhos a serem seguidos.

Aos meus companheiros do LABSECI, Ana Caroline, Bruno França, Ernesto Franklin e José Pinheiro, pela ajuda, por todos os ensinamentos que compartilhamos e momentos de descontração.

A todos os professores e colaboradores do Programa de Pós-Graduação em Engenharia de Eletricidade e da Universidade Federal do Maranhão, que de alguma forma contribuíram com o desenvolvimento deste trabalho.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo suporte financeiro.

Eu não falhei. Apenas descobri 10 mil maneiras que não funcionam.

Thomas Edison

RESUMO

Muitos usuários que possuem a mobilidade prejudicada não conseguem operar uma cadeira de rodas motorizada com segurança, sem prejudicar a si mesmos e/ou outros por conta de variações no sistema, como por exemplo alteração do centro de massa do sistema ou alterações nas condições físicas da via a qual trafega. Tendo em vista contribuir para atenuar o problema de mobilidade, esta dissertação tem como principal objetivo apresentar o desenvolvimento de uma metodologia para projeto de controladores dinâmicos para cadeira de rodas motorizadas. Especificamente, duas propostas são apresentadas, que se baseiam em modelos para sintonia de controladores proporcional, integral e derivativo (PID), com propriedades de otimalidade e adaptabilidade. Estas propriedades são inseridas nos controladores PID por meio das abordagens de aprendizado de máquina, controles ótimo e adaptativo. A primeira proposta é relacionada com a propriedade de otimalidade, em que o projeto do controlador baseia-se em modelos no plano-s que são customizados para forma em espaço de estados para aplicação do projeto linear quadrático (LQR). Os referidos modelos são exclusivamente desenvolvidos para sintonia de controladores PID. Desta forma, o controlador deve lidar não só com as métricas convencionais de desempenho, mas também devem atender os seus custos operacionais. A segunda proposta associa as propriedades de adaptabilidade ao controlador PID baseado na teoria Redes Neurais Artificiais (RNA), para resolver problemas vinculados às variações paramétricas do sistema, visto que um algoritmo de controle do tipo PID convencional não responde bem aos distúrbios do processo. As propostas de sintonia são avaliadas em modelo matemático da cadeira de rodas em duas situações. Na primeira, o sinal de referência adotado é uma trajetória linear e o segundo caso é uma trajetória circular, os testes apresentam resultados satisfatórios.

Palavras-chave: Cadeira de Rodas Motorizada, Controlador PID, Sintonia Ótima, Controle Adaptativo, Redes Neurais Artificiais.

ABSTRACT

Many users with impaired mobility are unable to safely operate a motorized wheelchair without harming themselves and/or others due to parametric system variations, such as changing the center of mass of the system or changes in the physical road conditions in which it travels. In order to contribute to attenuate the problem of mobility, this dissertation has as main objective to present development of a methodology for the design of dynamic controllers for motorized wheelchairs. Specifically, two proposals are presented, which are based on proportional, integral and derivative (PID) controller tuning models, with optimality and adaptability properties. These properties are entered into the PID controllers through the machine learning, optimal and adaptive control approaches. The first proposal is related to the optimality property, the controller design is based on s-plane models that are customized to state-space form for linear quadratic regulator design (LQR) application, such models are exclusively developed for PID controllers tuning. Thus, the controller must deal not only with conventional performance metrics, but also must meet its operating costs. The second proposal associates the adaptive properties to the PID controller based on Artificial Neural Networks (ANN) theory, to solve problems related to the parametric system variations, since a conventional PID control algorithm does not respond well to the process disturbances. The tuning proposals are evaluated in mathematical wheelchair model in two situations. In the first, the reference signal adopted is a linear trajectory and the second case is a circular trajectory. The tests present satisfactory results.

Keywords: Powered Wheelchair, PID Controller, Optimal Tuning, Adaptive Control, Artificial Neural Networks.

LISTA DE FIGURAS

| | |
|---|----|
| Figura 2.1 – Cadeira de Rodas Manual. | 8 |
| Figura 2.2 – Diagrama de Blocos de uma Cadeira de Rodas Motorizada..... | 8 |
| Figura 2.3 – Ilustração de um Robô Móvel de Tração Diferencial Triciclo (Siciliano, et al., 2009)..... | 10 |
| Figura 2.4 – Ilustração de um Robô Móvel de Tração Diferencial Quadriciclo (Siciliano, et al., 2009)..... | 11 |
| Figura 2.5 – Ilustração de um Robô Móvel de Tração Diferencial Uniciclo (Siciliano, et al., 2009)..... | 11 |
| Figura 2.6 – Geometria de uma Cadeira de Rodas..... | 12 |
| Figura 2.7 – Centro de Rotação Instantâneo de uma Cadeira de Rodas. Adaptado de (Moslehi, 2011) | 14 |
| Figura 2.8 – Sistema Dinâmico de Uma Cadeira de Rodas Motorizada. | 15 |
| Figura 3.1 – Diagrama de Blocos do Controle de Posição Utilizando o Modelo Cinemático. | 18 |
| Figura 3.2 – Diagrama de Blocos do Sistema Dinâmico em Malha Fechada. | 18 |
| Figura 3.3 – Diagrama de Blocos do Sistema de Controle de Posicionamento Completo..... | 18 |
| Figura 3.4 – Diagrama de Blocos do Sistema em Malha Fechada da Cadeira de Rodas Motorizada. | 21 |
| Figura 3.5 – Diagrama de Blocos de um Sistema Dinâmico em Espaço de Estados..... | 23 |
| Figura 3.6 – Diagrama de Blocos de um Sistema Dinâmico em Malha Fechada em Variáveis de Estados. | 24 |
| Figura 3.7 – Estrutura Generalizada de um Neurônio. (a)Neurônio Natural (Kaiadi, 2006). (b) Neurônio Artificial (Haykin, 2008)..... | 27 |
| Figura 3.8 – Diagrama de Blocos de um Controlador Adaptativo na Forma Direta..... | 31 |
| Figura 3.9 – Diagrama de Blocos de um Sistema de Controle Adaptativo Indireto. | 31 |
| Figura 4.1 – Diagrama de Blocos do Sistema de Controle Dinâmico com PI Ótimo. | 36 |
| Figura 4.2 – Diagrama de Blocos do Sistema em Malha Fechada no Plano- s | 39 |
| Figura 4.3 – Sistema em Malha Fechado Utilizando o Controlador PID-RNA..... | 42 |
| Figura 4.4 – Estrutura Controlador PID-RNA com Conversor de Estado 1. | 44 |
| Figura 4.5 – Estrutura Controlador PID-RNA com Conversor de Estado 2. | 46 |

| | |
|---|----|
| Figura 5.1 – Velocidades Angulares das Rodas Esquerda e Direita para uma Trajetória de Referência Pontual Sistema para o Modelo de Sintonia de Segunda Ordem..... | 51 |
| Figura 5.2 – Lei de Controle das Rodas Esquerda e Direita para uma Trajetória de Referência Pontual para o Modelo de Sintonia de Segunda Ordem. | 52 |
| Figura 5.3 – Trajetória da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Segunda Ordem. | 53 |
| Figura 5.4 – Velocidades Angulares das Rodas Esquerda e Direita para uma Trajetória de Referência Circular para o Modelo de Sintonia de Segunda Ordem. | 54 |
| Figura 5.5 – Lei de Controle das Rodas Esquerda e Direita para uma Trajetória de Referência Circular para o Modelo de Sintonia de Segunda Ordem. | 55 |
| Figura 5.6 – Trajetória Circular da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Segunda Ordem..... | 56 |
| Figura 5.7 – Velocidades Angulares das Rodas Esquerda e Direita para uma Trajetória de Referência Pontual Sistema para o Modelo de Sintonia de Terceira Ordem. | 59 |
| Figura 5.8 – Lei de Controle das Rodas Esquerda e Direita para Uma Trajetória De Referência Pontual Para o Modelo de Sintonia de Terceira Ordem. | 60 |
| Figura 5.9 – Trajetória da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Terceira Ordem. | 61 |
| Figura 5.10 – Velocidades Angulares das Rodas Esquerda e Direita para Uma Trajetória De Referência Circular para o Modelo de Sintonia de Terceira Ordem. | 62 |
| Figura 5.11 – Lei de Controle das Rodas Esquerda E Direita Para Uma Trajetória De Referência Circular para o Modelo de Sintonia de Terceira Ordem..... | 63 |
| Figura 5.12 – Trajetória Circular da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Terceira Ordem. | 64 |
| Figura 5.13 – Resposta ao Degrau para o Sistema em Malha Fechada Aplicando PID-RNA. | 65 |
| Figura 5.14 – Convergência dos Ganhos do Controlador PID-RNA. | 66 |
| Figura 5.15 – Comparativo entre o Esforço de Controle do Controlador PID-RNA e PID Clássico. | 66 |
| Figura 5.16 – Velocidades angulares rodas esquerda e direita PID-RNA a um ponto..... | 67 |
| Figura 5.17 – Esforço de controle controlador esquerda e direita PID-RNA a um ponto. | 68 |
| Figura 5.18 – Acompanhamento da rede dos controladores esquerdo e direito PID-RNA a um ponto..... | 69 |
| Figura 5.19 – Trajetória plano (x, y) PID-RNA a um ponto | 70 |
| Figura 5.20 – Velocidades angulares rodas esquerda e direita PID-RNA trajetória circular. | 71 |

| | |
|--|-----|
| Figura 5.21 – Esforço de controle controladores esquerdo e direito PID-RNA trajetória circular..... | 72 |
| Figura 5.22 – Acompanhamento da rede controlador esquerdo e direito PID-RNA trajetória circular. | 73 |
| Figura 5.23 – Trajetória plano (x, y) PID-RNA circular. | 74 |
| Figura A.1 – Tipos de Conexão entre Bobinas de Campo e a Bobina de Armadura em Motores CC com Escovas. | 83 |
| Figura A.2 – Diagrama de Blocos de um Motor CC. | 85 |
| Figura A.3 – Diagrama de um Circuito Equivalente de um Motor BLDC de Três Fases..... | 86 |
| Figura A.4 – Estrutura de Controle de Velocidade de um Motor BLDC. | 86 |
| Figura A.5 – Diagrama de Blocos do Modelo Matemático de um Motor BLDC. | 89 |
| Figura A.6 – Exemplos de Sensores Utilizados no Sistema de Orientação de Cadeira de Rodas Motorizada. (a) Encoder Rotativo Óptico. (b) Bussola Digital. (c) Acelerômetro. | 91 |
| Figura C.1 – Diagrama de Blocos de Identificação do Sistema. | 105 |
| Figura C.2 – Algoritmo do filtro de Kalman. (Brown, 2012) | 107 |
| Figura C.3- Módulo de aquisição de dados NI USB-6212 M, Fonte alimentação regulável Minipa, motor CC com sensor de rotação e tacogerador. | 108 |
| Figura C.4 - Tensão medida e filtrada. | 109 |
| Figura C.5 - Processo de Identificação saída da planta: Estimada, Medida e Gerada pela Função Transferência estimada. | 110 |
| Figura C.6 - Convergência dos parâmetros ao longo das interações..... | 110 |
| Figura C.7 - Gráfico do erro de estimação de parâmetros. | 111 |

LISTA DE TABELAS

| | |
|--|----|
| Tabela 5.1 - Parâmetros de Sintonia e Ganhos do Controlador..... | 50 |
| Tabela 5.2 – Relação de amortecimento e constante para os sistemas em malha aberta e fechada. | 57 |
| Tabela 5.3 - Parâmetros de Sintonia e ganhos do controlador. | 58 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|-------------|--|
| ANN | <i>Artificial Neural Network</i> |
| ARE | <i>Algebraic Riccati Equation</i> |
| BLDC | <i>Brushless Direct Current</i> |
| CA | Corrente Alternada |
| CC | Corrente Contínua |
| DC | <i>Direct Current</i> |
| EAR | Equação Algébrica de Riccati |
| EDO | Equação Diferencial Ordinária |
| emf | <i>Electromotive force</i> |
| LQR | <i>Linear Quadratic Regulator</i> |
| MIMO | <i>Multiple Input Multiple Output</i> |
| MPL | Modulação por Largura de Pulso |
| MQR | Mínimos Quadrados Recursivos |
| PD | Programação Dinâmica |
| PID | Proporcional Integral Derivativo |
| PMDC | <i>Permanent Magnet Direct Current</i> |
| PWM | <i>Pulse-Width Modulation</i> |
| RLQ | Regulador Linear Quadrático |
| RLS | <i>Recursive Least Square</i> |
| RNA | Rede Neuronal Artificial |
| SISO | <i>Single Input Single Output</i> |
| WMR | <i>Wheeled Mobile Robot</i> |

SUMÁRIO

| | |
|---|-----------|
| LISTA DE FIGURAS | X |
| LISTA DE TABELAS..... | XIII |
| LISTA DE ABREVIATURAS E SIGLAS | XIV |
| CAPÍTULO 1 - INTRODUÇÃO..... | 1 |
| 1.1 Visão Geral..... | 1 |
| 1.2 Justificativa..... | 2 |
| 1.3 Objetivos..... | 3 |
| 1.3.1 Objetivos Gerais..... | 3 |
| 1.3.2 Objetivos Específicos..... | 4 |
| 1.4 Estado da Arte..... | 4 |
| 1.5 Organização do Trabalho | 5 |
| CAPÍTULO 2 - SISTEMA DE CONTROLE DO PROCESSO..... | 7 |
| 2.1 Descrição do Processo..... | 7 |
| 2.2 Robôs móveis com rodas | 9 |
| 2.3 Cinemática de cadeiras de rodas motorizadas | 11 |
| 2.4 Dinâmica de cadeiras de rodas motorizadas | 13 |
| CAPÍTULO 3 - SISTEMA DE CONTROLE PID-ADAPTATIVO ÓTIMO..... | 17 |
| 3.1 Preâmbulo..... | 17 |
| 3.2 Controlador Cinemático..... | 19 |
| 3.3 Controlador Dinâmico – Controlador PID | 20 |
| 3.4 Controlador Ótimo Linear Quadrático | 22 |
| 3.5 Controle via Redes Neurais Artificiais | 26 |
| 3.5.1 Neurônio Artificial..... | 27 |
| 3.5.2 Treinamento de uma Rede Neuronal Artificial..... | 28 |
| 3.5.3 Aplicações..... | 30 |
| CAPÍTULO 4 - SINTONIA ÓTIMA PARA CONTROLADORES PID VIA LQR E PID-RNA | 33 |
| 4.1 Modelos de Sistemas Dinâmicos para Sintonia LQR | 33 |
| 4.1.1 Modelo de sintonia de segunda ordem..... | 34 |
| 4.1.2 Modelo de sintonia de terceira ordem..... | 37 |
| 4.1.3 Modelo de sintonia de n^{a} ordem | 40 |

| | | |
|---|--|------------|
| 4.1.4 | Heurística $QR - 1$ para sintonia LQR..... | 41 |
| 4.2 | Controlador PID Adaptativo Baseado em RNA | 41 |
| 4.2.1 | Treinamento | 43 |
| 4.2.1 | Conversor de estados 1 | 43 |
| 4.2.2 | Conversor de estados 2 | 45 |
| CAPÍTULO 5 | - EXPERIMENTOS COMPUTACIONAIS..... | 49 |
| 5.1 | Experimentos Computacionais PID-LQR..... | 49 |
| 5.1.1 | PID-LQR para modelo de sintonia de segunda ordem | 50 |
| 5.1.2 | PID-LQR para modelo de sintonia de terceira ordem..... | 56 |
| 5.2 | Experimentos Computacionais PID-RNA | 64 |
| 5.2.1 | Experimentos computacionais para atuador em malha fechada | 65 |
| 5.2.2 | Experimentos computacionais para sistema cinemático e dinâmico referência pontual..... | 67 |
| 5.2.3 | Experimentos computacionais para sistema cinemático e dinâmico referência circular. | 70 |
| CAPÍTULO 6 | - CONCLUSÃO | 75 |
| 6.1 | Considerações Finais | 75 |
| 6.2 | Trabalhos Futuros | 76 |
| REFERÊNCIAS BIBLIOGRÁFICAS..... | | 77 |
| APÊNDICE A. SISTEMA ATUADOR E SENSOR | | 82 |
| A.1 | Elemento atuador..... | 82 |
| A.1.1 | Motores de Corrente Contínua com escovas | 82 |
| A.1.2 | Motores de Corrente Contínua sem escovas..... | 85 |
| A.2 | Sensores | 89 |
| APÊNDICE B. ALGORITMOS MATLAB | | 93 |
| B.1 | Algoritmo de Simulação Sintonia LQR Controlador PID..... | 93 |
| B.2 | Algoritmo de Simulação Sintonia LQR Controlador PID Simulink | 97 |
| B.3 | Algoritmo de Simulação Controlador PID-RNA..... | 98 |
| B.4 | Funções MATLAB© - Modelos matemáticos | 102 |
| B.4.1 | Função WC_model_d.m..... | 102 |
| B.4.2 | Função gain_lqr.m..... | 102 |
| B.4.3 | Função C_control.m | 103 |
| B.4.4 | Função D_wchair.m..... | 103 |
| B.4.5 | Função C_wchair.m..... | 104 |

| | |
|---|------------|
| B.4.6 Função sat_effort.m | 104 |
| APÊNDICE C. IDENTIFICAÇÃO DE SISTEMAS..... | 105 |
| C.1 Identificação do Sistema Eletromecânico | 105 |
| C.1.1. Algoritmo dos mínimos quadrados recursivos (RLS) | 106 |
| C.1.2. Filtro de Kalman | 106 |
| C.2 Descrição do Sistema de Ensaios | 107 |
| C.3 Tratamento dos sinais..... | 108 |
| C.4 Identificação do sistema | 109 |

Capítulo 1

INTRODUÇÃO

Neste capítulo será apresentada uma breve introdução que fornecerá uma visão geral e a motivação necessária para o trabalho de dissertação desenvolvido nesta dissertação. Depois de declarar o problema relevante, serão sintetizadas as abordagens de controle e algumas das suas deficiências serão apontadas. Os temas específicos abordados neste trabalho serão apresentados e ao final do capítulo é apresentado como o restante do trabalho está organizado.

1.1 Visão Geral

A evolução da tecnologia vem se tornando uma aliada aos portadores de deficiência física, pois suas inovações estão cada vez mais voltadas para atender suas necessidades. Além disso, a própria tecnologia assistiva possibilitou um avanço considerável em equipamentos dos mais variados tipos, atendendo as mais diversas necessidades.

A necessidade de muitas pessoas portadoras de deficiência física pode ser suprida com cadeiras de rodas manuais, mas para outros é muito difícil ou até mesmo impossível, utilizar cadeiras de rodas independentemente (Simpson, 2005). Nos últimos anos foram introduzidos uma variedade de métodos para desenvolvimento de sistemas de cadeiras de rodas motorizadas para atender pessoas com deficiência (Simpson, et al., 2004).

A cadeira de rodas motorizada é um instrumento médico usado para que as pessoas com deficiência física nos membros inferiores possam se locomover (Caporal, et al., 2015). Para facilitar a mobilidade dos usuários de cadeira de rodas, de acordo com (Bates, 2002), sinais ou ações geradas a partir da cabeça podem ser possíveis dispositivos de entrada para todo o nível

de lesões. Neste trabalho considera-se que cadeirantes com níveis de lesão baixa são aqueles que podem utilizar a mão e/ou cabeça para gerar comandos de entradas, entretanto quando este nível aumenta, geralmente a única solução são comando provenientes da cabeça.

O controle de velocidade da dinâmica de uma cadeira de rodas motorizada é indispensável quando se almeja torná-la mais confortável para os usuários (Clearesta, et al., 2013). O conforto pode ser garantido através da aplicação de um controlador adaptativo, o qual se adapta as mudanças ou perturbações na dinâmica do sistema quase sempre presentes no ambiente, tais como inclinação da via, peso do cadeirante, irregularidades da via, dentre outras.

Neste trabalho de dissertação é apresentado o projeto de um controlador ótimo de sintonia *offline* e um projeto de um controlador adaptativo para sintonia *online* aplicado para controle do sistema dinâmico de uma cadeira de rodas motorizada. O método é avaliado através de simulações em ambiente MATLAB®.

1.2 Justificativa

Os problemas de mobilidade devido à idade, doença ou lesão podem contribuir para uma perda severa de independência das pessoas, portanto, muitas destas dependem de dispositivos auxiliares para ajudá-los a alcançar um grau aceitável de mobilidade. Vários dispositivos que auxiliam na mobilidade estão disponíveis para o paciente, desde dispositivos muito simples até muito complexos (Vanacker, 2008). Entre os assistentes de mobilidade tem-se a bengala que pode ser considerada um dos dispositivos mais simples para tal. Os dispositivos de assistência mais complexos geralmente precisam de uma fonte de energia e normalmente fornecem mais assistência, logo quanto mais assistências estes tipos de dispositivos fornecem aos usuários, mais complexo se torna o sistema (Wang & Li, 2010).

Em alguns casos existem pessoas que perderam o controle dos braços e pernas, como resultado de lesão da medula espinhal em um nível mais elevado ou distúrbios do sistema nervoso, desta forma estas pessoas não podem usar a cadeira de rodas convencionais, que depende da força muscular dos braços para mover a cadeira de rodas. Ao mesmo tempo que não podem usar a cadeira de rodas motorizada controlada por *joystick* e ainda precisam da ajuda de outra pessoa para mover sua cadeira de rodas de um lugar para outro, ainda que consigam utilizá-la, uma cadeira de rodas motorizada pode se tornar perigosa para o seu usuário e/ou para

as pessoas ao redor, devido a velocidade máxima atingida (entre 10 e 15 km/h) e também ao seu peso (podem chegar a pesar 120 kg em conjunto com o peso do usuário) (Amori, 1992).

A aplicação de sistemas de controle em cadeiras de rodas motorizada tem como propósito superar os problemas de segurança associados à sua condução e também para auxiliar os usuários atenuando a falta de controle, para tal, vários projetos desenvolveram algoritmos no campo de controladores inteligentes (Wang, et al., 2009). O controlador geralmente vem na forma de uma unidade de computação *on-board* que possui uma gama de sensores como periféricos. Através de seus sensores, o controlador percebe o meio ambiente e o usuário (o *joystick* também é um dos sensores do controlador). A partir desses dados, então, pode-se tentar ajustar a comportamento dinâmico do sistema como um todo. Normalmente, o controlador atua diretamente sobre os motores de cadeiras de rodas. O *joystick* apenas está conectado aos motores através do controlador, de modo que o usuário tem influência indireta no movimento da estrutura.

A segurança e a robustez são aspectos especialmente sensíveis em uma implementação onde o piloto está situado no interior do sistema e além disso possui capacidades cognitivas e físicas limitadas (Kelouwani, et al., 2013). A robustez do sistema é retratada na medida em que pode-se atenuar as situações de alterações do meio em que trafega, por exemplo, uma cadeira de rodas pode ser bloqueada devido a alguns critérios de segurança, o protótipo deve permitir ao piloto poder retomar o controle a qualquer momento no modo manual (Fehr, et al., 2000).

1.3 Objetivos

Os objetivos deste trabalho de dissertação estão categorizados em gerais e específicos. Desta forma, distinguem-se pontos abrangentes a serem alcançados para que sejam logrados os fins essenciais.

1.3.1 Objetivos Gerais

Propor uma abordagem de projeto de controlador acessível, que possam ser facilmente adaptados em uma classe de máquinas móveis, para melhoria de desempenho do rastreamento da direção de referência. Desenvolvendo um sistema de deslocamento através de um

controlador dinâmico que seja capaz de conduzir uma cadeira de rodas motorizada de um ponto a outro de forma segura e confortável.

1.3.2 Objetivos Específicos

- Levantar modelo matemático de uma plataforma móvel de tração diferencial;
- Projetar controlador dinâmico e cinemático para cadeiras de rodas motorizada;
- Levantar modelo matemático de um controlador PID de sintonia ótima, para controle do sistema dinâmico.
- Pesquisar projetos de sistemas de controle através de estudo bibliográfico.
- Propor uma metodologia que permita projetar e implementar um sistema de controle automático para o posicionamento de uma cadeira de rodas elétrica.
- Desenvolver metodologias para controladores ótimo-adaptativo, o qual fornece as velocidades necessárias para rastrear o caminho de referência;
- Avaliar o projeto dos algoritmos propostos em plataforma do tipo MATLAB[®] e verificar o seu desempenho em face a convergência.

1.4 Estado da Arte

Um importante ponto no controle de robôs móveis é que a maioria dos controladores projetados atualmente são baseados na cinemática do sistema, contudo, para execução de tarefas que requerem deslocamento em altas velocidades ou transporte de cargas, é fundamental considerar a dinâmica dos veículos. Nesse caso, alguns controladores são propostos para compensar a dinâmica, como por exemplo em (Martins, et al., 2008) e (Cruz, et al., 2010), que propõem um controlador adaptativo seguidor de trajetória baseado na dinâmica de um robô móvel. Nestes trabalhos, existe um ponto em comum, ambos projetam seus controladores cinemáticos como sendo o modelo inverso da cinemática dos robôs, e este controlador fornece os sinais de referência para o controlador dinâmico, o que será detalhado nos capítulos a seguir.

A maior parte dos controladores que realizam a compensação da dinâmica de robôs móveis presentes na literatura geram sinais de torque (Cruz, et al., 2010). No entanto, robôs móveis comerciais usualmente recebem sinais de velocidade como comandos, como é o caso dos robôs Pioneer da empresa *Mobile Robots*, e dos robôs *Khepera* da *K-Team Corporation*.

Em (Solea, et al., 2015) propõe-se um controlador seguidor baseado em cinemática de robôs móveis de rodas (WMR – *Wheel Mobile Robot*) com aquisição de dados a partir de *encoders* aplicado a cadeira de rodas motorizada. Neste tipo de controlador é provida a robustez as curvas de erro de estimação, além de se propor um sistema de navegação baseado na movimentação da íris (olho) do usuário, com desvio de obstáculos.

Já em (Clearesta, et al., 2013), (Cruz, et al., 2012) e (Martins, et al., 2008), é proposto um controlador dinâmico adaptativo, no qual sua estabilidade é provada por Lyapunov baseado em modelos cinemáticos e dinâmicos de uma cadeira de rodas motorizada e a avaliação do método é realizada através de experimentos computacionais.

Para desenvolvimento deste trabalho de dissertação, também foram utilizadas bibliografias nas áreas de controladores ótimos e adaptativos para sistema dinâmicos. Em (He, et al., 2000), apresenta-se um algoritmo de sintonia de controladores PID através da teoria do Regulador Linear Quadrático, onde o processo de sintonia ocorre através da seleção das matrizes de ponderação Q e R associadas à frequência natural (ω_n) e ao fator de amortecimento (ζ) de um modelo de referência de segunda ordem, afim de se obter o comportamento desejado.

Em (Rivera-Mejía, et al., 2012) utiliza-se um controlador PID baseado em uma Rede Neuronal Artificial, afim de se garantir a adaptabilidade do sistema de controle. O desempenho do método é avaliado com base nos critérios de controle clássico, avaliando a resposta ao degrau usando modelos de planta de primeira e segunda ordem com mínima configuração de hardware.

1.5 Organização do Trabalho

No Capítulo 2 são fornecidas as explicações sobre o modelo matemático de uma cadeira de rodas motorizada e apresenta sua semelhança em relação a robôs móveis de tração diferencial. Neste capítulo, os modelos dinâmico e cinemático são expostos, em conjunto, com as restrições não-holonômicas do sistema e suas limitações. No Capítulo 3 é apresentado os principais conceitos relacionados a metodologia aplicada ao sistema de controle aplica em uma cadeira de rodas motorizada, que é composto por um controlador cinemático e controlador dinâmico. Ainda nesse capítulo são expostos os principais conceitos relacionados a controladores ótimos e Redes Neurais Artificiais aplicadas em sistemas de controle. O Capítulo 4 expõe o projeto do sistema de controle, de forma a iniciar com a apresentação do primeiro método de sintonia ótima de controladores PID e em seguida é apresentado a estrutura

do controlador PID adaptativo baseado em Redes Neuronais Artificiais. No Capítulo 5 é avaliada a metodologia mediante a análise dos resultados obtidos através de sua aplicação. As conclusões e propostas para trabalhos futuros, são apresentadas no Capítulo 6. Em anexo são disponibilizados os algoritmos desenvolvidos em ambiente MATLAB[®], para coleta e avaliação dos resultados (Apêndice B), de maneira mais específica nesta seção são fornecidos os algoritmos de sintonia LQR de controladores PID aplicado em uma cadeira de rodas motorizada incluindo seus comportamentos dinâmico e cinemático. Ainda nesta seção, é fornecido o diagrama de blocos montado utilizando a ferramenta MATLAB[®] e Simulink[®] dos sistemas dinâmico e cinemático de uma cadeira de rodas motorizada. O algoritmo de controle PID-RNA também é disponibilizado nesta seção. No Apêndice C é apresentado um método experimental para identificação de sistemas aplicando o algoritmo dos mínimos quadrados recursivos (*Recursive Least Square* – RLS) para identificação dos parâmetros de um motor de Corrente Continua.

Capítulo 2

SISTEMA DE CONTROLE DO PROCESSO

Neste capítulo é realizada a descrição o processo utilizado para avaliação da presente proposta, através do levantamento do modelo matemático, onde divide-se em cinemático e dinâmico, baseado nas principais metodologias empregadas nos trabalhos pesquisados.

2.1 Descrição do Processo

O levantamento do modelo matemático é a parte mais importante para qualquer projeto do sistema de controle de sistema dinâmicos e para isto é necessário entender o funcionamento de todo o sistema e este modelo, é dividido em modelo dinâmico e cinemático. Podem ser obtidos através de modelagem matemática ou através de técnicas de identificação de sistemas (no Apêndice C é apresentado um algoritmo para identificação de sistemas, baseado no estimador dos Mínimos Quadrados Recursivos). Normalmente, uma cadeira de rodas tem duas rodas diferenciais que a movem e duas rodas giratórias (rodas castor¹) que ajustam o movimento da cadeira de rodas, que são identificados na Figura 2.1.

No modelo dinâmico estão inseridos os componentes de aceleração em sua representação, de forma a resultar a variação da velocidade angular das rodas, levando-se em

¹Roda orientável não-centralizada, ou *castor wheel*, é uma roda orientável com relação à estrutura, sua principal função é dar estabilidade à estrutura mecânica como roda de direção (Siciliano, et al., 2009).

consideração constantes de massa, atrito e inercia, por exemplo, e esta energia mecânica é resultante do fornecimento de energia elétrica aos motores (Hou, et al., 2012).



Figura 2.1 – Cadeira de Rodas Manual.

As cadeiras de rodas motorizada geralmente utilizam dois motores de corrente contínua e estes requerem um sistema eletrônico capaz de comandar e ajustar de forma independente a velocidade de cada motor diante de uma superfície na qual a cadeira de rodas está se movendo.

O modelo cinemático tem como variáveis de entrada as velocidades da cadeira de rodas (escalar e angular) e como saída a posição e a direção da cadeira, e como constantes tem-se o diâmetro de suas rodas e o seu centro de gravidade. Os modelos cinemático e dinâmico são representados pela Figura 2.2.

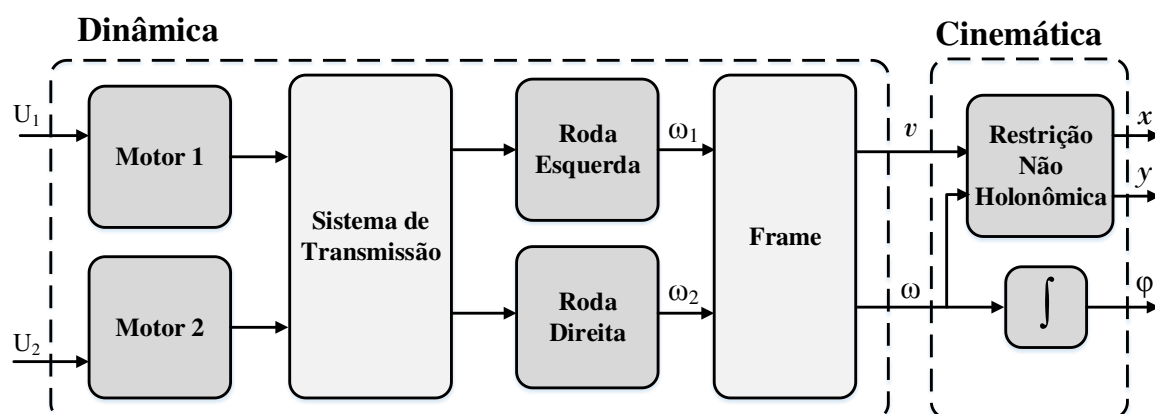


Figura 2.2 – Diagrama de Blocos de uma Cadeira de Rodas Motorizada.

onde U_1 representa a tensões de entrada do motor esquerdo e U_2 a tensão de entrada do direito, as velocidades angulares das rodas esquerda e direita são representadas por ω_1 e ω_2 , respectivamente, v representa a velocidade escalar da cadeira de rodas, ω é a velocidade angular

da cadeira de rodas, x e y são as posições da cadeira em relação ao eixo x e y e φ é ângulo da cadeira em relação ao eixo x .

A estrutura mecânica de uma cadeira de rodas motorizada assemelha-se as estruturas utilizadas em robôs móveis de tração diferencial (Clearesta, et al., 2013). Na área da robótica a principal característica de um robô é a sua estrutura mecânica. Os robôs podem ser classificados como aqueles com base fixa, manipuladores de robôs e aqueles com uma base móvel, robôs móveis. A seguir são apresentadas as características geométricas dos robôs móveis com rodas.

2.2 Robôs móveis com rodas

A robótica pode ser definida como um conjunto de conhecimentos teóricos e práticos que permitem conceber, executar e automatizar sistemas baseados em estruturas metálicas com várias articulações, dotados de um certo grau de "inteligência" e destinados à produção industrial ou à substituição do homem em várias tarefas (Júnior, 2000). Um sistema robótico pode ser descrito como o que é capaz de receber informações, de compreender o seu entorno através do uso de modelos, de formular e executar planos e de controlar ou supervisionar sua operação.

Os robôs móveis de modo geral são uma plataforma mecânica equipada com um sistema de locomoção capaz de transitar através de um determinado ambiente de trabalho, dotado de um certo nível de autonomia por sua deslocação carregando cargas ou cumprindo uma missão (Laut, 2011). Os robôs móveis que operam em ambientes não estruturados, ou seja, ambientes com obstáculos imprevisíveis (pessoas ou animais por exemplo), se deparam com incertezas significativas no posicionamento e identificação de objetos e até mesmo o posicionamento do robô em relação ao ambiente que está situado (Ruzajj & Poonguzhali, 2012). Os robôs móveis são caracterizados pela sua capacidade de se locomover pelo ambiente, que podem ser classificados de uma forma mais simples, como sendo aéreos, aquáticos, terrestres, etc. e evidentemente, um robô móvel pode ser projetado para se mover por vários meios.

A maioria dos robôs móveis distribuem seus sistemas de tração e direção nos eixos de suas rodas de acordo com as exigências de velocidade, facilidade em efetuar manobras e características do terreno (Romero, et al., 2014). A precisão e a velocidade com que o robô deve chegar ao seu destino, implica ter um sistema de tração confiável e um sistema de direção que proporciona maior estabilidade ao robô. Estes sistemas são responsáveis por determinar as

características do sistema de tração e direção, não apenas em relação à técnica a ser aplicada, mas também o número e o tipo de rodas necessárias para que se obtenha uma estável e de acordo com o arranjo das rodas na estrutura, os robôs móveis são classificados em triciclos, quadriciclos e unicyclos (Siciliano, et al., 2009).

O robô do tipo triciclo é composto por duas rodas fixas convencionais no mesmo eixo e uma roda orientável convencional que concentra as funções de direção de tração, que pode ser ilustrado pela Figura 2.3. A estrutura mecânica e os circuitos eletrônicos de controle são simples, para aplicações industriais, esta configuração é adequada para o transporte de cargas pesadas a uma baixa velocidade (Júnior, 2000).

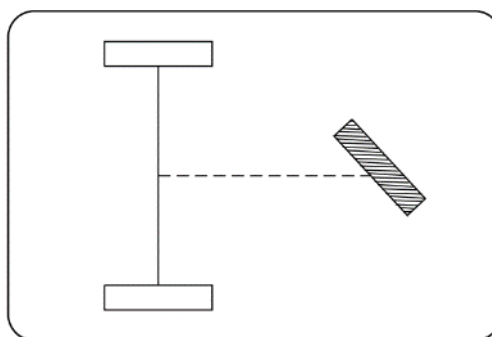


Figura 2.3 – Ilustração de um Robô Móvel de Tração Diferencial Triciclo (Siciliano, et al., 2009).

A configuração de um robô triciclo possui um problema associado ao posicionamento do seu centro de gravidade, em alguns casos, dentro dos limites da superfície do balanço, definidos pelas três rodas, quando o veículo está em movimento isso produz uma perda de tração no veículo causando erros de estimação de posição do robô.

O robô móvel do tipo quadriciclo possui duas rodas fixas montadas em um eixo traseiro e duas rodas orientáveis montadas em um eixo dianteiro. A representação deste tipo de robô é ilustrada pela Figura 2.4. Como no caso anterior, um motor fornece tração (nas rodas dianteiras ou traseiras) enquanto outro motor altera a orientação das rodas dianteiras em relação ao veículo. Vale ressaltar que, para evitar o deslizamento, ou arrastamento de uma das rodas, as duas rodas dianteiras devem ter uma inclinação diferente quando o veículo se move ao longo de uma curva, então a roda interna em relação ao raio de curvatura de sua trajetória é ligeiramente mais inclinada em relação à externa (Siciliano, et al., 2009). Isso é garantido pelo uso de um artifício específico chamado direção Ackerman aplicado também na área automotiva.

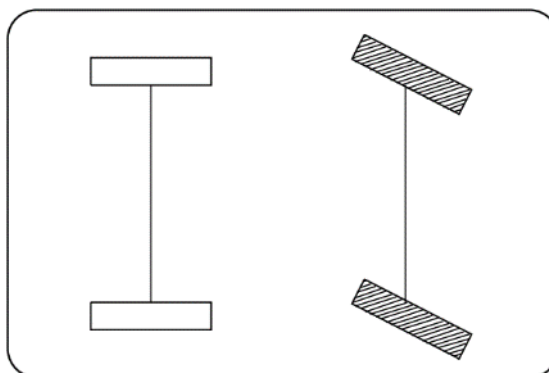


Figura 2.4 – Ilustração de um Robô Móvel de Tração Diferencial Quadriciclo (Siciliano, et al., 2009).

O modelo mais utilizado pelos pesquisadores para o teste de novas metodologias de controle é o do tipo unicyclo, por ter uma cinemática simples e sua estrutura consiste em duas rodas fixas convencionais no mesmo eixo, controladas independentemente e uma roda louca que dá estabilidade, conhecidos também como robôs móveis de tração diferencial (Alves, et al., 2011). O sistema de direção de tração associado a este tipo de configuração permite que a velocidade angular e escalar de referência sejam tratadas de forma independente.

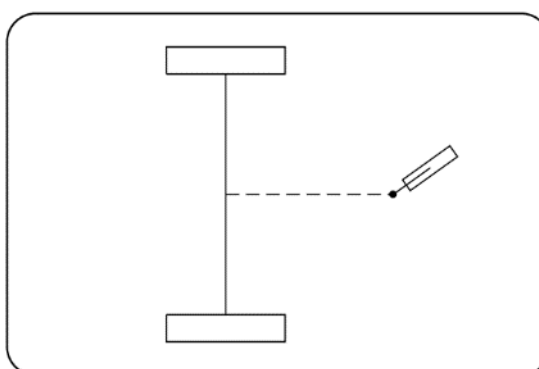


Figura 2.5 – Ilustração de um Robô Móvel de Tração Diferencial Unicyclo (Siciliano, et al., 2009).

2.3 Cinemática de cadeiras de rodas motorizadas

Os veículos com rodas são geralmente submetidos a algum tipo de restrição em sua movimentação (Siciliano, et al., 2009), por exemplo, um carro pode chegar a qualquer ponto final em seu plano, mas nunca pode se mover lateralmente, portanto, dependendo da seu destino final, é necessário realizar uma série de manobras (como estacionamento em uma vaga entre dois carros) para alcançar o estado desejado. Uma cadeira de rodas motorizada pode ser

caracterizada da mesma forma, e seu posicionamento pode ser descrito através de duas coordenadas, x e y , e um ângulo de direção φ , isto é ilustrado pela Figura 2.6.

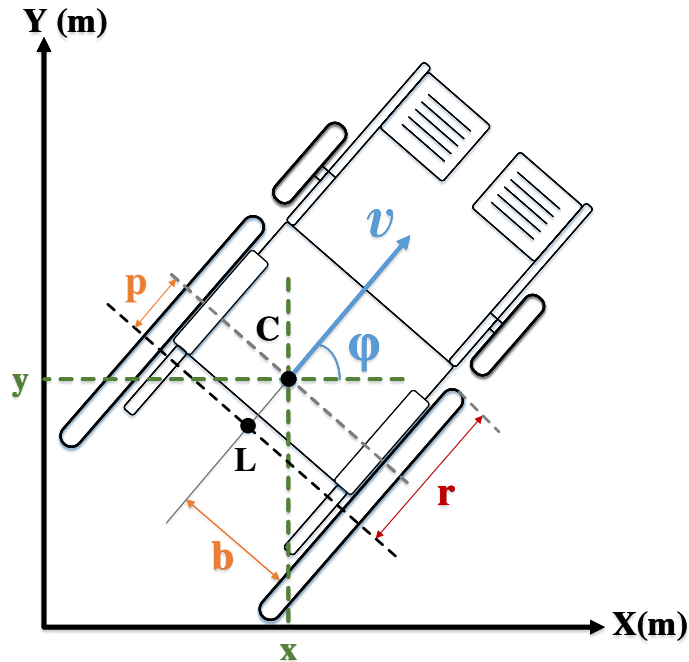


Figura 2.6 – Geometria de uma Cadeira de Rodas.

sendo p a distância entre o ponto C e o ponto L , r é o raio da roda, d é a distância do ponto L a cada roda, C é o centro de massa do sistema, v é a velocidade de avanço e ω é a velocidade angular da cadeira de rodas, como ilustrado na Figura 2.6.

A cadeira de rodas pode se locomover ao longo de um caminho que pode fornecer estas informações através de um vetor, dado por

$$q = [x \quad y \quad \varphi]^T. \quad (2.1)$$

Por uma questão de simplicidade optou-se por não referenciar as equações no domínio do tempo. Por exemplo, $x(t) = x$ e para diferenciação $dx(t)/dt = \dot{x}$.

A movimentação do vetor q está sujeita a uma restrição não-holonômica, que significa que o modelo cinemático do sistema não é capaz de se movimentar, instantaneamente, em qualquer direção e sua movimentação é descrita de acordo com a Eq. (2.2) (Laut, 2011).

$$\dot{x} \cdot \sin \varphi = \dot{y} \cdot \cos \varphi. \quad (2.2)$$

A Eq. (2.2) modela a restrição que impede o robô de se mover instantaneamente em qualquer direção, mas não o restringe de chegar em qualquer posição no plano em que está se

movendo (Siciliano, et al., 2009). A cinemática de robôs móveis de tração diferencial relacionam a velocidade escalar e a angular, respectivamente $[v \ \omega]^T$ em velocidades cartesianas $\dot{q} = [\dot{x} \ \dot{y} \ \dot{\varphi}]^T$, onde a componente x da velocidade escalar pode ser expressa como $\dot{x} = v \cdot \cos \varphi$ e a componente de y como $\dot{y} = v \cdot \sin \varphi$, a partir disso reescreve-se o sistema como

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi & -p \cdot \sin \varphi \\ \sin \varphi & p \cdot \cos \varphi \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (2.3)$$

A posição da cadeira de rodas é representada pelo vetor $h = [x \ y]^T$, assim

$$\dot{h} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos \varphi & -p \cdot \sin \varphi \\ \sin \varphi & p \cdot \cos \varphi \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = M \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad (2.4)$$

onde

$$M = \begin{bmatrix} \cos \varphi & -p \cdot \sin \varphi \\ \sin \varphi & p \cdot \cos \varphi \end{bmatrix}, \quad (2.5)$$

Neste tipo de abordagem, de obtenção do modelo cinemático, assume-se que qualquer velocidade v ou ω imprimida pela cadeira é instantaneamente atingida e os efeitos dinâmicos da cadeira de rodas são desconsiderados. Então para uma melhor representação do modelo matemático de uma cadeira de rodas é necessário gerar as velocidades angulares das rodas através de um modelo dinâmico para daí então serem geradas as velocidades cartesianas.

2.4 Dinâmica de cadeiras de rodas motorizadas

As entradas em um modelo cinemático não representam diretamente as entradas reais (isto é, força ou torque), pois omite-se a dinâmica do sistema para se lidar apenas com o modelo cinemático (Alves, et al., 2011). Com posse do modelo cinemático é necessário inserir a dinâmica fornecidas pelas rodas ao sistema, então defini-se ω_1 e ω_2 como sendo as velocidades angulares das rodas esquerda e direita, respectivamente, uma característica importante do modelo apresentado é que suas entradas são velocidades angulares das rodas (Cruz, et al., 2012).

O modelo proposto incorpora a dinâmica dos atuadores e de seus servos, então a velocidade escalar de cada uma das rodas, considerando que as rodas não deslizem ao girar

(destracionar) (Moslehi, 2011) (Cruz, et al., 2010) (Martins, et al., 2008), podem ser descritas como

$$v_i = r \cdot \omega_i, \quad (2.6)$$

sendo $i = e$ para a roda esquerda e $i = d$ para roda direita. As velocidades escalar e angular da cadeira podem ser definidas respectivamente como

$$v = \frac{(v_e + v_d)}{2}, \quad (2.7)$$

e

$$\omega = \frac{(v_d - v_e)}{2 \cdot b}. \quad (2.8)$$

Quando $i = e$ refere-se a esquerda e $i = d$ a direita. As Eq. (2.7) e (2.8) podem ser reescritas de modo que forneça o vetor de velocidades $[v \ \omega]^T$, necessários para determinação do vetor \dot{q} na Eq. (2.3), para o modelo dinâmico dado por

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \frac{r}{2} \begin{bmatrix} 1 & 1 \\ -1/b & 1/b \end{bmatrix} \begin{bmatrix} \omega_e \\ \omega_d \end{bmatrix} = N \begin{bmatrix} \omega_e \\ \omega_d \end{bmatrix}. \quad (2.9)$$

Como citado anteriormente, uma cadeira de rodas motorizada possui a mesma dinâmica robôs móveis de tração diferencial e esse comportamento pode ser ilustrado pela Figura 2.7, considerando que as rodas não percam tração.

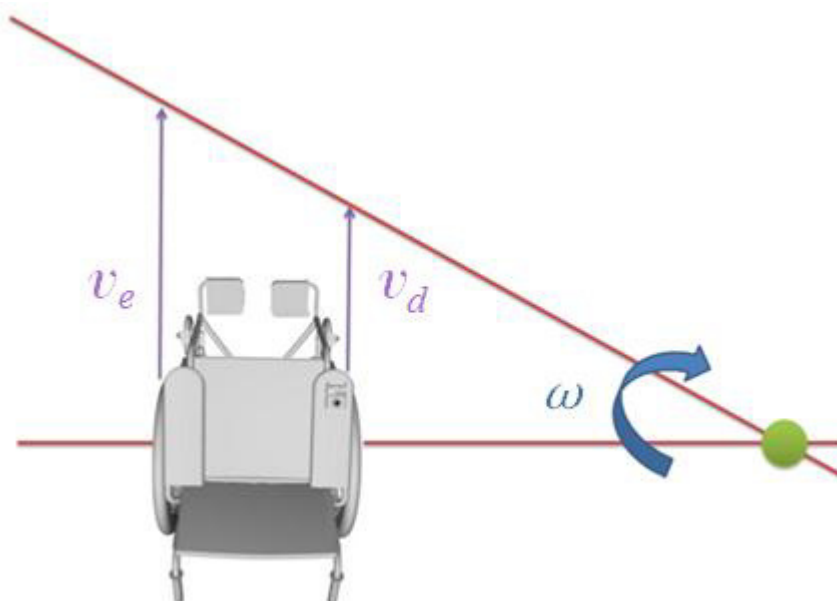


Figura 2.7 – Centro de Rotação Instantâneo de uma Cadeira de Rodas. Adaptado de (Moslehi, 2011)

As velocidades angulares das rodas esquerda e direita são fornecidas por dois motores elétricos, por intermédio de um sistema de transmissão, representado pela Figura 2.8.

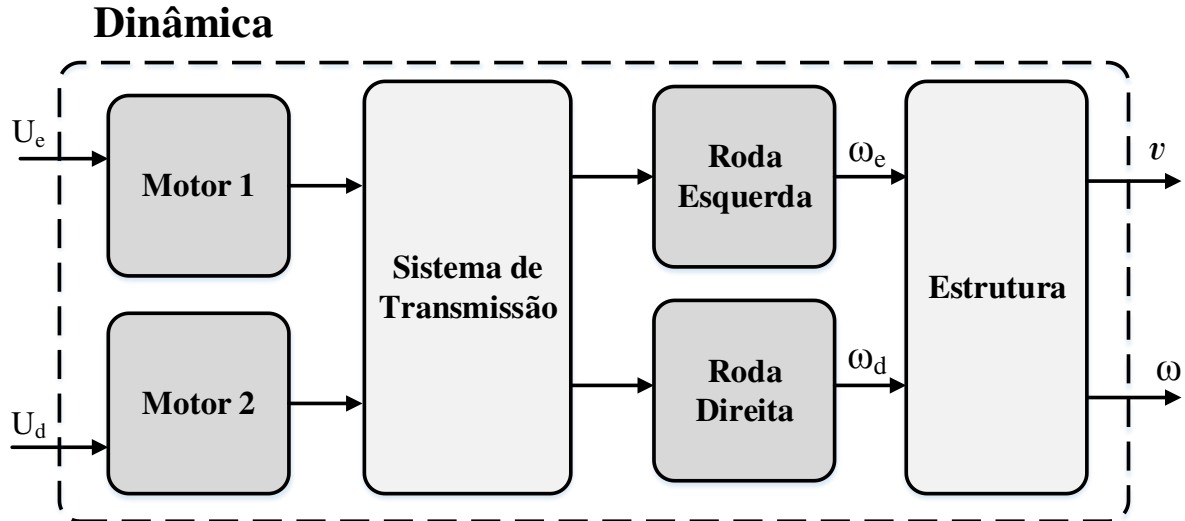


Figura 2.8 – Sistema Dinâmico de Uma Cadeira de Rodas Motorizada.

O método clássico para representação dinâmica de robôs móveis é o método de Lagrange que incorpora os conceitos de trabalho e energia para aquisição indireta das equações de movimento, isto se dá através da diferença entre a energia cinética e a potencial, dado por (Cruz, et al., 2010) (Martins, et al., 2008)

$$\mathcal{L}(q, \dot{q}) = E_{\tau}(q, \dot{q}) - E_u(q) = \frac{1}{2} \dot{q}^T I(q) \dot{q} - E_u(q), \quad (2.10)$$

sendo $E_{\tau}(q, \dot{q})$ e $E_u(q)$ as energias cinemática e potencial, respectivamente e $I(q)$ é a matriz de inércia do sistema mecânico.

A equação de Lagrange que representa a dinâmica é dada por

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right)^T - \left(\frac{\partial \mathcal{L}}{\partial q} \right)^T = 0, \quad (2.11)$$

esta é a forma geral da equação de Lagrange para sistemas holonômicos, no caso de sistemas não-holonômicos a equação anterior deve ser reescrita como

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right)^T - \left(\frac{\partial \mathcal{L}}{\partial q} \right)^T = S(q) \tau + V(q) \lambda, \quad (2.12)$$

onde $\tau \in \mathbb{R}^{o \times 1}$ é o vetor de torques de entrada, onde o é o número de entradas, $S(q) \in \mathbb{R}^{n \times o}$ é uma matriz de transformação das entradas dado pela Eq. (2.3), $\lambda \in \mathbb{R}^{m \times 1}$ é vetor de multiplicadores de Lagrange que representa a forças de restrição e $V(q) \in \mathbb{R}^{m \times n}$ é a matriz associada as restrições.

De acordo com as Eqs. (2.10) e (2.11) o modelo dinâmico de um sistema mecânico não-holonômicos é dado por

$$I(q)\ddot{q} + n(q, \dot{q}) = S(q)\tau + V(q)\lambda, \quad (2.13)$$

e

$$V(q)\dot{q} = 0. \quad (2.14)$$

De acordo com as Eqs (2.13) e (2.14), o modelo dinâmico da cadeira de rodas é dado por

$$\begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi & -p \cdot \sin \varphi \\ \sin \varphi & p \cdot \cos \varphi \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \tau_l \\ \tau_a \end{bmatrix} + \begin{bmatrix} \sin \varphi \\ -\cos \varphi \\ 0 \end{bmatrix} \lambda, \quad (2.15)$$

e

$$[\sin \varphi \quad -\cos \varphi \quad 0] \dot{q} = 0, \quad (2.16)$$

onde τ_l e τ_a representam a força escalar e o torque angular da cadeira de rodas respectivamente, aplicando a derivada na Eq. (2.4), para simplificar a Eq.(2.13) tem-se que

$$\dot{v} = \frac{\tau_l}{m}, \quad (2.17)$$

e

$$\dot{\omega} = \frac{\tau_a}{I}. \quad (2.18)$$

De forma similar as Eqs. (2.7) e (2.8), as equações de torque escalar e angular possuem a mesma relação entre as rodas direita e esquerda, dessa forma podem ser reescritas como

$$\tau_l = \frac{(\tau_e + \tau_d)}{2}, \quad (2.19)$$

e

$$\tau_a = \frac{l(\tau_d - \tau_e)}{2 \cdot b}. \quad (2.20)$$

onde τ_1 e τ_2 representam os torques das rodas esquerda e direita, os quais são gerados pelos atuadores (motores elétricos) de cada uma das rodas.

Capítulo 3

SISTEMA DE CONTROLE PID- ADAPTATIVO ÓTIMO

Este capítulo é dedicado a apresentação dos sistemas de controle de uma cadeira de rodas motorizada. A arquitetura e a formulação do controlador cinemático baseada no modelo cinemático inverso do sistema é apresentada. Para o controlador dinâmico, propõe-se a aplicação de um controlador Proporcional Integral e Derivativo, onde é apresentado alguns conceitos referentes a metodologia utilizada neste trabalho, tais como Regulador Linear Quadrático e Redes Neurais Artificiais, que serão aplicados no projeto do controlador apresentado na próxima seção.

3.1 Preâmbulo

O controle de movimento de máquinas móveis com tração diferencial é separado em controles cinemático e dinâmico. Os controladores atuam na velocidade do sistema de modo que seu posicionamento, representado pelo vetor $q = [x \ y \ \varphi]^T$, siga uma trajetória de referência, representada pelo vetor $q^r = [x^r \ y^r \ \varphi^r]^T$. Desta maneira, a estrutura do controlador, então é decomposta em controle cinemático, para controlar a posição do robô, e pelo controle dinâmico, que controla o vetor de velocidades do robô, ilustrados pelos diagramas de blocos nas Figura 3.1 e Figura 3.2.

Inicialmente, concentra-se apenas no modelo cinemático, assumindo que existe um rastreamento de velocidade perfeito, desta forma, a dinâmica do veículo é negligenciada e

considera-se apenas o sistema de direção conforme apresentado na Eq. (2.3), onde o vetor $V^r = [v^r \ \omega^r]^T$ representa as velocidades escalar e angular de referência da cadeira de rodas, respectivamente.

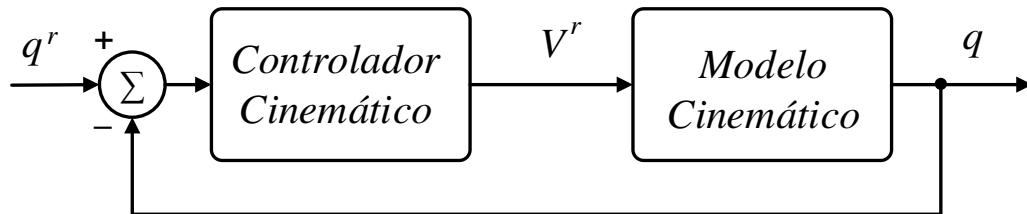


Figura 3.1 – Diagrama de Blocos do Controle de Posição Utilizando o Modelo Cinemático.

O projeto para controladores dinâmico é realizado levando-se em consideração a aceleração angular das rodas e momentos de inercia.

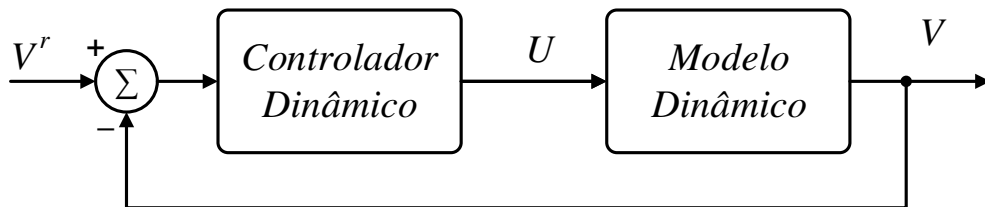


Figura 3.2 – Diagrama de Blocos do Sistema Dinâmico em Malha Fechada.

Ainda na Figura 3.2, $U = [U_e \ U_d]$ representa as tensões de alimentação dos motores esquerdo e direito. O sistema completo é ilustrado pelo diagrama de blocos da Figura 3.3.

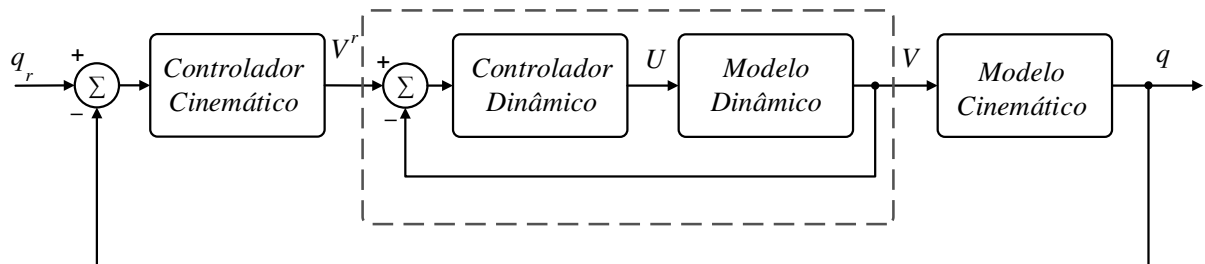


Figura 3.3 – Diagrama de Blocos do Sistema de Controle de Posicionamento Completo.

O controle de movimento de cadeira de rodas motorizada foi estudado por muitos autores na última década e como mencionado anteriormente, eles são cada vez mais utilizados em ampla gama de aplicações. O presente trabalho de dissertação foi realizado para projetar controladores de movimento, levando-se em consideração toda dinâmica do sistema, no entanto, o conhecimento exato sobre os valores de todos os parâmetros é praticamente inconcebível se considerar que durante o movimento esses parâmetros podem mudar devido a fricção superficial, carga adicional ou entre outros, daí o problema torna-se mais complexo.

Estes fatos justificam a necessidade de implementação de um controlador de forma a atenuar tais distúrbios

3.2 Controlador Cinemático

O controlador cinemático, aplicado em robôs móveis tem como objetivo controlar a sua posição e em alguns casos sua orientação (φ), gerando os sinais de referência para o controlador dinâmico a uma dada trajetória percorrida pela cadeira de rodas e baseia-se na cinemática do sistema dado pela Eq. (2.4). Então para um dado ponto de referência necessita-se gerar rotações de referência para as rodas, a inversa da cinemática do sistema, de acordo com (Cruz, et al., 2010), é representado por

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = M^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\frac{1}{p} \sin \varphi & \frac{1}{p} \cos \varphi \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}. \quad (3.1)$$

Ainda de acordo com (Cruz, et al., 2010) o controlador cinemático é dado por

$$\begin{bmatrix} v^r \\ \omega^r \end{bmatrix} = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\frac{1}{p} \sin \varphi & \frac{1}{p} \cos \varphi \end{bmatrix} \begin{bmatrix} x^r + \tanh\left(\frac{k_x}{I_x} e_x\right) \\ y^r + \tanh\left(\frac{k_y}{I_y} e_y\right) \end{bmatrix}, \quad (3.2)$$

As velocidades escalar e angular de referência da cadeira de rodas são representadas por v^r , ω^r , e_x e e_y são os erros de posicionamento em relação as coordenadas x e y respectivamente, k_x e k_y são os ganhos do controlador, onde $k_x > 0$ e $k_y > 0$, I_x e I_y são as constantes de saturação e x^r e y^r são as coordenadas de referências.

O modelo de controlador representado na Eq. (3.2) fornece as velocidades escalar e angular de referência para o sistema. As velocidades angulares de cada roda são obtidas a partir da inversa do modelo matemático dado pela Eq. (2.9). Desta forma manipulando-se a dada equação tem-se que as velocidades das esquerda e direita são dadas por

$$\begin{bmatrix} \omega_e \\ \omega_d \end{bmatrix} = N^{-1} \begin{bmatrix} v \\ \omega \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & -b \\ 1 & b \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad (3.3)$$

substituindo a Eq. (3.2) na Eq. (3.3), as velocidades angulares de referência são dadas por

$$\begin{bmatrix} \omega_e^r \\ \omega_d^r \end{bmatrix} = N^{-1} \cdot M^{-1} \begin{bmatrix} x^r + \tanh\left(\frac{k_x}{I_x} e_x\right) \\ y^r + \tanh\left(\frac{k_y}{I_y} e_y\right) \end{bmatrix}, \quad (3.4)$$

os valores de velocidade angular de referência da roda esquerda e direita são representados por ω_e^r e ω_d^r , respectivamente.

3.3 Controlador Dinâmico – Controlador PID

A sintonia de controladores dinâmicos geralmente necessita de um modelo preciso da planta para sintonizar os seus parâmetros, sem este modelo, deve-se usar um controlador mais simples, o qual seja possível de se ajustar os seus parâmetros *online*. Desta forma, a melhor escolha é, um controlador Proporcional Integral e Derivativo (PID) e sua saída é dada por

$$u(t) = K_P e(t) + K_I \int e(t) dt + K_D \frac{de(t)}{dt}. \quad (3.5)$$

Atualmente, controladores PID são amplamente utilizados no controle dinâmico de robôs móveis e em sistemas de controle industrial em geral (Majid, et al., 2016) (He, et al., 2000) (Widyotriatmo, et al., 2012). No entanto, sua capacidade de lidar com algumas propriedades complexas do processo, como não-linearidades e parâmetros variáveis no tempo, é conhecida por ser ineficaz (Vilanova & Visioli, 2012). Recentemente, algumas pesquisas foram conduzidas para projetar controladores dinâmicos não-lineares (Anvari, 2013) (Cruz, et al., 2010). A utilização de controladores não-lineares, melhoram significativamente a estabilidade do sistema (Majid, et al., 2016), no entanto, estes tipos de controladores têm uma estrutura mais complexa, o que torna o projeto e a implementação mais difíceis.

O controlador dinâmico proposto no presente trabalho é composto por um controlador PID para cada uma das rodas, que tem por objetivo controlar o movimento angular da cadeira de rodas, diminuindo o erro de regime estacionário por conta de distúrbios e incertezas do sistema. Por conveniência aplica-se a transformada de Laplace no modelo do controlador dado

pela Eq. (3.5) e efetuando algumas simplificações, tem-se que sua representação no plano- s é dada por

$$U_i(s) = \left(K_{P_i} + K_{I_i} \frac{1}{s} + K_{D_i} s \right) \cdot E_i(s), \quad (3.6)$$

quando $i = e$ refere-se ao lado esquerdo e $i = d$ ao direito. Desta forma, reescrevendo a Eq. (3.6) em função do integrador $1/s$, tem-se que

$$C_{PID}(s) = \left(K_{P_i} s + K_{I_i} + K_{D_i} s^2 \right) \frac{1}{s}, \quad (3.7)$$

sendo $C_{PID}(s) = U_i(s)/E_i(s)$. Ao se customizar o diagrama de blocos do sistema ilustrado pela Figura 2.2 para organização da estrutura do sistema em malha fechada, o diagrama de blocos ilustrado na Figura 3.4 é obtido.

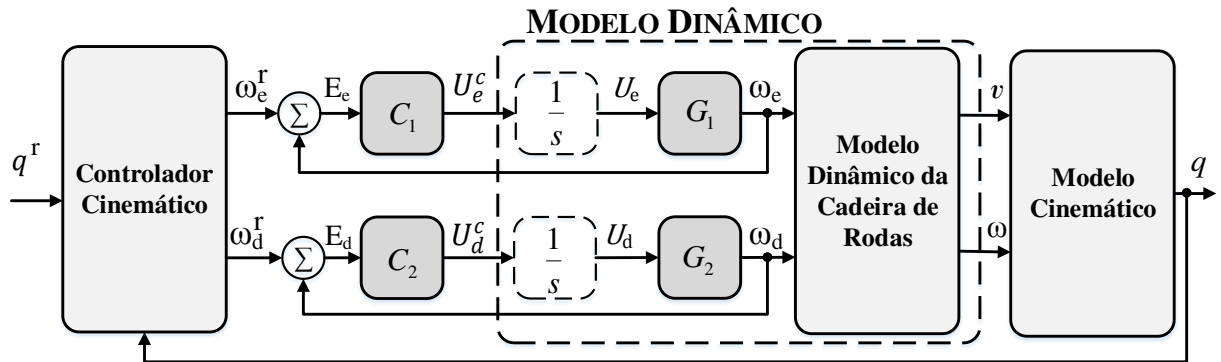


Figura 3.4 – Diagrama de Blocos do Sistema em Malha Fechada da Cadeira de Rodas Motorizada.

O projeto do controlador é realizado considerando-se que o integrador da Eq. (3.7) é acrescentado ao modelo matemático do atuador. Desta forma tem-se que a relação entre a saída do controlador e a saída da planta é dada por

$$\frac{U_i^c(s)}{\omega_i(s)} = \frac{1}{s} G_i(s), \quad (3.8)$$

portanto, a nova estrutura do controlador obtida é dada por

$$\frac{U_i^c(s)}{E_i(s)} = K_{P_i} s + K_{I_i} + K_{D_i} s^2, \quad (3.9)$$

substituindo-se $E_i(s) = \omega_i^r(s) - \omega_i(s)$ na Eq. (3.9) tem-se que

$$\frac{U_i(s)}{(\omega_i^r(s) - \omega_i(s))} = K_{P_i}s + K_{I_i} + K_{D_i}s^2,$$

desenvolvendo a relação anterior, tem-se que

$$U_i(s) = K_{P_i}s\omega_i^r(s) + K_{I_i}\omega_i^r(s) + K_{D_i}s^2\omega_{ref}^i(s) - K_{P_i}s\omega_i(s) - K_{I_i}\omega_i(s) - K_{D_i}s^2\omega_i(s). \quad (3.10)$$

Aplicando a transformada inversa de Laplace na Eq. (3.10), tem-se que

$$u_i(t) = K_{P_i} \frac{d\omega_i^r}{dt} + K_{I_i}\omega_i^r(t) + K_{D_i} \frac{d^2\omega_i^r}{dt^2} - K_{P_i} \frac{d\omega_i}{dt} - K_{I_i}\omega_i(t) - K_{P_i} \frac{d^2\omega_i}{dt^2}. \quad (3.11)$$

A Eq. (3.11) no espaço de estados é dada por

$$u_i(t) = [-K_{I_i} \quad -K_{P_i} \quad -K_{D_i}] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \check{r}, \quad (3.12)$$

sendo $K_{PID_i} = [K_{I_i} \quad K_{P_i} \quad K_{D_i}]$, então $\check{r} = K_{PID}^i \cdot [r(t) \quad \dot{r}(t) \quad \ddot{r}(t)]$ representa o sinal de entrada do sistema em malha fechada e as variáveis de estado x_{1_i} , x_{2_i} e x_{3_i} correspondem respectivamente a ω_i , $\dot{\omega}_i$ e $\ddot{\omega}_i$. A equação anterior é formada pelo numerador da função transferência do controlador PID e o denominador foi inserido como parte da planta, conforme constatado pela Eq. (3.8). Através disto tem-se a forma adequada para sintonia do controlador por meio de técnicas de controle ótimo, de forma mais específica o Regulador Linear Quadrático.

3.4 Controlador Ótimo Linear Quadrático

O projeto de sistemas de controle geralmente é um processo de tentativa e erro no qual vários métodos de análise são usados iterativamente para determinar os parâmetros “aceitáveis” de projeto de um sistema (Kirk, 2004). O desempenho de um sistema dinâmico geralmente é definido através de especificações no domínio do tempo e da frequência, tais como tempo de subida (t_r), tempo de estabilização (t_s), *overshoot*, margem de ganho e fase e largura de banda. Os critérios de desempenho devem ser rigorosamente satisfeitos quando empregados em sistemas complexos, como por exemplo sistemas com múltiplas entradas e múltiplas saídas. Desta forma, a teoria de controle ótimo é uma abordagem direta para a síntese desses sistemas complexos, que foi viabilizada pelo desenvolvimento do computador digital.

A teoria de controle clássico não é efetiva quando aplicada em sistema com múltiplas entradas e múltiplas saídas, de forma que seus estados variem em tempo mínimo ou variem usando uma quantidade mínima de energia (Lewis, et al., 2012). Neste tipo de problema, a teoria de controle ótimo serve como uma excelente estrutura matemática que tem como objetivo alcançar um comportamento desejável de várias saídas através de várias entradas. Para tanto, este tipo de controlador necessita de um modelo matemático capaz de descrever o comportamento do sistema dinâmico da melhor forma possível.

Um modelo matemático que representa um dado comportamento de um sistema dinâmico e possui seus parâmetros invariantes no tempo geralmente é modelado em forma de Equações Diferenciais Ordinárias (EDO), que por sua vez podem ser descritas em função de transferência, como representado nas Eqs. (A.4) e (A.6) do Capítulo 2, ou descritas no espaço de estado dado por

$$\dot{x}(t) = Ax(t) + Bu(t), \quad (3.13)$$

sendo $x \in \mathbb{R}^{n \times 1}$ é o vetor de estados, o vetor $u \in \mathbb{R}^{m \times 1}$ representa as entradas ou lei de controle. A matriz $A \in \mathbb{R}^{n \times n}$ representa a matriz de transição de estados, $B \in \mathbb{R}^{n \times m}$ é a matriz de entradas e m representa o número de entradas e n número de estados. E a saída para este sistema dinâmico é representada por

$$y(t) = Cx(t) + Du(t), \quad (3.14)$$

sendo $C \in \mathbb{R}^{l \times n}$ é a matriz de saída e $D \in \mathbb{R}^{n \times m}$ é matriz alimentação direta. O sistema apresentado pelas Eqs. (3.14) e (3.15) é ilustrado pela Figura 3.5.

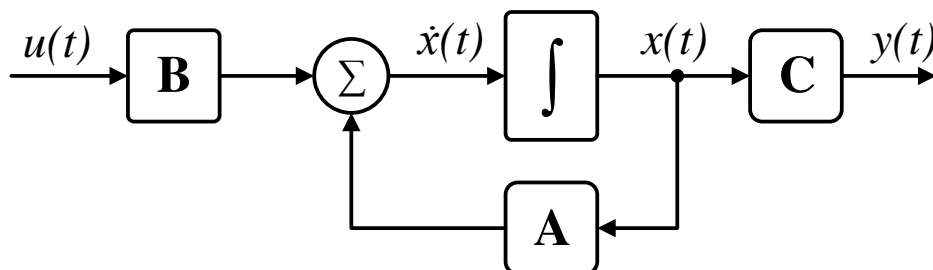


Figura 3.5 – Diagrama de Blocos de um Sistema Dinâmico em Espaço de Estados.

O objetivo da teoria de controle ótimo é determinar o sinal de controle para que um sistema possa atender às suas restrições físicas e minimizar uma função custo, ou seja, resolver um problema de otimização que leve o estado $x(t)$ do sistema para a trajetória desejada $x_d(t)$ minimizando a função custo, além de minimizar a lei de controle, reduzindo assim o uso dos atuadores do sistema (Lewis, et al., 2012).

Assumindo que todos os n estados estão disponíveis para realimentação e são completamente controláveis, então existe uma matriz de realimentação K , de modo que a lei de controle do sistema em malha fechada é dada por

$$u(t) = -K \cdot x(t) . \tag{3.15}$$

Através Eqs (3.13), (3.14) e (3.15) nota-se que o modelo do sistema em malha fechada é dado por

$$\dot{x}(t) = (A - BK)x(t) + Br(t) , \tag{3.16}$$

e a representação em diagrama de blocos é ilustrada pela Figura 3.6.

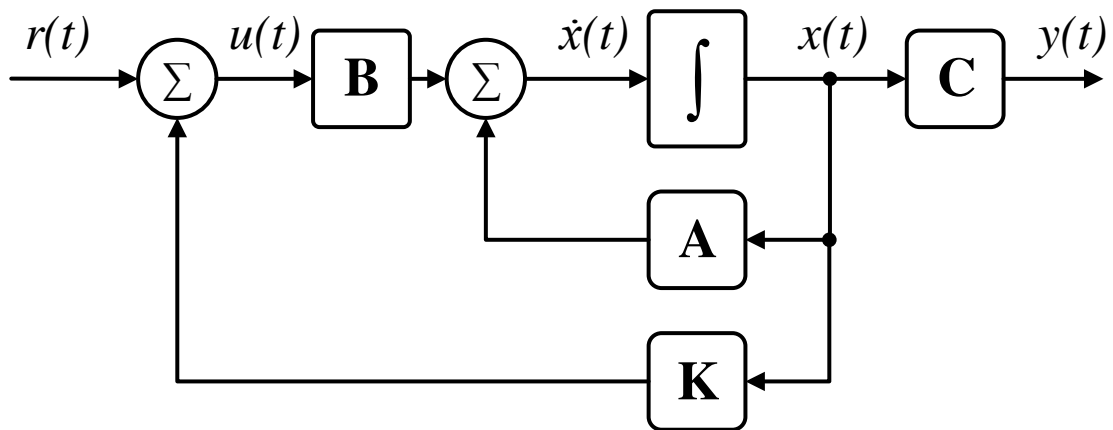


Figura 3.6 – Diagrama de Blocos de um Sistema Dinâmico em Malha Fechada em Variáveis de Estados.

O Regulador Linear Quadrático, ou em inglês *Linear Quadratic Regulator* (LQR) é um dos métodos de realimentação de estados mais simples e utilizado. Trata-se de um controlador ótimo onde o índice de desempenho (J) é uma integral temporal da soma da energia transitória e da energia de controle. Desta forma o levantamento do modelo do índice de desempenho J é necessário, levando-se em consideração as especificações, a necessidade do projeto e as restrições físicas que limitam o sistema. A integral é expressa em função do tempo e visa minimizar esse o índice de desempenho que é dado por

$$J(x, Q, u, R) = \frac{1}{2} \int_0^{\infty} (x^T Q x + u^T R u) dt , \tag{3.17}$$

sendo $Q > 0$ é uma matriz de ponderação de estados semidefinida positiva, $R > 0$ é uma matriz de ponderação de entrada positiva. Em (Da Fonseca Neto, et al., 2010) representa a função de otimização que minimiza o índice de desempenho como

$$\min_u J(x, Q, u, R), \quad (3.18)$$

sujeita a restrição dada pela Eq. (3.13).

A solução ótima para o problema LQR para um horizonte infinito, aplicado a Eq. (3.15), proporciona a estabilidade assintótica ao sistema se o par (A, B) for estabilizável e o par (A, \sqrt{Q}) seja detectável (Lewis, et al., 2012), ou seja, os estados instáveis devem ser detectados através da função de custo para que possam ser controlados. Por isso, exige-se que o par (A, \sqrt{Q}) seja detectável, isto é, todos os estados instáveis são observáveis. Além disso, o par (A, B) deve ser estabilizável, isto é, todos os estados instáveis são controláveis, para garantir que os estados instáveis possam ser estabilizados.

As matrizes de ponderação Q e R geralmente são matrizes constantes e diagonais, sendo $Q \geq 0$ e $R > 0$. Para que os valores de resultado da integral da expressão $x^T Q x$, dada pela função custo na Eq. (3.17), sejam pequenos e não negativos a matriz Q deve ser semidefinida positiva. Devido à natureza quadrática das matrizes de ponderação, grandes erros são penalizados mais do que pequenos erros. Além disso, a partir da expressão $u^T R u$, ainda na Eq. (3.17), nota-se claramente que quanto maior o esforço de controle maior será a energia. Esta energia deve assumir valores positivos, desta forma a matriz R deve ser definida positiva. (Naidu, 2003).

Considerando o sistema dado nas Eqs. (3.13) e (3.14) com a função custo dada pela Eq. (3.17), a lei de controle ótima é dada por

$$u^*(t) = -K^* \cdot x(t), \quad (3.19)$$

onde K^* é a matriz de ganhos de realimentação de estados para o sistema em malha fechada e é dada por

$$K^* = -R^{-1} B^T P, \quad (3.20)$$

sendo P é a matriz solução da Equação Diferencial Algébrica de Riccati, ou em inglês *Differential Algebraic Riccati Equation* (ARE), dada numericamente por (Lewis, et al., 2012)

$$-\dot{P} = A^T P + P A - P B R^{-1} B^T P + Q. \quad (3.21)$$

Utilizando métodos de Programação Dinâmica, pode-se solucionar a equação diferencial de Riccati dada pela Eq. (3.21), quando o horizonte tende ao infinito os valores de

P convergem (estabilizam), ou seja, a matriz $\dot{P} = 0$. Logo a Equação Algébrica de Riccati é dada por

$$A^T P + PA - PBR^{-1}B^T P + Q = 0. \quad (3.22)$$

Desta forma o LQR é ótimo em um horizonte infinito e pode ser sintonizado através matrizes de ponderação Q e R . A matriz Q , de maneira mais específica pode ser considerada uma penalidade de estado (erro) e a matriz R como uma penalidade na aplicação da entrada. Outra forma é penalizar as saídas (y), dada pela Eq. (3.14), admitindo-se que

$$Q = C^T Q_y C, \quad (3.23)$$

onde Q_y é a penalidade de saída. De forma genérica, o fator dado por

$$\beta = QR^{-1}, \quad (3.24)$$

onde os valores de β podem ser selecionados para proporcionar um controle rápido, selecionando-se somente os valores da matriz Q , ou obter um baixo consumo de energia pelo sinal de entrada, pela seleção apenas da matriz R (Grimstad, 2009).

3.5 Controle via Redes Neuronais Artificiais

As Redes Neuronais Artificiais (RNA) são modelos estimulados baseados na estrutura neuronal do cérebro. Ao contrário dos modelos matemáticos tradicionais, que são programados, as RNA aprendem as relações entre entradas e saídas selecionadas. Com as Redes Neuronais, um terá acesso a um forte ambiente de modelagem que permite testar e explorar modelos simulados mais rápido e fácil do que nunca. A presente proposta propõe a implementação de um controlador dinâmico adaptativo baseado em RNA, logo, alguns conceitos básicos necessitam ser discutidos, para um melhor entendimento da proposta.

A unidade básica de uma RNA é o neurônio, que são células eletricamente excitáveis e podem gerar sinais elétricos através do seu corpo celular, estes sinais são somados no corpo da célula após serem multiplicados por um coeficiente (peso), de tal forma que, quando este somatório atinge um valor limiar, o neurônio "dispara" e um sinal conhecido como o potencial de ação é enviado através do axônio para as terminações sinápticas. As sinapses são as junções através das quais os neurônios enviam sinais uns aos outros, onde cada um destes pode se conectar com aproximadamente 1000 outros neurônios (Kaiadi, 2006). A Figura 3.7 mostra uma

analogia entre um neurônio biológico e um neurônio artificial, com os seus quatro componentes principais.

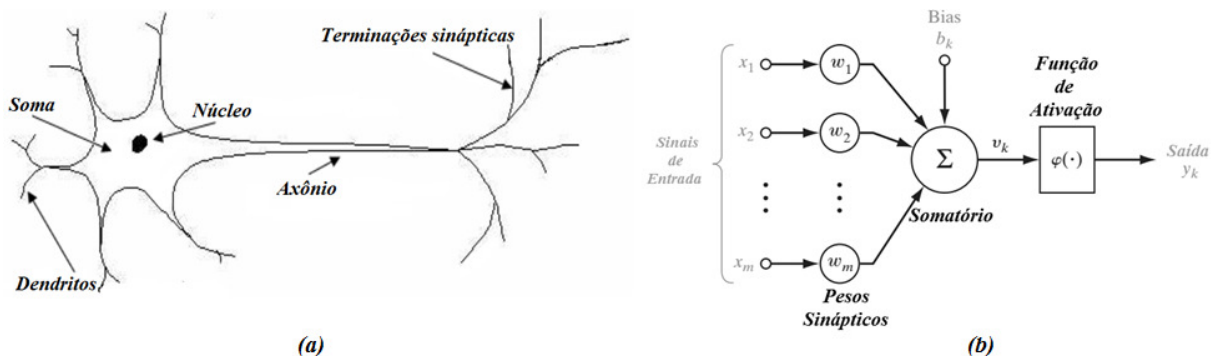


Figura 3.7 – Estrutura Generalizada de um Neurônio. (a) Neurônio Natural (Kaiadi, 2006). (b) Neurônio Artificial (Haykin, 2008).

O sinal transmitido entre neurônios é um potencial elétrico na natureza igual que é exatamente o mesmo sinal para todos os neurônios a que está conectado. De uma forma mais ampla, a RNA é um modelo matemático com parâmetros ajustáveis que se propõe a imitar esta estrutura biológica, e é treinado a partir de um conjunto de exemplos de entrada-saída (Kaiadi, 2006).

3.5.1 Neurônio Artificial

Um neurônio é essencial para uma RNA, visto que é uma unidade de processamento de informação. Uma RNA é organizada de muitas maneiras, ou seja, os seus neurônios podem estar interligados de formas de diferente, onde essas novas estruturas baseiam-se em conexões provenientes de dados de entrada ou de saídas de outros neurônios, que são designadas por pesos sinápticos que podem ser fixos ou ajustáveis, em um somador para efetuar a soma de todas as conexões de entrada do neurônio e em uma função de ativação responsável por limitar a saída do neurônio (Haykin, 2008).

De acordo com a Figura 3.7, pode-se constatar que as entradas são representadas por x_m , e cada uma destas entradas é multiplicada pelos pesos de conexão especificados, que são representados por w_m . Além disso, há uma entrada extra conhecida como *Bias* (b_k) e é responsável pela polarização, permitindo ao neurônio ter uma saída diferente de zero. No próximo passo, algumas operações matemáticas (principalmente somas) são realizadas e os produtos desta operação são alimentados através de funções de ativação não-linear ($\varphi(\cdot)$) para gerar saída, se a entrada combinada exceder um limite, ela irá ativar e enviar uma saída (Kaiadi, 2006). A saída enviada é determinada pela função de ativação e geralmente é escolhida como

sendo valores entre 0 e 1 ou -1 e 1. Como a derivada da função de ativação é frequentemente usada no treinamento da rede, é conveniente que a derivada seja expressa em termos do valor da função original, pois assim os cálculos adicionais são minimizados de forma a calcular a derivada desta função (Haykin, 2008).

O resultado produzido pelo neurônio artificial ilustrado pela Figura 3.7 é sintetizado por duas equações dadas por

$$v_k = \sum_{i=1}^n w_i \cdot x_i + b_k, \quad (3.25)$$

e

$$y_k = \varphi(v_k). \quad (3.26)$$

3.5.2 Treinamento de uma Rede Neuronal Artificial

A RNA é uma abordagem dentro do campo da aprendizagem de máquina, que basicamente desenvolve algoritmos que solucionam problemas e fazem previsões. A característica mais importante de uma RNA é a de poder aprender modelos matemáticos experimentalmente, sem ter que determinar explicitamente as relações matemáticas que vinculam as soluções com o problema, portanto, não é programada, mas sim "treinada" através de um processo de aprendizagem baseado em dados empíricos (Bishop, 2006). Dos métodos de treinamento, pode-se destacar três principais métodos de aprendizado: aprendizagem supervisionada, aprendizagem não supervisionada e aprendizagem por reforço.

A aprendizagem supervisionada, é aquela em que os algoritmos são usados para minimizar o erro preditivo da rede em um conjunto finito de dados, que pode ser do tipo *online*. A atualização dos parâmetros ocorre de forma incremental usando uma amostra de cada vez, ou por batelada, na qual a correção dos pesos sinápticos é feita na totalidade do erro das amostras coletadas; se o treinamento for bem sucedido, a rede aprende a reconhecer a relação que liga as variáveis de entrada às variáveis de saída e é capaz de responder corretamente aos estímulos que não estavam presentes no conjunto de treinamento (Haykin, 2008) (Bishop, 2006).

A aprendizagem não supervisionada, é aquela onde a informação sobre o ambiente externo é recebida sem fornecer qualquer indicação de valores de saída, na tentativa de agrupar esses dados reconhecendo padrões ou padrões implícitos.

A aprendizagem por reforço, é um método de treinamento realista e flexível em relação aos anteriores, em que a rede interage diretamente com o ambiente externo que responde através de estímulos positivos ou negativos ("recompensas" ou "punições") que orientam o algoritmo na fase de aprendizagem; a rede é então adaptada de modo a aumentar as chances de obter "recompensas" e diminuir aqueles de receber "punições".

3.5.2.1 Algoritmos de aprendizado

O algoritmo de aprendizagem é o método que as RNA utilizam para aprender com dados armazenados. Existem vários algoritmos de aprendizado disponíveis e dentre eles, o algoritmo *back-propagation* (retroalimentação) é o mais utilizado. A fim de minimizar o custo de uma predição, é necessário analisar as entradas, a função custo e os pesos da rede, visto que contribuem para o cálculo do custo. Onde alterações na entrada para obter uma saída correta está fora de questão, uma vez que o objetivo aqui é fazer com que a rede use a resposta real do sistema para dar uma resposta aceitável. A função de ativação está fora do nosso controle quando a aprendizagem está em andamento, portanto, é o peso que precisa ser alterado para minimizar o custo (Marsland, 2009).

A teoria geral do algoritmo de *back-propagation* geralmente é implementada por softwares de forma iterativa. O resultado desse processo de aprendizagem é passado para o neurônio de saída, onde este resultado de aprendizagem é comparado ao resultado desejado, se o erro não for suficientemente pequeno, o processo de aprendizagem na camada oculta continuará até obter resultados satisfatórios.

As equações matemáticas detalhadas e os antecedentes do algoritmo de aprendizado *back-propagation* estão disponíveis em muitos livros didáticos de Rede de Neuronais Artificiais (Haykin, 2008) (Marsland, 2009) (Bishop, 2006).

Cada método de aprendizagem discutidos anteriormente resultam em uma regra que ajusta os pesos das conexões entre unidades, de acordo com alguma regra de modificação. Praticamente todas as regras de aprendizagem para modelos deste tipo podem ser consideradas como uma variante da regra de aprendizagem *Hebbian* sugerida por Hebb (Hebb, 1949). De acordo com a regra de *Hebb*, o ajuste aplicado no peso sináptico w_{kj} em um instante de tempo n é dado por

$$\Delta w_{jk} = \eta y_k(n) x_j(n), \quad (3.27)$$

onde y_k representa as atividades pós-sinápticas, x_j as atividades pré-sinápticas e η é uma constante positiva a qual determina a taxa de aprendizagem (Haykin, 1994). Outra regra de treinamento comumente empregada, utiliza a diferença entre a entrada atual (pré-sináptico) e o sinal desejado ao invés de somente as atividades pré-sinápticas (x_j), e é dada por

$$\Delta w_{jk} = \eta y_k(n) (d_j(n) - x_j(n)), \quad (3.28)$$

sendo d_j o sinal de ativação desejado fornecido por um “professor”. Isto é chamado de Regra Delta e dentre uma de suas variações obtêm-se a seguinte expressão

$$\Delta w_{jk} = \eta (d_j(n) - x_j(n)) y_k(n) x_j(n). \quad (3.29)$$

A Eq. (3.29) é uma versão combinada da Regra de *Hebb* e a Regra Delta, sendo denominada como Regra de *Hebb* Supervisionada (Sheng, et al., 2002) (Liu, 2004) (Liu, 2008).

3.5.3 Aplicações

Uma RNA pode ser implementada como um controlador em um sistema em uma forma direta ou indireta. Em estratégias de controle direto a saída da rede é o próprio esforço de controle, onde os parâmetros do controlador são ajustados diretamente em função da saída do processo (Åström & Wittenmark, 1995), para este tipo de configuração normalmente a RNA necessita de m treinamento *online*, a representação dessa estrutura de controle é apresentada pelo diagrama de blocos na Figura 3.8.

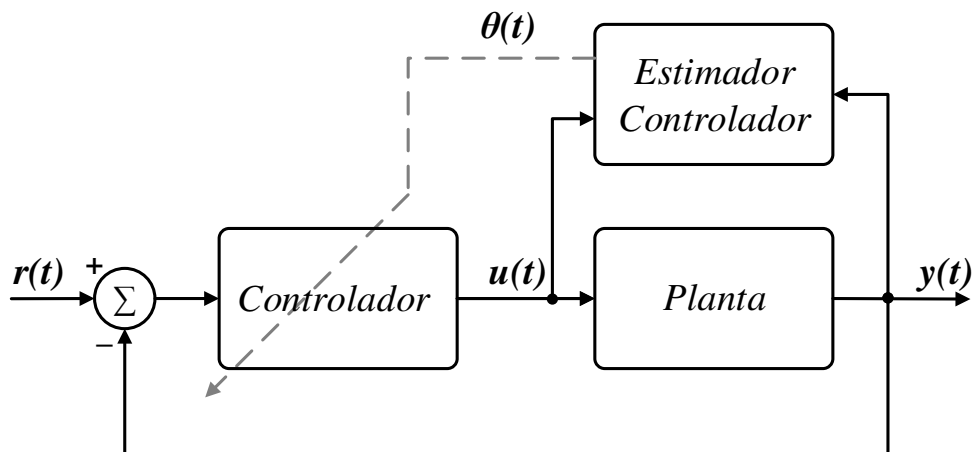


Figura 3.8 – Diagrama de Blocos de um Controlador Adaptativo na Forma Direta.

A implementação de controladores adaptativos na forma indireta, ou também conhecida como auto ajustável (Åström & Wittenmark, 1995), no caso de RNA, é aquela em que é responsável pela sintonia ou compensação do controlador fornecendo assistência quando necessário, neste caso quando é implementado é conhecido também como compensador.

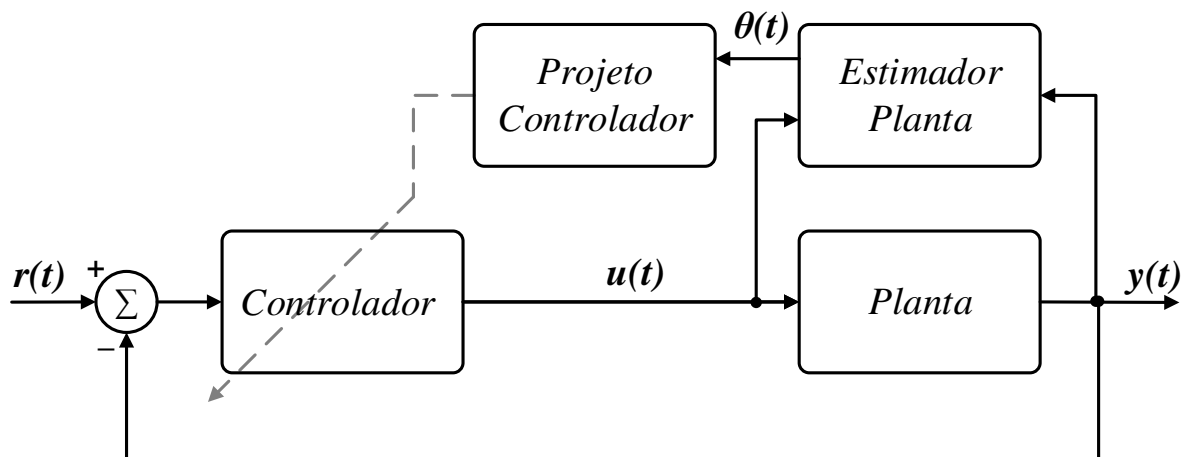


Figura 3.9 – Diagrama de Blocos de um Sistema de Controle Adaptativo Indireto.

Ao se comparar as duas formas de aplicação de RNA em sistema de controle, é notável que poucas pesquisas foram até o momento realizadas o estudo do desempenho das RNA como controladores, a maioria das aplicações das RNA envolvem controladores indiretos, onde a RNA é usada como um estimador de modelo ou como um compensador (Ruano, 1992).

As RNA têm sido amplamente aplicadas nas áreas de controle por realimentação de estados, sistemas de controle não linear, identificação de sistemas e outros. Como a Rede Neuronal Artificial tem uma forte capacidade de aproximação e auto adaptação, é amplamente utilizada em vários problemas de controle. A fim de evitar as anomalias do controlador PID clássico, a RNA e o PID são combinados para o controle. O controlador PID baseado em Redes

Neuronais Artificiais (PID-RNA) tem estrutura simples e demanda pouco tempo para o cálculo, além disso, possui uma forte capacidade de auto ajuste. Todas essas características tornam-no adequado para o controle em tempo real (Liu, 2008) (Haykin, 2008).

A característica mais interessante das RNA, e também a fonte de sua reputação como ferramentas poderosas para resolver diversos problemas, é a sua alta capacidade de mapeamento de sistemas não-lineares, permitindo-lhes aprender os comportamentos subjacentes aos dados adquiridos desses sistemas.

Capítulo 4

SINTONIA ÓTIMA PARA CONTROLADORES PID VIA LQR E PID-RNA

A abordagem de projeto do controlador de sintonia ótima é apresentada. Os modelos matemáticos utilizados para projeto do controlador foram modelados no plano-s e em seguida são efetuadas manipulações matemáticas, afim de se obter a representação do sistema em variáveis de estados. Em seguida, o algoritmo de controle adaptativo baseado em Redes Neurais Artificiais é apresentado para garantir a adaptabilidade do sistema de controle relacionada a variação dos parâmetros do sistema e rejeição a distúrbios. O principal critério de projeto do sistema controle da cadeira de rodas motorizadas é propor um sistema não complexo e de baixo custo que pode ser implementado a uma cadeira de rodas motorizada para melhorar sua mobilidade e garantir um percurso seguro.

4.1 Modelos de Sistemas Dinâmicos para Sintonia LQR

Muitos problemas de engenharia hoje são conduzidos pela necessidade de encontrar trajetórias ótimas. Esses problemas não se limitam a um único setor, mas abrangem uma ampla gama de áreas, incluindo aeronáutica, robótica, economia e química. De acordo com as formulações das seções anteriores, assume-se que todos os estados do sistema em estudo estão disponíveis para que eles sejam realimentados para o controlador.

O método de sintonia LQR é uma associação da implementação do controle da cadeira de rodas baseada em diagrama de blocos no plano- s e seu método de projeto em espaço de estado. A implementação engloba uma realização de controle de alto nível do controlador PI clássico em uma planta em malha fechada, como exposto na Figura 4.1. Em termos de projeto do controlador, a abordagem encontra-se em métodos de espaço de estado e controle ótimo.

Os principais passos do método da sintonia LQR são descritos a seguir. O primeiro passo é desenvolver o sistema de controle do sistema em espaço de estado que é equivalente ao sistema de controle da Figura 4.1. O segundo passo é o cômputo dos ganhos proporcional e integral do controlador, dado na Eq. (3.6), que são os ganhos PI representados nesta Figura 4.1. E, por fim, definir o método associado com as especificações de controle a fim de se obter a operação desejada.

4.1.1 Modelo de sintonia de segunda ordem

O modelo matemático do sistema atuador em estudo representa a dinâmica do motor CC com escovas acoplado ao sistema de transmissão e com seu circuito de potência, tendo como entrada um sinal de tensão representado por $u(t)$, de nível lógico, e sua saída uma velocidade angular representado por $y(t)$ e tem sua função de transferência dada por

$$G_i(s) = \frac{b_{0i}}{s + a_{1i}}, \quad (4.1)$$

onde i representa os parâmetros para o sistema e , lado esquerdo, e d o sistema do lado direito.

Para fins de sintonia a ação integral do controlador é inserida no modelo da planta conforme apresentado no Capítulo 3 e ilustrado na Figura 3.4, com isso tem-se que o modelo de sintonia de segunda ordem obtido é dado por

$$F_i(s) = \frac{1}{s} \left(\frac{b_{0i}}{s + a_{1i}} \right),$$

logo

$$F_i(s) = \frac{b_{0i}}{s^2 + a_{1i}s}, \quad (4.2)$$

sendo $F_i(s)$ é a função de transferência da planta com o integrador obtido a partir do controlador.

Considerando que $F_i(s)$ possui como entrada $U_i(s)$ e saída $Y_i(s)$, pode-se escrever esta função transferência da seguinte forma

$$\frac{\omega_i(s)}{U_i^c(s)} = \frac{b_{0i}}{s^2 + a_{1i}s},$$

logo

$$\omega_i(s)s^2 + \omega_i(s)a_{1i}s = b_{0i}U_i^c(s). \quad (4.3)$$

Aplicando a transformada inversa de Laplace na Eq. (4.3), tem-se

$$\frac{d^2\omega_i(t)}{dt} + a_{1i}\frac{d\omega_i(t)}{dt} = b_{0i} \cdot u_i^c(t), \quad (4.4)$$

colocando na descrição de espaço de estados obtém-se

$$\frac{d^2\omega_i(t)}{dt} = -a_{1i}\frac{d\omega_i(t)}{dt} + b_{0i} \cdot u_i^c(t), \quad (4.5)$$

sendo

$$\begin{aligned} x_{1i} &= \omega_i(t), \\ \dot{x}_{1i} &= \frac{d\omega_i(t)}{dt} = x_2, \end{aligned} \quad (4.6)$$

e

$$\ddot{x}_{1i} = \frac{d^2\omega_i(t)}{dt} = \dot{x}_2.$$

Substituindo a Eq. (4.5) na Eq. (4.6), o sistema de equações é dado por

$$\begin{cases} \dot{x}_{1i} = x_{2i} \\ \dot{x}_{2i} = -a_{1i} \cdot x_{2i} + b_{0i} \cdot u_i^c. \end{cases} \quad (4.7)$$

O modelo da planta com ação integral em malha aberta resulta em um sistema de segunda ordem e sua representação completa em espaço de estado na forma matricial, ilustrado na Figura 4.1, incluindo a dinâmica dos dois atuadores é dada por

$$\begin{bmatrix} \dot{x}_e \\ \dot{x}_d \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & a_{1e} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & a_{1d} \end{bmatrix} \begin{bmatrix} x_e \\ x_d \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b_{0e} & 0 \\ 0 & 0 \\ 0 & b_{0d} \end{bmatrix} \begin{bmatrix} u_e^c \\ u_d^c \end{bmatrix}, \quad (4.8)$$

$$\begin{bmatrix} \omega_e \\ \omega_d \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_e \\ x_d \end{bmatrix},$$

sendo $x_e = [x_{1e} \quad x_{2e}]^T$ e $x_d = [x_{1d} \quad x_{2d}]^T$. Observa-se que a matriz pode ser separada em blocos fornecendo os modelos dos atuadores esquerdo e direito.

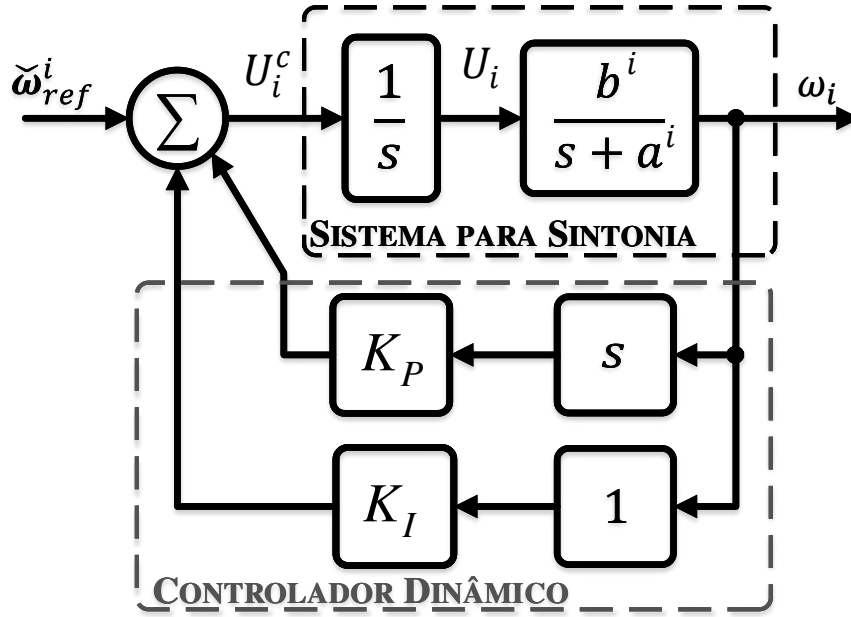


Figura 4.1 – Diagrama de Blocos do Sistema de Controle Dinâmico com PI Ótimo.

4.1.1.1 Cálculo dos ganhos de realimentação de estados de segunda ordem

A lei de controle $u_i^c(t)$ na forma de espaço de estado, de acordo com as Eqs. (3.9) e (3.11), que apresenta a relação entre o plano- s e a representação em espaço de estado. Conseqüentemente, é dada por

$$u_i^c(t) = K_{I_i} \omega_r^r(t) + K_{P_i} \dot{\omega}_i^r(t) - K_{I_i} x_{1_i} - K_{P_i} x_{2_i}. \quad (4.9)$$

Uma forma simplificada da lei de controle da Eq. (4.9), considerando os dois atuadores é dada por

$$\begin{bmatrix} u_e^c \\ u_d^c \end{bmatrix} = - \begin{bmatrix} K_{I_e} & K_{P_e} & 0 & 0 \\ 0 & 0 & K_{I_d} & K_{P_d} \end{bmatrix} \begin{bmatrix} x_e \\ x_d \end{bmatrix} + \begin{bmatrix} \tilde{\omega}_e^r \\ \tilde{\omega}_d^r \end{bmatrix}. \quad (4.10)$$

Com base no sistema dado pela Eq. (4.8), o sistema é representado em blocos na Figura 4.1, onde apresenta-se o modelo da planta com a ação integral, sendo este o modelo de sintonia necessário para a obtenção dos ganhos de realimentação de estados.

A representação em espaço de estados do sistema em malha fechada é dada por

$$\begin{bmatrix} \dot{x}^1 \\ \dot{x}^2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -bK_{I_e} & -(a_{1_e} + b_{0_e}K_{P_e}^1) & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -b_{0_d}K_{I_d} & -(a_{1_d} + b_{0_d}K_{P_d}) \end{bmatrix} \begin{bmatrix} x^1 \\ x^2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b_{0_e} & 0 \\ 0 & 0 \\ 0 & b_{0_d} \end{bmatrix} \begin{bmatrix} \tilde{\omega}_e^r \\ \tilde{\omega}_d^r \end{bmatrix} \quad (4.11)$$

sendo $\tilde{\omega}_r^1$ e $\tilde{\omega}_r^2$ as velocidades angulares de referencias das rodas esquerda e direita, respectivamente.

O modelo matemático do controlador foi desenvolvido, transformando os ganhos K_P e K_I , de um controlador PI, em ganhos de realimentação do estado.

4.1.2 Modelo de sintonia de terceira ordem

O mesmo procedimento também foi realizado para um modelo de segunda ordem para avaliação do método, o que resultou em um controlador PID com três ganhos de realimentação de estados K_P , K_I e K_D . De forma generalizada este modelo no plano-s, é expresso como

$$G(s) = \frac{b_{0i}}{a_{0i}s^2 + a_{1i}s + a_{2i}}, \quad (4.12)$$

onde $a_0^i = 1$, tem-se que

$$G(s) = \frac{b_{0i}}{s^2 + a_{1i}s + a_{2i}}, \quad (4.13)$$

removendo a ação integral do controlador e inserindo na planta, tem-se que

$$F_i(s) = \frac{\omega_i(s)}{U_i^c(s)} = \frac{b_{0i}}{s^3 + a_{1i}s^2 + a_{2i}s}, \quad (4.14)$$

onde

$$U_i^c(s) = (K_{P_i}s + K_{I_i} + K_{D_i}s^2)E_i(s), \quad (4.15)$$

logo

$$\omega_i(s)s^3 + \omega_i(s)a_{1i}s^2 + \omega_i(s)a_{2i}s = b_{0i}U_i^c(s), \quad (4.16)$$

aplicando a transformada inversa de Laplace

$$\frac{d^3\omega_i(t)}{dt^3} + a_{1i}\frac{d^2\omega_i(t)}{dt^2} + a_{2i}\frac{d\omega_i(t)}{dt} = b_{0i}u_i^c(t), \quad (4.17)$$

em variáveis de estado tem-se que

$$\begin{aligned} x_1 &= \omega_i(t), \\ x_2 &= \dot{x}_1 = \frac{d\omega_i(t)}{dt}, \end{aligned} \quad (4.18)$$

$$x_3 = \dot{x}_2 = \frac{d^2 \omega_i(t)}{dt^2},$$

e

$$\dot{x}_3 = \frac{d^3 \omega_i(t)}{dt^3}.$$

A Eq. (4.17) em variáveis de estados é dada por

$$\dot{x}_{3i} + a_{1i}x_{3i} + a_{2i}x_{2i} = b_{0i}u_i^c(t). \quad (4.19)$$

O sistema representado em espaço de estados levando-se em consideração a dinâmica dos dois atuadores é dado por

$$\begin{bmatrix} \dot{x}_e \\ \dot{x}_d \end{bmatrix} = \begin{bmatrix} A_1 & \mathbf{0} \\ \mathbf{0} & A_2 \end{bmatrix} \begin{bmatrix} x_e \\ x_d \end{bmatrix} + \begin{bmatrix} B_1 & \mathbf{0} \\ \mathbf{0} & B_2 \end{bmatrix} \begin{bmatrix} u_1^c(t) \\ u_2^c(t) \end{bmatrix}, \quad (4.20)$$

sendo $x_e = [x_{e1} \quad x_{e2} \quad x_{e3}]$ e $x_d = [x_{d1} \quad x_{d2} \quad x_{d3}]$. Observa-se que a matriz de transição de estados e a matriz de entrada, respectivamente, são dadas por

$$A_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -a_{2i} & -a_{1i} \end{bmatrix} \quad (4.21)$$

$$B_i = [0 \quad 0 \quad b_{0i}]^T.$$

O sistema representado pela Eq. (4.20) é utilizado para o cálculo dos ganhos de realimentação de estados através da teoria do LQR. Desta forma, se faz necessário a associação destes ganhos aos ganhos do controlador PID, então a Eq. (3.6) sem o integrador no plano- s é dada por

$$C(s) = \frac{U_i(s)}{E_i(s)} = K_{D_i}s^2 + K_{P_i}s + K_{I_i}, \quad (4.22)$$

sendo $E_i(s) = \omega_i^r(s) - \omega_i(s)$, onde $\omega_i^r(s)$ é o sinal da entrada de referência, aplicando a transformada inversa de Laplace na Eq. (4.22), tem-se que

$$u_i^c(t) = -K_{D_i} \frac{d^2 \omega_i(t)}{dt^2} - K_{P_i} \frac{d\omega_i(t)}{dt} - K_{I_i} \omega_i(t) + \check{\omega}_i^r(t), \quad (4.23)$$

em variáveis de estados a Eq. (4.23) é dada por

$$\begin{bmatrix} u_e^c \\ u_d^c \end{bmatrix} = - \begin{bmatrix} K_{I_e} & K_{P_e} & K_{D_e} & 0 & 0 & 0 \\ 0 & 0 & 0 & K_{I_d} & K_{P_d} & K_{D_d} \end{bmatrix} \begin{bmatrix} x_e \\ x_d \end{bmatrix} + \begin{bmatrix} \check{\omega}_e^r \\ \check{\omega}_d^r \end{bmatrix}, \quad (4.24)$$

O sistema em malha fechada é ilustrado na Figura 4.2, a qual apresenta a estrutura do modelo no plano-s.

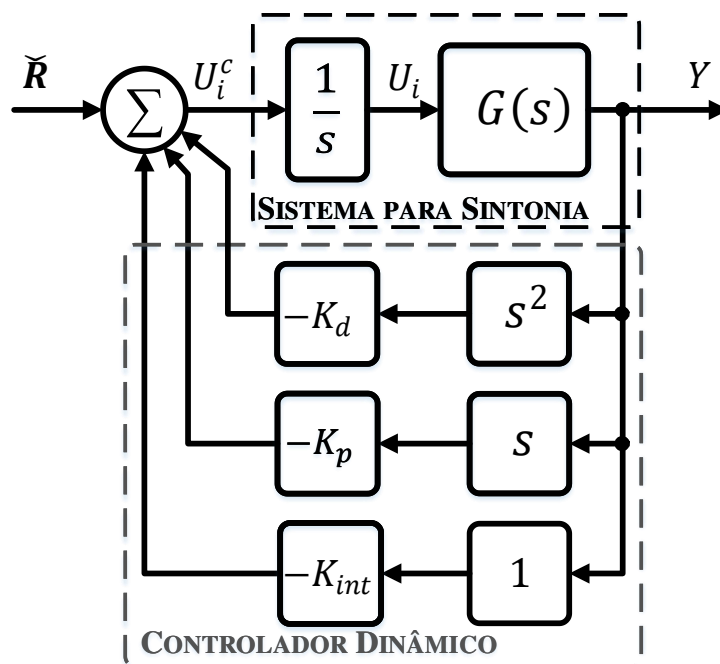


Figura 4.2 – Diagrama de Blocos do Sistema em Malha Fechada no Plano-s.

Na estrutura do modelo do sistema em malha fechada, na Figura 4.2, $\check{R} = \check{\omega}_r^i(s)$ e $Y = \omega(s)$ e observa-se que os ganhos de realimentação de estados podem ser diretamente obtidos através da saída, o que viabiliza sua implementação, dispensando a utilização de um observador de estados, simplificado a estrutura do modelo.

4.1.2.1 Cálculo dos ganhos de realimentação de estados terceira ordem

Durante o processo da seleção da matriz Q , nota-se que ao se calcular os ganhos do controlador pelo LQR, o valor do ganho de realimentação referente a x_3 , ganho derivativo K_D , torna-se pequeno em relação aos demais ganhos (K_P e K_I). Ao se desenvolver a ARE (Eq.(3.21)), em partes de forma que o ganho K_D seja igual a zero, é obtida a seguinte relação

$$A^T P + PA = \begin{bmatrix} 0 & p_{11} - a_2 p_{31} & p_{21} - a_1 p_{31} \\ p_{11} - a_2 p_{31} & 2 * (p_{21} - a_2 p_{32}) & p_{22} + p_{31} - a_1 p_{32} - a_2 p_{33} \\ p_{21} - a_1 p_{31} & p_{22} + p_{31} - a_2 p_{33} - a_1 p_{32} & 2 * (p_{32} - a_1 p_{33}) \end{bmatrix}. \quad (4.25)$$

Para segunda parte da ARE, tem-se que

$$-PBR^{-1}B^T P = -PBK, \quad (4.26)$$

então

$$-PBK = - \begin{bmatrix} p_{31}b_0K_I & p_{31}b_0K_P & p_{31}b_0K_D \\ p_{32}b_0K_I & p_{32}b_0K_P & p_{32}b_0K_D \\ p_{33}b_0K_I & p_{33}b_0K_P & p_{33}b_0K_D \end{bmatrix}. \quad (4.27)$$

Por fim tem-se a matriz Q , dada por

$$Q = \begin{bmatrix} q_{11} & 0 & 0 \\ 0 & q_{22} & 0 \\ 0 & 0 & q_{33} \end{bmatrix}. \quad (4.28)$$

Desta forma, realizando as manipulações matemáticas, tem-se que

$$2p_{32} - 2a_1p_{33} - p_{33}b_0K_D + q_{33} = 0 \quad (4.29)$$

$$K_D = \frac{2p_{32} - 2a_1p_{33}}{p_{33}b_0} + \frac{q_{33}}{p_{33}b_0}$$

De acordo com a equação expressa acima, nota-se que o valor do ganho K_d está em função de a_1 , p_{32} , p_{33} , b_0 e q_{33} , logo, para que seja aproximadamente ou igual a zero isto dependerá destas variáveis.

4.1.3 Modelo de sintonia de n^a ordem

A metodologia aplicada nos modelos matemáticos de primeira e segunda ordem, baseia-se inicialmente no levantamento do modelo matemático no plano- s e em seguida é manipulado matematicamente de forma a se obter sua representação no espaço de estados, viabilizando a aplicação da teoria de controladores ótimos, no caso o LQR.

Desta maneira analisando a estrutura dos modelos apresentado pela Eq. (4.8) (Araújo Filho & da Fonseca Neto, 2017) e o modelo de sintonia de terceira ordem apresentado na Eq. (4.20), o sistema para controladores tipo PID com derivadas dos sinais de saída de ordem $(n - 1)$ é dado por

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & \vdots & 0 & \ddots & \vdots \\ \vdots & 0 & 0 & \cdots & 1 \\ 0 & -a_n & \cdots & -a_2 & -a_1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_0 \end{bmatrix} u_i(t). \quad (4.30)$$

A matriz de transição de estados, matriz A , da Eq. (4.30) é composta por um vetor de zeros de ordem $(n \times 1)$ na primeira coluna, e em sua ultima linha, exceto o primeiro elemento, é composta por um vetor que por sua vez é formado pelos coeficientes do denominador da função tranferência da planta do coeficiente do termo de menor ordem (s^0) até o termo a_1

(s^{n-1}) , considerando que o termo $a_0 = 1$, o bloco acima desta ultima linha é composto por uma matriz identidade de ordem $(n - 1) \times (n - 1)$.

4.1.4 Heurística QR^{-1} para sintonia LQR

A heurística estabelecida para sintonia LQR do controlador PID é baseada nas especificações de projeto para avaliar o desempenho do método. As métricas estabelecidas são baseadas nas especificações de controle, onde estas especificações devem atender as métricas técnicas, tais como: w_n , ζ e T_s , bem como, as especificações de operação, como: consumo mínimo de energia que está associado ao integrador da Eq. (3.17).

Com base nas Eqs. (3.19) e (3.21), pode-se observar que a matriz de ganho K^* tem forte influência das matrizes Q e R do índice de desempenho dado pela Eq. (3.17). Referente a dependência paramétrica e os limites numéricos de projetos estabelecidos das matrizes Q e R , o modelo de sintonia ótima é baseada nas Eqs. (3.19) e (3.21) que é dada por

$$K^*(Q, R) \approx f_1(R) + f_2(Q, R), \quad (4.31)$$

onde f_1 e f_2 representa a influências dessas matrizes nos valores da matriz de ganho. A escolha de Q e R estabelecerá as principais diretrizes para se determinar os ganhos do controlador que satisfaçam as especificações de projeto. Essa formulação é demonstrada em (Da Fonseca Neto, et al., 2010) no tempo discreto. Ainda, observa-se nas Eqs. (3.20) e (3.22) e conforme apresentado na Eq. (3.24), que a matriz de ganhos de realimentação ótimos K^* depende das matrizes de ponderação. Conseqüentemente, a lei de controle dada pelas Eqs. (4.9) e (4.24) escrita como

$$u_i(Q, R) = -K_{QR_i}x + \check{\omega}_i^r. \quad (4.32)$$

4.2 Controlador PID Adaptativo Baseado em RNA

O controlador PID clássico possui pouca flexibilidade, pois seus parâmetros são invariantes, na maioria dos casos não é possível obter bons resultados quando aplicado a sistemas com grandes níveis de perturbação. Como a Rede Neuronal Artificial tem uma forte capacidade de adaptação, é amplamente utilizada em vários problemas de controle (Liu, 2008).

A fim de evitar comportamentos indesejáveis do PID clássico, a RNA e o PID são combinados para o fornecer o esforço de controle. Portanto, o controlador PID da rede neuronal surge e sua representação é dada pelo diagramas de blocos na Figura 4.3.

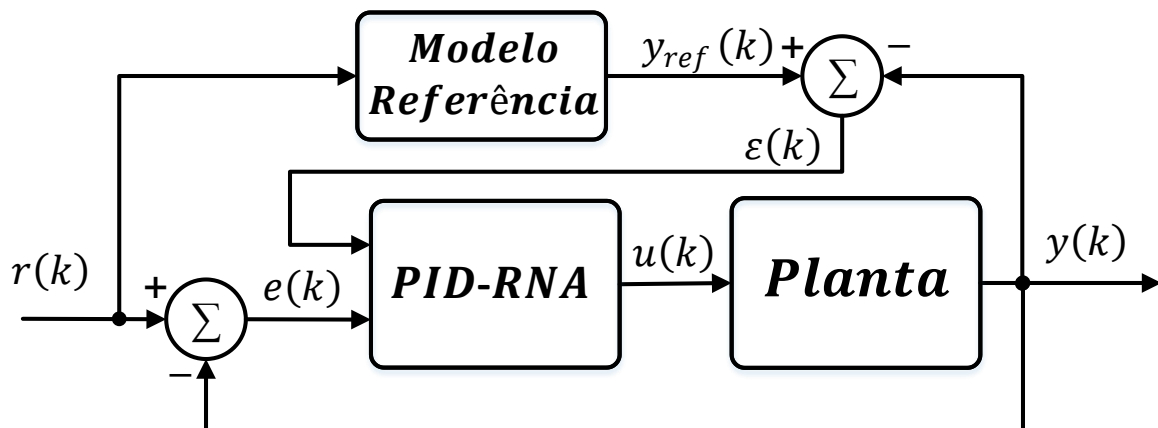


Figura 4.3 – Sistema em Malha Fechado Utilizando o Controlador PID-RNA.

A aplicação de um controlador PID clássico em plantas que possuem grandes variações paramétricas e/ou não-linearidades na maioria dos casos é desvantajoso ou até mesmo inviável. A vantagem da aplicação de um PID-RNA, neste caso, é a sua habilidade de se adaptar a estes comportamentos indesejados.

A saída do controlador PID digital, para implementação do algoritmo de controle, pode ser obtida aplicando a transformada Z na saída do controlador PID no tempo contínuo, Eq. (3.5), e é dada por (Ogata, 1996)

$$U(z) = \left(K_P + \frac{K_I}{(1 - z^{-1})} + K_D(1 - z^{-1}) \right) E(z), \quad (4.33)$$

onde os estados de entradas podem ser definidos como

$$x_i(k) = [x_1(k) \quad x_2(k) \quad x_3(k)]^T, \quad (4.34)$$

sendo o vetor de pesos da rede neuronal é dado por

$$w_i = [w_1 \quad w_2 \quad w_3], \quad (4.35)$$

onde os ganhos do controlador PID-RNA, são dados por

$$\begin{aligned} K_P &= K_c \cdot w_1 \\ K_I &= K_c \cdot w_2 \\ K_D &= K_c \cdot w_3 \end{aligned} \quad (4.36)$$

sendo K_c o valor de ganho dos pesos e tem influência na velocidade de seguimento da entrada de referência do sistema de controle. O valor de K_c é determinado de forma empírica

observando-se os resultados. Para uma resposta mais rápida necessita-se de um ganho maior e para uma resposta mais lenta um ganho menor deve ser selecionado.

4.2.1 Treinamento

O treinamento da rede baseia-se na regra *Hebbiana* Supervisionada (Liu, 2008), conforme apresentada na Seção 3.5, a qual incrementa os pesos a cada iteração através de um fator dado por

$$\Delta w_i(k) = \eta_i \cdot \varepsilon(k) \cdot u(k) \cdot x_i(k), \quad (4.37)$$

sendo $i = 1, 2, 3$ e η_i representa o fator de aprendizagem e $\eta_1 = \eta_P$, $\eta_2 = \eta_I$ e $\eta_3 = \eta_D$ responsáveis por ponderar o treinamento dos valores dos ganhos proporcional, integral e derivativo, respectivamente e $\varepsilon(k)$ é a diferença entre a saída do modelo de referência e a saída medida da planta dado por $\varepsilon(k) = y_{ref}(k) - y(k)$.

O vetor de pesos w_i , de acordo com a Eq.(4.37), está sujeito a variações numéricas ao longo das iterações. Afim de limitar os pesos da rede, para valores finitos, (Rivera-Mejía, et al., 2012) a seguinte regra foi acrescentada

$$\bar{w}_i(k) = \frac{w_i(k)}{\left\| \sum_{i=1}^3 w_i(k) \right\|}. \quad (4.38)$$

Desta maneira, de acordo com a Eq. (4.37) a atualização dos pesos é dada por

$$w_i(k+1) = w_i(k) + \eta_i \cdot \varepsilon(k) \cdot u(k) \cdot x_i(k), \quad (4.39)$$

A topologia do controlador PID-RNA é apresentada na Figura 4.3, a qual ilustra o método de forma geral o algoritmo de controle e a regra de treinamento da rede, para conclusão da RNA é necessário se obter o vetor de estados para o cálculo da saída do controlador, que pode ser obtidos de duas formas apresentas a seguir.

4.2.1 Conversor de estados 1

O vetor de estados para implementação da rede neuronal pode ser obtido aplicando a transformada Z inversa na Eq. (4.33), para obtenção de uma aproximação de $u(k)$ no tempo discreto, tem-se

$$u(k) = K_p e(k) + K_I \sum e(k) \cdot \Delta t + K_D \frac{\Delta e(k)}{\Delta t}, \quad (4.40)$$

onde $\Delta e(k) = e(k) - e(k - 1)$ e $\Delta t = T_s$ representa a taxa de amostragem. Desta forma para obtenção da estrutura do PID-RNA necessita-se obter os sinais de entrada da rede, onde a Eq. (4.40) é reescrita como

$$u(k) = K_p x_1(k) + K_I x_2(k) + K_D x_3(k), \quad (4.41)$$

onde o estado ponderado pelo ganho proporcional é dado por

$$x_1(k) = e(k), \quad (4.42)$$

e o estado ponderado pelo ganho integral, é dado por

$$x_2(k) = \sum e(k) \cdot \Delta t = \sum e(k) \cdot T_s, \quad (4.43)$$

e por fim, o estado ponderado pelo ganho derivativo é dado por

$$x_3(k) = \frac{\Delta e(k)}{\Delta t} = \frac{(e(k) - e(k - 1))}{T_s}. \quad (4.44)$$

As Eqs (4.42), (4.43) e (4.44) expressam o conversor de estados, logo, a saída da rede neuronal é dada por

$$u(k) = K_c \cdot \sum_{i=1}^3 x_i(k) \cdot w_i(k). \quad (4.45)$$

O algoritmo de controle para este conversor de estados pode ser ilustrado pela Figura 4.4, na qual é ilustrada a estrutura do controlador PID-RNA em diagrama de blocos.

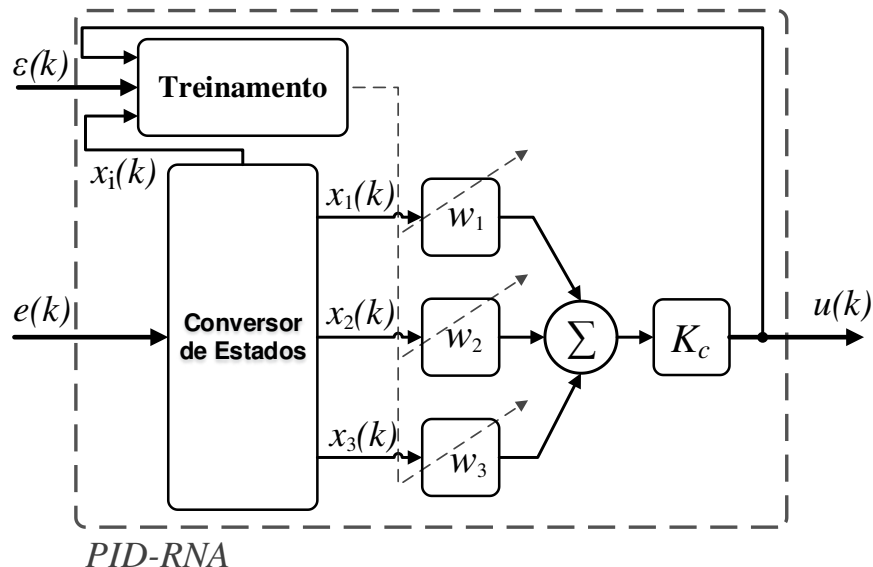


Figura 4.4 – Estrutura Controlador PID-RNA com Conversor de Estado 1.

4.2.2 Conversor de estados 2

Outra forma de obtenção dos estados da RNA é reescrevendo a Eq. (4.33) da seguinte forma

$$U(z) = \left(\frac{K_I + K_P(1 - z^{-1}) + K_D(1 - z^{-1})^2}{(1 - z^{-1})} \right) E(z), \quad (4.46)$$

ainda,

$$U(z)(1 - z^{-1}) = (K_I + K_P(1 - z^{-1}) + K_D(1 - z^{-1})^2)E(z), \quad (4.47)$$

aplicando a transformada Z inversa, para obtenção de uma aproximação de $u(k)$ no tempo discreto, tem-se que

$$\frac{\Delta u(k)}{\Delta t} = \left(K_I e(k) + K_P \frac{\Delta e(k)}{\Delta t} + K_D \frac{\Delta^2 e(k)}{\Delta t^2} \right), \quad (4.48)$$

sendo $\Delta u(k) = u(k) - u(k - 1)$ e $\Delta^2 e(k)$ é dado por

$$\Delta^2 e(k) = \Delta e(k) - \Delta e(k - 1), \quad (4.49)$$

desta forma

$$\Delta^2 e(k) = e(k) - e(k - 1) - (e(k - 1) - e(k - 2)), \quad (4.50)$$

logo

$$\Delta^2 e(k) = e(k) - 2e(k - 1) + e(k - 2), \quad (4.51)$$

então o termo $\Delta^2 e(k)/\Delta t^2$ é dado por

$$\frac{\Delta^2 e(k)}{\Delta t^2} = \left(\frac{e(k) - e(k - 1)}{T_s} - \left(\frac{e(k - 1) - e(k - 2)}{T_s} \right) \right) \cdot \frac{1}{T_s}. \quad (4.52)$$

A Eq. (4.48) pode ser simplificada através de manipulações matemáticas e ser reescrita como

$$\Delta u(k) = \left(K_P \frac{\Delta e(k)}{\Delta t} + K_I e(k) + K_D \frac{\Delta^2 e(k)}{\Delta t^2} \right) \Delta t, \quad (4.53)$$

simplificando a equação acima tem-se que

$$u(k) = u(k - 1) + K_P \Delta e(k) + K_I e(k) \Delta t + K_D \frac{\Delta^2 e(k)}{\Delta t}, \quad (4.54)$$

desta maneira, a equação anterior pode ser manipulada em função dos seus estados e é dada por

$$u(k) = u(k - 1) + (K_P x_1(k) + K_I x_2(k) + K_D x_3(k)), \quad (4.55)$$

esta equação leva em consideração a organização dos ganhos K_P , K_I e K_D no vetor de pesos da rede neuronal apresentado pelas Eqs. (4.35) e (4.36), assim o estado ponderado pelo ganho proporcional é dado por

$$x_1(k) = \Delta e(k), \quad (4.56)$$

e o estado ponderado pelo ganho integral é dado por

$$x_2(k) = e(k) \cdot T_s, \quad (4.57)$$

por fim, o estado ponderado pelo ganho derivativo é dado por

$$x_3(k) = \frac{\Delta^2 e(k)}{T_s} = \frac{e(k) - 2e(k - 1) + e(k - 2)}{T_s}. \quad (4.58)$$

A fim de se obter uma estrutura mais simples o estado $x_3(k)$ pode ser reescrito como

$$x_3(k) = \left(\frac{e(k) - e(k - 1)}{T_s} - \left(\frac{e(k - 1) - e(k - 2)}{T_s} \right) \right) = x_1(k) - x_1(k - 1). \quad (4.59)$$

As Eqs. (4.55) - (4.59) expressam o conversor de estados ilustrado na Figura 4.5, logo, a saída da rede neuronal, é dada por

$$u(k) = u(k - 1) + K_c \cdot \sum_{i=1}^3 x_i(k) \cdot w_i(k). \quad (4.60)$$

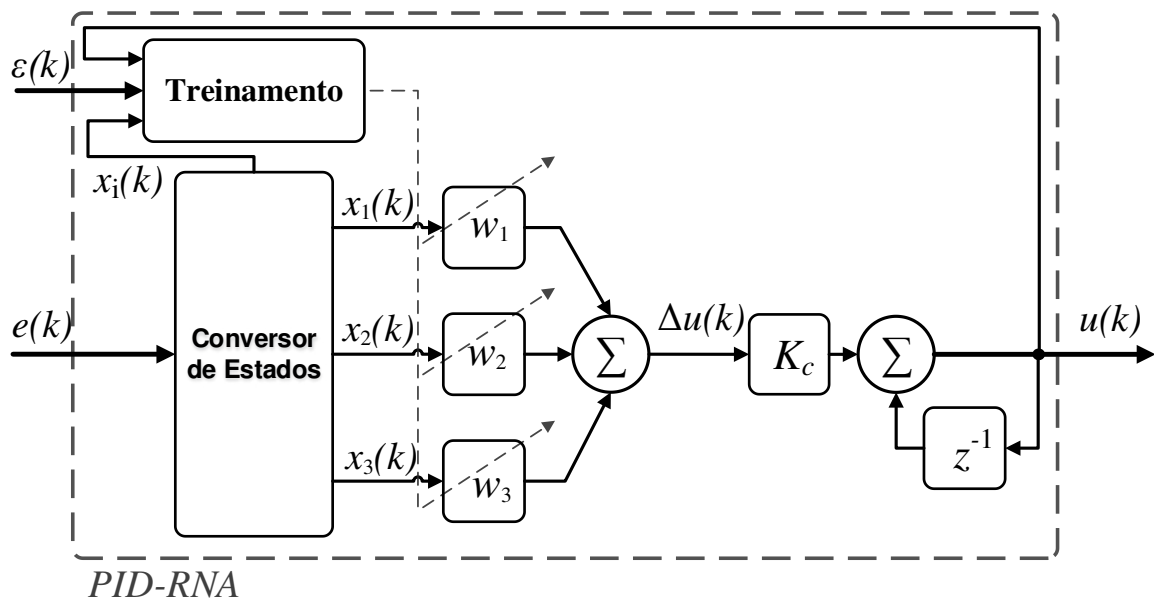


Figura 4.5 – Estrutura Controlador PID-RNA com Conversor de Estado 2.

Nas Figura 4.4 e Figura 4.5, são apresentados os conversores de estados, que geram as entradas da rede e são descritos no primeiro conversor pelas Eqs. (4.42)-(4.44) e para o segundo pelas Eqs. (4.57)-(4.59). A adaptação é feita através do treinamento da rede baseada na Eq. (4.39). O algoritmo de controle ilustrado pela Figura 4.4, é descrito de forma mais detalhada pelo Algoritmo PID-RNA, dado por

Algoritmo PID-RNA 1

- **Inicialização dos parâmetros**
 - 1 T_s ← Período de Amostragem.
 - 2 ζ ← Fator de Amortecimento – Referência.
 - 3 ω_n ← Frequência Natural – Referência.
 - 4 η_i ← Vetor do Fator de Aprendizagem.
 - 5 $w_i(0)$ ← Vetor de Pesos Inicial.
 - 6 K_c ← Coeficiente Proporcional da Rede.
- **Processo Iterativo**
 - 7 Medição $y(k)$
 - 8 Medição $r(k)$
 - 9 Gera $y_{ref}(k, \zeta, \omega_n, T_s)$
 - 11 $e(k) = r(k) - y(k)$
 - 10 $\varepsilon(k) = y_{ref}(k) - y(k)$
- **Conversão de Estados**
 - 14 Seleciona Conversor (i)
 - 15 Para $i = 1 \rightarrow$ Eqs. (4.42) - (4.44)
 - 16 Para $i = 2 \rightarrow$ Eqs. (4.56) - (4.59)
- **Normalização dos Pesos**
 - 20
$$w_i(k) = \frac{w_i(k)}{\left\| \sum_{i=1}^3 w_i(k) \right\|}$$
- **Saída da Rede**
 - 18 Para $i = 1 \rightarrow$ Eqs. (4.45)
 - Para $i = 2 \rightarrow$ Eqs. (4.60)
- **Treinamento da Rede – Regra Hebb**
 - 22 $w_1(k+1) = w_1(k) + \eta_P \cdot \varepsilon(k) \cdot u(k) \cdot x_1(k)$
 - 23 $w_2(k+1) = w_2(k) + \eta_I \cdot \varepsilon(k) \cdot u(k) \cdot x_2(k)$
 - 24 $w_3(k+1) = w_3(k) + \eta_D \cdot \varepsilon(k) \cdot u(k) \cdot x_3(k)$
- **Fim processo iterativo**

Desta forma é visível que o ajustes dos ganhos são realizados automaticamente o que possibilita que o controlador se adapte as perturbações. De modo geral, o algoritmo pode ter sua resposta ajustada através do ganho K_c e do fator de aprendizagem, que baseia-se em uma heurística através da observação da resposta do sistema e da análise da convergência dos valores dos pesos ao longo das iterações.

Capítulo 5

EXPERIMENTOS COMPUTACIONAIS

Neste capítulo são apresentados os resultados obtidos através da proposta exposta nos capítulos anteriores, através de simulações realizadas em ambiente MATLAB[®]. Para avaliação do método dois modelos matemáticos foram utilizados para diversas trajetórias de referências, afim de se observar o comportamento cinemático e dinâmico do sistema e a influência deste modelo.

5.1 Experimentos Computacionais PID-LQR

O projeto proposto foi avaliado baseando-se em seu modelo equivalente discreto, ou também chamado de emulado. Conforme exposto no Capítulo 3, o projeto foi estruturado no tempo contínuo e nesta etapa de avaliação os modelos obtidos a partir da proposta foram digitalizados e em seguida a experimentação computacional foi realizada, levando-se em consideração a abordagem apresentada em (Franklin, et al., 2010). Nota-se que para esta proposta, o projeto do controlador é feito de forma *off-line*, ou seja, primeiramente realiza-se o projeto de acordo com a metodologia proposta e em seguida aplica-se os resultados obtidos são implementados no sistema de controle em malha fechada apresentado. Para mais detalhes, o algoritmo utilizado para simulações está disponível no Apêndice B.

O processo de sintonia ocorre fixando o valor da matriz R , neste caso $R = 1$, e variando-se os valores da matriz Q afim de se obter os ganhos de realimentação de estado através da teoria do LQR. Algumas matrizes Q foram selecionadas para o cálculo dos ganhos de realimentação de estados K e a partir daí observou-se o comportamento do sistema a uma dada trajetória de referência.

5.1.1 PID-LQR para modelo de sintonia de segunda ordem

O modelo matemático de primeira ordem utilizado para avaliação desta metodologia é dado por

$$G(s) = \frac{9.28}{s + 3.57}. \quad (5.1)$$

Na Tabela 5.1 são apresentados os valores dos ganhos do controlador K_p e K_I , o coeficiente de amortecimento (ζ) e os polos do sistema em malha fechada. O método de sintonia proposto permite a seleção das matrizes de ponderação Q e R , baseado na relação apresentada na Eq. (4.31) que pode ser aproximada por

$$K \approx R^{-1}Q. \quad (5.2)$$

A proposta do projeto do controlador é reduzir o erro de posicionamento a uma dada trajetória de referência através da rotação das rodas esquerda e direita. Nesta primeira etapa é utilizado um ponto como referência localizados nas coordenadas (35, 35) metros.

Tabela 5.1 - Parâmetros de Sintonia e Ganhos do Controlador.

| Q | K_p | K_{int} | ζ | <i>Pólos</i> |
|--|-------|-----------|---------|--------------|
| $Q_1 = \begin{bmatrix} 0.1 & 0 \\ 0 & 1 \end{bmatrix}$ | 0.71 | 0.31 | 1 | -0.2950 |
| | | | | -9.9681 |
| $Q_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0.1 \end{bmatrix}$ | 0.29 | 1.00 | 1 | -2.3032 |
| | | | | -4.0379 |
| $Q_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ | 0.78 | 1.00 | 1 | -0.9367 |
| | | | | -9.9284 |
| $Q_4 = \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix}$ | 0.96 | 3.16 | 1 | -3.1031 |
| | | | | -9.4774 |
| $Q_5 = \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix}$ | 2.83 | 1.00 | 1 | -0.3139 |
| | | | | -29.6270 |
| <i>Método de Ziegler-Nichols</i> | 0.404 | 4.36 | 0.58 | -3.68 + 5.2i |
| | | | | -3.68 - 5.2i |

Baseados nos dados fornecidos pela Tabela 5.1, nota-se que para qualquer valor de Q o sistema permanece estável, pelo fato de todos seus polos obtidos possuem parte real negativa, analisando os dados de forma mais específica, observa-se que quando se utiliza Q_4 para sintonizar o controlador os dois polos do sistema em malha fechada estão mais distantes da origem, fazendo com que o sistema apresente uma resposta mais rápida e isto pode ser observado na Figura 5.1, que apresenta as velocidades angulares das rodas esquerda e direita com a aplicação do controlador dinâmico proposto.

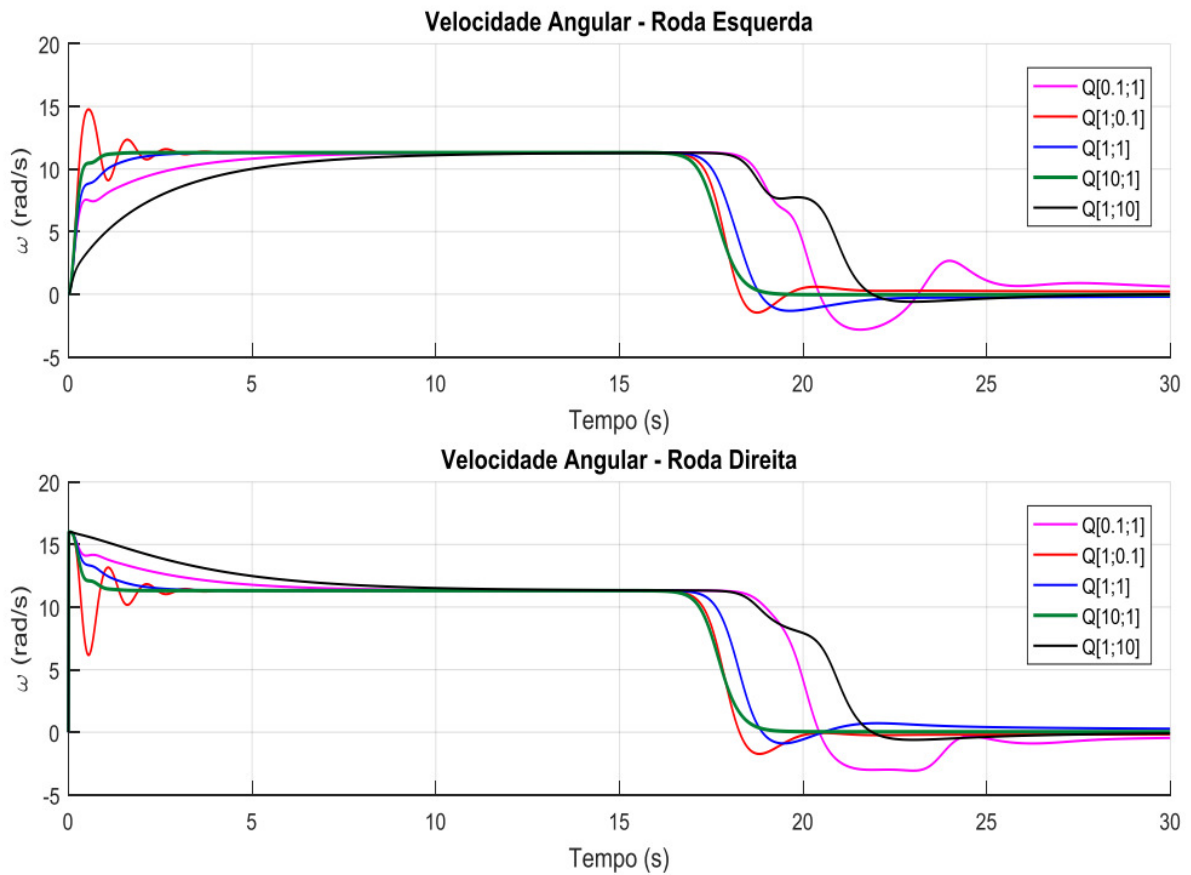


Figura 5.1 – Velocidades Angulares das Rodas Esquerda e Direita para uma Trajetória de Referência Pontual Sistema para o Modelo de Sintonia de Segunda Ordem.

A cadeira de rodas é projetada para seguir determinada trajetória de referência de forma suave, desta maneira, é necessário determinar os ganhos de realimentação de estados com base nesta premissa. Além da resposta do sistema, se faz necessário analisar o esforço de controle proposto (Figura 5.2), com base na tensão aplicada em cada um dos motores, o que possibilita analisar a viabilidade do projeto.

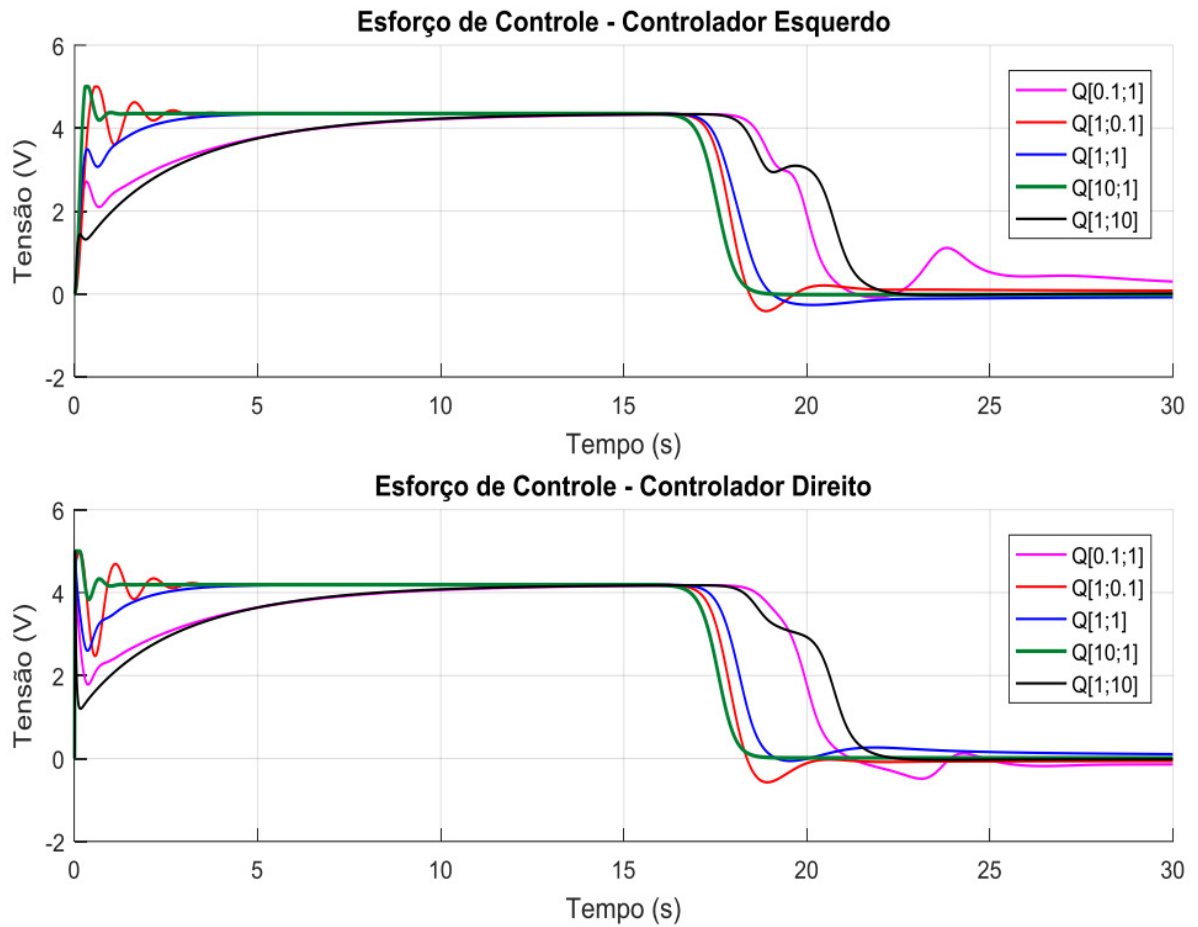


Figura 5.2 – Lei de Controle das Rodas Esquerda e Direita para uma Trajetória de Referência Pontual para o Modelo de Sintonia de Segunda Ordem.

A posição da cadeira de rodas a dadas rotações de suas rodas, pode ser ilustrada pela Figura 5.3, na qual possui um ponto de referência a 35 metros de distância do eixo x e 35 metros de distância do eixo y .

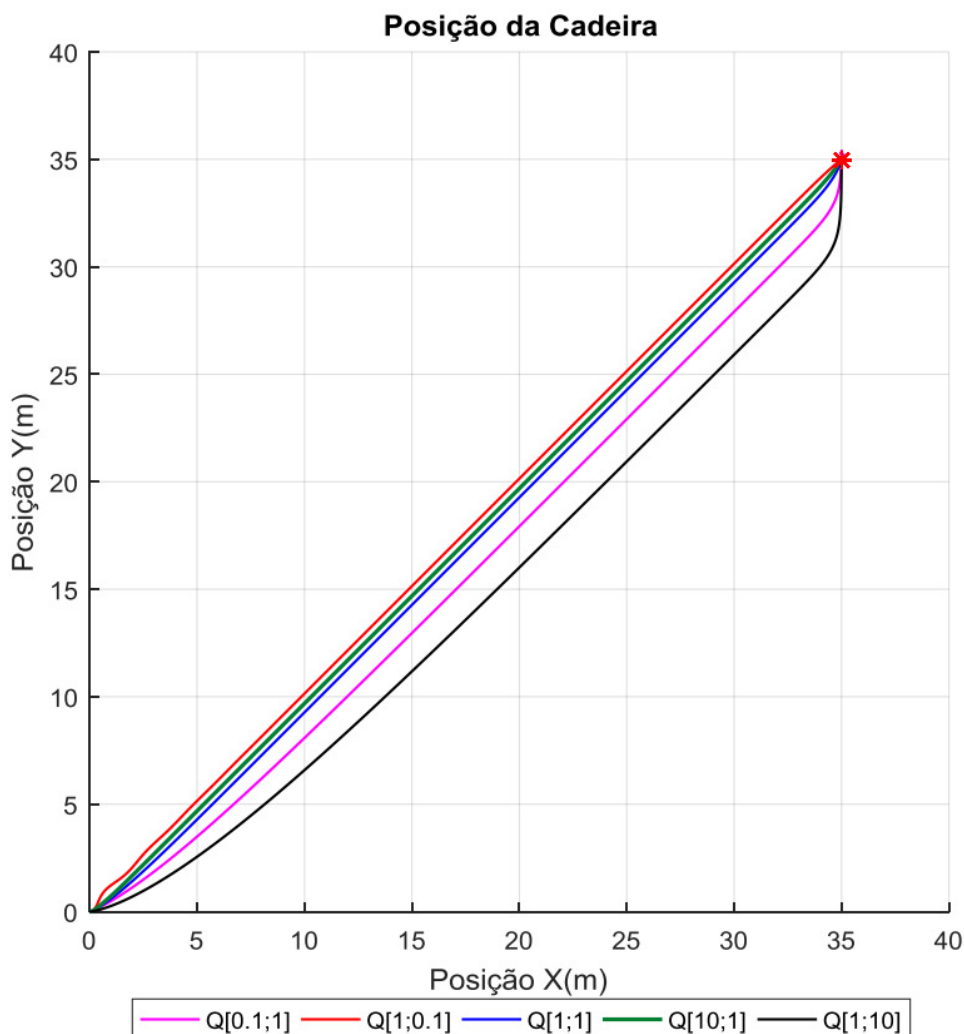


Figura 5.3 – Trajetória da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Segunda Ordem.

O asterisco ilustrado pela Figura 5.3 representa o ponto de destino final. Nota-se que a resposta obtida para Q_4 é a mais rápida e sem oscilações em sua trajetória, o que garante uma menor distância percorrida e conseqüentemente menor gasto de energia. Em relação ao tempo, para este valor de Q o sistema obteve uma resposta de 18 segundos, como mostrado nas figuras anteriores.

Na segunda etapa, novas simulações são realizadas, para os valores apresentados na Tabela 5.1, tendo uma circunferência como trajetória de referência, e, esta deverá possuir raio de 5 metros e velocidade angular de 0.2 rad/s . Na Figura 8 é apresentado a velocidade angular das rodas.

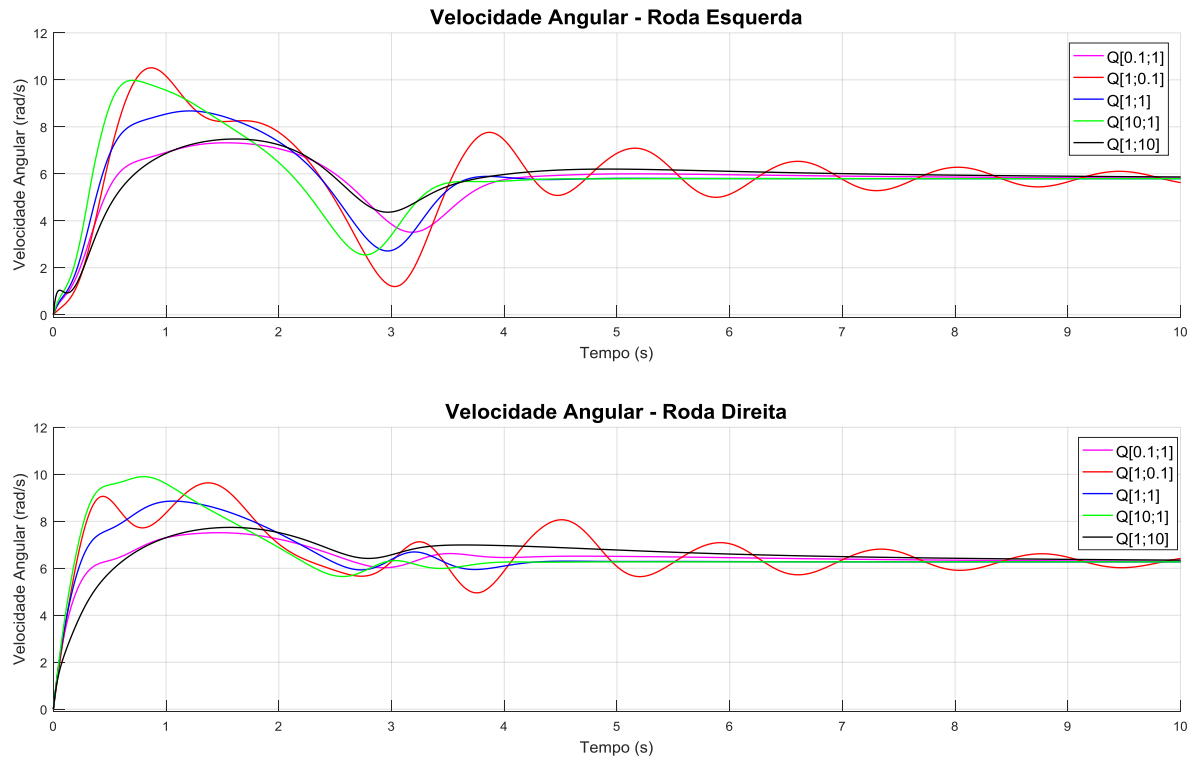


Figura 5.4 – Velocidades Angulares das Rodas Esquerda e Direita para uma Trajetória de Referência Circular para o Modelo de Sintonia de Segunda Ordem.

As velocidades angulares para Q_4 estabilizam em aproximadamente 4 segundos, sem oscilações. A roda direita está a uma velocidade angular maior em comparação a roda esquerda durante o estado estacionário, permitindo que a cadeira de rodas siga a referência circular. O mesmo ocorre com a lei de controle como ilustrado na Figura 5.5.

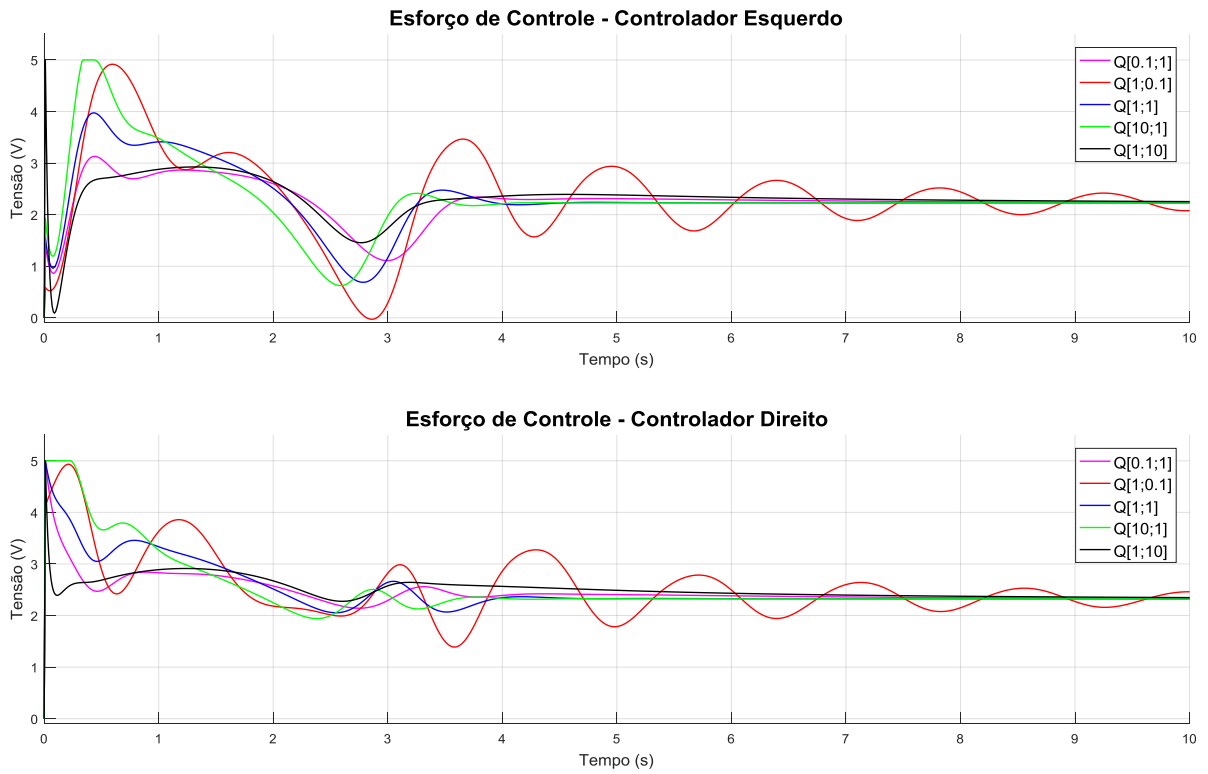


Figura 5.5 – Lei de Controle das Rodas Esquerda e Direita para uma Trajetória de Referência Circular para o Modelo de Sintonia de Segunda Ordem.

A lei de controle apresentada permanece dentro de uma faixa de tensão aceitável a qual torna o projeto do controlador factível e sua resposta pode ser observada na Figura 5.6, a qual mostra a trajetória da cadeira de rodas.

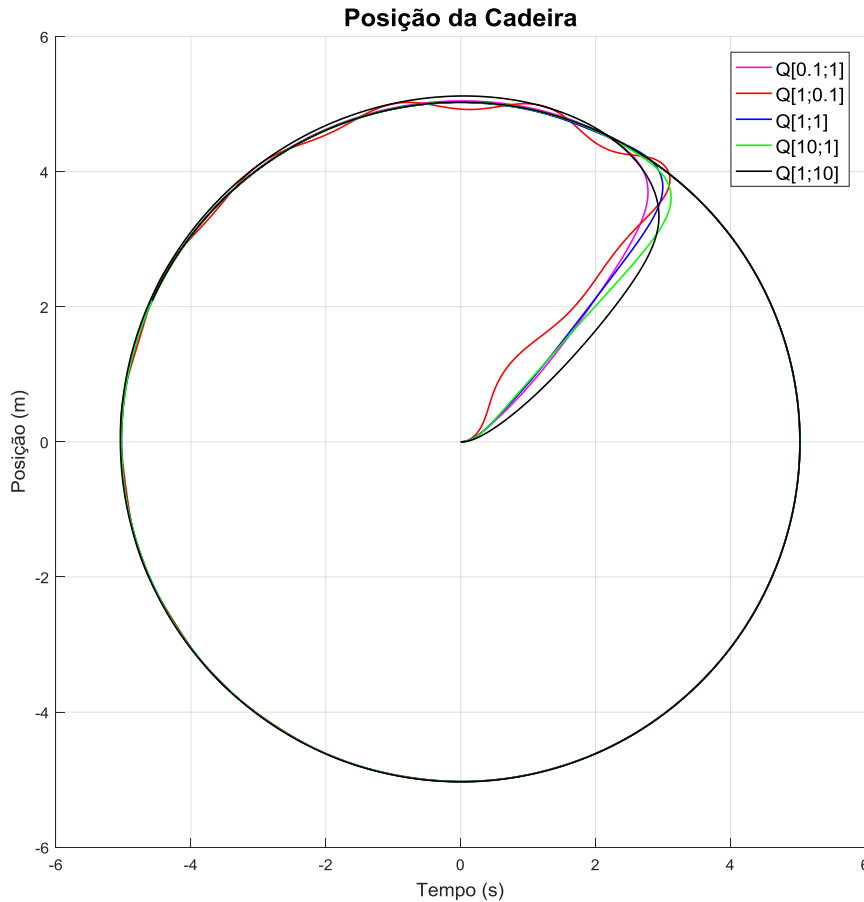


Figura 5.6 – Trajetória Circular da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Segunda Ordem.

5.1.2 PID-LQR para modelo de sintonia de terceira ordem

O modelo matemático de segunda ordem utilizado para avaliação desta metodologia é dado por

$$G(s) = \frac{4613}{s^2 + 496.8s + 16500} , \tag{5.3}$$

substituindo os coeficientes da Eq. (5.3) na Eq. (4.20), tem-se que o sistema em espaço de estados é representado por

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -16500 & -496.8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 4613 \end{bmatrix} u(t) . \tag{5.4}$$

Durante o processo da seleção da matriz Q, nota-se que ao se calcular os ganhos do controlador pelo LQR, o valor do ganho de realimentação referente a x_3 , ganho derivativo K_d , torna-se pequeno em relação aos demais ganhos (K_p e K_i). Então para $q_{33} = 0$, portanto, a matriz Q é dada por

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (5.5)$$

Assim, a solução da ARE é representada por

$$P = \begin{bmatrix} 242.97 & 7.31 & 0.01 \\ 7.31 & 0.23 & 0.00 \\ 0.01 & 0.00 & 0.00 \end{bmatrix}, \quad (5.6)$$

Os valores de p_{32} e p_{33} estão próximos de zero e os ganhos ótimos calculados são dados por

$$K = [1.00 \quad 0.16 \quad 0.00], \quad (5.7)$$

onde o valor de K_d está próximo de zero, caracterizando assim o controlador como sendo um controlador PI, pois $K_p \neq 0$, $K_i \neq 0$ e $K_d \cong 0$. Assim o sistema em malha fechada em espaço de estados é dado por

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -4613 & -17267 & -498 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 4613 \end{bmatrix} r(t). \quad (5.8)$$

A seguir na Tabela 5.2 é apresentada a relação do fator de amortecimento e constante de tempo dos polos para o sistema em malha aberta e em malha fechada.

Tabela 5.2 – Relação de amortecimento e constante para os sistemas em malha aberta e fechada.

| Matrizes de Transição de Estados | Pólos | ζ | τ |
|--|--------|---------|----------|
| $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -16500 & -496.8 \end{bmatrix}$ | 0 | 1 | ∞ |
| | -35.8 | | 0.0279 |
| | -461.0 | | 0.0022 |
| $A_{MF} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -4613 & -17267 & -498 \end{bmatrix}$ | -0.9 | 1 | 3.71 |
| | -10.1 | | 0.0269 |
| | -485.7 | | 0.0022 |

Os resultados apresentados na tabela acima, mostram para este caso o sistema em malha fechada é estável. E para a avaliação do método experimentos computacionais foram realizados utilizando a mesma metodologia aplicada em sistemas de primeira ordem (seção 5.1.1).

Tabela 5.3 - Parâmetros de Sintonia e ganhos do controlador.

| Q | K_p | K_{int} | ζ | <i>Pólos</i> |
|---|-------|-----------|---------|---------------------|
| $Q_1 = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ | 0.71 | 0.31 | 1 | -0.2950 |
| | | | | -10.1625 |
| | | | | -486.3486 |
| $Q_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ | 0.29 | 1.00 | 1 | -2.2974 |
| | | | | -4.0946 |
| | | | | -490.4143 |
| $Q_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ | 0.78 | 1.00 | 1 | -0.9360 |
| | | | | -10.1466 |
| | | | | -485.7236 |
| $Q_4 = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ | 0.97 | 3.16 | 1 | -3.0774 |
| | | | | -9.7957 |
| | | | | -483.9331 |
| $Q_5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ | 2.83 | 1.00 | 1 | -0.3138 |
| | | | | -31.6287 |
| | | | | -464.8637 |
| <i>Método de Ziegler-Nichols</i> | 0.48 | 5.12 | 0.58 | $-4.0562 + 5.6500i$ |
| | | | 0.58 | $-4.0562 + 5.6500i$ |
| | | | 1 | -488.6936 |

Com base nesta tabela, analisa-se a resposta do sistema com os valores de ganhos gerados pelo LQR para cada Q fornecido, onde estas podem ser representadas pelas figuras a seguir. As primeiras simulações são realizadas tendo como trajetória um ponto e em seguida utiliza-se uma trajetória circular, onde são avaliados os sinais de velocidade angular das rodas, as saídas dos controladores e o posicionamento da cadeira de rodas em relação ao eixo x e y .

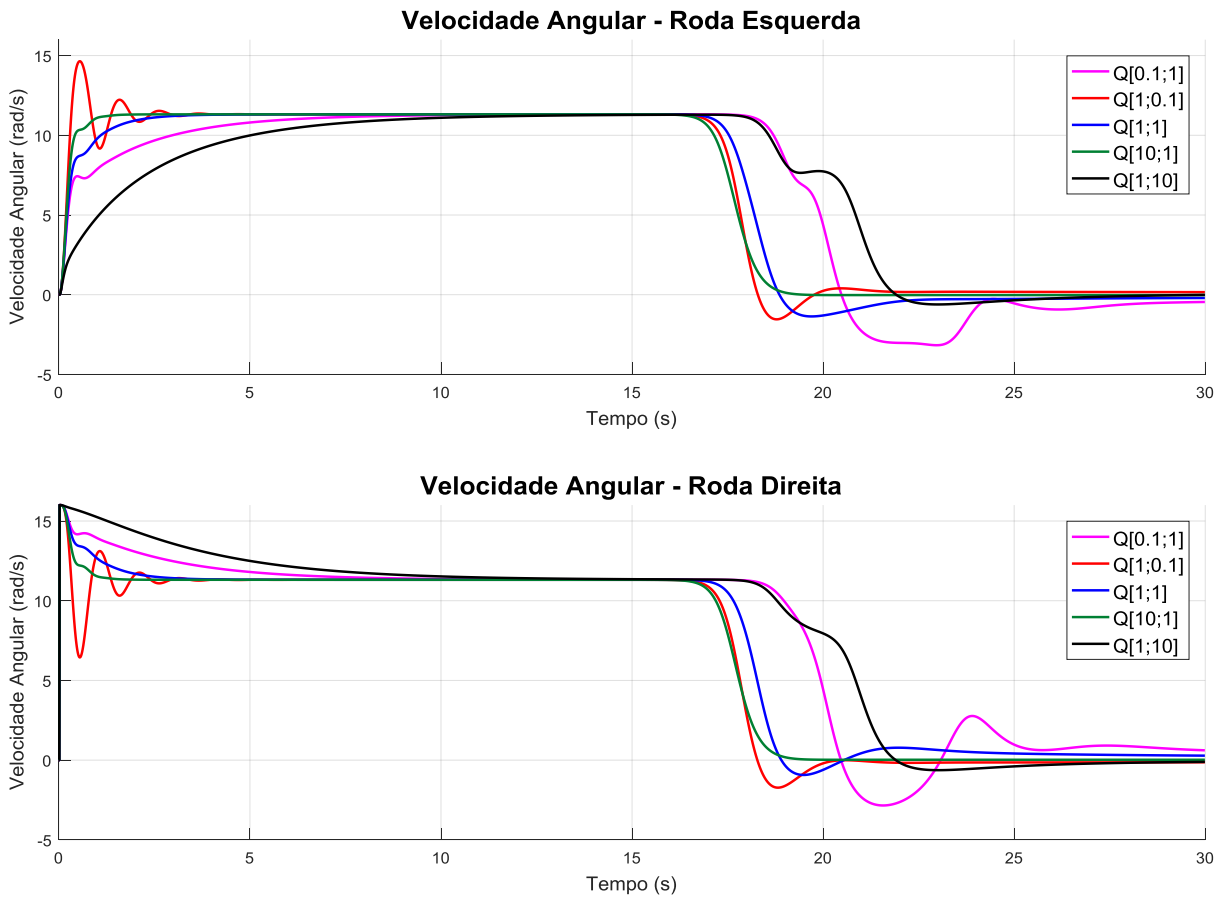


Figura 5.7 – Velocidades Angulares das Rodas Esquerda e Direita para uma Trajetória de Referência Pontual Sistema para o Modelo de Sintonia de Terceira Ordem.

Ao se analisar a resposta para o sistema em malha fechada, nota-se que o comportamento do sistema é semelhante ao apresentado pela Figura 5.1, exceto no intervalo de tempo entre 15 – 25 segundos, no qual o modelo de sintonia de terceira ordem apresenta comportamento distinto, principalmente para os sistema que utilizaram os valores de $Q[1; 1]$ e $Q[0.1; 1]$.

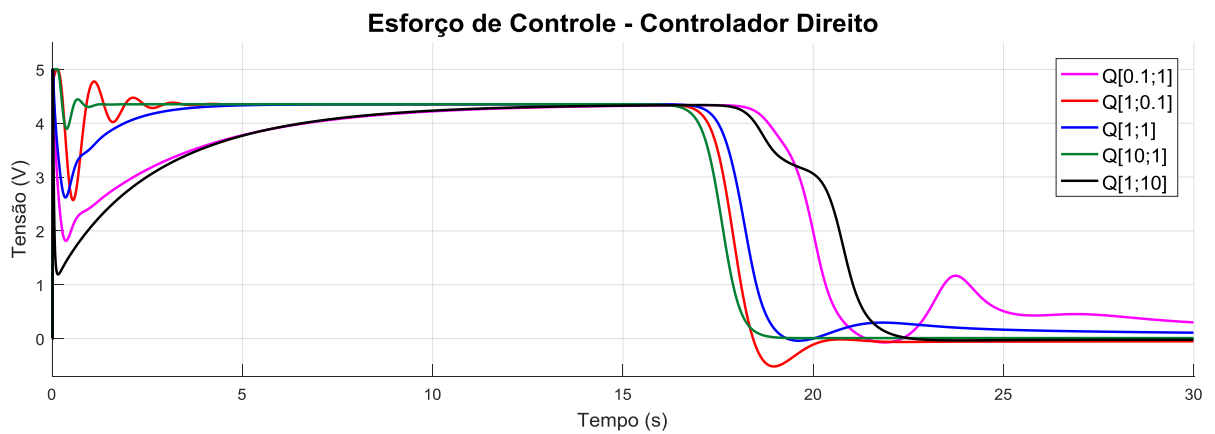
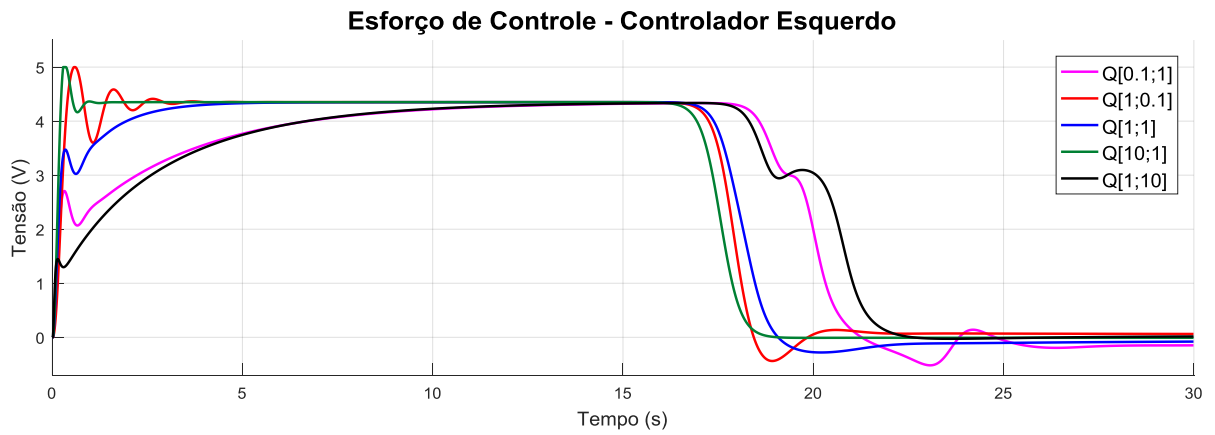


Figura 5.8 – Lei de Controle das Rodas Esquerda e Direita para Uma Trajetória De Referência Pontual Para o Modelo de Sintonia de Terceira Ordem.

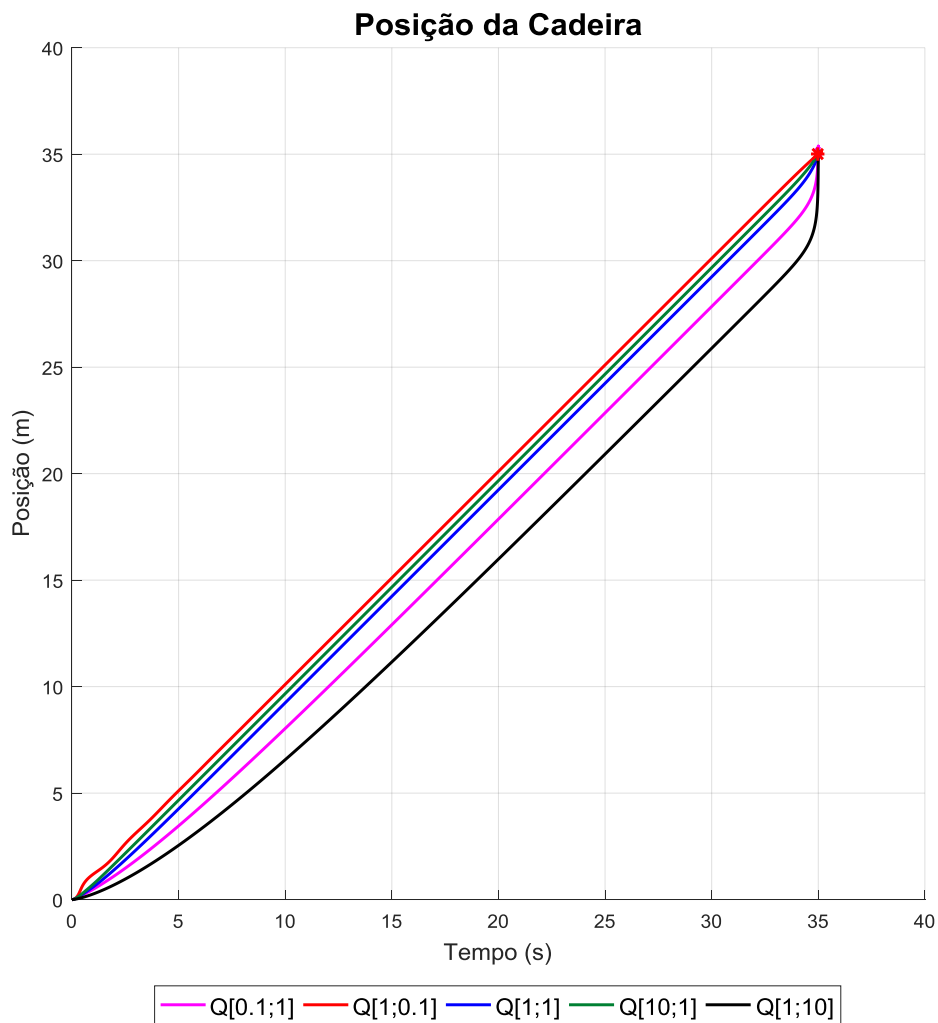


Figura 5.9 – Trajetória da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Terceira Ordem.

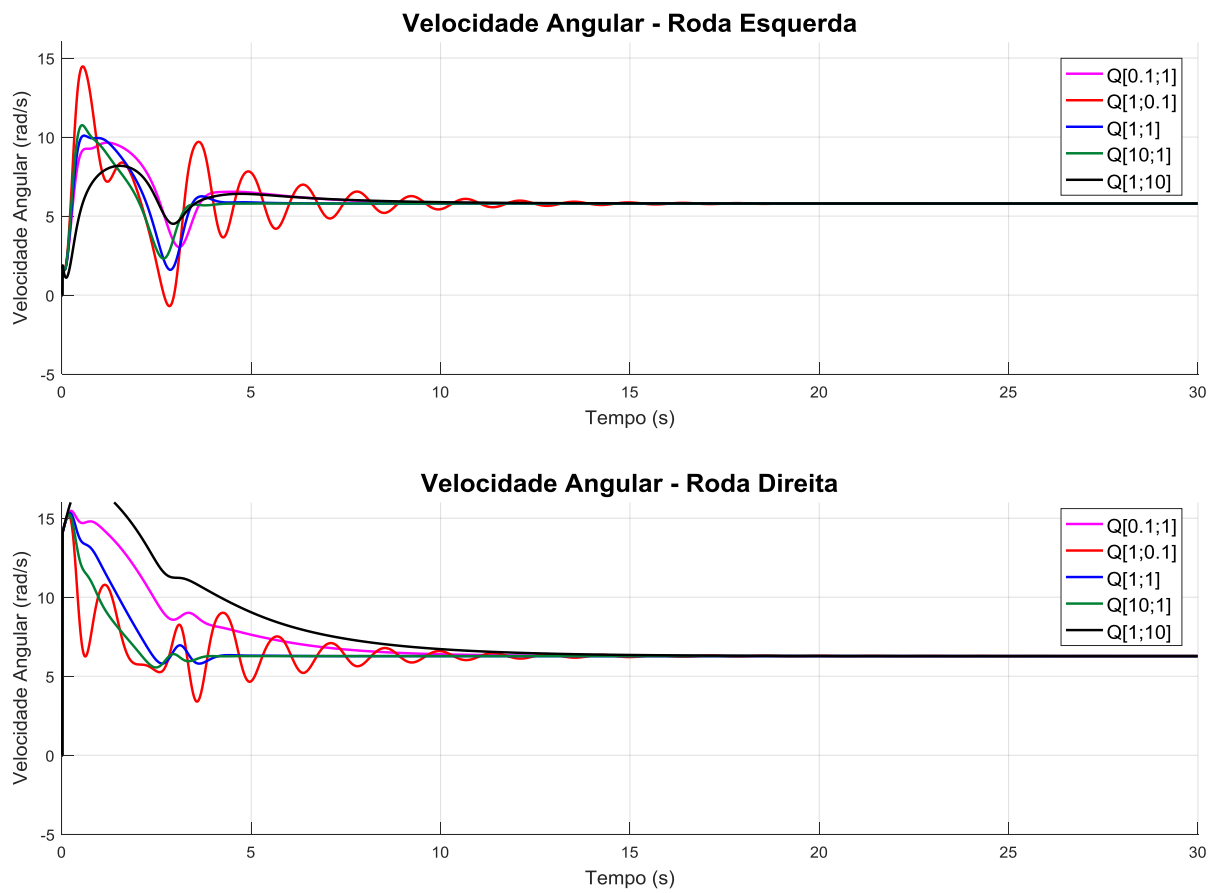


Figura 5.10 – Velocidades Angulares das Rodas Esquerda e Direita para Uma Trajetória De Referência Circular para o Modelo de Sintonia de Terceira Ordem.

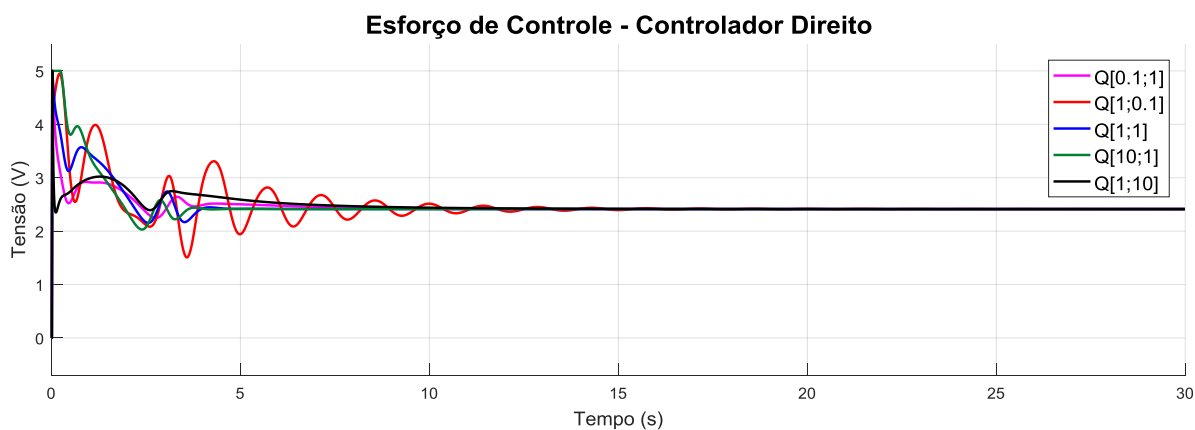
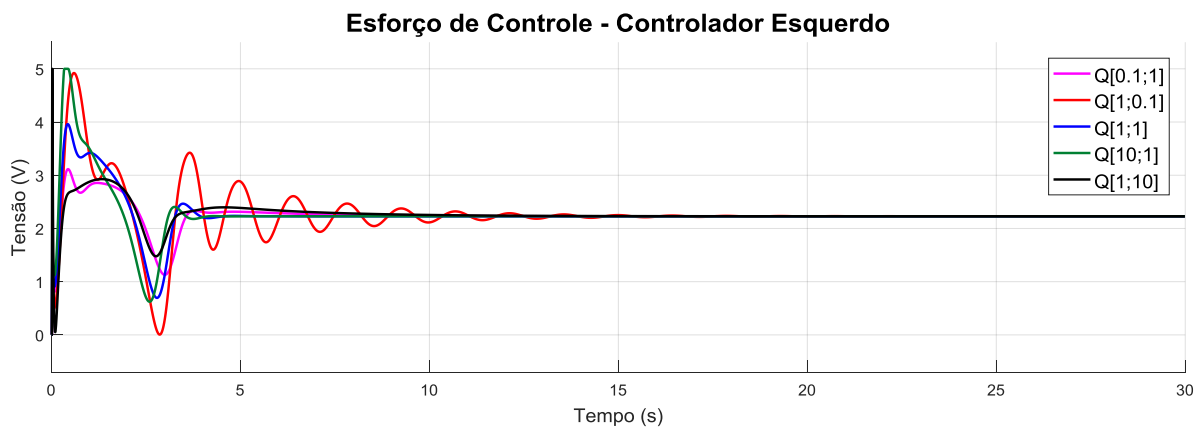


Figura 5.11 – Lei de Controle das Rodas Esquerda E Direita Para Uma Trajetória De Referência Circular para o Modelo de Sintonia de Terceira Ordem.

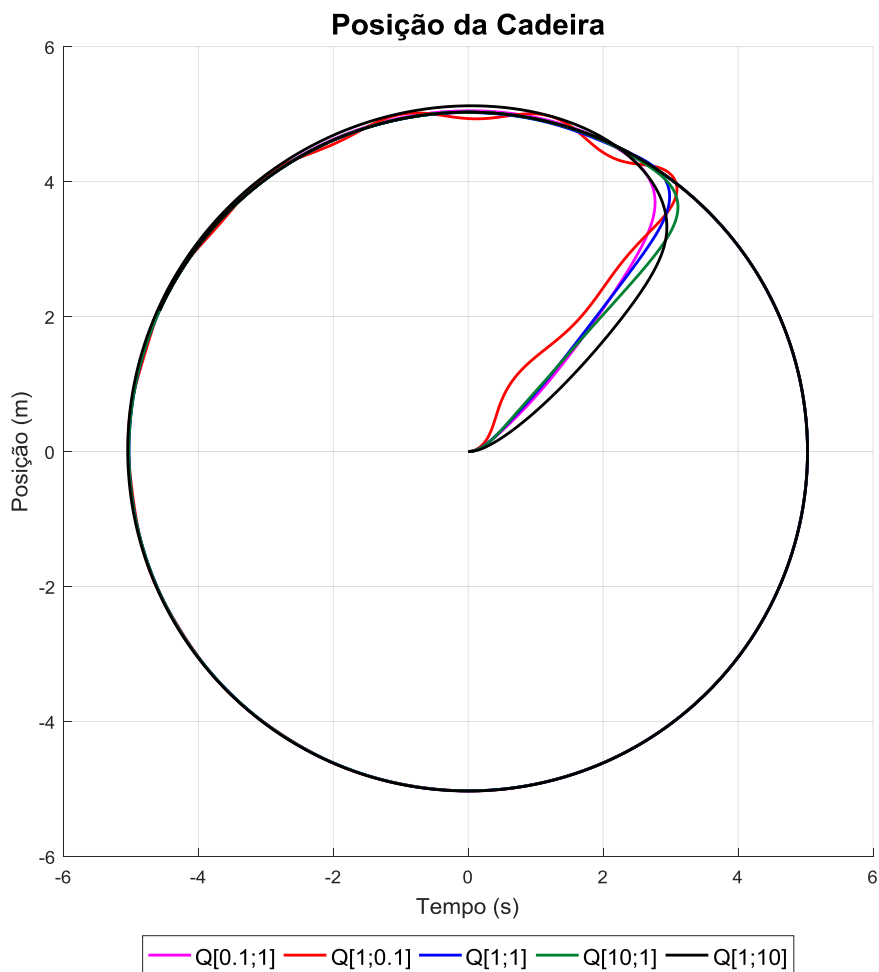


Figura 5.12 – Trajetória Circular da Cadeira de Rodas Motorizada para o Modelo de Sintonia de Terceira Ordem.

5.2 Experimentos Computacionais PID-RNA

Os resultados foram obtidos através do algoritmo de controle proposto, onde a resposta do sistema é avaliada para uma entrada do tipo degrau unitário, ilustrado pela Figura 5.13. Ao longo do processo de simulação um sinal de distúrbio periódico foi aplicado para analisar a as respostas dos dois sistemas de controle, PID-RNA e PID Clássico. Em seguida, o modelo do controlador PID-RNA é implementado no sistema considerando o comportamento dinâmico e o cinemático, aplicando-se uma trajetória de referência do tipo pontual e outra do tipo circular.

5.2.1 Experimentos computacionais para atuador em malha fechada

Os experimentos computacionais iniciais, foram realizados levando-se em consideração somente a dinâmica dos atuadores. Como mencionado anteriormente, o modelo matemático do atuador tem como entrada um sinal de tensão e saída a velocidade angular. A avaliação do controlador foi realizada através da análise da resposta do sistema após a adição de um sinal de perturbação periódico e comparada com um controlador PID clássico. Para inicialização do algoritmo utilizou-se $\eta_i = [0.1 \quad 0.1 \quad 0.001]$ e $K_c = 2.2$, para o modelo de referência tem-se $\zeta = 1$ e $t_s = 0.9s$ e taxa de amostragem utilizada para o sistema $T_s = 0.001s$.

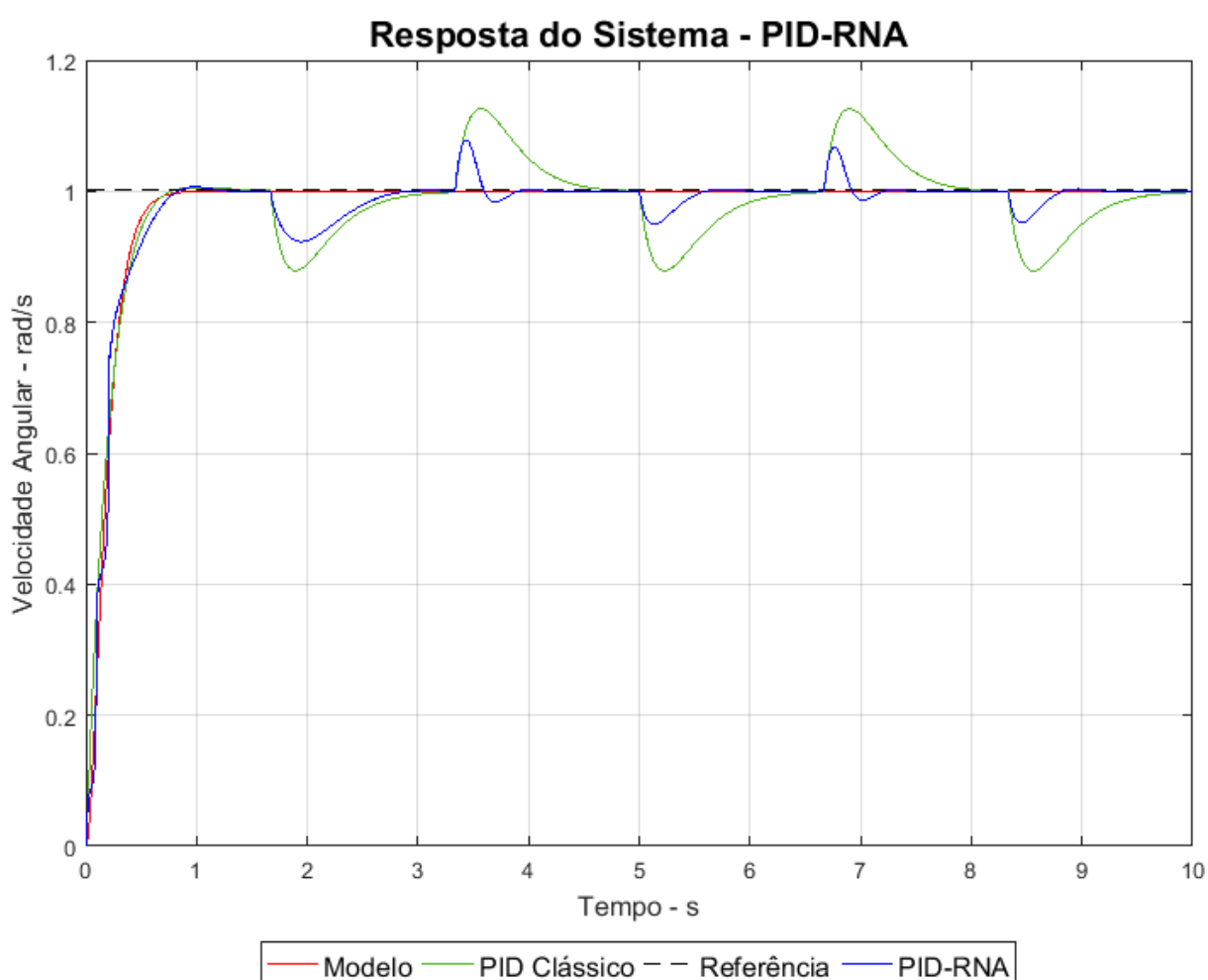


Figura 5.13 – Resposta ao Degrau para o Sistema em Malha Fechada Aplicando PID-RNA.

Na figura acima nota-se que nas primeiras iterações a saída apresenta um comportamento oscilatório, pois inicialmente os pesos da rede ainda se ajustam afim de seguir o modelo de referência, isto é ilustrado pela Figura 5.14, que apresenta a convergência dos parâmetros com a utilização da rede neuronal. Estas observações são confirmadas na Figura 5.15, na qual pode ser observada a saída do controlador.

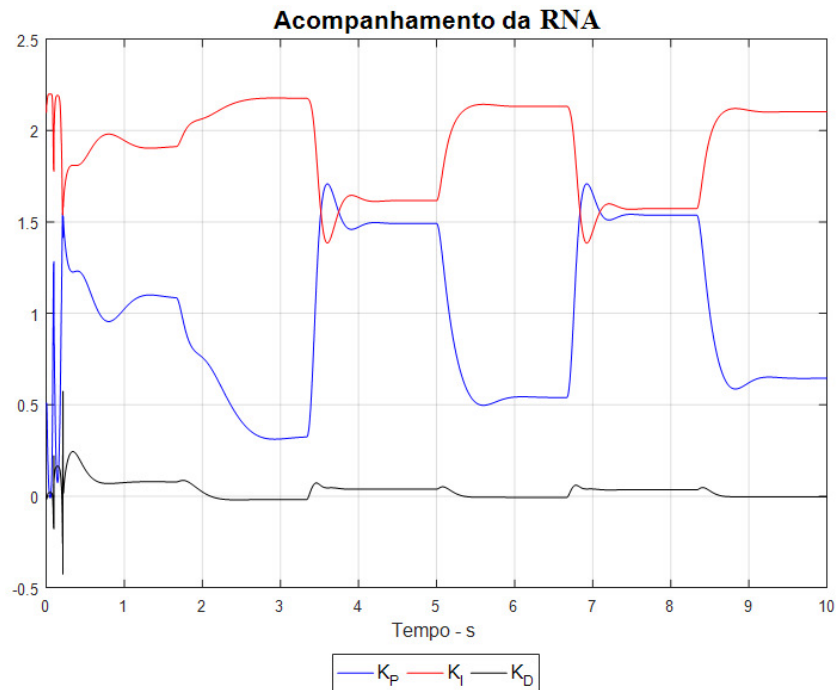


Figura 5.14 – Convergência dos Ganhos do Controlador PID-RNA.

A partir da Figura 5.14, nota-se que inicialmente os parâmetros variam rapidamente e depois durante os sinais de distúrbios inseridos convergem para pontos específicos.

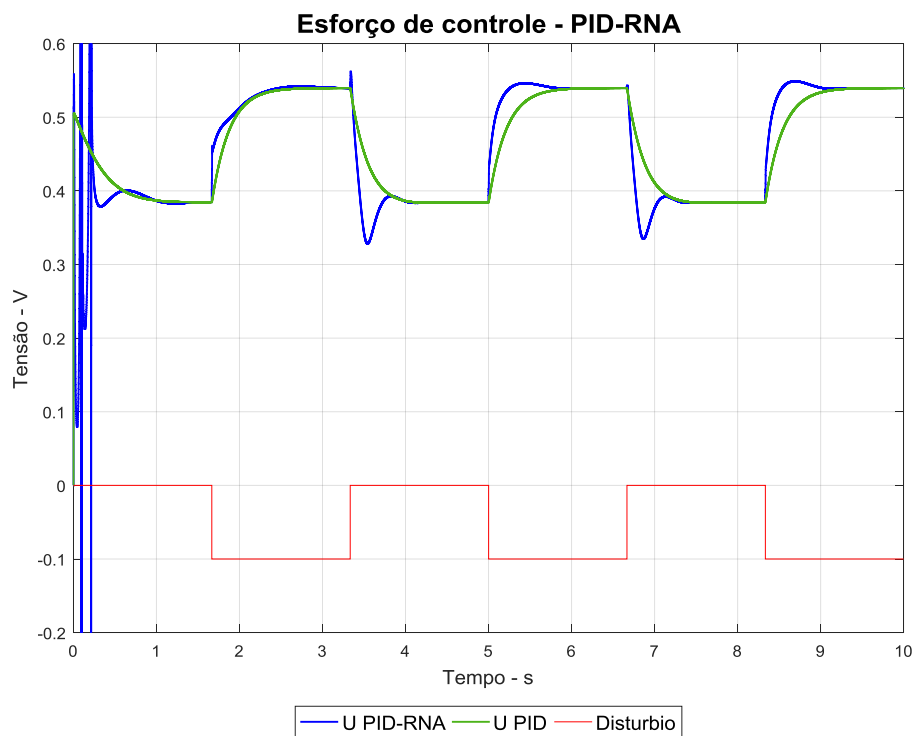


Figura 5.15 – Comparativo entre o Esforço de Controle do Controlador PID-RNA e PID Clássico.

Como pode ser visto na Figura 5.15, o esforço de controle é maior nos instantes iniciais, isso se dá as condições iniciais atribuídas e aos primeiros ajustes do controlador. Observa-se que os maiores valores de esforço de controle não ultrapassam os 5 Volts. Nos sistemas de controle significa que a fonte pode ter atingido o seu valor máximo.

5.2.2 Experimentos computacionais para sistema cinemático e dinâmico referência pontual.

A análise realizada aqui nesta seção, leva em consideração a dinâmica e a cinemática do sistema, para uma melhor avaliação do controlador. O sinal de entrada do sistema é representado por um ponto o qual a cadeira de rodas deve alcançar. Da mesma forma que foi realizado no experimento anterior, perturbações foram inseridas nas duas rodas com períodos diferentes. Na Figura 5.16 é analisada a velocidade angular das rodas esquerda e direita e são apresentas os sinais de referência gerado pelo modelo cinemático e outro sinal gerado pelo modelo de referência do algoritmo de controle PID-RNA. Para inicialização do algoritmo utilizou-se $\eta_i^1 = \eta_i^2 = [0.3 \ 0.03 \ 0.0001] \cdot 10^{-3}$ e $K_c^1 = K_c^2 = 2$, onde 1 representa o controlador esquerdo e 2 o controlador direito, para o modelo de referência para os dois controladores tem-se $\zeta = 1$ e $t_s = 0.9s$ e taxa de amostragem utilizada para o sistema $T_s = 0.01s$.

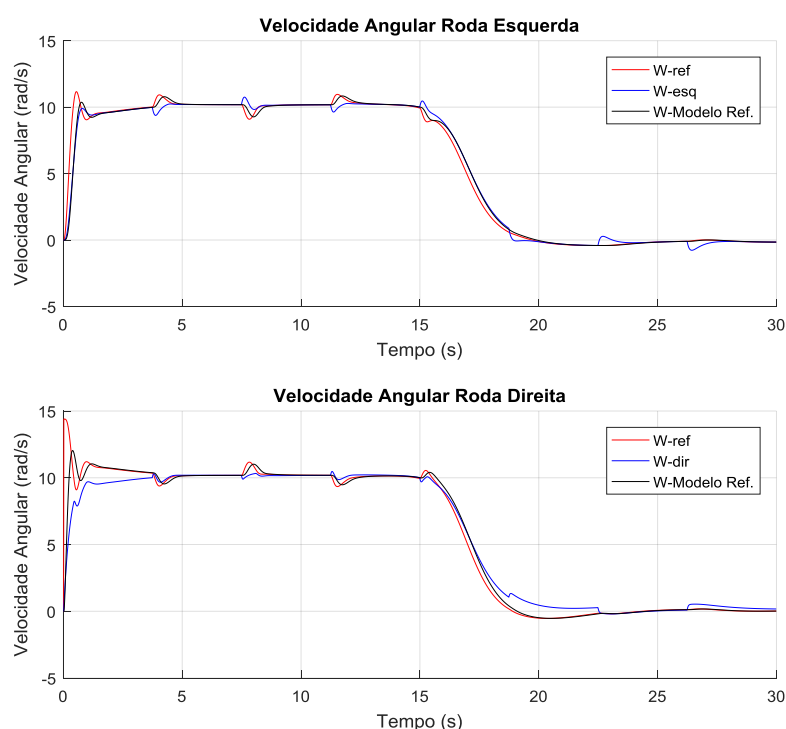


Figura 5.16 – Velocidades angulares rodas esquerda e direita PID-RNA a um ponto.

Com os resultados apresentados das Figura 5.16, nota-se a influência do sinal de perturbação sobre a resposta do sistema, este sinal de perturbação é apresentado na Figura 5.17 em conjunto com o esforço de controle dos dois controladores. A resposta do sistema apresenta bons resultados e tende a seguir as trajetórias de referência assim como projetado.

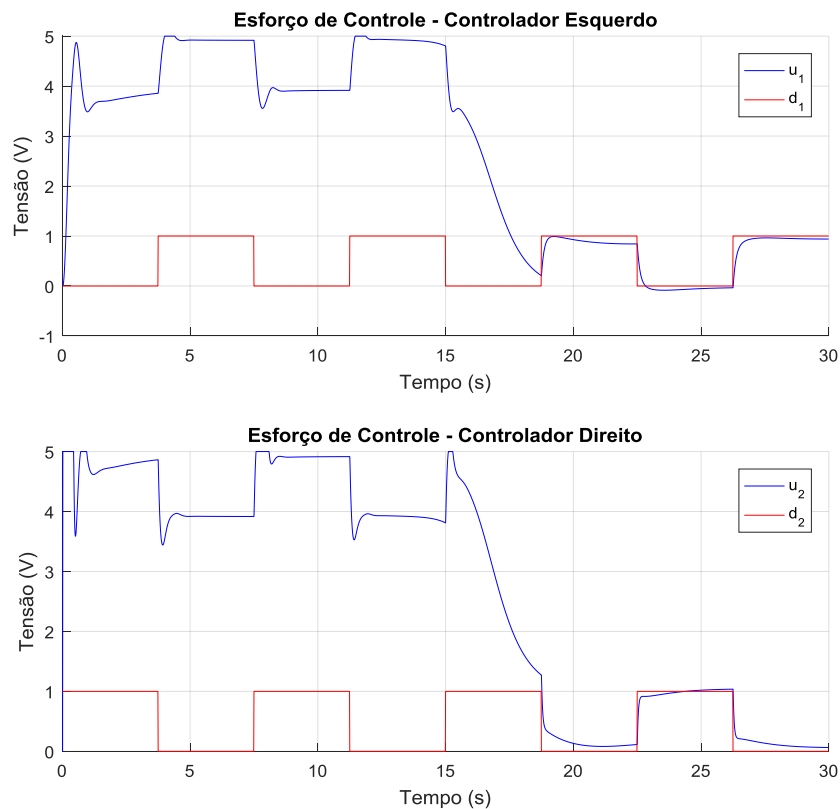


Figura 5.17 – Esforço de controle controlador esquerda e direita PID-RNA a um ponto.

A Figura 5.17 apresenta o esforço de controle dos controladores esquerdo e direito, em conjunto com o sinal de distúrbio inserido, afim de se obter resultados mais próximos do real, uma saturação foi inserida no esforço de controle, $u_{sat} = \pm 5V$. Os controladores apresentam uma boa resposta em relação as perturbações inseridas. Nota-se que ao chegar em 20s o esforço de controle tende a zero, pois a trajetória de referência do sistema é um ponto. Logo ao chegar no ponto de referência a cadeira deve estar parada, isto é ilustrado pela Figura 5.19. Nas primeiras interações o esforço de controle apresenta um comportamento anormal, devido a adaptação dos parâmetros neste período de tempo.

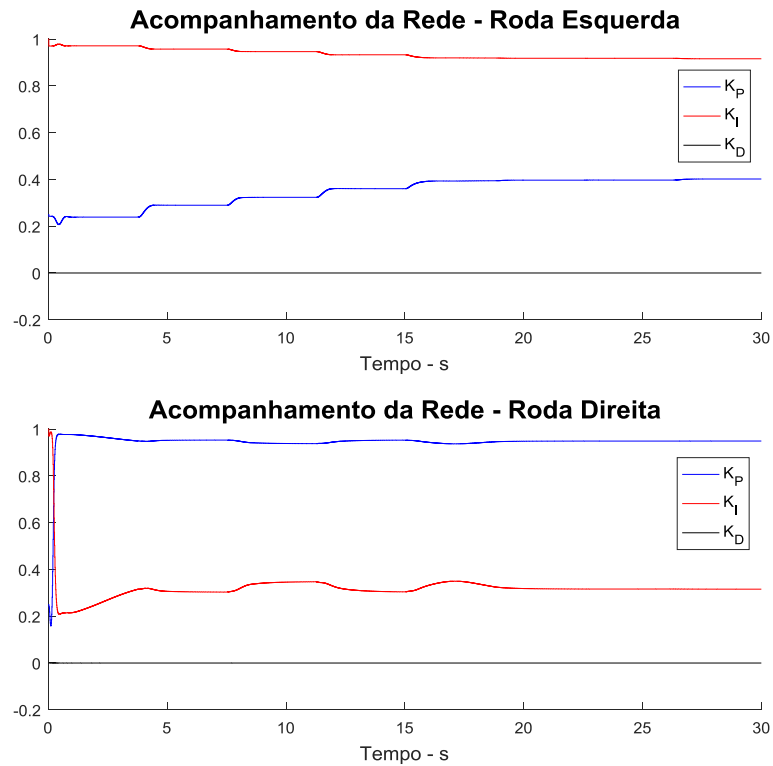


Figura 5.18 – Acompanhamento da rede dos controladores esquerdo e direito PID-RNA a um ponto

A Figura 5.18 apresenta a convergência dos ganhos do controlador ao longo das iterações. Observa-se que durante os sinais de perturbação os parâmetros variam com intuito de obter uma melhor resposta. E na Figura 5.19 é ilustrado o comportamento do sistema em relação ao plano (x, y) tendo como sinal de entrada um ponto localizado em $(30,30)$ metros.

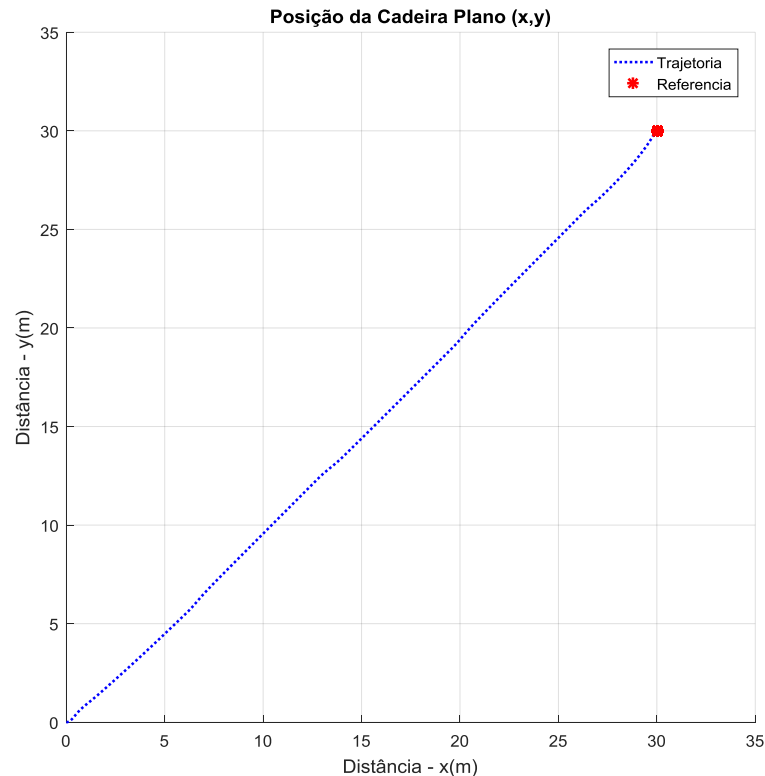


Figura 5.19 – Trajetória plano (x, y) PID-RNA a um ponto

Na Figura 5.19 é ilustrado o deslocamento da cadeira de rodas em relação ao plano (x, y) , através do qual é possível o comportamento do sistema como um todo. Ao longo de sua trajetória a cadeira de rodas apresenta pequenas variações devido as perturbações inseridas, mas ainda assim o sistema apresenta uma resposta aceitável.

5.2.3 Experimentos computacionais para sistema cinemático e dinâmico referência circular.

Os experimentos computacionais realizados nesta seção foram realizados levando-se em consideração o sistema cinemático e dinâmico e tendo como sinal de referência uma trajetória circular, com raio de 5 metros. Os mesmos valores de inicialização do algoritmo foram utilizados para esta trajetória de referência, $\eta_i^1 = \eta_i^2 = [0.3 \quad 0.03 \quad 0.0001] \cdot 10^{-3}$ e $K_c^1 = K_c^2 = 2$, para o modelo de referência para os dois controladores tem-se $\zeta = 1$ e $t_s = 0.9s$ e taxa de amostragem utilizada para o sistema $T_s = 0.01s$.

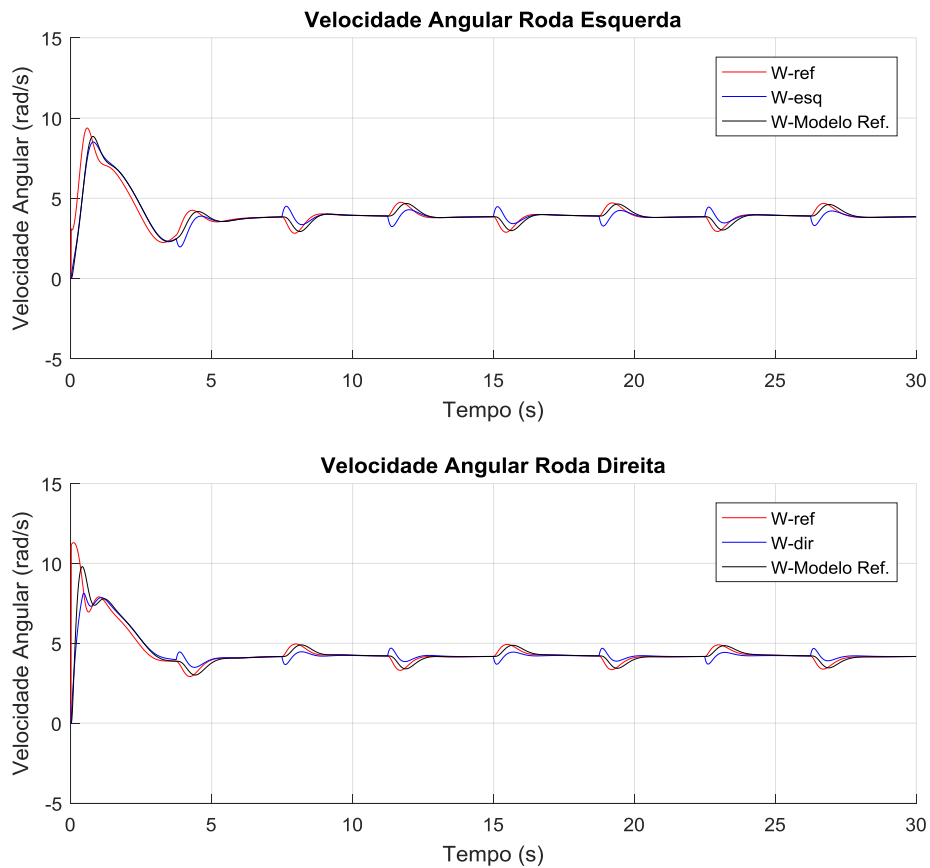


Figura 5.20 – Velocidades angulares rodas esquerda e direita PID-RNA trajetória circular.

As velocidades angulares das rodas esquerda e direita após os 5 segundos apresentam comportamento estável, ainda neste período nota-se que as rodas possuem uma pequena diferença em suas velocidades devido a trajetória de referência circular. Em relação as perturbações, a resposta do sistema dinâmico apresenta um bom comportamento e o controlador tende a rejeitar o sinal de distúrbio inserido. Isto é confirmado pela Figura 5.21, que apresenta a resposta do controlador e o sinal de distúrbio.

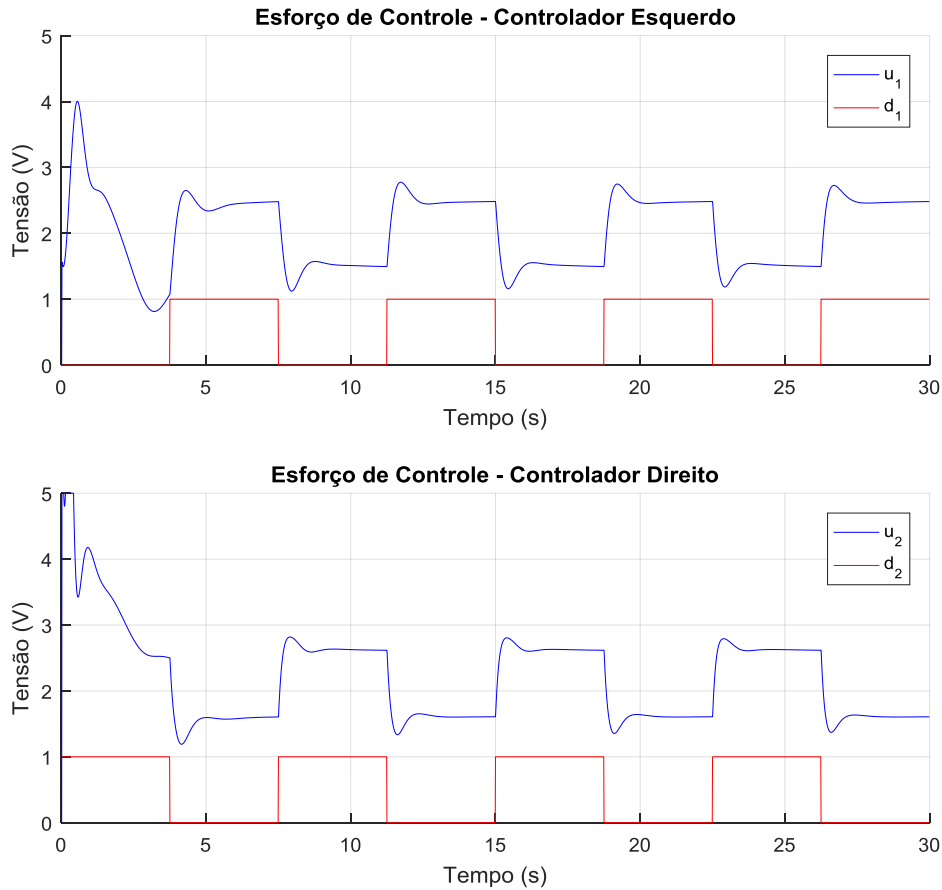


Figura 5.21 – Esforço de controle controladores esquerdo e direito PID-RNA trajetória circular.

Durante os instantes iniciais, os controladores apresentam uma resposta anormal, devido ao ajuste dos parâmetros, nota-se através da Figura 5.21. Assim como realizado no experimento anterior, foi inserida uma saturação na saída do controlador, $u_{sat} = \pm 5V$. Durante as perturbações os controladores variam em torno de um ponto específico e afim de compensar os distúrbios inseridos, este comportamento é confirmado ao se observar a Figura 5.22 a qual apresenta a convergência dos ganhos dos controladores ao longo das iterações.

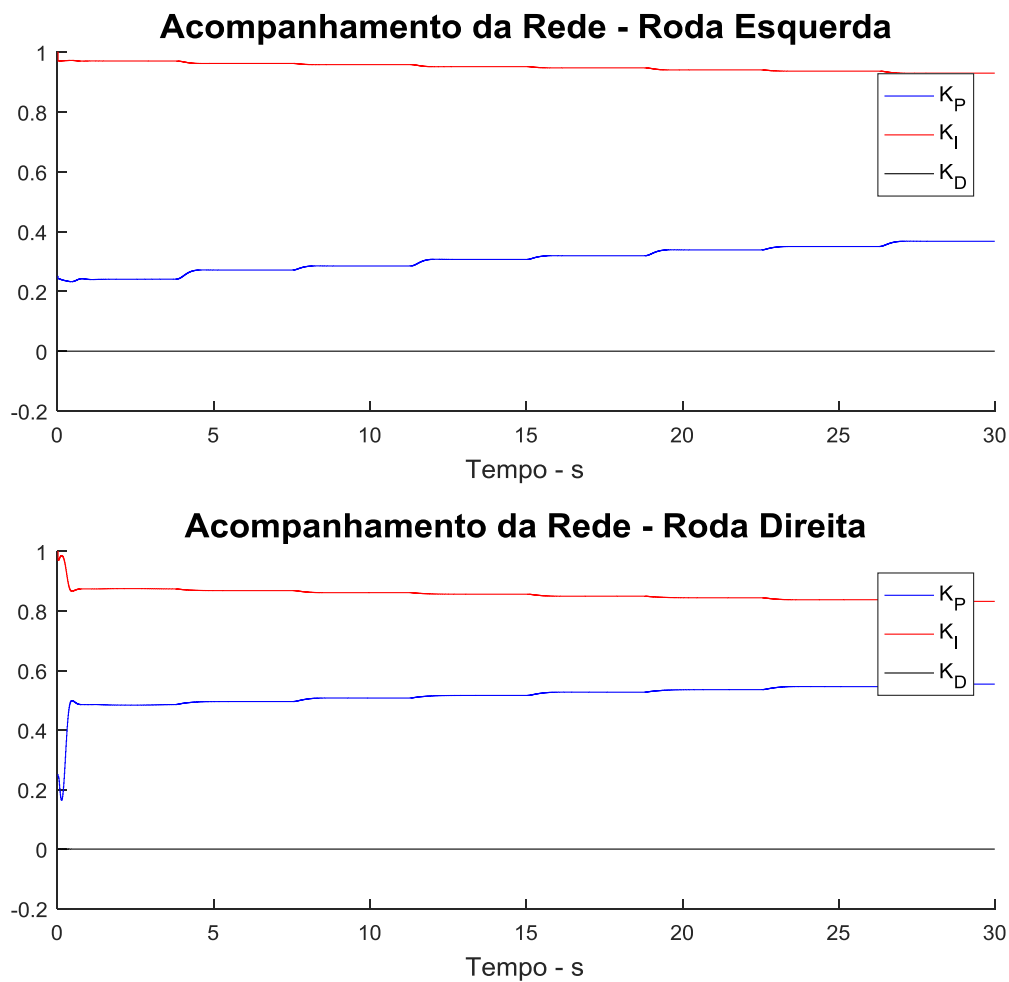


Figura 5.22 – Acompanhamento da rede controlador esquerdo e direito PID-RNA trajetória circular.

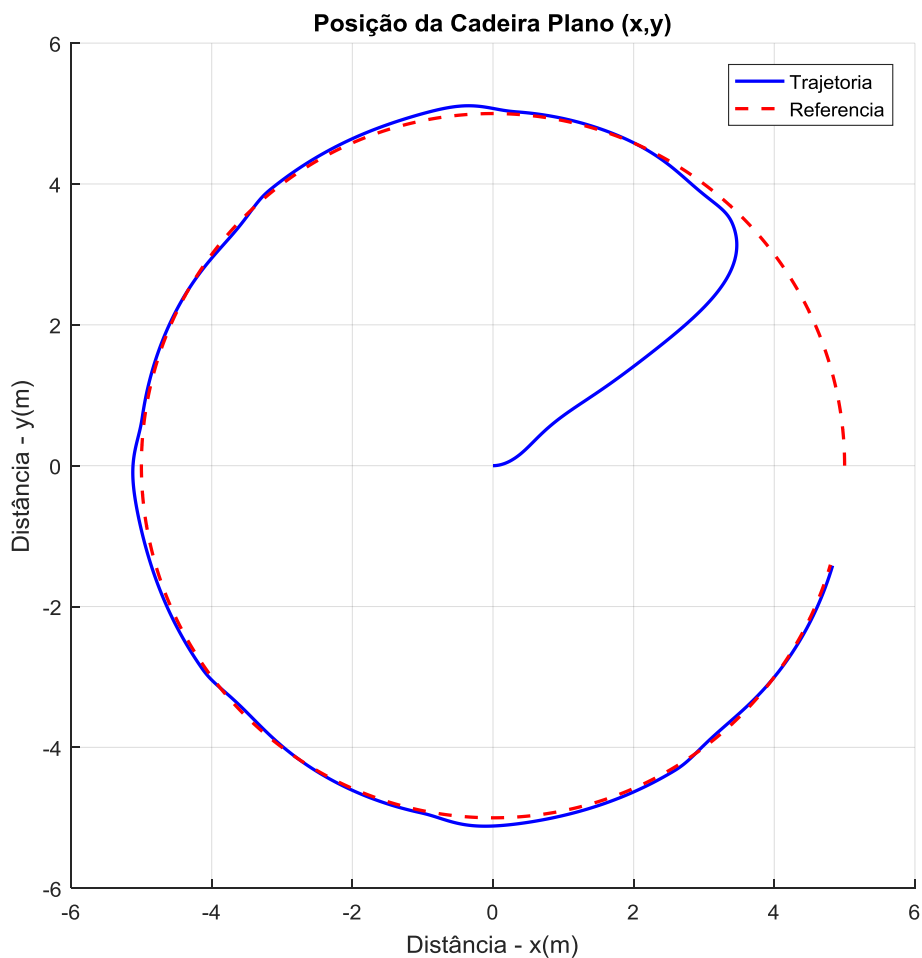


Figura 5.23 – Trajetória plano (x, y) PID-RNA circular.

Capítulo 6

CONCLUSÃO

Neste capítulo são expostas as conclusões obtidas com o desenvolvimento do trabalho proposto. E em seguida são apresentados temas para trabalho futuros.

6.1 Considerações Finais

Em conclusão, o presente trabalho desenvolveu modelos matemáticos para sintonia de controlador dinâmico de cadeira de rodas motorizada, que combina a simplicidade e a robustez de um controlador PID com otimização e margem de ganho e de fase garantidos pela sintonia LQR. Portanto, a função de transferência do sistema em malha fechada foi representada no espaço de estados, onde o integrador do PID foi inserido no modelo da planta e o numerador do controlador foi considerado como detentor dos ganhos de realimentação de saída. Em primeiro lugar, os parâmetros da planta da função de transferência foram estimados pela medição de dados fornecendo num primeiro momento um modelo de primeira ordem.

O desempenho da resposta do sistema com a aplicação da presente proposta, mostraram a eficiência do método, onde os valores do esforço de controle não excederam o limite imposto pela implementação do mundo real, mostrando o que viabiliza a sua implementação em uma cadeira de rodas motorizada.

Ao contrário do modelo anterior, o qual o projeto é realizado *off-line*, o segundo modelo funciona de forma *on-line* e baseia-se em princípios de Redes Neurais Artificiais para representar o controlador PID que é associado com abordagens do controle adaptativo direto por modelo de referência. Para treinamento, utilizou-se o algoritmo de treinamento baseado na regra de Hebb. O método proposto mostrou bom desempenho, devido ao fato de que apresentou

uma rejeição satisfatória ao distúrbio, isto é, propriedade de adaptação para liderar com situações imprevisíveis. Durante o desenvolvimento do projeto, ficou evidente a necessidade de validar o modelo matemático utilizado. Esta ação fornecerá os meios para uma melhor concepção e análise dos resultados. Conseqüentemente, obtém-se uma maior confiabilidade na implementação no sistema real.

Ao analisar os resultados, é possível concluir que o método é eficiente e o algoritmo desenvolvido atendeu às especificações de projeto. Além de implementar o controlador PID-RNA proposto, ele tem a flexibilidade de ser implantado em outras plantas.

6.2 Trabalhos Futuros

No decorrer do desenvolvimento deste trabalho de dissertação vários itens se apresentaram como oportunidades para estudos adicionais. A seguir, uma descrição das principais ideias as quais necessitariam mais atenção aos detalhes e conseqüentemente mais tempo e recursos disponíveis para o autor. Espera-se que este trabalho inspire o leitor com inúmeras ideias adicionais.

- Avaliar a resposta da metodologia proposta através da implementação em hardware;
- Pesquisar e desenvolver metodologia de projeto de sensores para o desvio de obstáculos que permitam a cadeira de rodas e cadeirante trafeguem em ambientes com segurança e evitando assim possíveis colisões;
- Propor outros algoritmos de controle para o posicionamento da cadeira de rodas motorizada, levando em conta que o centro de massa do sistema variável, simulando assim quando a postura do usuário muda sobre o assento da cadeira;
- Implementar planejadores da trajetória que trabalham diretamente com habilidades de percepção e motoras, ao invés de trabalhar com dados planejados pré-programados.

REFERÊNCIAS BIBLIOGRÁFICAS

Alves, S. F. R. et al., 2011. Conceptual Bases of Robot Navigation Modeling, Control and Applications.. Em: *Advances in Robot Navigation*. Rijeka: InTech.

Amori, R., 1992. *Vocomotion - An intelligent voice-control system for powered wheelchairs*. Toronto, s.n.

Anderson, B. D. O. & Moore, J. B., 1990. *Optimal Control*. New York: Dover Publications.

Anvari, I., 2013. Non-holonomic Differential Drive Mobile Robot Control & Design: Critical Dynamics and Coupling Constraints. *Dissertação de Mestrado*. Arizona State University.

Araújo Filho, E. M. & da Fonseca Neto, J. V., 2017. Optimal Tuning of Dynamic Controller Via LQR in a Powered Wheelchair. *American Journal of Engineering Research*, 6(11), pp. 44-53.

Åström, K. J., 2005. *Advanced PID Control*. s.l.:ISA.

Åstrom, K. J. & Hagglund, T., 2001. The future of pid control. *Control engineering practice*, 9(11), p. 1163–1175.

Åström, K. J. & Wittenmark, B., 1995. *Adaptive Control*. 2nd ed. Massachusetts: Addison Wesley Longman Publishing Co..

Bates, R., 2002. *A computer input device selection methodology for users with high-level spinal cord injuries*. s.l., s.n.

Bishop, C. M., 2006. *Pattern Recognition and Machine Learning*. s.l.:Springer.

Brown, R. G., 2012. *Introduction to Random Signals and Applied Kalman Filtering with MATLAB exercises*. s.l.: John Wiley & Sons, Inc.

Caporal, R. M. et al., 2015. Digital Controller for an Electric Wheelchair Based on a Low-Cost Hardware. *IEEE LATIN AMERICA TRANSACTIONS*, OCTOBER, 13(10).

Clearesta, E., Wardhana, A. A., Widyotriatmo, A. & Suprijanto, 2013. Adaptive Control for Velocity Control of an Electric Wheelchair. *3rd International Conference on Instrumentation Control and Automation (ICA)*, pp. 222-226.

Cruz, C. D. L., Bastos, T. F. & Carelli, R., 2010. Adaptive motion control law of a robotic wheelchair. *Control Engineering Practice*, pp. 113-125.

Cruz, C. D. L., Cardoso, W. & Bastos, C. T. F., 2012. Dynamic Model-Based Adaptive Posture Controller for Robotic Wheelchairs. *Journal of Medical and Biological Engineering*, 32(1).

Da Fonseca Neto, J. V., Abreu, I. S. & Silva, F. N. d., 2010. Neural–Genetic Synthesis for State-Space Controllers Based on Linear Quadratic Regulator Design for Eigenstructure Assignment. *IEEE Transactions on Systems, Man, and Cybernetics*, 40(2), pp. 266-285.

Fehr, L., Langbein, W. & Skaar, S., 2000. Adequacy of power wheelchair control interfaces for persons with severe disabilities: a clinical survey. *Journal of Rehabilitation Research and Development*, 37(3), p. 353–360.

Fitzgerald, A., KINGSLEY, C. & UMANS, S., 2003. *Electric Machinery*. 6^a ed. s.l.:McGraw-Hill.

Franklin, G. F., Powell, J. D. & Emami-Naeini, A., 2010. *Feedback Control of Dynamic Systems*. 6^a ed. New Jersey: Addison-Wesley.

Grimstad, B., 2009. Studies In Static Output Feedback Control. *Tese de Mestrado em Norwegian University of Science and Technology*.

Haykin, S., 1994. *Neural Networks: A Comprehensive Foundation..* s.l.:Macmillan College Publishing Company.

Haykin, S., 2008. *Neural Networks and Learning machines*. 3^a ed. New Jersey: Prentice Hall.

Hebb, D. O., 1949. *The Organization of Behavior A Neuropsychological Theory*. s.l.:John Wiley if Sons, Inc.

He, J.-B., Wang, Q.-G. & Lee, T.-H., 2000. PI/PID controller tuning via LQR approach. *Chemical Engineering Science*, 55(13), pp. 2429-2439.

Hou, R., Shi, X. & Krishnamurthy, M., 2012. Design and implementation of a novel power assisted drivetrain for a wheelchair. *IEEE Transportation Electrification Conference and Expo (ITEC)*, pp. 1-6.

Júnior, C. d. S., 2000. Controle Adaptativo de Robôs Móveis. *Dissertação de Mestrado - ITA*.

Kaiadi, M., 2006. Artificial Neural Networks Modelling for Monitoring and Performance Analysis of a Heat and Power Plant. *Thesis for the Degree of Master of Science*, Department of Energy Sciences(LUND UNIVERSITY).

Kelouwani, S., Ouellette, C. & Cohen, P., 2013. Stable and Adaptive Control for Wheeled Mobile Platform. *Intelligent Control and Automation*, Issue November, pp. 391-405.

Kirk, D. E., 2004. *Optimal Control Theory. An Introduction..* s.l.:Dover Publications, Inc..

Krishnan, R., 2003. *Electric Motor Drives Modeling, Analysis and Control*. Singapore: Pearson Education.

Laut, J., 2011. *Tese. A Dynamic Parameter Identification Method for Migrating Control Strategies Between Heterogeneous Wheeled Mobile Robots*. s.l.:Worcester Polytechnic Institute.

Lewis, F. L., Vrabie, D. L. & Syrmos, V. L., 2012. *Optimal Control*. s.l.:John Wiley & Sons, Inc..

Liu, J. K., 2004. *Advance PID Control & MATLAB Simulation*. s.l.:Beijing: electronic industry press.

Liu, X., 2008. *Single Neuron Self-tuning PID Control for Welding Molten Pool Depth*. Chongqing, s.n., pp. 7922-7925.

Ljung, L., 1999. *System Identification: Theory for the User*. University of Linköping. New Jersey.: P TR Prentice Hall.

Majid, N. A., Mohamed, Z. & Basri, M. A. M., 2016. Velocity Control of a Unicycle Type of Mobile Robot Using Optimal PID Controller. *Jurnal Teknologi (Sciences & Engineering)*, Issue 78, p. 7–14.

Marsland, S., 2009. *Machine Learning: An Algorithmic Perspective*. s.l.:Chapman & Hall/CRC..

Martins, F. N. et al., 2008. An adaptive dynamic controller for autonomous mobile robot trajectory tracking. *Control Engineering Practice*, Issue 18, p. 1354– 1363.

Miller, G., Brown, T. & Randolph, W., 1985. Voice controller for wheelchairs. Volume 6, p. 597–600.

Moslehi, H. R., 2011. *Design and Development of Fuzzy Logic Operated Microcontroller Based Smart Motorized Wheelchair*. *Dissertação de Mestrado*. Halifax, Nova Scotia: Dalhousie University.

Moslehi, H. R. & gu, J., 2011. Design and development of fuzzy logic operated smart motorized wheelchair.. *IEEE CCECE*, pp. 869-873.

Nada, D., Bousbia-Salah, M. & Bettayeb, M., 2016. Multi-sensor Data Fusion for Wheelchair Position Estimation with Unscented Kalman Filter. *International Journal of Automation and Computing*.

Naidu, D., 2003. *Optimal control systems*. s.l.:CRC Press.

Nilsson, C. & Modrack, D., 2013. Universal Embedded Motor Control. *Tese de Mestrado*. Chalmers University of Technology. Goteborg, Sweden .

Nisbet, P. D., 2002. *Who's intelligent? Wheelchair, driver or both?*. Glasgow, s.n.

Ogata, K., 1996. *Discrete-Time Control Systems*. s.l.:Prentice Hall Inc.

Ogata, K., 2008. *Engenharia de Controle Moderno*. 5ª ed. São Paulo(São Paulo): Prentice Hall.

Rivera-Mejía, J., León-Rubio, A. & Arzabala-Contreras, E., 2012. PID Based on a Single Artificial Neural Network Algorithm for Intelligent Sensors.. *Journal of Applied Research and Technology*, 10(2), pp. 262-282.

Romero, R., Osório, F., Prestes, E. & Wolf, D., 2014. *Robótica Móvel*. s.l.:LTC Editora.

Ruano, A. E. d. B., 1992. Applications of Neural Networks to Control Systems. *Dissertação Mestrado. Universidade de Wales*, February.

Ruzajj, M. F. & Poonguzhali, S., 2012. Design and implementation of low cost intelligent wheelchair. *International Conference on Recent Trends in Information Technology*, pp. 468-471.

Salem, F. A., 2013. Mechatronics Design of Small Electric Vehicles; Research and Education. *International Journal of Mechanical & Mechatronics Engineering*, 13(01).

Sen, P. C., 2013. *Principles of Electric Machines and Power Electronics*. Kingston: Wiley.

Sheng, Q. et al., 2002. Design and Implementation of An Adaptive PID Controller Using SingleNeuron Learning Algorithm. *Proceedings of the 4* World Congress on Intelligent Control and Automation*, 10-14 June, pp. 2279-2283.

Siciliano, B., Sciavicco, L., Villani, L. & Oriolo, G., 2009. *Robotics Modelling, Planning and Control*. London : Springer.

Siegwart, R. & Nourbakhsh, I. R., 2004. *Introduction to Autonomous Mobile Robots*. London, England: The MIT Press.

Silva, I. N. d. et al., 2017. *Artificial Neural Networks A Practical Course*. 1 ed. São Carlos: Springer.

Simpson, R., 2005. Smart wheelchairs: a literature review. 42(4), p. 423–436.

Simpson, R. et al., 2004. The smart wheelchair component system. *Journal of Rehabilitation Research and Development*, 41(3), p. 429–442.

Solea, R., Filipescu, A. & Jr, A. F., 2015. *Wheelchair Control and Navigation Based on Kinematic Model and Iris Movement*. Siem Reap, s.n.

Taghizadeh, S., 2010. Control of a Pneumatic System with Adaptive Neural Network Compensation. *Tese de Doutorado. Queen's University. Kingston, Ontario, Canada*.

Tomari, M. R. M., Kobayashi, Y. & Kuno, Y., 2012. *Development of Smart Wheelchair System for a User with Severe Motor Impairment*. s.l., s.n.

Tsuchiya, K., Urakubo, T. & Tsujita, K., 2002. Motion Control of a Nonholonomic System Based on the Lyapunov Control Method. *Journal of Guidance, Control and Dynamics*, 25(2), pp. 285-290.

Tzafestas, S. G., 2014. *Introduction to Mobile Robot Control*. s.l.:Elsevier.

Vanacker, G., 2008. Adaptive Steering Behaviour Modelling For Power Wheelchair Control. *Katholieke Universiteit Leuven*, Julho.

Vilanova, R. & Visioli, A., 2012. *PID Control in the Third Millennium. Lessons Learned and New Approaches*. 3^a ed. London: Springer.

Wang, H. & Li, G., 2010. Motion Control and Trajectory Tracking Control for a Mobile Robot via Disturbance Observer;. *WSEAS Transactions on Systems*, 9(1), pp. 31-41.

Wang, H. et al., 2009. Real-time model based electrical powered wheelchair control. *Medical Engineering & Physics*, Issue 31.

Widyotriatmo, A., Rauzanfiqr, S. K. & Suprijanto, 2012. *A Modified PID Algorithm for Dynamic Control of an Automatic Wheelchair*.. Bandung, s.n., pp. 23-26.

Apêndice A

SISTEMA ATUADOR E SENSOR

A.1 Elemento atuador

As aplicações que envolvem movimento exigem energia mecânica, e esta por sua vez pode ser obtida através de conversores de energia eletromecânica. Nestes tipos de conversores a energia elétrica é convertida em energia mecânica, ou vice-versa. Em sua grande maioria, os robôs e as cadeiras de rodas motorizadas utilizam motores elétricos como conversores de energia, então para tal, se faz necessário o estudo da sua dinâmica bem como a sua aplicação no sistema em estudo.

Os motores elétricos em geral podem ser classificados em motores de Corrente Contínua (CC), Corrente Alternada (CA) e síncronos (Sen, 2013). As cadeiras de rodas motorizadas em sua grande maioria utilizam motores CC com e sem escovas (Simpson, 2005), que são subclasses de motores CC e síncronos.

A.1.1 Motores de Corrente Contínua com escovas

O projeto do motor CC com escovas é o mais simples e o mais antigo no campo de motores elétricos, embora tenha uma série de desvantagens, é barato e ainda é amplamente utilizado para aplicações de controle de torque e velocidade variável.

Os motores CC com escovas ainda podem ser classificados de acordo com projeto do seu estator, naqueles cujo estator consiste em bobinas de campo e naqueles que usam ímãs permanentes no estator. A forma de alimentação da bobina de campo e a bobina de armadura pode ter três classificações básicas (Krishnan, 2003). A primeira forma de alimentação é a do

tipo *shunt* ou em derivação (paralelo), em que as bobinas de campo estão conectadas em paralelo com as bobinas da armadura através das escovas. A segunda forma é a alimentação em série, em que as bobinas de campo estão em série com as bobinas na armadura. E a terceira forma é a híbrida, com bobinas de campo separadas, onde uma está em série e a outra está em paralelo com as bobinas da armadura, as representações destas formas de alimentação são ilustradas pelas Figura A.1.

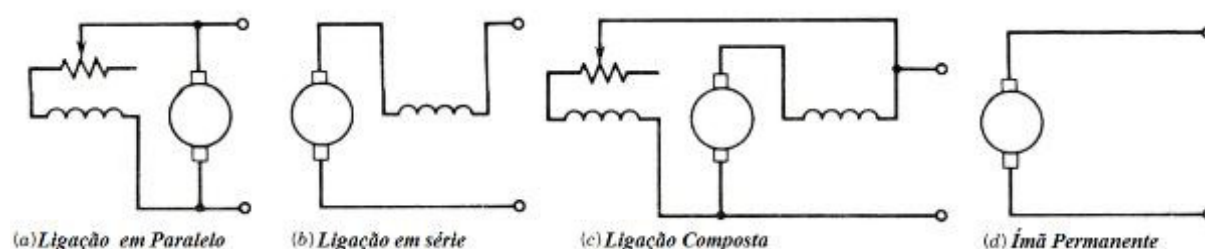


Figura A.1 – Tipos de Conexão entre Bobinas de Campo e a Bobina de Armadura em Motores CC com Escovas.

Os motores CC com escovas do tipo de ligação em paralelo podem gerar torque elevado, sem redução de velocidade com o aumento da corrente elétrica do motor e possuem um nível médio de torque de partida com velocidade constante (Fitzgerald, et al., 2003), por isso são amplamente utilizados em aplicações que incluem tornos, aspiradores, transportadores e trituradores. Já os tipos motores com ligação em série o controle de velocidade é realizado através da variação de tensão de alimentação, no entanto, esse tipo controle de velocidade ineficaz e, à medida que o torque aumenta a velocidade diminui, mas como vantagem têm um torque elevado de partida (Sen, 2013).

Os motores de Corrente Contínua de Ímã Permanente (PMDC - *Permanent Magnet Direct Current*) são utilizados em uma variedade de aplicações, como por exemplo desde dispositivos com bateria, como cadeiras de rodas e ferramentas elétricas, até transportadores e abridores de portas, equipamentos de soldagem, sistemas de raios-X e tomográficos e equipamentos de bombeamento. Estes tipos de motores elétricos geralmente são melhor opção para controle de movimento e aplicações de transmissão de energia devido ao seu tamanho compacto, a ampla faixa de velocidade em que opera, a capacidade de se adaptar a uma variedade de fontes de energia ou a segurança por conta de sua operação em baixas tensões (Sen, 2013). A sua capacidade de produzir um elevado torque a baixa velocidade torna-os substitutos adequados para motoredutores em muitas aplicações, devido à sua curva de velocidade-torque eles são especialmente adequados em aplicações que necessitam de

velocidade ajustável (Fitzgerald, et al., 2003) e em aplicações de controle de servo onde o motor funcionará a menos de 5000 rpm.

Os motores DC de ímã permanente são equivalentes aos motores de corrente contínua controlados por armadura ilustrados pela Figura A.1, de fato, em ambos o fluxo magnético é constante. Em motores DC convencionais, o torque desejado pode ser obtido aumentando a corrente armadura (i_a) e/ou a corrente de campo (i_e). Por outro lado, os motores com ímã permanente podem ser controlados por meio através de i_a .

O controle de velocidade em motores PMDC é realizada, de forma similar aos demais motores CC com escovas através da variação da tensão aplicada à armadura. Os sensores de rotação enviam as informações ao controlador para variar a sua tensão de saída afim de se manter a velocidade o mais próximo possível do valor de referência ajustado.

Nos últimos anos, os conversores de estado sólido foram utilizados para controlar a velocidade dos motores de CC.

Para a utilização de um motor CC para uma devida aplicação deve-se, portanto, conhecer a relação entre torque e velocidade da máquina. A partir da Figura A.1 a relação velocidade angular e a tensão de alimentação obtida através de um modelo matemático é dada por

$$e_a(t) = R_a i_a(t) + L_a \frac{di_a}{dt} + e_b(t), \quad (\text{A.1})$$

onde e_a é tensão de entrada, R_a é a resistência da armadura, i_a é a corrente de armadura e L_a é a indutância da armadura e

$$e_b(t) = K_b \frac{d\theta}{dt}, \quad (\text{A.2})$$

onde e_b é a tensão contra eletromotriz (emf - *electromotive force*), K_b é uma constante do motor e θ a posição angular do eixo.

Aplicando a transformada de Laplace, a função de transferência que representa a velocidade angular é dada por

$$\omega(s) = G(s)U_a(s), \quad (\text{A.3})$$

onde ω representa a velocidade angular do rotor, $U_a = e_a$ representa a tensão de entrada dada pela Eq.(A.2) e a função de transferência $G(s)$ é dada por

$$G(s) = \frac{K_m}{\frac{JT_a}{K_a K_t K_b} s^2 + \frac{J}{K_a K_t K_b} s + 1}, \quad (\text{A.4})$$

onde $K_m = 1/K_b$, $T_a = L_a/R_a$, $K_a = 1/R_a$ e K_t é a constante de torque do motor. O diagrama de blocos que representa o modelo matemático expresso pela Eq.(A.4) é apresentado pela Figura A.2.

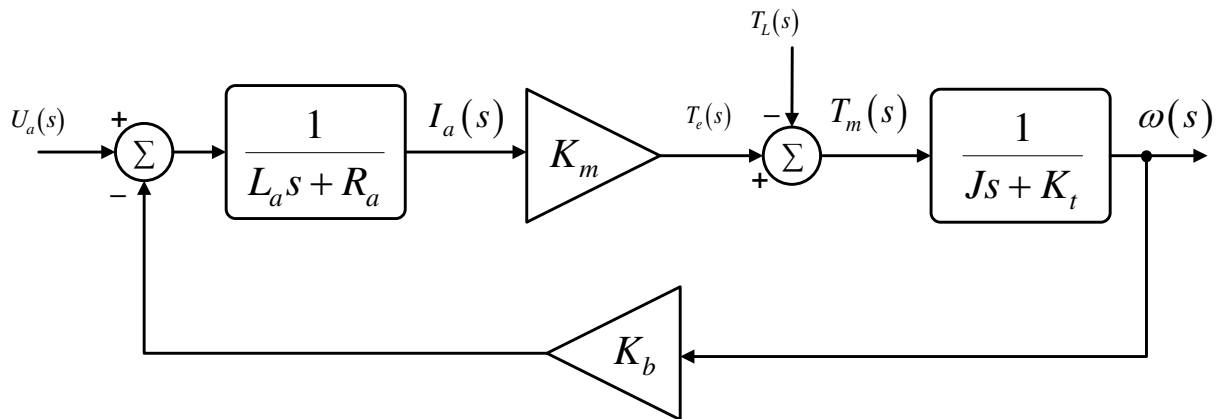


Figura A.2 – Diagrama de Blocos de um Motor CC.

sendo T_L representando o torque de cargas mecânicas externas, T_e representando o torque gerado pela força contra eletromotriz, T_m o torque do rotor e I_a a corrente que passa pela bobina do motor.

A.1.2 Motores de Corrente Contínua sem escovas

Os motores de Corrente Contínua sem escovas são motores síncronos com ímãs permanentes que operam em modo de autocontrole (usando um sensor de posição de rotor e um inversor para controlar a corrente dos enrolamentos do estator) é mais popularmente conhecido como motores BLDC (*Brushless Direct Current*). A estrutura interna de um motor BLDC é semelhante a um motor CC escovado, pois a armadura está em seu estator e os ímãs estão no rotor e suas características de operação também são similares a um motor CC escovado (Sen, 2013).

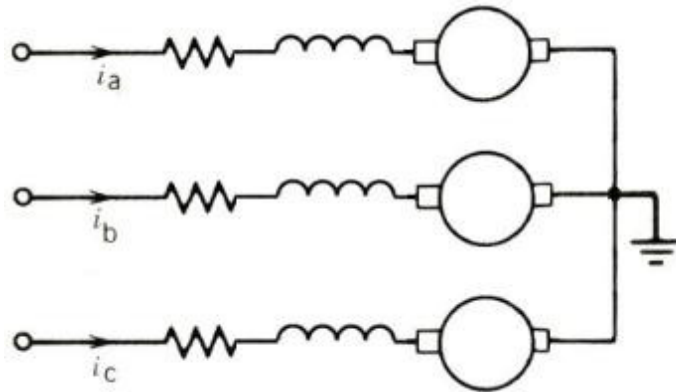


Figura A.3 – Diagrama de um Circuito Equivalente de um Motor BLDC de Três Fases.

O sensor de posição e os comutadores de estado sólido no circuito de potência inversor desempenham o papel das escovas e o comutador mecânico do motor CC. A principal diferença entre os motores BLDC e os motores CC com escovas de ímãs permanentes, é que nestes tipos de motores sem escovas os ímãs permanentes são montados junto ao rotor, desta forma suas bobinas são posicionadas no estator, onde são comutadas eletronicamente através de um circuito inversor que pode ser ilustrado pela Figura A.4.

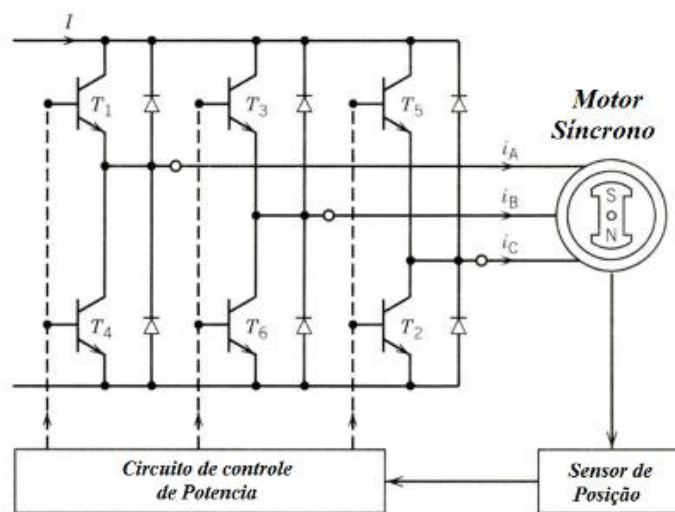


Figura A.4 – Estrutura de Circuito Inversor para Controle de um Motor BLDC.

Uma configuração típica de inversor alimentado a partir de uma fonte de CC, como ilustrado na Figura A.4, pode ser utilizada para um sistema de acionamento BLDC. O circuito inversor é composto por transistores representados por T , que controlam a corrente de alimentação do circuito representada por I afim de se obter as correntes nas bobinas A , B e C representada respectivamente por i_A , i_B e i_C . O circuito de controle de potencia é responsável por acionar ou desacionar os terminais *Gate*, ou Base no caso de transistores BJT, de acordo com o sinal de posição do sensor. Os pares de transistores T_1 e T_4 , T_3 e T_6 , T_5 e T_2 fazem parte

do circuito inversor de meia ponte que é responsável por controlar a corrente em função da amplitude e sentido de cada uma das bobinas.

O motor BLDC opera de forma similar a um motor síncrono, onde os transistores do circuito inversor são comutados de acordo com o sinal da tensão da força contra eletromotriz. Nos circuitos de controle de potência para estes tipos de motores pode ser utilizado um inversor que emprega transistores de auto-comutação, tais como BJT, IGBT, MOSFET e outros. O estado “Liga-Desliga” destes tipos contatos podem ser controlados pelos seus sinais de porta (*Gate*) do transistor, os quais são obtidos a partir dos sensores de posição do rotor. O sensor de posição do rotor irá controlar os instantes de liga e desliga para os “contatos”, de modo que o ângulo entre o campo do rotor e o campo do estator seja de 90°, semelhante a um motor com escovas (Sen, 2013). Se o inversor é alimentado por uma fonte de tensão CC, a modulação de largura de pulso (PWM) de cada contato pode fornecer a regulação da corrente do motor (Nilsson & Modrack, 2013).

A forma de onda do sinal da força contra eletromotriz dos motores BLDC podem ter basicamente um formato trapezoidal ou senoidal. No tipo trapezoidal a força contra eletromotriz tem formato trapezoidal e, para operação sem flutuações de torque, a corrente de fase requerida é uma onda quase quadrada de 120°. No tipo senoidal, a força contra eletromotriz é senoidal e, para uma operação livre de oscilações de torque, a corrente de fase requerida é senoidal (Sen, 2013).

A obtenção do modelo matemático de motores do tipo BLDC baseia-se no princípio de que as correntes induzidas no rotor devido a campos harmônicos do estator e as perdas são negligenciadas (Krishnan, 2003). A relação corrente-tensão é dada pelas propriedades elétricas das bobinas e estas são dadas pelo seu circuito equivalente, como mostrado na Figura A.3, onde cada enrolamento do motor consiste em um resistor em série com um indutor. Uma fonte de tensão, que é a orientação de referência é oposta à tensão de alimentação, que ocasiona uma força contra eletromotriz. As três fases deste tipo de motor têm valores de impedância e resistência iguais, isto é, $R_a = R_b = R_c = R$ e $L_a = L_b = L_c = L$, o que implica em uma carga equilibrada. Aplicando a Lei de Kirchhoff das tensões em nas três fases o sistema de equações obtidos é dado por

$$\begin{bmatrix} v_{an} \\ v_{bn} \\ v_{cn} \end{bmatrix} = R \cdot \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + L \cdot \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix}, \quad (\text{A.5})$$

aplicando a transformada de Laplace na Eq. (3.1) e reorganizando os resultados o sistema de equações que representam o sistema no plano- s é dado por

$$\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = \frac{1}{Ls + R} \cdot \left(\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} - \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \right), \quad (\text{A.6})$$

A intensidade da força contra eletromotriz depende do valor de sua constante (k_e), da velocidade angular do rotor (ω_m) e a forma de onda é definida como uma função da posição angular do rotor (θ_m) de entrada (Nilsson & Modrack, 2013) e esta relação é dada por

$$e_i = k_e \cdot \omega_m \cdot f(\theta_m), \quad (\text{A.7})$$

sendo $i = (a, b, c)$ e e_i representa a força contra eletromotriz para cada uma das bobinas.

A função $f(\Omega)$ neste caso possui um formato trapezoidal e é obtida por uma *lookup-table*, que fornece o valor desta função para a posição angular (θ_m) referente a cada uma das três fases. Observa-se que a frequência do sinal da Eq. (A.7) e a frequência do sinal (Ω_e) são relacionadas pelo número de pares de pólos (p_m) nos ímãs permanentes do rotor, logo a frequência do ângulo elétrico é dada por

$$\Omega_e = \omega_m \cdot p_m, \quad (\text{A.8})$$

O passo final no levantamento do modelo matemático é modelar a dinâmica do rotor, ou seja, estabelecer uma relação entre a corrente elétrica e a velocidade angular, portanto, o torque aplicado externamente é desconsiderado e apenas uma carga inercial constante é levada em consideração, desta forma a equação mecânica é dada por

$$T_e = \frac{e_a \cdot i_a + e_b \cdot i_b + e_c \cdot i_c}{\omega_r}, \quad (\text{A.9})$$

A simulação do motor é realizada incluindo o torque de carga no modelo, para isto realiza-se a subtração entre o torque da carga e o torque gerado eletricamente, de modo que o resultado é dado por

$$T_m = T_e - T_L, \quad (\text{A.10})$$

sendo T_m o torque gerado pela rotação do motor e T_L o torque gerado por uma carga mecânica. Desta forma tem-se que a relação entre o torque e a velocidade angular é dada por

$$T_m = K_b \cdot \omega_m - J \frac{d\omega_m}{dt}, \quad (\text{A.11})$$

sendo K_b a constante de amortecimento e J o momento de inercia do rotor. A função de transferência que modela o torque em função da velocidade angular é dada por

$$\omega_m(s) = \frac{1}{J \cdot s + K_b} \cdot T_m. \tag{A.12}$$

As equações anteriores modelam o comportamento de um motor BLDC, desta maneira é realizada uma relação entre as mesmas e montado o diagrama de blocos do sistema (Nilsson & Modrack, 2013), que é ilustrado pela a Figura A.5.

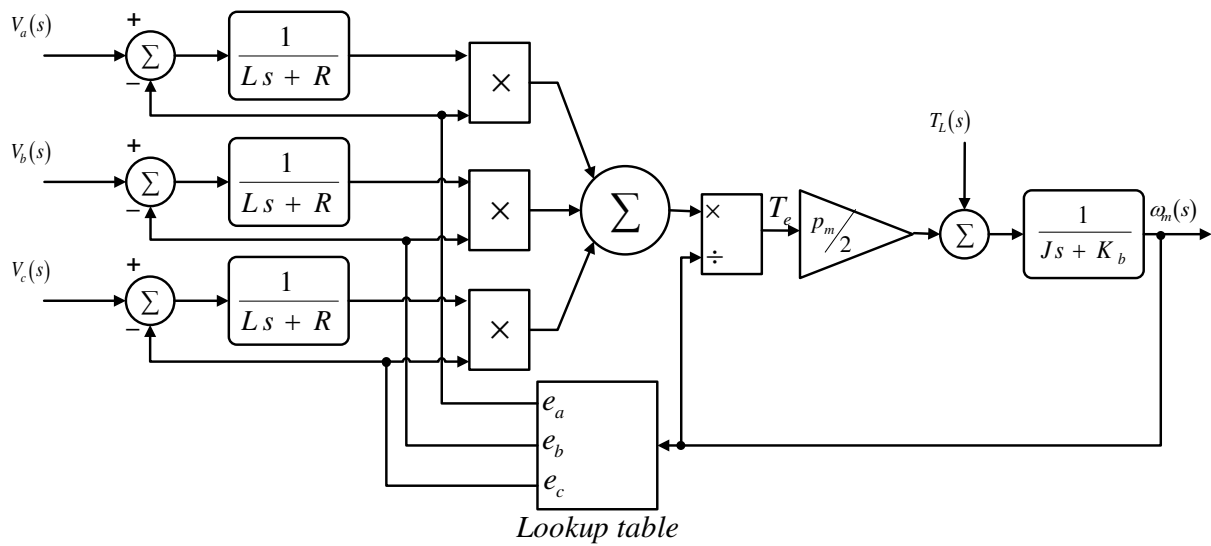


Figura A.5 – Diagrama de Blocos do Modelo Matemático de um Motor BLDC.

Os motores do tipo BLDC tem várias vantagens quando relacionados com motores DC escovados, mas as principais são a eficiência energética e uma grande relação peso potência. Porém, pode ser notado através dos parágrafos anteriores, motores deste tipo de configuração possuem um preço e grau de complexidade elevados por conta do seu circuito eletrônico de potência responsável por seu funcionamento (Sen, 2013). Devido a estas desvantagens muitas cadeiras de rodas motorizadas disponíveis no mercado utilizam motores CC com escovas.

A.2 Sensores

O uso de sensores em cadeiras de rodas motorizadas é de suma importância para o fechamento da malha de controle, o que torna sua operação eficiente e automática em aplicações no mundo real (Tzafestas, 2014).

O sistema de orientação é uma das aplicações mais utilizada em cadeiras de rodas motorizada e pode proporcionar vantagens aos seus usuários devido à sua simplicidade. Este tipo de ferramenta possibilita a localização e a movimentação da cadeira em qualquer momento e é indispensável para sistemas de controle.

Em sua grande maioria os sensores utilizados em robôs móveis são usados para medir valores simples como a temperatura do sistema ou a velocidade de rotação dos motores, já outros sensores mais sofisticados podem ser usados para adquirir informações sobre o ambiente ou mesmo para medir diretamente a posição global de um robô (Siegwart & Nourbakhsh, 2004).

O pré-requisito para uma orientação precisa é a disponibilidade de sensores confiáveis de alta resolução e infelizmente, os sensores da vida real praticamente disponíveis possuem várias imperfeições devido a diferentes causas (Tzafestas, 2014). O ruído do sensor é causado principalmente pelas variações ambientais que não podem ser capturadas pelo robô. Uma falha dos sensores aplicado em robôs é o *aliasing*, isto é, o fato de que as leituras dos sensores não são únicas, implica que (mesmo que não exista ruído), a quantidade disponível de informações na maioria dos casos não é suficiente para identificar a posição do robô a partir de uma única leitura do sensor. Para minimizar o efeito de ruído e *alias* na prática, técnicas especiais de processamento de sensores devem ser empregadas, e assim obter uma estimativa precisa da posição do robô ao longo do tempo. Essas técnicas incluem métodos de probabilidade e teoria da informação (estimadores bayesianos, filtros de Kalman, Filtro de Kalman Estendido, aproximadores fuzzy/neuronal, etc.). No Apêndice C é apresentado um algoritmo do Filtro de Kalman aplicado em sensores de rotação, que tem por intuito atenuar o ruído gerado pelo sensor.

Os sons têm capacidades de largura de banda espacial muito baixas e estão sujeitos a ruídos devido à dispersão de ondas e, portanto, o uso da detecção do alcance do laser é preferido. Mas, apesar de os sensores laser possuírem uma largura de banda muito maior, eles ainda são afetados pelo ruído. Além disso, os lasers têm um campo de visão restrito, a menos que sejam tomadas medidas especiais (por exemplo, incorporação de espelhos rotativos no projeto).

Ao se mover, um robô móvel frequentemente encontrará características imprevisíveis do ambiente o que é considerado preocupante. Os sensores normalmente utilizados na orientação de robôs são medidores de alcance através de tecnologias como sonar, laser e infravermelho, radar, sensores táteis, bússolas e GPS (Alves, et al., 2011). Desta maneira, a determinação da localização do robô utilizando um único sensor é insuficiente para capturar todas as características relevantes de ambientes reais. Então, é necessário combinar dados de múltiplos sensores em um processo conhecido como fusão. A fusão de dados de múltiplos sensores é uma tecnologia que permite combinar informações de vários sensores, onde o

objetivo é ter uma melhor precisão e dar um resultado mais preciso do que o que poderia ser obtido pelo uso de um único sensor. O sistema de sensores geralmente utilizado em cadeiras de rodas motorizadas consiste em dois odômetros, uma bússola digital (magnetômetro) e um acelerômetro (Nada, et al., 2016), ilustrados pela Figura A.6.

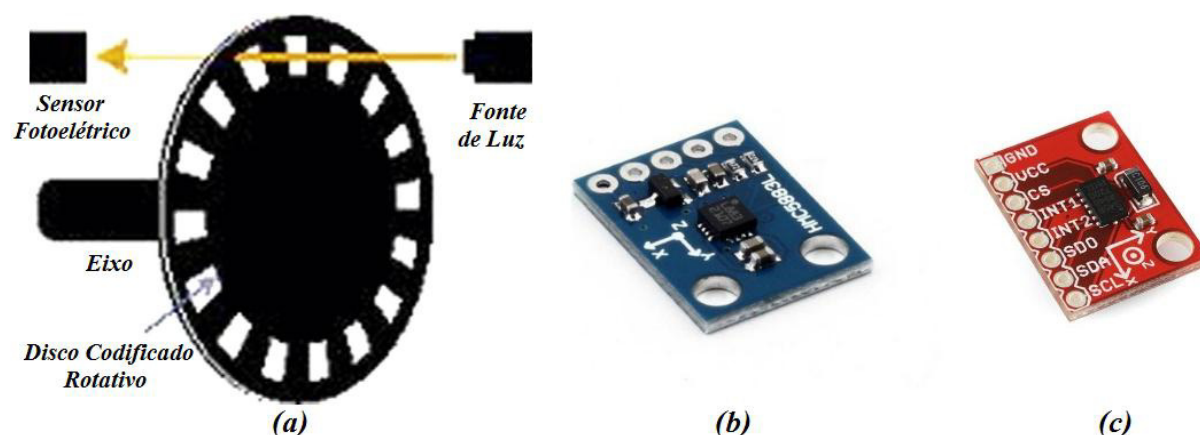


Figura A.6 – Exemplos de Sensores Utilizados no Sistema de Orientação de Cadeira de Rodas Motorizada. (a) Encoder Rotativo Óptico. (b) Bússola Digital. (c) Acelerômetro.

Os odômetros são dispositivos utilizados para determinação da posição (x, y) e a direção do sistema através de *encoders* rotativos. Estes tipos de dispositivos são montados na própria roda fornecendo desta maneira informações sobre a sua velocidade angular, e por integração é fornecida a posição angular. A resolução da medida da velocidade das rodas aumenta em uma proporção correspondente à do sistema de transmissão de movimento, onde sua precisão depende da qualidade do projeto mecânico e do desempenho do processamento dos sinais elétricos (Tzafestas, 2014). O *encoder* óptico é o dispositivo mais popular para este tipo de aplicação e é constituído basicamente por um bloqueador de luz que produz um certo número de pulsos de onda senoidal ou quadrada para cada rotação do eixo, sua estrutura básica é ilustrada pela Figura A.6.(a).

A bússola digital possui dois semicondutores de efeito Hall em ângulos retos, fornecendo dois eixos de direção de campo magnético, produzindo assim uma das oito direções possíveis da bússola, ilustrado pela Figura A.6.(b). Estes tipos de instrumentos são baratos, mas também sofrem de uma série de desvantagens, uma delas é sua baixa resolução e as fontes internas de erro incluem a não-linearidade do sensor básico e os erros de polarização sistemática no nível de semicondutores (Siegwart & Nourbakhsh, 2004). Para atenuar os problemas gerados pela aplicação destes componentes, um sistema de filtragem de sinais é necessário, no entanto o tempo de resposta do sensor é aumentado.

O acelerômetro é utilizado para medir a aceleração escalar utilizando o princípio da lei fundamental do movimento que é $F = m \cdot a$, onde F é a força (N), m é a massa (kg) e a é a aceleração (m/s^2). De maneira mais específica, este sensor consiste na igualdade entre a massa inercial do sensor de força sísmica e a força de polarização aplicada à massa, medido através de um circuito de medição de posição e então é convertido em um sinal. Os acelerômetros podem medir em até três eixos ortogonais, X, Y e Z. A aceleração é obtida geralmente em unidades de metros por segundo ao quadrado (m/s^2) ou G-force (g), onde $g = 9.8m/s^2$. Os acelerômetros são utilizados para detectar tanto a aceleração estática (por exemplo, gravidade) quanto a aceleração dinâmica, por serem afetados pela aceleração da gravidade, um acelerômetro pode ser utilizado para orientação em relação à superfície da Terra.

Apêndice B

ALGORITMOS MATLAB

B.1 Algoritmo de Simulação Sintonia LQR Controlador PID

```
% -----  
%                               Sistema de Controle                               %  
% -----  
%                               Cadeira de Rodas Motorizada                       %  
%                               Sintonia Ótima Controladores PID                 %  
% -----  
%                               Desenvolvimento: Evandro Martins                 %  
%                               Supervisão: Prof. João Viana                     %  
%                               Data: 12/07/2017                                %  
% -----  
%                               clc;clear;close all;                             %  
% ----- Gerar Modelo - Cinemático/Dinâmico ----- %  
Tf = 30;           % Tempo final (s)  
Ts = 0.01;        % Intervalo de amostragem (s)  
t = 0:Ts:Tf-Ts;   % Intervalo de tempo (-)  
N = length(t);    % Quantidade de amostras (-)  
a = 0.2;          % Distância do CG para eixo de tração (m)  
r = 0.25;         % Raio da roda (m)  
b = 0.2;          % Distância entre rodas/2 (m)  
% ----- Constantes - Controlador Cinemático ----- %  
Kxy = [2 2];      % Ganho do controlador em [x, y]  
Ixy = [2 2];      % Constante de saturação em [x, y]  
% ----- Referencias ----- %  
xy_r1 = [35 35];  % Ponto de referência [x y] (m)  
dxy_ref = zeros(N,2); % Velocidade escalar ref. (m/s)  
xy_ref = ones(N,1)*xy_r1; % Monta Vetor  
% ----- Função de Transferência - Motor+Driver ----- %  
[numc_l,denc_l] = WC_model_d(Ts); % [Num,Den] Motor Esquerdo  
[numc_r,denc_r] = WC_model_d(Ts); % [Num,Den] Motor Direito  
% ----- Parametros Controlador ----- %  
Q = diag([10 7 0]); % Matriz de ponderação de estados  
R = 1;              % Matriz de ponderação de entrada  
K = gain_lqr(Q,R)   % Ganhos Ótimos K = [Ki Kp Kd]  
s_sup = 5;          % Tensão máxima de saída do controlador (V)  
% ----- FT - Controlador PID ----- %  
Cs_l = pid(K(2),K(1)); C_l = c2d(Cs_l,Ts); % F.T. controlador esquerdo  
[num_cl,den_cl] = tfdata(C_l);  
den_cl = den_cl{:}; num_cl = num_cl{:};  
Cs_r = pid(K(2),K(1)); C_r = c2d(Cs_r,Ts); % F.T. controlador direito  
[num_cr,den_cr] = tfdata(C_r);
```

```

den_cr = den_cr{:}; num_cr = num_cr{:};
% ----- Condições Iniciais ----- %
q      = length(denc_l);
zz     = zeros(1,q); % Vetor auxiliar para valores iniciais
exy    = [zz' zz']; % Erro de regime [x y] (m)
U_r    = [zz' zz']; % Lei de controle ref. [v_escalor w_angular]
W_r    = [zz' zz']; % Lei de controle ref. [w_esq w_dir]
V      = zz; W      = zz; % V (V. Escalar) (m/s); W (V. Angular) (rad/s)
yl     = zz; yr     = zz; % V. Angular Inicial Roda Esquerda; Direita; (rad/s)
ec_l   = zz; ec_r   = zz; % Erro de Regime Roda Esquerda; Direita; (rad/s)
uc_r   = zz; uc_l   = zz; % Esforço de controle Roda Esquerda; Direita; (v)
dxy    = [zz' zz']; % Velocidade inicial (m/s)
xy     = [zz' zz']; % Posição (m)
psi    = zz; % Angulo inicial (rad)
v      = 0.04; % Amplitude do ruído de medição
for k=q:N
% ----- Controlador Cinemático ----- %
[exy(k,:), U_r(k,:), W_r(k,:)] = ...
    C_control(a, r, b, xy(k-1,:), xy_ref(k-1,:), dxy_ref(k-1,:), psi(k-1), Ixy, Kxy);
% ----- Controlador PID ----- %
ec_l(k) = W_r(k,1) - yl(k-1); % Erro
uc_l(k) = ec_l(k)*num_cl(1)+ec_l(k-1)*num_cl(2) -uc_l(k-1)*den_cl(2);

ec_r(k) = W_r(k,2) - yr(k-1); % Erro
uc_r(k) = ec_r(k)*num_cr(1)+ec_r(k-1)*num_cr(2)-uc_r(k-1)*den_cr(2);
% ----- Saturação ----- %
uc_l(k) = sat_effort(uc_l(k), s_sup);
uc_r(k) = sat_effort(uc_r(k), s_sup);
% ----- Dinâmica Motores L + R ----- %
el(k) = z*randn; % Ruído
yl(k) = -denc_l(2)*yl(k-1) -denc_l(3)*yl(k-2) ...
    +numc_l(2)*uc_l(k) +numc_l(3)*uc_l(k-1) + el(k); % Saída Roda Esq.

er(k) = z*randn; % Ruído
yr(k) = -denc_r(2)*yr(k-1) -denc_r(3)*yr(k-2) ...
    +numc_r(2)*uc_r(k) +numc_r(3)*uc_r(k-1) + er(k); % Saída Roda Dir.
% ----- Dinâmica Cadeira de Rodas ----- %
[V(k), W(k)] = D_wchair(yl(k), yr(k), r, b); % [(m/s), (rad/s)]
% ----- Cinemática Cadeira de Rodas ----- %
psi(k) = psi(k-1) + W(k)*Ts; % Integrador Posição Angular (rad)
dxy(k,:) = C_wchair(V(k), psi(k));
xy(k,:) = xy(k-1,:) + dxy(k,:)*Ts; % Integrador Posição no plano[x, y] (m)
end

figure(1)
hold on
title('Posição da Cadeira Plano (x,y)')
plot(xy(:,1), xy(:,2), '--b')
plot(xy_ref(:,1), xy_ref(:,2), '*r')
grid on
ylabel('Distância - y(m)')
xlabel('Distância - x(m)')
axis('square')
legend('Trajetoria', 'Referencia');
hold off

figure(2)
subplot(2,1,1)
hold on
title('Posição da Cadeira - x')
plot(t, xy_ref(:,1), '-r')

```

```

    plot(t,xy(:,1),'b')
    grid on
    legend('X Ref.','X Pos. ');
    ylabel('Posição (m)')
    xlabel('Tempo (s)')
hold off
subplot(2,1,2)
hold on
title('Posição da Cadeira - y')
plot(t,xy_ref(:,2),'-r')
plot(t,xy(:,2),'b')
grid on
legend('Y Ref.','Y Pos. ');
ylabel('Posição (m)')
xlabel('Tempo (s)')
hold off

```

figure(3)

```

subplot(2,1,1)
hold on
title('Velocidade Angular Roda Esquerda')
plot(t,W_r(:,1),'-r')
plot(t,y_l,'b')
grid on
ylabel('Velocidade Angular (rad/s)')
xlabel('Tempo (s)')
legend('W-ref','W-esq');
axis([0 Tf 0 20])
subplot(2,1,2)
hold on
title('Velocidade Angular Roda Direita')
plot(t,W_r(:,2),'-r')
plot(t,y_r,'b')
grid on
ylabel('Velocidade Angular (rad/s)')
xlabel('Tempo (s)')
legend('W-ref','W-dir');
axis([0 Tf 0 20])

```

figure(4)

```

subplot(2,1,1)
hold on
title('Esforço de Controle - Controlador Esquerdo')
plot(t,uc_l,'b')
grid on
ylabel('Tensão (V)')
xlabel('Tempo (s)')
subplot(2,1,2)
hold on
title('Esforço de Controle - Controlador Direito')
plot(t,uc_r,'b')
grid on
ylabel('Tensão (V)')
xlabel('Tempo (s)')

```

figure(5)

```

subplot(2,1,1)
hold on
title('Velocidade Escalar - Cadeira de Rodas')
plot(t,V,'b')
plot(t,U_r(:,1),'r')

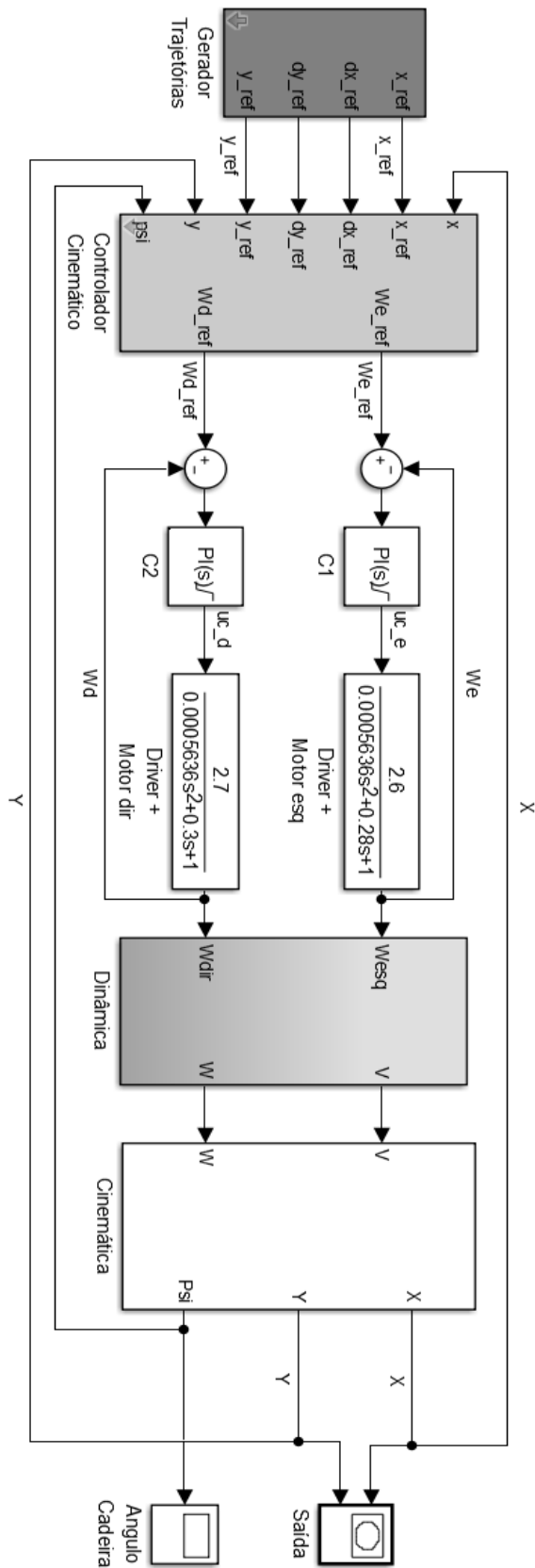
```



```
grid on
ylabel('Velocidade Escalar (V)')
xlabel('Tempo (s)')
legend('V-real', 'V-ref');

subplot(2,1,2)
hold on
title('Velocidade Angular - Cadeira de Rodas')
plot(t,W, 'b')
plot(t,U_r(:,2), 'r')
grid on
ylabel('W Angular (rad/s)')
xlabel('Tempo (s)')
legend('W-real', 'W-ref');
```

B.2 Algoritmo de Simulação Sintonia LQR Controlador PID Simulink



B.3 Algoritmo de Simulação Controlador PID-RNA

```

% -----
%                               Sistema de Controle                               %
% -----
%                               Cadeira de Rodas Motorizada                       %
%                               PID-RNA                                           %
% -----
%                               Desenvolvimento: Evandro Martins                 %
%                               Supervisão: Prof. João Viana                     %
%                               Data: 20/10/2017                                %
% -----
%                               clc;clear;close all; load ref_circ; load ref_eight; load ref_circ_2;
% ----- Gerar Modelo - Cinemático/Dinâmico ----- %
Tf = 30;           % Tempo final (s) 300s p/ circ
Ts = 0.01;        % Intervalo de amostragem (s)
t = 0:Ts:Tf-Ts;  % Intervalo de tempo (-)
N = length(t);   % Quantidade de amostras (-)
a = 0.2;         % Distancia do CG para eixo de tração (m)
r = 0.25;        % Raio da roda (m)
b = 0.2;         % Distância entre rodas/2 (m)
% ----- Constantes - Controlador Cinemático ----- %
Kxy = [1 1];     % Ganho do controlador em [x, y]
Ixy = [1.8 1.8]; % Constante de saturação em [x, y]
% ----- Referencias ----- %
% xy_ref = [ref_eigth(1:N,1) ref_eigth(1:N,4)]; % "Oito" de referência (m)
% dxy_ref = [ref_eigth(1:N,2) ref_eigth(1:N,3)]; % Veloc. referência (m/s)

xy_ref = [ref_circ_2(1:N,1) ref_circ_2(1:N,4)]; % Circ. de referência (m)
dxy_ref = [ref_circ_2(1:N,2) ref_circ_2(1:N,3)]; % Vel. referência (m/s)

% xy_r1 = [30 30];           % Ponto de referência [x y] (m)
% dxy_ref = zeros(N,2);     % Velocidade escalar ref. (m/s)
% xy_ref = ones(N,1)*xy_r1; % Monta Vetor

d1 = gensig('square',Tf/4,Tf,Ts); d1=1*d1; % Disturbance
d2 = gensig('square',Tf/4,Tf,Ts); d2=1*d2; d2=~d2; % Disturbance
% ----- Reference Model Design ----- %
zeta = 1;           % Damping Ratio
ts = 0.9;          % Settling time
pd = 10;           % Desired Pole
wn = pd/(zeta*ts); % Natural frequency
% ----- Função de Transferência - Motor+Driver ----- %
[numc_l,denc_l] = WC_model_d(Ts); % [Num,Den] Motor Esquerdo
[numc_r,denc_r] = WC_model_d(Ts); % [Num,Den] Motor Direito

[num_gdr,den_gdr] = REF_model_d(wn,zeta,Ts); % Discrete Model Referece
% ----- Parametros Controlador ----- %
s_sup = 5;         % Tensão máxima de saída do controlador (V)
% ----- Parameters ----- %
n = [0.0003 0.00003 0.00000001]; % Learning rate
% ----- Condições Iniciais ----- %
q = length(den_gdr); % Delay constant
zz = zeros(1,q); % Vetor auxiliar para valores iniciais
exy = [zz' zz']; % Erro de regime [x y] (m)
U_r = [zz' zz']; % Lei de controle ref. [v_escalare w_angular]
W_r = [zz' zz']; % Lei de controle ref. [w_esq w_dir]
V = zz; W = zz; % V (V. Escalar) (m/s); W (V. Angular) (rad/s)

```

```

y1 = zz; yr = zz; % V. Angular Inicial Roda Esquerda; Direita; (rad/s)
ym1 = zz; ym2 = zz;
ec_l = zz; ec_r = zz; % Erro de Regime Roda Esquerda; Direita; (rad/s)
uc_r = zz; uc_l = zz; % Esforço de controle Roda Esquerda; Direita; (v)
dxy = [zz' zz']; % Velocidade inicial (m/s)
xy = [zz' zz']; % Posição (m)
psi = zz; % Angulo inicial (rad)
v = 0.04; % Amplitude do ruído de medição
e1 = zz; ee1 = zz;
e2 = zz; ee2 = zz;
x1 = zeros(3,q); x2 = zeros(3,q);
K1 = 2;
K2 = 2;
w1 = (1/K1)*[0.5 2 0]; w1 = [w1;w1;w1];
w2 = (1/K2)*[0.5 2 0]; w2 = [w2;w2;w2];

for k=q:N
% ----- Controlador Cinemático ----- %
[exy(k,:),U_r(k,:),W_r(k,:)] = ...
C_control(a,r,b,xy(k-1,:),xy_ref(k-1,:),dxy_ref(k-1,:),psi(k-1),Ixy,Kxy);
% ----- Controlador PID ----- %
% ec_l(k)= W_r(k,1) - y1(k-1); % Erro

ee1(k) = W_r(k,1) - y1(k-1); % Error: Referencia -
Medido;
e1(k) = ym1(k-1) - y1(k-1); % Error: Desejado - Medido;
x1(1,k) = ee1(k); % NN State 1 - Error
x1(2,k) = x1(2,k-1) + ee1(k)*Ts; % NN State 2 - Integral error
x1(3,k) = (ee1(k-1) - ee1(k-2))/Ts; % NN State 3 - Derivative error

w1(k,:) = w1(k, :)/(sqrt(w1(k,:) *w1(k,:)' )); % Normalização
uc_l(k) = (K1*(w1(k,:) *x1(:,k)));

w1(k+1,1) = w1(k,1) + n(1)*e1(k)*uc_l(k)*x1(1,k); % Weight 1 Update: Kp
w1(k+1,2) = w1(k,2) + n(2)*e1(k)*uc_l(k)*x1(2,k); % Weight 2 Update: Ki
w1(k+1,3) = w1(k,3) + n(3)*e1(k)*uc_l(k)*x1(3,k); % Weight 3 Update: Kd

ee2(k) = W_r(k,2) - yr(k-1); % Error: Referencia - Medido;
e2(k) = ym2(k-1) - yr(k-1); % Error: Desejado - Medido;
x2(1,k) = ee2(k); % NN State 1 - Error
x2(2,k) = x2(2,k-1) + ee2(k)*Ts; % NN State 2 - Integral error
x2(3,k) = (ee2(k-1) - ee2(k-2))/Ts; % NN State 3 - Derivative error

w2(k,:) = w2(k, :)/(sqrt(w2(k,:) *w2(k,:)' )); % Normalização
uc_r(k) = (K2*(w2(k,:) *x2(:,k)));

w2(k+1,1) = w2(k,1) + n(1)*e2(k)*uc_r(k)*x2(1,k); % Weight 1 Update: Kp
w2(k+1,2) = w2(k,2) + n(2)*e2(k)*uc_r(k)*x2(2,k); % Weight 2 Update: Ki
w2(k+1,3) = w2(k,3) + n(3)*e2(k)*uc_r(k)*x2(3,k); % Weight 3 Update: Kd

% ----- Saturação ----- %
uc_l(k) = sat_effort(uc_l(k),s_sup);
uc_r(k) = sat_effort(uc_r(k),s_sup);
% ----- Dinâmica Motores L + R ----- %
e1(k) = 0*randn; % Ruído
y1(k) = -denc_l(2)*y1(k-1) -denc_l(3)*y1(k-2) ...
+numc_l(2)*(uc_l(k)-d1(k)) +numc_l(3)*(uc_l(k-1)-d1(k-1)); % Saída
Roda Esq.

```

```

ym1(k) = -den_gdr(2)*ym1(k-1) -den_gdr(3)*ym1(k-2) ...
         +num_gdr(2)*W_r(k-1,1) +num_gdr(3)*W_r(k-2,1);

er(k) = 0*randn; % Ruído
yr(k) = -denc_r(2)*yr(k-1) -denc_r(3)*yr(k-2) ...
         +numc_r(2)*(uc_r(k)-d2(k)) +numc_r(3)*(uc_r(k-1)-d2(k-1)); % Saída
Roda Dir.

ym2(k) = -den_gdr(2)*ym2(k-1) -den_gdr(3)*ym2(k-2) ...
         +num_gdr(2)*W_r(k-1,2) +num_gdr(3)*W_r(k-2,2);
% ----- Dinâmica Cadeira de Rodas ----- %
[V(k), W(k)] = D_wchair(yl(k),yr(k),r,b); % [(m/s), (rad/s)]
% ----- Cinemática Cadeira de Rodas ----- %
psi(k) = psi(k-1) + W(k)*Ts; % Integrador Posição Angular (rad)
dxy(k,:) = C_wchair(V(k),psi(k));
xy(k,:) = xy(k-1,:) + dxy(k,:)*Ts; % Integrador Posição no plano[x, y] (m)
end

disp(' NNA-PID Controller Gains 1')
disp([' Kp ', ' Ki ', ' Kd '])
disp([ K1*w1(k,1), K1*w1(k,2), K1*w1(k,3)])

disp(' NNA-PID Controller Gains 2')
disp([' Kp ', ' Ki ', ' Kd '])
disp([ K2*w2(k,1), K2*w2(k,2), K2*w2(k,3)])

aux = 1:fix(Tf/Ts); % Aux. Vetor
figure(1)
hold on
title('Posição da Cadeira Plano (x,y)')
plot(xy(:,1),xy(:,2),'-b','linewidth',1.5)
plot(xy_ref(:,1),xy_ref(:,2),'--r','linewidth',1.5)
grid on
ylabel('Distância - y(m)')
xlabel('Distância - x(m)')
axis('square')
legend('Trajetoria','Referencia');
hold off

figure(2)
subplot(2,1,1)
hold on
title('Posição da Cadeira - x')
plot(t,xy_ref(:,1),'-r')
plot(t,xy(:,1),'b')
grid on
legend('X Ref.','X Pos. ');
ylabel('Posição (m)')
xlabel('Tempo (s)')
hold off
subplot(2,1,2)
hold on
title('Posição da Cadeira - y')
plot(t,xy_ref(:,2),'-r')
plot(t,xy(:,2),'b')
grid on
legend('Y Ref.','Y Pos. ');
ylabel('Posição (m)')
xlabel('Tempo (s)')

```

```

hold off

figure(3)
subplot(2,1,1)
hold on
title('Velocidade Angular Roda Esquerda')
plot(t,W_r(:,1),'-r')
plot(t,yL,'b')
plot(t,ym1,'k')
grid on
ylabel('Velocidade Angular (rad/s)')
xlabel('Tempo (s)')
legend('W-ref','W-esq','W-Modelo Ref.');
```

```

axis([0 Tf -5 15])
subplot(2,1,2)
hold on
title('Velocidade Angular Roda Direita')
plot(t,W_r(:,2),'-r')
plot(t,yR,'b')
plot(t,ym2,'k')
grid on
ylabel('Velocidade Angular (rad/s)')
xlabel('Tempo (s)')
legend('W-ref','W-dir','W-Modelo Ref.');
```

```

axis([0 Tf -5 15])

figure(4)
subplot(2,1,1)
hold on
title('Esforço de Controle - Controlador Esquerdo')
plot(t,uc_l,'b')
plot(t,d1(aux),'r')
grid on
legend('u_1','d_1');
ylabel('Tensão (V)')
xlabel('Tempo (s)')
subplot(2,1,2)
hold on
title('Esforço de Controle - Controlador Direito')
plot(t,uc_r,'b')
plot(t,d2(aux),'r')
grid on
legend('u_2','d_2');
ylabel('Tensão (V)')
xlabel('Tempo (s)')
```

```

figure(5)
subplot(2,1,1)
hold on
title('Velocidade Escalar - Cadeira de Rodas')
plot(t,V,'b')
plot(t,U_r(:,1),'r')
grid on
ylabel('Velocidade Escalar (V)')
xlabel('Tempo (s)')
legend('V-real','V-ref');
```

```

subplot(2,1,2)
hold on
title('Velocidade Angular - Cadeira de Rodas')
plot(t,W,'b')
plot(t,U_r(:,2),'r')
```

```

grid on
ylabel('W Angular (rad/s)')
xlabel('Tempo (s)')
legend('W-real', 'W-ref');

figure(6)
subplot(2,1,1)
hold on
stairs(t,w1(aux,1),'-b','linewidth',0.5)
stairs(t,w1(aux,2),'-r','linewidth',0.5)
stairs(t,w1(aux,3),'-k','linewidth',0.5)
legend('\fontsize{10}K_P', '\fontsize{10}K_I', '\fontsize{10}K_D')
xlabel('\fontsize{12}Tempo - s')
title('\fontsize{16}Acompanhamento da Rede - Roda Esquerda')
hold off
subplot(2,1,2)
hold on
stairs(t,w2(aux,1),'-b','linewidth',0.5)
stairs(t,w2(aux,2),'-r','linewidth',0.5)
stairs(t,w2(aux,3),'-k','linewidth',0.5)
legend('\fontsize{10}K_P', '\fontsize{10}K_I', '\fontsize{10}K_D')
xlabel('\fontsize{12}Tempo - s')
title('\fontsize{16}Acompanhamento da Rede - Roda Direita')
hold off

```

B.4 Funções MATLAB© - Modelos matemáticos

B.4.1 Função WC_model_d.m

```

function [num,den] = WC_model_d(Ts)
b = 2.6; % Dados do artigo (Solea,2015)
a0 = 0.0005636; % Dados do artigo (Solea,2015)
a1 = 0.28; % Dados do artigo (Solea,2015)
a2 = 1; % Dados do artigo (Solea,2015)

num_gs = b;
den_gs = [a0 a1 a2];
Gs = tf(num_gs,den_gs); % Modelo Real para simulação da saída

Gd = c2d(Gs,Ts); % Modelo Real Discreto
[num,den] = tfdata(Gd);
den=den{:};
num=num{:};

end

```

B.4.2 Função gain_lqr.m

```

function K = gain_lqr(Q,R)

a0 = 0.0005636; % Do artigo
b = 2.6/a0; % Do artigo
a2 = 9.3/a0; % Do artigo
a1 = 0.28/a0; % Do artigo
a0 = a0/a0;

```

```
A = [0 1 0;
      0 0 1;
      0 -a2 -a1];
```

```
B = [0 0 b]';
```

```
K = lqr(A,B,Q,R);
```

```
end
```

B.4.3 Função C_control.m

```
function [exy, U_r, W_r] = C_control(a,r,b,xy,xy_ref,dxy_ref,psi,Ixy,Kxy)
% ----- %
%                               Controlador Cinemático                               %
% ----- %
exy(1) = xy_ref(1) - xy(1);      % Erro de regime x
exy(2) = xy_ref(2) - xy(2);      % Erro de regime y

B_c = [1/r -b/r ;
        1/r  b/r]; % Matrix Transform: [U, V] to [U_esq, U_dir]

A_c = [ cos(psi)   sin(psi) ;
        -sin(psi)/a cos(psi)/a];

X_c = [(dxy_ref(1) + Ixy(1)*tanh((Kxy(1)/Ixy(1))*exy(1)));
        (dxy_ref(2) + Ixy(2)*tanh((Kxy(2)/Ixy(2))*exy(2)))]];

U_r = A_c*X_c;                  % Lei de controle ref. [v_escal;w_angular]

W_r = B_c*A_c*X_c;              % Lei de controle ref. [w_esq;w_dir]
```

```
end
```

B.4.4 Função D_wchair.m

```
function [V, W] = D_wchair(Wesq,Wdir,r,L)
% ----- %
%                               Sistema Dinâmico Cadeira de Rodas                               %
% ----- %
Vesq = Wesq*r;
Vdir = Wdir*r;
V = (Vdir+Vesq)/2;
W = (Vdir-Vesq)/(2*L);
```

```
end
```


B.4.5 Função C_wchair.m

```
function [dxy] = C_wchair(V,psi)
% ----- %
%                               Sistema Cinemático Cadeira de Rodas %
% ----- %
dxy(1) = V*cos(psi);
dxy(2) = V*sin(psi);
end
```

B.4.6 Função sat_effort.m

```
% ----- %
%                               Saturação Esforço de Controle %
% ----- %
function [uc] = sat_effort(u,sat)
uc = u;
if u > sat
uc = sat;
end
if u < -sat
uc = -sat;
end

end
```

Apêndice C

IDENTIFICAÇÃO DE SISTEMAS

C.1 Identificação do Sistema Eletromecânico

O levantamento do modelo matemático nada mais é do que uma abstração do processo real com a finalidade de caracteriza-lo. Geralmente o modelo dinâmico que represente o sistema é muito difícil de ser obtido devido à complexidade do processo, que se dá por dinâmicas desconhecidas (Ljung, 1999). Afim de simplificar a problemática de encontrar um modelo que melhor represente o sistema, se faz necessário aplicar o procedimento de identificação de sistema, que constrói um modelo matemático a partir de dados medidos diretamente do sistema real. Para a presente estudo utilizou-se o algoritmo estimador dos mínimos quadrados recursivos (RLS) afim de se estimar os parâmetros do sistema, conforme exposto pelos blocos na Figura C.1.

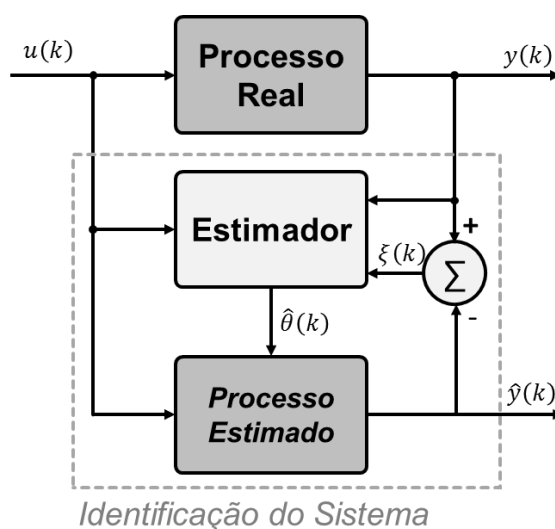


Figura C.1 – Diagrama de Blocos de Identificação do Sistema.

C.1.1. Algoritmo dos mínimos quadrados recursivos (RLS)

Em controladores adaptativos as observações são obtidas sequencialmente em tempo real. Espera-se que as computações sejam realizadas recursivamente e conseqüentemente o tempo de computação será menor. O cálculo dos mínimos quadrados recursivos pode ser organizado de tal maneira que os resultados obtidos em $(k - 1)$ podem ser usados para adquirir as estimativas no tempo k e (Åström & Wittenmark, 1995) o descrevem da seguinte forma

$$\hat{\theta}(k) = \hat{\theta}(k - 1) + K(k)\xi(k), \quad (13)$$

$$\xi(k) = y(k) - \varphi^T(k - 1)\hat{\theta}(k - 1), \quad (14)$$

$$K(k) = P(k - 1)\varphi(k - 1)(\lambda + \varphi^T(k - 1)P(k - 1)\varphi(k - 1))^{-1}, \quad (15)$$

$$P(k) = \frac{(I - K(k)\varphi^T(k - 1))P(k - 1)}{\lambda}, \quad (16)$$

sendo $\hat{\theta}(k)$ é o vetor de parâmetros estimados do modelo, o vetor $K(k)$ é o vetor de ganhos do estimador, $\varphi(k)$ representa o vetor de regressão, $\xi(k)$ é o erro de estimação, $y(k)$ é a saída da planta, $P(k)$ é a matriz de covariância do estimador e λ é o fator de esquecimento.

A Eq. (13) é o resultado desejado para a atualização recursiva do algoritmo RLS. Basicamente, apesar da grande complexidade do algoritmo RLS, seu objetivo é simplesmente calcular todos os componentes indicador que formam o novo vetor de ganho K_k os quais são utilizados para atualizar os coeficientes passados de $\hat{\theta}_{k-1}$ para os seus valores atuais $\hat{\theta}_k$ de acordo com os pré-requisitos do algoritmo dos mínimos quadrados.

C.1.2. Filtro de Kalman

O funcionamento do filtro pode ser conceituado basicamente através de dois estágios. O primeiro é chamado de estágio de predição, o qual produz um estado estimado “*a priori*”, a partir do estado anterior, utilizando um modelo de previsão a partir da evolução do sistema. O segundo estágio, conhecido como estágio de atualização, que leva em consideração as medições no sistema para produzir um estado estimado “*a posteriori*”, através da correção da estimativa “*a priori*” anterior. Esse segundo estágio do processo começa com um estado inicial \hat{x}_0 estimado e é repetido em um loop recursivamente até filtragem finalizar (Brown, 2012), como pode ser visto na Figura C.2.

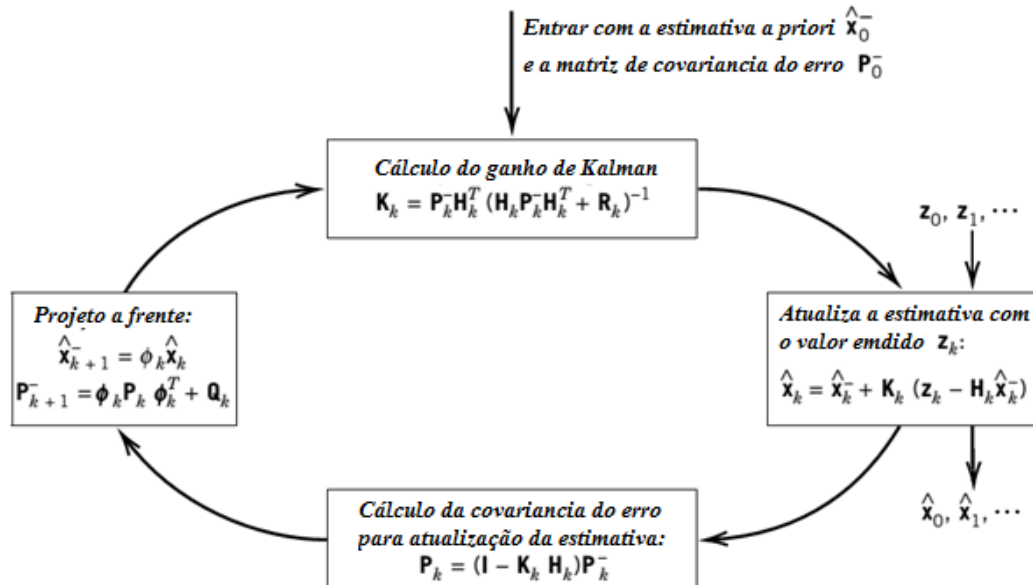


Figura C.2 – Algoritmo do filtro de Kalman. (Brown, 2012)

Nesta figura é possível verificar todos os processos que ocorrem durante a filtragem de Kalman, por estágio. Onde $\hat{\mathbf{x}}_k^-$ é o vetor de estado estimado “a priori”. $\hat{\mathbf{x}}_k$ é o vetor de estado estimado “a posteriori”, resultante da atualização de estimativa a partir das medidas \mathbf{z}_k . \mathbf{K}_k é a matriz de ganho ótimo de Kalman, que pondera a importância das novas medições introduzidas por \mathbf{z}_k na fase de atualização. \mathbf{P}_k é a matriz de covariâncias “a posteriori”. \mathbf{P}_{k+1}^- é a matriz de covariância, o que fornece a estimação do erro de covariância “a priori” (Brown, 2012). \mathbf{Q} é a matriz de covariância do processo que de certo modo pondera o peso das estimativas de processo. \mathbf{R} é a covariância do ruído observado, o qual reflete sobre o grau de confiança em cada uma das medições. As ponderações referidas se tornam maiores à medida que os elementos da matriz covariância tornam-se menores, o que significa que as grandezas estão se tornando mais confiáveis.

C.2 Descrição do Sistema de Ensaio

O sistema de aquisição de dados é composto por componentes capazes de medir as grandezas elétricas e mecânicas importantes da bancada, bem como processar esses sinais para obtenção de valores e resultados de interesse. A placa de aquisição de dados, mostrada na Figura C.3, modelo NI USB-6212 M, fabricada pela National Instruments (NI), é utilizada na conversão dos sinais analógicos em digitais para que sejam processados no software. Além da placa de aquisição, é utilizada uma fonte de tensão ajustável, para alimentação do motor.

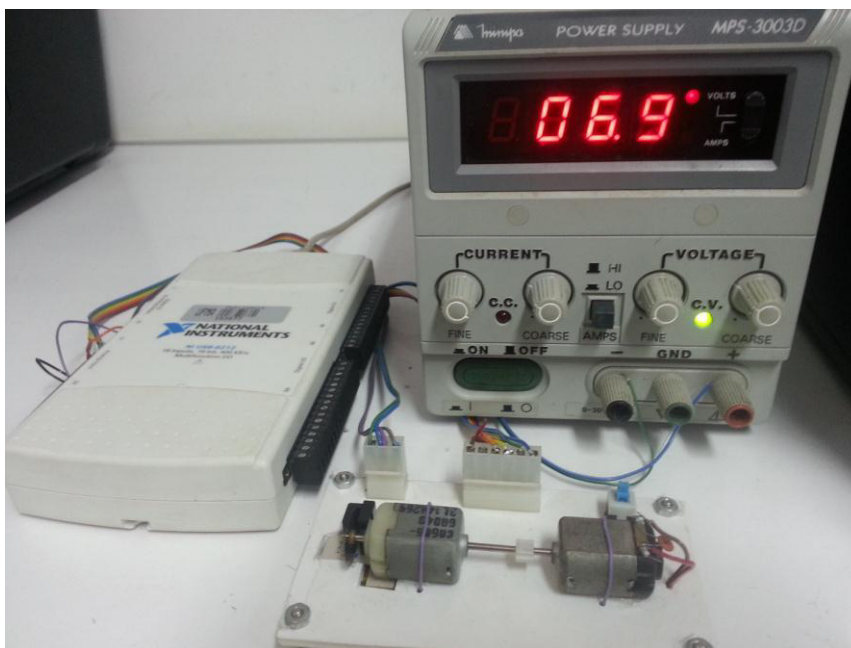


Figura C.3- Módulo de aquisição de dados NI USB-6212 M, Fonte alimentação regulável Minipa, motor CC com sensor de rotação e tacogerador.

Nesta imagem é possível se observar dois motores CC com seus eixos acoplados, o motor da esquerda é alimentado pela fonte de tensão e na outra ponta de seu eixo está acoplado um sensor ótico que faz a leitura de posição angular do eixo, podendo assim fornecer sua velocidade angular. O motor da direita funciona como tacogerador e a tensão gerada em seus terminais é adquirida através da placa de aquisição.

C.3 Tratamento dos sinais

Como abordado anteriormente, para esta etapa de tratamento de sinais foi utilizado um filtro de Kalman, afim de se remover erros apresentados durante a fase de aquisição. Este comportamento é representado pelo gráfico na Figura C.4 que relaciona o sinal medido pela placa de aquisição com o sinal estimado pelo filtro implementado.

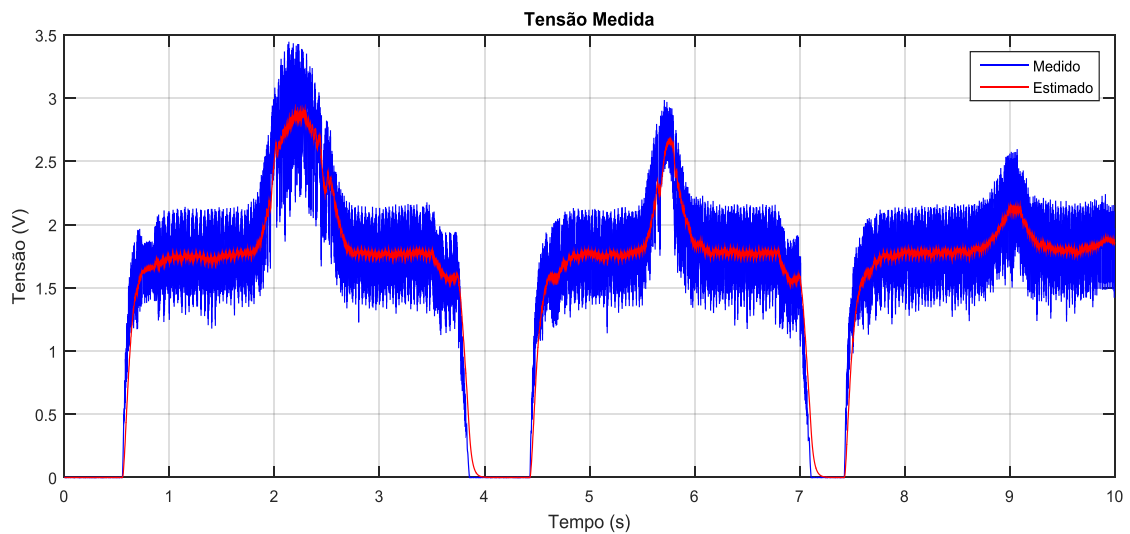


Figura C.4 - Tensão medida e filtrada.

A grande vantagem da utilização deste tipo de abordagem é que entre um instante de tempo e outro, é possível calcular uma nova estimativa do estado atual, o que viabiliza sua implementação junto a etapa de identificação, além melhorar os resultados durante esta fase.

C.4 Identificação do sistema

Para o início da etapa de identificação, se faz necessário a determinação da ordem do sistema, bem como a quantidade de polos e zeros. E para tal é necessário descrever o sistema por equações a diferença, para um sistema no tempo discreto com dois polos e nenhum zero, logo podemos escrever a saída do sistema como sendo

$$y(k) = [-y(k-1) \quad -y(k-2) \quad u(k-1)] * [a_1 \quad a_2 \quad b_0]^T \quad (17)$$

Com a estrutura do modelo do sistema já definida, é possível iniciar o algoritmo de identificação, que é representado pelo gráfico

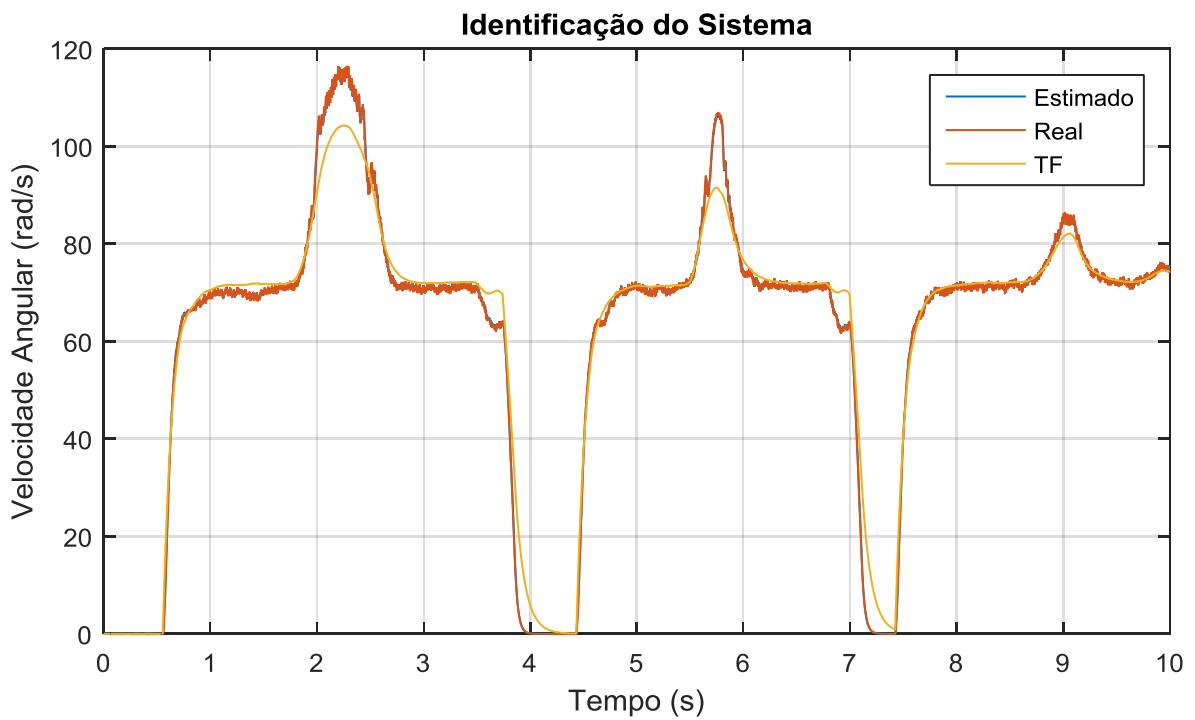


Figura C.5 - Processo de Identificação saída da planta: Estimada, Medida e Gerada pela Função Transferência estimada.

Na Figura C.6 é possível se verificar a convergência dos parâmetros ao longo das interações

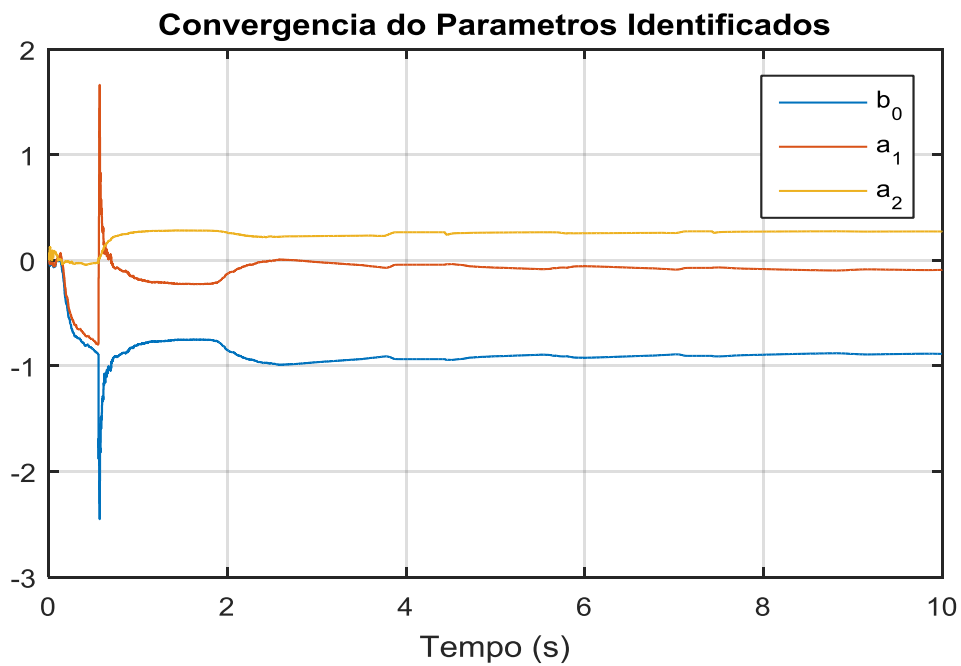


Figura C.6 - Convergência dos parâmetros ao longo das interações.

Como pode ser observado a partir dos seis segundos os parâmetros sofrem poucas alterações, mostrando que os parâmetros foram identificados e pode ser evidenciado pela Figura C.7, que apresenta o erro de estimação ao longo do processo de identificação do sistema.

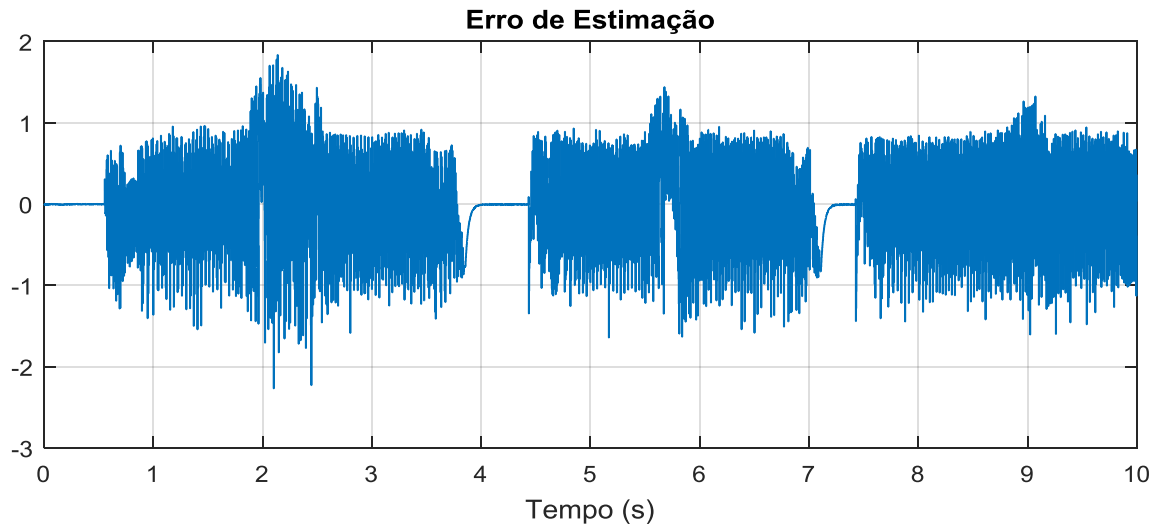


Figura C.7 - Gráfico do erro de estimação de parâmetros.

Analisando o gráfico do erro de estimação, é possível notar que o erro assume valores em um intervalo de dois à menos dois, levando em consideração o sinal de saída apresentado na Figura C.5, o que representa um erro de 1% nas regiões de transição, o que é aceitável.