



UNIVERSIDADE
ESTADUAL de LONDRINA

Alex Archela

**Técnicas de Modelagem, Estimação de
Parâmetros e Controle Embarcado Aplicados
em Veículos Elétricos com Trações Traseira
Independentes**

Londrina, PR

2019

Alex Archela

Técnicas de Modelagem, Estimação de Parâmetros e Controle Embarcado Aplicados em Veículos Elétricos com Trações Traseira Independentes

Dissertação de mestrado apresentada ao Departamento de Engenharia Elétrica da Universidade Estadual de Londrina, como requisito parcial para a obtenção do Título de MESTRE em Engenharia Elétrica.

Universidade Estadual de Londrina – UEL

Departamento de Engenharia Elétrica – DEEL

Programa de Pós-Graduação

Orientador: Prof. Dr. Leonimer Flávio de Melo

Londrina, PR

2019

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

Archela, Alex.

Técnicas de Modelagem, Estimação de Parâmetros e Controle Embarcado Aplicados em Veículos Elétricos com Trações Traseira Independentes / Alex Archela. - Londrina, 2019.
167 f. : il.

Orientador: Leonimer Flávio de Melo.

Dissertação (Mestrado em Engenharia Elétrica) - Universidade Estadual de Londrina, Centro de Tecnologia e Urbanismo, Programa de Pós-Graduação em Engenharia Elétrica, 2019.

Inclui bibliografia.

1. Controle digital - Tese. 2. Filtro de Kalman - Tese. 3. Identificação Paramétrica - Tese. 4. Hardware-in-the-loop - Tese. I. de Melo, Leonimer Flávio. II. Universidade Estadual de Londrina. Centro de Tecnologia e Urbanismo. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

Alex Archela

Técnicas de Modelagem, Estimação de Parâmetros e Controle Embarcado Aplicados em Veículos Elétricos com Trações Traseira Independentes

Dissertação de mestrado apresentada ao Departamento de Engenharia Elétrica da Universidade Estadual de Londrina, como requisito parcial para a obtenção do Título de MESTRE em Engenharia Elétrica.

BANCA EXAMINADORA

Prof. Dr. Leonimer Flávio de Melo
Universidade Estadual de Londrina

Prof. Dr. Ruberlei Gaino
Universidade Estadual de Londrina

Prof. Dr. Newton da Silva
Universidade Estadual de Londrina

Prof. Dr. Marcelo Favoretto Castoldi
Universidade Tecnológica Federal do Paraná

Londrina, PR
2019

Dedico este trabalho a Deus por me guiar e sustentar durante esta etapa da minha vida, à minha família pela compreensão e apoio que tornaram possível a conclusão desta monografia.

Agradecimentos

Agradeço ao meu orientador prof. Dr. Leonimer Flávio de Melo por todo o suporte, dedicação e orientação no desenvolvimento deste trabalho.

Ao departamento de Engenharia Elétrica da Universidade Estadual de Londrina pela oportunidade de ingressar ao programa de mestrado.

Ao apoio à pesquisa da CAPES por meio do recebimento de bolsa durante todo o período do desenvolvimento deste estudo.

Ao prof. Dr. Ruberlei Gaino e prof. Dr. Newton da Silva por aceitarem participar da minha banca de qualificação e pelas sugestões apresentadas que possibilitaram a melhora deste estudo.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Resumo

Este trabalho apresenta um estudo sobre técnicas de controle de velocidade e torque, identificação de sistema pela utilização de métodos gráficos e numéricos, levantamento de parâmetros de um motor em corrente contínua, voltados para a aplicação em veículos elétricos de rodas motorizadas independentes. O sistema apresentado é composto por uma roda suspensa acionada por um motor o qual é controlado por um sistema embarcado. Para execução dos experimentos e análises utilizou-se o sistema em *hardware-in-the-loop*, que possibilita a prototipagem rápida dos sistemas de controle. Os resultados teóricos e experimentais são apresentados em forma de gráficos, tabelas e discutidos quanto ao desempenho de cada uma das técnicas.

Palavras-chave: Controle digital. Filtro de Kalman. Identificação paramétrica. Prototipagem rápida. Hardware-in-the-Loop.

Abstract

This present work presents a study of the speed and torque control techniques, system identification by graphic and numeric methods, estimation of DC motor parameters, with focus on application on an independent motored wheels electric vehicle. The system described is formed by a suspended wheel driven by a DC motor, which is controlled by a embedded system. For the development of the experiments and analysis it was used the system in hardware in the loop, which provides the possibility of fast prototyping os the control systems. The results are presented with the use of graphs, tables and discussed about each performance.

Keywords: Digital control. Kalman filter. Parametric identification. Fast prototyping. Hardware-in-the-Loop.

Lista de ilustrações

Figura 1 – Exemplo do valor da força de atrito máxima em relação a taxa de escorregamento entre a roda e a superfície para diferentes pisos.	35
Figura 2 – Exemplo de movimento de guinada, gerado por diferentes valores de torque aplicados em cada roda ($F_2 + F_4 > F_1 + F_3$).	36
Figura 3 – Esquema de funcionamento do sistema em <i>hardware-in-the-loop</i>	37
Figura 4 – Modelo elétrico de um motor em corrente contínua.	41
Figura 5 – Esquema de controle global-local para veículo elétrico movido a quatro motores independentes em cada roda.	46
Figura 6 – Diagrama de blocos do controlador PID em malha fechada atuante sobre uma planta qualquer.	54
Figura 7 – Exemplo de resposta à entrada degrau para aplicação do primeiro método de Ziegler-Nichols.	55
Figura 8 – Diagrama de blocos de um observador de estados em malha aberta baseado no modelo da planta.	60
Figura 9 – Diagrama de blocos de um observador de estados em malha fechada.	61
Figura 10 – Diagrama de blocos de um sistema genérico em <i>hardware-in-the-loop</i>	63
Figura 11 – Roda suspensa acoplada ao eixo do motor por meio de um conjunto coroa-corrente-pinhão, para realização testes de controle de velocidade e torque.	66
Figura 12 – Esquema elétrico de ponte H e seus circuitos de acionamento.	67
Figura 13 – Posicionamento do sensor de efeito Hall e ímãs ao redor do eixo do motor elétrico em corrente contínua.	68
Figura 14 – Ilustração do microcontrolador embarcado <i>BeagleBone Black</i> com os pinos usados e suas funções.	69
Figura 15 – Diagrama de blocos do funcionamento do sistema em <i>hardware-in-the-loop</i>	71
Figura 16 – Interface de configuração dos parâmetros para comunicação com o <i>hardware</i>	72
Figura 17 – Interface de configuração dos parâmetros para geração de código.	73
Figura 18 – Ilustração para execução do código em HIL.	74
Figura 19 – Ilustração do relatório gerado a partir do gerador de código do <i>Simulink</i>	74
Figura 20 – Diagrama de blocos para determinação do valor da velocidade média de rotação do motor a cada 100 ms.	75
Figura 21 – Código para captação dos dados do <i>Simulink</i>	76
Figura 22 – Diagrama de blocos para determinação do valor da velocidade instantânea de rotação do motor pelo segundo método.	77
Figura 23 – Função no <i>MATLAB</i> para correção da velocidade medida.	77
Figura 24 – Valores da constante K_e e valor médio obtidos experimentalmente em função da tensão.	79

Figura 25 – Curva da velocidade em função do tempo para a entrada degrau.	82
Figura 26 – Diagrama de blocos do controlador PID embarcado, com realimentação, normalização e limitador para geração do sinal de controle.	84
Figura 27 – Diagrama de blocos do controlador PID embarcado, com realimentação, normalização e limitador para geração do sinal de controle.	85
Figura 28 – Diagrama de blocos do sistema de controle de direção a partir do sinal de controle gerado pelo PID.	87
Figura 29 – Diagrama de blocos do sistema de acionamento dos pinos <i>PWM horário</i> e <i>PWM anti-horário</i> com proteção contra curto-circuito.	88
Figura 30 – Função no <i>MATLAB</i> mostrada na Figura 29 para proteção contra curto-circuito durante a mudança de sentido.	88
Figura 31 – Aplicação do filtro de Kalman para observação dos estados e realimentação do valor da velocidade observado.	89
Figura 32 – Interface do filtro de Kalman com os respectivos parâmetros.	91
Figura 33 – Diagrama de blocos para simulação da resposta do espaço de estados, com conversão de rad/s para RPM.	94
Figura 34 – Velocidade angular medida e estimada em função do tempo para uma entrada degrau com valor igual a 12 V.	94
Figura 35 – Erro entre velocidade angular medida e estimada em função do tempo para uma entrada degrau com valor igual a 12 V.	95
Figura 36 – Torque estimado pelo observador de estados por meio do modelo matemático da planta em função do tempo para uma entrada degrau com valor igual a 12 V.	95
Figura 37 – Reta de auxílio para Método de Ziegler/Nichols sobre o gráfico de velocidade em função do tempo para uma entrada degrau de valor igual a 12 V. . .	96
Figura 38 – Código para identificação paramétrica segundo o método de Ziegler/Nichols.	97
Figura 39 – Velocidade medida e estimada pelo método de Ziegler/Nichols, em função do tempo.	98
Figura 40 – Erro entre a velocidade medida e a estimada pelo método de Ziegler/Nichols, em função do tempo.	98
Figura 41 – Código para identificação paramétrica segundo o método de Hägglund. . . .	99
Figura 42 – Velocidade medida e estimada pelo método de Hägglund, em função do tempo.	100
Figura 43 – Erro entre a velocidade medida e a estimada pelo método de Hägglund, em função do tempo.	100
Figura 44 – Código para identificação paramétrica segundo o método de Smith.	101
Figura 45 – Velocidade medida e estimada pelo método de Smith, em função do tempo. .	102
Figura 46 – Erro entre a velocidade medida e a estimada pelo método de Smith, em função do tempo.	102

Figura 47 – Código para identificação paramétrica segundo o método de Sundaresan/Krishnaswamy.	103
Figura 48 – Velocidade medida e estimada pelo método de Sundaresan/Krishnaswamy, em função do tempo.	104
Figura 49 – Erro entre a velocidade medida e a estimada pelo método de Sundaresan/Krishnaswamy, em função do tempo.	104
Figura 50 – Código para identificação paramétrica segundo o método de índices de desempenho.	106
Figura 51 – Velocidade medida e estimada pelo método de índices de desempenho, em função do tempo.	107
Figura 52 – Erro entre a velocidade medida e a estimada pelo método de índices de desempenho, em função do tempo.	107
Figura 53 – Velocidade medida e estimada pelo método de Mollenkamp, em função do tempo.	108
Figura 54 – Código para identificação paramétrica segundo o método de Mollenkamp. .	109
Figura 55 – Erro entre a velocidade medida e a estimada pelo método de Mollenkamp, em função do tempo.	110
Figura 56 – Código para identificação paramétrica segundo o método dos mínimos quadrados.	111
Figura 57 – Interface do <i>APP System Identification</i> com seus respectivos parâmetros. . .	112
Figura 58 – Interface do <i>APP System Identification</i> com seus respectivos parâmetros. . .	113
Figura 59 – Velocidade medida e estimada pelo método ARMA de segunda ordem, em função do tempo.	113
Figura 60 – Erro entre a velocidade medida e a estimada pelo método ARMA de segunda ordem, em função do tempo.	114
Figura 61 – Valor do sinal de controle não normalizado em função do tempo.	117
Figura 62 – Velocidade de rotação do motor obtida pela leitura do sensor pelo método da velocidade média, em função do tempo.	118
Figura 63 – Velocidade de rotação do motor obtida pela leitura do sensor pelo método da velocidade instantânea, em função do tempo.	118
Figura 64 – Comparação entre as velocidades obtidas pela leitura do sensor pelo primeiro e segundo método, em função do tempo.	119
Figura 65 – Resultado simulado da tensão do controlador PID para modelo obtido pela identificação paramétrica pelo método ARMA em função do tempo.	120
Figura 66 – Resultado simulado da velocidade de rotação do motor para modelo obtido pela identificação paramétrica pelo método ARMA em função do tempo. . .	120
Figura 67 – Resultado simulado da tensão do controlador PID segundo o modelo matemático do sistema, em função do tempo.	121

Figura 68 – Resultado simulado da velocidade de rotação do motor segundo o modelo matemático do sistema, em função do tempo.	121
Figura 69 – Velocidade de rotação do motor em função do tempo, para o controle PID ajustado para o modelo identificado pelo método ARMA.	122
Figura 70 – Valor do sinal de controle em função do tempo, para o controlador PID ajustado para o modelo identificado pelo método ARMA.	123
Figura 71 – Sinal de controle do ciclo ativo do PWM em função do tempo, para o PID ajustado para o modelo identificado pelo método ARMA.	123
Figura 72 – Velocidade de rotação do motor em função do tempo, para o controle PID ajustado para o modelo matemático.	124
Figura 73 – Valor do sinal de controle em função do tempo, para o controlador PID ajustado para o modelo matemático.	124
Figura 74 – Sinal de controle do ciclo ativo do PWM em função do tempo, para o controle PID ajustado para o modelo matemático.	125
Figura 75 – Velocidade em função do tempo para um valor de frequência de amostragem igual a 1 kHz.	126
Figura 76 – Sinal de erro entre a velocidade requerida e a medida, em função do tempo, um valor de frequência de amostragem igual a 1 kHz.	126
Figura 77 – Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz.	127
Figura 78 – Sinal de controle do ciclo ativo do PWM horário em função do tempo, um valor de frequência de amostragem igual a 1 kHz.	127
Figura 79 – Sinal de controle do ciclo ativo do PWM anti-horário em função do tempo, um valor de frequência de amostragem igual a 1 kHz.	128
Figura 80 – Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado.	129
Figura 81 – Sinal de erro entre a velocidade requerida e a medida à saída do sistema, em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado.. . . .	129
Figura 82 – Sinal de controle gerado pelo PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado.	130
Figura 83 – Sinal de controle do ciclo ativo do PMW em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado.	130
Figura 84 – Velocidade de rotação do motor em função do tempo, com aplicação do filtro de Kalman.	131
Figura 85 – Erro entre a velocidade medida e real, em função do tempo, com aplicação do filtro de Kalman.	132
Figura 86 – Sinal de controle em função do tempo, com aplicação do filtro de Kalman.	132

Figura 87 – Erro entre a velocidade requerida e observada, em função do tempo, com aplicação do filtro de Kalman.	133
Figura 88 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, com aplicação do filtro de Kalman.	133
Figura 89 – Velocidade de rotação do motor em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.	134
Figura 90 – Erro entre a velocidade medida e estimada, em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.	134
Figura 91 – Sinal de controle em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.	135
Figura 92 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.	136
Figura 93 – Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman.	136
Figura 94 – Erro entre o valor da velocidade medida e o estimado pelo filtro de Kalman em função do tempo, um valor de frequência de amostragem igual a 1 kHz. .	137
Figura 95 – Erro entre a velocidade requerida e a estimada, em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman.	137
Figura 96 – Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman.	138
Figura 97 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman. .	138
Figura 98 – Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.	139
Figura 99 – Erro entre o valor da velocidade medida e o estimado pelo filtro de Kalman em função do tempo, um valor de frequência de amostragem igual a 1 kHz, para uma entrada de onda quadrada.	140
Figura 100 – Erro entre a velocidade requerida e a estimada, em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.	140
Figura 101 – Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.	141
Figura 102 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.	141

Figura 103–Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.	142
Figura 104–Erro entre o valor da velocidade medida e o estimado pelo filtro de Kalman em função do tempo, um valor de frequência de amostragem igual a 1 kHz, para uma entrada de onda quadrada.	143
Figura 105–Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.	143
Figura 106–Sinal de controle do ciclo ativo dos PWMs em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada. . .	144
Figura 107–Sinal de Torque gerado pelo motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.	144
Figura 108–Sinal do torque estimado com filtro de Kalman em função do tempo.	145
Figura 109–Erro entre o torque requerido e o estimado pelo filtro de Kalman em função do tempo.	146
Figura 110–Sinal de controle do ciclo ativo do PWM em função do tempo.	146
Figura 111–Sinal de velocidade medido e estimado pelo filtro de Kalman, em função do tempo.	149
Figura 112–Sinal do torque estimado pelo filtro de Kalman em função do tempo.	150
Figura 113–Erro entre o torque requerido e estimado pelo filtro de Kalman, em função do tempo.	150
Figura 114–Sinal de controle em função do tempo.	151
Figura 115–Sinal de controle do ciclo ativo dos PWMs em função do tempo.	151
Figura 116–Velocidade de rotação do motor em RPM, em função do tempo, com alteração de cargas.	152
Figura 117–Sinal do ruído de medição eliminado pelo filtro de Kalman em RPM, em função do tempo, com alteração de cargas.	153
Figura 118–Tensão de alimentação em V, em função do tempo, com alteração de cargas.	153
Figura 119–Sinal de PWM para acionamento do controlador, em função do tempo, com alteração de cargas.	154
Figura 120–Sinal de controle do ciclo ativo dos PWMs em função do tempo, com alteração de cargas.	154
Figura 121–Projeto do controle de velocidade por torque, com filtro de Kalman.	163

Lista de tabelas

Tabela 1	– Valores de K_p , T_I e T_d para controladores P, PI e PID, conforme a primeira regra de Ziegler-Nichols, baseado na resposta ao degrau.	55
Tabela 2	– Valores de K_p , T_I e T_d para controladores P, PI e PID, conforme a segunda regra de Ziegler-Nichols, baseado na constante crítica do controle proporcional.	56
Tabela 3	– Valores de tensão, corrente e velocidade de rotação obtidos experimentalmente e valores de K_e respectivamente calculados.	79
Tabela 4	– Valores de corrente e velocidade de rotação obtidos experimentalmente para determinação do valor de B_m	80
Tabela 5	– Parâmetros que compõem o modelo matemático da planta do sistema determinados experimentalmente.	80
Tabela 6	– Valores de tensão, corrente e velocidade de rotação para determinação das constantes do sistema.	81
Tabela 7	– Valores das constantes de força eletromotriz, de torque eletromagnético e de viscosidade experimentalmente obtidos.	81
Tabela 8	– Funções de transferência e ITSE para cada método de identificação paramétrica de primeira ordem.	105
Tabela 9	– Funções de transferência e ITSE para cada método de identificação paramétrica de segunda ordem.	110
Tabela 10	– Funções de transferência e ITSE para os métodos de identificação paramétrica de primeira e segunda ordem gráficos, determinísticos, através do modelo matemático e ARMA.	115

Lista de abreviaturas e siglas

α	Autovalores
$\mathbf{\Gamma}$	Matriz de entrada discretizada
Δ_d	Polinômio característico
ϵ	Vetor de erros
ϵ	Erro
$\dot{\epsilon}$	Derivada do erro
θ	Tempo de atraso
κ	Valor de estabilização
μ	Polo desejado
ν	Ruído de medição
ξ	Constante de amortecimento
π	Constante matemática
τ	Constante de tempo
Φ	Matriz de estado para alocação de polos
Φ	Matriz de estado discretizada
ϕ	Ângulo
ψ	Vetor reposta de identificação
ω	Velocidade angular
ω'	Derivada de primeira ordem da velocidade angular
ω_{motor}	Velocidade angular do eixo do motor
ω_n	Frequência natural
ω_{roda}	Velocidade angular da roda
$\mathbf{A}$	Matriz de estado

$\tilde{\mathbf{A}}$	Matriz de estado realimentada
$\mathbf{A}_c$	Matriz de estado de Frobenius
ABS	Sistema frenagem anti-bloqueio (<i>antiblock braking system</i>)
AR	Autoregressivo (<i>autoregressive</i>)
ARMA	Média móvel autorregressiva (<i>autoregressive moving average</i>)
ARMAX	Auto-regressivo de médias móveis com modelo de inputs exógenos (<i>Auto-regressive moving average model with exogenous inputs</i>)
$\mathbf{B}$	Matriz de entrada
$\mathbf{B}_c$	Matriz de entrada de Frobenius
B_m	Coefficiente de viscosidade
$\mathbf{C}$	Matriz de saída
c.c.	Corrente contínua
c.c.-c.c.	Corrente contínua para corrente contínua
$\mathbf{D}$	Matriz de transmissão direta
e	Sinal de erro
e	Força eletromotriz
e	Constante de Euler
EHRPWM	Enhanced High-Resolution Pulse Width Modulation
$\mathbf{F}$	Força de tração
$\mathbf{F}$	Matriz de identificação
F_{roda}	Força de tração máxima na roda
G_p	Função de transferência
HIL	Hardware-in-the-Loop
$\mathbf{I}$	Matriz identidade
I	Corrente constante
i	Índice

i_a	Corrente de armadura do motor
i'_a	Derivada de primeira ordem da corrente de armadura do motor
I_{roda}	Constante de inércia da roda
ITSE	Integral do tempo multiplicado pelo erro ao quadrado (<i>Integral of the Time multiplied by the Square Error</i>)
J_m	Momento de inércia do motor
J_L	Momento de inércia da roda
K	Ganho do filtro de Kalman
K	Ganho do controlador
k	Índice do vetor discretizado
k	Vetor de ganhos ótimos
k	Ganho ótimo
K_{cr}	Ganho crítico
K_d	Ganho do derivador
K_e	Constante de força eletromotriz
K_I	Ganho do integrador
K_p	Ganho da função de transferência
K_t	Constante de torque eletromagnético
L	Tempo de atraso do controlador
L	Ganho do observador de estados
L_a	Indutância da armadura do motor
m	Massa da roda
MA	Média móvel (Moving Average)
MA	Vantagem mecânica (Mechanical advantage)
MOSFET	Transistor de efeito de campo metal-óxido-semicondutor (Metal Oxide Semiconductor Field Effect Transistor)
N	Número de termos da média móvel

n	Número de polos desejados
n	Velocidade de rotação
$n_{NomRoda}$	Velocidade de rotação máxima da roda
P	Potência nominal do motor
P	Erro de covariância
$\tilde{P}$	Predição do erro de covariância
P_{cr}	Período crítico de oscilação
PID	Proporcional integral derivativo
PWM	Modulação por largura de pulso (Pulse Width Modulation)
Q	Covariância do ruído da planta
r	Sinal de referência
r	raio da roda
R	Covariância do ruído de medição
R_a	Resistência da armadura do motor
s	Eixo do Plano-s
T	Período de amostragem
$\mathbf{T}$	Matriz de transformação
T	Constante de tempo do controlador
t	Tempo
T_d	Constante de tempo do derivador
t_d	Tempo de atraso
T_i	Constante de tempo do integrador
T_L	Torque da carga sobre o eixo do motor
T_m	Torque eletromagnético
T'_m	Derivada de primeira ordem do torque eletromagnético
T_{motor}	Torque gerado pelo motor

t_p	Tempo de pico
T_{roda}	Torque gerado na roda
t_s	Tempo de estabilização
u	Entrada da planta
V	Tensão constante
v	Tensão de alimentação do motor
V_{in}	Erro compensado
v_{vel}	Velocidade máxima gerada pela roda
w	Ruído da planta
w_c	Matriz de controlabilidade
x	Constante de tempo de Mollenkamp
$\mathbf{x}$	Vetor de estados da planta
$\tilde{x}$	Vetor de estados estimado da planta
$\dot{\mathbf{x}}$	Vetor das derivadas de primeira ordem dos estados da planta
$\tilde{\dot{x}}$	Vetor das derivadas de primeira ordem dos estados estimados da planta
y	Direção no espaço cartesiano
y	Saída da planta
$\mathbf{y}$	Vetor de saída da planta
z	Eixo do Plano-z
ZOH	Retentor de ordem nula (<i>Zero Order Holder</i>)

Sumário

1	INTRODUÇÃO	35
1.1	Objetivo geral	38
1.2	Objetivos específicos	38
1.3	Justificativas	38
1.4	Descrição da estrutura do trabalho	39
2	MODELAGEM MATEMÁTICA E MÉTODOS DE CONTROLE	41
2.1	Modelo dinâmico de um motor em corrente contínua	41
2.2	Espaço de estados de um motor em corrente contínua	43
2.3	Discretização do espaço de estados	44
2.4	Tração de veículos elétricos	45
2.4.1	Arquitetura de controle global-local	46
2.5	Identificação paramétrica	47
2.5.1	Determinação dos parâmetros do motor	47
2.5.2	Método determinístico de identificação paramétrica	48
2.5.3	Método heurístico de identificação paramétrica	51
2.6	Projeto do controlador proporcional integral derivativo	53
2.6.1	Método de Ziegler-Nichols	54
2.6.2	Alocação de polos	56
2.6.3	PID discreto	59
2.7	Projeto do observador de estados	59
2.8	Filtro de Kalman	61
2.9	Sistema em <i>Hardware-in-the-Loop</i>	63
2.10	Conclusão do capítulo	64
3	DESCRIÇÃO DO SISTEMA	65
3.1	Projeto mecânico motor-roda	65
3.2	Descrição do sistema elétrico	66
3.2.1	Driver ponte-H do motor	67
3.2.2	Sensor de velocidade do motor	68
3.3	Sistema embarcado	69
3.4	Sistema de <i>Hardware-in-the-Loop</i>	70
3.4.1	Configuração do <i>MatLab</i>	71
3.4.2	Configuração do <i>Simulink</i>	72
3.5	Leitura do sensor de velocidade	75
3.5.1	Método da velocidade média	75

3.5.2	Método da velocidade instantânea	76
3.6	Modelagem da planta	78
3.6.1	Identificação dos parâmetros do motor	78
3.6.2	Identificação do sistema completo	81
3.6.3	Identificação paramétrica por meio da entrada degrau	82
3.7	Projeto do controlador PID	82
3.7.1	Ajuste de parâmetros do controlador PID	85
3.7.2	Sistema de acionamento	85
3.7.3	Sistema de controle de direção	86
3.7.4	Sistema de proteção	88
3.8	Aplicação do filtro de Kalman	89
3.9	Conclusão do capítulo	92
4	IDENTIFICAÇÃO PARAMÉTRICA DO SISTEMA	93
4.1	Modelagem matemática com análise parcial	93
4.2	Métodos experimentais determinísticos de primeira ordem	96
4.2.1	Método de Ziegler/Nichols	96
4.2.2	Método de Hägglund	99
4.2.3	Método de Smith	101
4.2.4	Método de Sundaresan/Krishnaswamy	103
4.3	Análise dos resultados dos métodos de primeira ordem	105
4.4	Métodos experimentais determinísticos de segunda ordem	106
4.4.1	Método por índices de desempenho	106
4.4.2	Método de Mollenkamp	108
4.4.3	Análise dos resultados dos métodos de segunda ordem	110
4.5	Método de mínimos quadrados	111
4.6	Conclusão do capítulo	114
5	RESULTADOS	117
5.1	Medição de velocidade	117
5.2	Controle PID	119
5.2.1	Aplicação ao sistema físico	122
5.2.2	Controle PID para a frequência de amostragem reduzida	125
5.2.3	Controle PID para o sistema remodelado	128
5.3	Controle PID com filtro de Kalman	131
5.3.1	Controle com filtro de Kalman e redução da frequência de amostragem.136	
5.3.2	Controle com filtro de Kalman e redução da frequência de amostra- gem e remodelagem da planta para o sistema completo	142
5.3.3	Controle de torque	145
5.4	Conclusão do capítulo	147

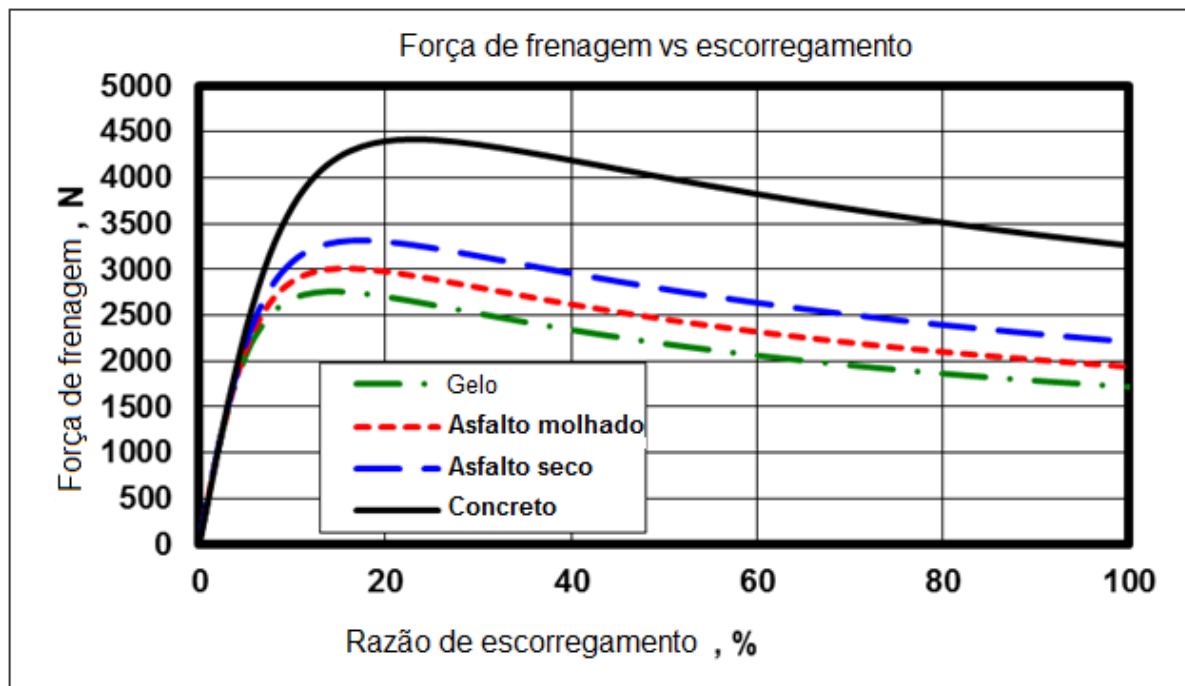
6	CONTROLE GLOBAL-LOCAL	149
6.1	Atuação do controle sob cargas variadas	152
6.2	Conclusão do capítulo	155
7	CONCLUSÕES	157
7.1	Sugestões para trabalhos futuros	158
	REFERÊNCIAS	159
	APÊNDICE A – CONTROLADOR COMPLETO	163
	ANEXO A – BEAGLEBONE BLACK DATASHEET	165
	ANEXO B – SENSOR DE EFEITO HALL	167

1 Introdução

A substituição dos veículos à combustão interna por veículos elétricos tem sido vista, cada vez mais, como uma realidade futura. Uma prova disto é que países como Noruega e França já possuem planos para proibição destes meios de transporte a partir de 2025 e 2030, respectivamente. Por este motivo a comunidade científica tem realizado muitas pesquisas para o desenvolvimento destes carros, levando em conta questões como segurança, eficiência energética, estabilidade e desempenho (YIN; HORI, 2010; MAGALLAN; ANGELO; GARCIA, 2011).

Uma das grandes dificuldades no controle de tração de um veículo é o fato do atrito, entre o pneu e a superfície que o veículo se desloca apresentar um comportamento não linear e incerto (KUNTANAPREEDA, 2014). Este atrito depende de fatores como o tipo do pneu, superfície de contato, carga sobre ele, material da rua, condições desta, escorregamento, dentre outros. Um exemplo de atrito em função do tipo de piso e razão de escorregamento pode ser visto na Figura 1.

Figura 1 – Exemplo do valor da força de atrito máxima em relação a taxa de escorregamento entre a roda e a superfície para diferentes pisos.



Fonte: adaptado de (WANG; FUJIMOTO; HARA, 2017).

Este problema também afeta robôs móveis em aplicações de menor escala, quando estes são submetidos a alto torque ou se movendo sobre superfícies escorregadias, fazendo com que o controle de velocidade se torne uma tecnologia limitada para forças de tração muito menores que o atrito.

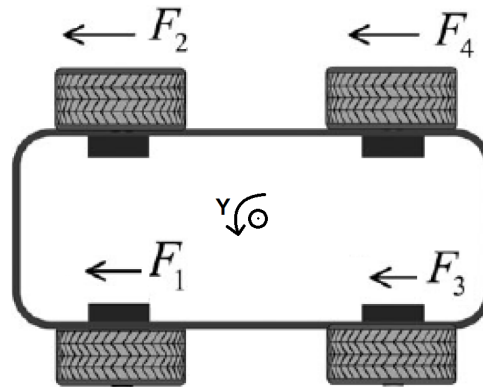
Outra solução para se aumentar a segurança em veículos elétricos é focar a distribuição

de forças em busca de uma maior estabilidade ou, ainda, ajustes melhores de controles que respondem mais rapidamente, possuem menor erro de estado estacionário, ou apresentam menor oscilação do sistema.

A arquitetura de quatro motores de tração independentes, um para cada roda do veículo, apresentada como uma possível solução para aumento da estabilidade de um veículo por (MUTOH, 2012) e (KUNTANAPREEDA, 2014) é baseada, usualmente, no controle de torque, direta ou indiretamente.

Assim, cada motor é controlado de forma independente, como em (LI et al., 2015), que propõe uma estratégia de distribuição de torque ótimo para as rodas de um veículo elétrico, considerando a força de tração e a razão do momento de guinada testando o sistema sobre uma superfície escorregadia. A guinada é o nome dado para a rotação de um objeto em relação do eixo vertical ilustrado na Figura 2.

Figura 2 – Exemplo de movimento de guinada, gerado por diferentes valores de torque aplicados em cada roda ($F_2 + F_4 > F_1 + F_3$).



Fonte: adaptado de (WANG; FUJIMOTO; HARA, 2017).

No cenário atual, este movimento ocorre quando o motorista vira o volante alterando o ângulo entre a frente do carro e as rodas dianteiras. Este movimento gera um aumento no atrito estático na direção y , porém uma diminuição no atrito na direção $y+\phi$.

Este desequilíbrio gerado faz com que as rodas internas sejam forçadas a obter uma rotação menor que as rodas externas, deste modo, por se tratar de um sistema passivo, as rodas internas perdem energia até que as velocidades estabilizem, formando um ângulo ϕ de curvatura.

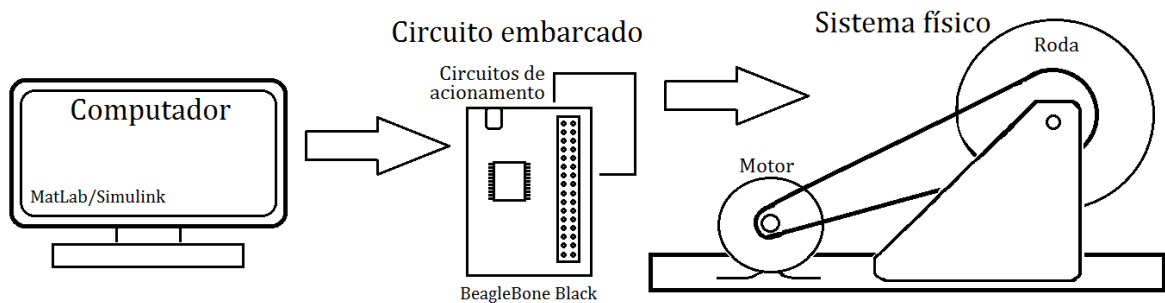
Com a implementação de motores independentes para cada roda, todo o sistema mecânico de diferencial existente nos carros atuais pode ser retirado diminuindo, assim, o peso total de veículo, o custo de construção e eliminando as limitações geradas pela implementação mecânica deste. Além disto, com a utilização deste sistema, é possível que as rodas possuam um torque em sentidos opostos, assim, o grau de liberdade do sistema cresce e a energia pode

ser distribuída de forma mais eficiente.

Este trabalho apresenta um estudo comparativo abordando diversos sistemas de controle de torque e velocidade com o objetivo de aplicação a veículos móveis com rodas independentes. Para o desenvolvimento desta pesquisa, uma roda é acoplada ao eixo de um motor por meio de uma coroa e corrente, e submetida aos controladores projetados de modo a simular acelerações, frenagens e variações na velocidade.

O sistema é composto por um processo de *hardware-in-the-loop* (HIL), ou seja, o sistema real é acionado por um sistema embarcado que comunica diretamente com o ambiente virtual de programação, conforme ilustra a Figura 3. O controle embarcado é realizado por meio de um microcontrolador *BeagleBone Black*, programado pelo *software Matlab/Simulink*, permitindo facilidade de projeto, ajuste do controlador e observação dos resultados.

Figura 3 – Esquema de funcionamento do sistema em *hardware-in-the-loop*.



Fonte: Próprio autor.

Como é possível observar no esquema de funcionamento da Figura 3 o projeto do sistema é realizado no computador por meio do ambiente de *MatLab/Simulink*. Neste, é possível realizar ajustes em tempo real e observar os resultados obtidos em forma de gráficos em tempo real. O projeto do controlador feito com a utilização dos blocos do *Simulink* é transformado em código compatível ao microcontrolador embarcado e transferido a este automaticamente.

O sistema embarcado é dedicado para execução do controlador, estimador de estados, leitura dos sensores e acionamento do motor, por meio de circuitos de tratamento de sinais, acionamento e interfaces realizando o controle da física real.

Desta maneira os resultados colhidos são efetivamente do motor e roda reais, os quais sofrem efeitos de perturbações reais, tais como atrito entre as partes móveis, oscilação da corrente, ruídos elétricos. Entretanto sem perder a velocidade de ajustes de parâmetros possibilitada pela simulação.

1.1 Objetivo geral

O objetivo deste trabalho é desenvolver um estudo comparativo entre diferentes técnicas de controle de torque e velocidade de um motor em corrente contínua acoplado a uma roda suspensa, para aplicação em um veículo elétrico com sistema de tração com rodas independentes de baixo custo e o menos invasivo possível de forma a se otimizar os sistemas de controle para tais situações.

1.2 Objetivos específicos

Os objetivos específicos são:

- Construir um sistema com uma roda suspensa de forma possibilitar a atuação do controlador no dois sentidos de rotação e também submetê-la a diferentes cargas;
- Criar um ambiente de estudo em *hardware-in-the-loop* com a utilização do *Simulink*;
- Fazer a identificação paramétrica do sistema para aplicação das técnicas de controle;
- Projetar sistema de controlador PID e filtro de Kalman para controle de velocidade e torque do motor;
- Testar as técnicas de controle implementadas no *Simulink* e avaliar seus desempenhos no âmbito de estabilidade e desempenho;
- Fazer testes físicos com o sistema para comparação de resultados.

1.3 Justificativas

Com o aumento na utilização de motores elétricos na tração de veículos o desenvolvimento de melhores sistemas de controle propiciam um aumento na segurança, eficiência energética, desempenho e custo. Portanto as técnicas aplicadas para o controle destes motores possuem um papel fundamental nestas características.

- Segurança: a melhora do sistema de controle propicia uma resposta a transição mais segura, mantendo o atrito do sistema roda-superfície mais estável. Além disto, caso haja algum distúrbio súbito sobre o carro, cada roda pode apresentar uma resposta específica e otimizada buscando a estabilidade do carro.
- Eficiência energética: uma melhora no controle permite o acionamento dos motores de maneira ótima adaptando-se rapidamente a situações de grandes perdas energia -como curvas, trabalhando integrados a um sistema de recuperação de energia diretamente roda a roda.

- Desempenho: o controle atuante pode ser projetado de modo a buscar o melhor índice de escorregamento, em busca do maior atrito, resultando na possibilidade de um torque maior de acionamento nas rodas, deste modo é possível transformar a maior potência possível gerada no motor em energia cinética para movimentação do veículo. De modo semelhante a adaptação do veículos à curvas e manobras súbitas permite este a operar em uma velocidade maior com a mesma segurança.
- Custo: um sistema ativo, tem a capacidade de se adequar de forma mais eficiente aos distúrbios gerados pelo ambiente, podem até mesmo agir previamente, como no caso do distúrbio gerado pelo motorista na entrada de uma curva. Como resultado o sistema estará mais estável resultando em menor desgaste de pneus e peças.

Este trabalho apresenta um estudo comparativo entre diferentes técnicas de controle para acionamento de rodas, sob variadas circunstâncias no intuito de apresentar os pontos positivos e negativos de cada técnica apresentada e uma análise sobre quais as melhores técnicas para cada aplicação.

1.4 Descrição da estrutura do trabalho

No Capítulo 2 apresenta-se o embasamento teórico das técnicas usadas para o desenvolvimento dos sistemas de controle, bem como filtros, interfaces de acionamento, observadores, dentre outros. O Capítulo 3 descreve em detalhes os componentes que fazem parte do sistema, bem como o funcionamento e ajustes destes, especificando o material usado, os procedimentos e técnicas adotadas. No Capítulo 4 é tratada a identificação paramétrica do sistema, que possibilita o projeto dos controladores e observadores usados. O Capítulo 5 dispõe os resultados obtidos e discussões sobre os dados levantados. O trabalho se conclui no Capítulo 7 com as considerações finais.

2 Modelagem matemática e métodos de controle

Este capítulo apresenta os conceitos e técnicas de controle usadas em motores de corrente contínua, tais como a modelagem de um motor c.c., a discretização do espaço de estados para aplicação em um sistema de controle digital. Além do projeto do estimador de estados, filtro de Kalman e projeto do controle de torque.

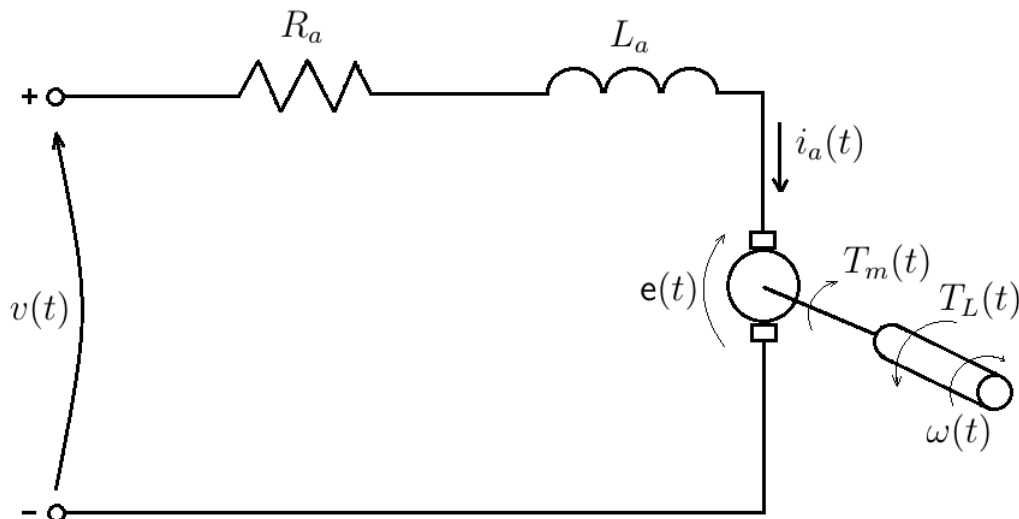
2.1 Modelo dinâmico de um motor em corrente contínua

O modelo elétrico do motor é importante para determinação da planta do sistema que deseja-se controlar, uma vez que somente a modelagem matemática possibilita a observação de estados sem a presença de sensores específicos.

Ainda que existam outras formas de se controlar um sistema sem obter o modelo matemático deste, como por exemplo, métodos heurísticos de controle, neste trabalho escolhe-se utilizar o modelo de espaço de estados do motor, pela possibilidade de se controlar estados observáveis ainda que na ausência de sensores específicos.

A Figura 4 mostra o esquema elétrico de um modelo de parâmetros de um motor c.c. linearizado onde:

Figura 4 – Modelo elétrico de um motor em corrente contínua.



Fonte: Próprio autor.

$v(t)$ – tensão de alimentação (V);

$i_a(t)$ – corrente da armadura (A);

R_a – resistência da armadura (Ω);

L_a – indutância da armadura (H);

$e(t)$ —força eletromotriz (V); $\omega(t)$ —velocidade angular (rad/s);
 $T_m(t)$ —torque eletromagnético (Nm); $T_L(t)$ —carga (Nm).

Para modelar o motor encontram-se equações diferenciais que expressam as características do motor e relacionam a tensão aplicada na entrada do mesmo com o torque e velocidade do rotor (OGATA, 2010), assim para o desenvolvimento deste equacionamento aproxima-se para um sistema linear. Desta forma, considerando a malha apresentada na Figura 4 obtêm-se a equação 2.1.

$$v(t) = R_a i_a(t) + L_a i'_a(t) + e(t) \quad (2.1)$$

Onde

$$e(t) = K_e \omega(t) \quad (2.2)$$

e que a constante de força eletromotriz K_e dada em (V/rad.s⁻¹).

Aplicando-se a segunda lei de Newton no movimento rotacional do eixo do motor chega-se em 2.3.

$$J_m \omega'(t) = T_m(t) - T_L(t) - B_m \omega(t) \quad (2.3)$$

Onde J_m é o momento de inércia da rotação do motor (Kg.m²) e B_m é o coeficiente de viscosidade (Kg.m²/s) e

$$T_m(t) = K_t i_a(t) \quad (2.4)$$

no qual a constante de torque eletromagnético K_t é dada em (Nm/A). Substituindo 2.2 e 2.4 em 2.1, tem-se 2.5

$$v(t) = R_a \frac{T_m(t)}{K_t} + L_a \frac{T'_m(t)}{K_t} + K_e \omega(t). \quad (2.5)$$

Considerando $\omega(t)$ e $T_m(t)$ estados do sistema, é possível obter as matrizes do espaço de estados do motor c.c..

É importante ressaltar que o modelo linear apresenta uma robustez suficiente quando se trabalha com um sistema mecânico que gira apenas para uma única direção, portanto para este caso os efeitos não lineares dominantes como *dead zone* (zona morta) e fricção de Coulomb podem ser aproximados para uma dinâmica linear (KARA; EKER, 2004).

2.2 Espaço de estados de um motor em corrente contínua

Para descrição do sistema em espaço de estados busca-se uma equação que descreve a saída do sistema e os seus próximos estados, baseado nos valores atuais de entrada e estados atuais, como a equação de Euler apresentadas nas equações 2.6 e 2.7.

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \quad (2.6)$$

$$y = \mathbf{C}\mathbf{x} + Du \quad (2.7)$$

Sabendo-se que $\omega(t)$ (velocidade de rotação do eixo do motor) e $T_m(t)$ (torque eletromagnético gerado) são os estados do sistema, que $v(t)$ (tensão aplicada na entrada do motor) é a única entrada controlável do sistema, que $T_L(t)$ (torque aplicado na carga) é considerada nula para o sistema em questão, as equações 2.3 e 2.5 podem ser reescritas conforme as equações 2.8 e 2.9.

$$\omega'(t) = -\frac{B_m}{J_m}\omega(t) + \frac{1}{J_m}T_m(t) \quad (2.8)$$

$$T_m'(t) = -\frac{K_t K_e}{L_a}\omega(t) - \frac{R_a}{L_a}T_m(t) + \frac{K_t}{L_a}v(t) \quad (2.9)$$

Como as equações resultantes são lineares não é necessário que o processo seja linearizado, portanto, reescrevendo as equações 2.8 e 2.9 como 2.6 e 2.7 resultam-se nas equações de estados 2.10 e 2.11,

$$\begin{bmatrix} \omega'(t) \\ T_m'(t) \end{bmatrix} = \begin{bmatrix} -B_m/J_m & 1/J_m \\ -K_e K_t/L_a & -R_a/L_a \end{bmatrix} \begin{bmatrix} \omega(t) \\ T_m(t) \end{bmatrix} + \begin{bmatrix} 0 \\ K_t/L_a \end{bmatrix} \begin{bmatrix} v(t) \end{bmatrix} \quad (2.10)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \omega(t) \\ T_m(t) \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} \begin{bmatrix} v(t) \end{bmatrix}. \quad (2.11)$$

No qual a matriz $\mathbf{C}$ é dada pelo vetor $[1 \ 0]$ uma vez que o sistema é composto apenas por um sensor de velocidade e $\mathbf{D}$ é dada por $[0]$, pois não há ação direta da entrada do sistema sobre a saída do mesmo.

Apesar do espaço de estados obtido descrever as entradas e saídas do motor para uma aplicação contínua no tempo, estas matrizes não podem ser aplicadas em um sistema digital discreto. Consequentemente para a implementação de um sistema digital de controle é necessário obter o modelo discreto do espaço de estados descrito pelas equações 2.10 e 2.11 com um período de amostragem T .

2.3 Discretização do espaço de estados

Para a discretização parte-se do espaço de estados contínuo no tempo, descrito nas equações 2.6 e 2.7, porém considera-se um vetor de entrada que se altera apenas em um período regular de valor T . Logo o novo sistema é descrito segundo as equações 2.12 e 2.13.

$$\mathbf{x}[k+1] = \Phi(T)\mathbf{x}[k] + \Gamma(T)u[k] \quad (2.12)$$

$$y[k] = \mathbf{C}\mathbf{x}[k] + Du[k] \quad (2.13)$$

Onde, considerando a entrada do sistema constante durante o período de amostragem pode-se usar o método de aproximação por integração conforme mostram as equações 2.14 e 2.15.

$$\mathbf{x}((k+1)T) = e^{\mathbf{A}(k+1)T}\mathbf{x}[0] + e^{\mathbf{A}[k+1]T} \int_0^{(k+1)t} e^{-\mathbf{A}t}\mathbf{B}u(t)dt \quad (2.14)$$

$$\mathbf{x}((k)T) = e^{\mathbf{A}(k)T}\mathbf{x}[0] + e^{\mathbf{A}[k]T} \int_0^{(k)t} e^{-\mathbf{A}t}\mathbf{B}u(t)dt \quad (2.15)$$

Multiplicando todos os termos da equação 2.15 por $e^{\mathbf{A}T}$, obtém-se

$$e^{\mathbf{A}T}\mathbf{x}((k)T) = e^{\mathbf{A}(k+1)T}\mathbf{x}[0] + e^{\mathbf{A}[k+1]T} \int_0^{(k)t} e^{-\mathbf{A}t}\mathbf{B}u(t)dt \quad (2.16)$$

reescrevendo a equação 2.16 em função de $e^{\mathbf{A}(k+1)T}\mathbf{x}[0]$

$$e^{\mathbf{A}(k+1)T}\mathbf{x}[0] = e^{\mathbf{A}T}\mathbf{x}((k)T) - e^{\mathbf{A}[k+1]T} \int_0^{(k)t} e^{-\mathbf{A}t}\mathbf{B}u(t)dt \quad (2.17)$$

substituindo então a equação 2.17 na 2.14

$$\mathbf{x}((k+1)T) = e^{\mathbf{A}T}\mathbf{x}((k)T) + e^{\mathbf{A}[k+1]T} \left[\int_0^{(k+1)t} e^{-\mathbf{A}t}\mathbf{B}u(t)dt - \int_0^{kt} e^{-\mathbf{A}t}\mathbf{B}u(t)dt \right] \quad (2.18)$$

ou seja,

$$\mathbf{x}((k+1)T) = e^{\mathbf{A}T} \mathbf{x}(kT) + e^{\mathbf{A}[k+1]T} \left[\int_{kT}^{(k+1)T} e^{-\mathbf{A}t} \mathbf{B}u(t) dt \right]. \quad (2.19)$$

Logo, comparando a equação 2.19 com 2.12, obtém-se 2.20

$$\Phi(T) = e^{\mathbf{A}T}; \quad \Gamma(T) = \int_0^T e^{\mathbf{A}t} \mathbf{B} dt. \quad (2.20)$$

Portanto substituindo a equação 2.20 em 2.12 e mantendo-se 2.13, conhecendo-se, ainda, os valores das matrizes A, B, C e D apresentados nas equações 2.10 e 2.11 tem-se o espaço de estados correspondente ao motor c.c. para um sistema discreto, no qual T é o período de amostragem.

2.4 Tração de veículos elétricos

Motores elétricos podem possuir uma resposta muito rápida, um torque exato e capacidade de tração independente para cada roda de um veículo. Por este motivo, diversas pesquisas têm sido desenvolvidas com o intuito de obter melhores repostas para utilização destes motores em veículos elétricos. O uso desta tecnologia possibilita o desenvolvimento de freios ABS (*antiblock braking system*), sistema de controle de guinada, controle de aceleração, dentre outros.

No entanto, segundo (YIN; OH; HORI, 2009) para que estes sistemas de controle possuam uma operação eficiente é necessário que o controlador seja capaz de prover uma saída de torque exata. Assim, o controle de torque é parte primária para controle de veículos elétricos, podendo este operar como controle de tração e também ABS, uma vez que estes podem gerar aceleração e desaceleração com igual facilidade (MUTOH, 2012).

Devido entretanto a não linearidade e de difícil previsão do atrito máximo apresentado pelo conjunto roda-superfície, (MAGALLAN; ANGELO; GARCIA, 2011) utilizou um controle de tração baseado em um observador de escorregamento para estimar essa fricção máxima e, conseqüentemente, o torque máximo permitido. Controladores de modo de escorregamento tem sido muito utilizados para controle de sistemas não lineares e incertos, devido a sua robustez (KUNTANAPREEDA, 2014). O qual apresenta uma técnica baseada no controle de modo deslizante para obtenção de um fator de escorregamento desejado.

Além do controle de escorregamento em (HEIKKINEN; SERYKH; MINAV, 2017) é sugerida a utilização de controle fuzzy PID e em (MAUER, 1995) é proposto um controlador fuzzy para identificar as condições da pista, baseado no escorregamento anterior. Em (WILL; HUI; ZAK, 1998) foi desenvolvido um sistema híbrido de controle não linear que combina o observador em modo deslizante com um PID.

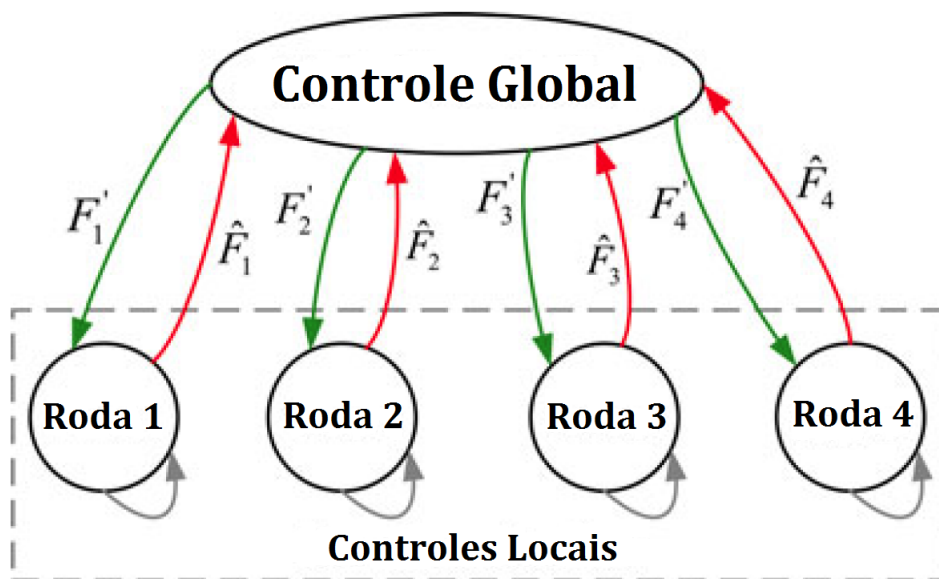
2.4.1 Arquitetura de controle global-local

Como afirma (WANG; FUJIMOTO; HARA, 2017), os coeficientes de atrito adotados pelas técnicas de controle por otimização da distribuição de torque são iguais para todas as rodas, este efeito faz com que as rodas que passem sobre uma parte da superfície com menor atrito tenham um escorregamento muito maior que desejado (esperado pelo controlador), tornando o sistema como um todo passível de apresentar algum grau de instabilidade não desejada em condições como esta.

Como sugestão é apresentada uma proposta de se equalizar o escorregamento em cada roda, desta forma, independente da força atrito atual sobre o pneu os quatro pneus continuarão estáveis em relação ao sistema.

Para execução deste sistema complexo é apresentado o esquema de controle global-local, como pode ser visto na Figura 5.

Figura 5 – Esquema de controle global-local para veículo elétrico movido a quatro motores independentes em cada roda.



Fonte: (Wang, 2017).

O controlador global (Figura 5) recebe dos controladores locais as forças de tração atuais e então por meio de alguma técnica de controle, como por exemplo a de distribuição de torque, calcula os valores das forças necessárias de saída e, então, retorna aos controladores locais a referência, ou seja, a força de tração desejada para estabilização do veículo.

Enquanto isso, cada controlador local, de modo independente controla a força de tração, por alguma técnica como, por exemplo, controle de modo deslizante, fazendo com que a força

desejada seja alcançada mesmo que diferentes coeficientes de atrito e condições atuem em cada roda. Esta técnica de controle demonstra um potencial promissor para resolução dos problemas de estabilidade e aceleração em veículos móveis com motores independentes.

Este trabalho pretende atuar no desenvolvimento de um controle local de torque baseado em um controle PID e observador de torque com filtro de Kalman, de forma a obter uma exatidão do torque desejado que satisfaça os requerimentos exigidos pelo projeto.

2.5 Identificação paramétrica

A identificação de sistema consiste na determinação de um modelo matemático de um processo, baseado em medidas das entradas e saídas deste (LJUNG, 2010). Este modelamento pode ser feito mediante ajuste de parâmetros ou do modelo, de forma que a saída do sistema seja o mais semelhante possível das amostras das medidas das saídas.

2.5.1 Determinação dos parâmetros do motor

Um das formas mais clássicas de se obter o modelo matemático de um sistema se dá pela obtenção do modelo dinâmico aproximado deste, seguido da estimação dos parâmetros que compõe este sistema. Como definido na Seção 2.1 o modelo matemático do motor utilizado é apresentado pelo esquema elétrico da Figura 4.

Desta forma, os parâmetros que devem ser obtidos para que o processo possa ser controlado são R_a , L_a , K_e , K_t , J_m e B_m . Os primeiros parâmetros a serem determinados são R_a e L_a , os quais são constantes características intrínsecas ao motor e, portanto, podem ser fornecidas nas descrições técnicas dadas pelo próprio fabricante ou medidas com um equipamento próprio em bancada.

Os parâmetros K_e , K_t , por sua vez, ainda que sejam intrínsecos ao motor, estes não são lineares, principalmente para valores de velocidade de rotação do eixo do motor próximos de zero. Para o desenvolvimento deste trabalho, considerou-se que os motores passarão a maior parte do tempo em rotação unidirecional, sem cruzamento pelo zero e, portanto, poderiam ser aproximados para valores constantes. Assim, para o caso destas constantes (K_e e K_t) não serem fornecidas pelo fabricante, é possível obtê-las com um experimento de bancada.

A constante K_e pode ser obtida da seguinte forma:

- determina-se os valores de R_a e L_a ;
- mantém-se uma tensão constante à entrada do motor com o eixo livre até que a velocidade de rotação estabilize;
- medem-se os respectivos valores de corrente e velocidade de rotação do eixo do motor;

- calcula-se o valor das constante conforme a equação 2.21, obtida pela substituição da equação 2.2 em 2.1,

$$K_e = \frac{V - RI}{\omega}. \quad (2.21)$$

Já a constante K_t pode ser obtida medindo-se a corrente de alimentação do motor constante e, com o auxílio de um dinamômetro, medir o torque gerado no eixo do motor. Em seguida substituindo os valores obtidos na equação 2.22

$$K_t = \frac{T_m}{I}, \quad (2.22)$$

entretanto muitos autores consideram K_e e K_t iguais, simplificando a obtenção do modelo.

Os parâmetros J_m e B_m são constantes que não dizem respeito necessariamente ao motor, mas ao sistema a que o motor atua. Para levantamento destas constantes é necessário não mais que o motor possua o eixo livre, mas que o sistema a ser controlado esteja completamente montado.

Para a obtenção do parâmetro B_m , pode-se usar a equação 2.3 ou substituir a equação 2.4 em 2.3, deste modo, mantendo novamente a tensão de entrada constante e medindo-se o valor da velocidade de rotação do motor e o valor da sua corrente de armadura, além de já conhecer o valor da constante K_t determina-se

$$B_m = \frac{K_t I}{\omega}. \quad (2.23)$$

A constante J_m atua sobre a aceleração do eixo do motor e, portanto, é mais difícil de ser obtida experimentalmente, no entanto este valor pode ser obtido matematicamente, por meio de equações clássicas do momento de inércia de um corpo, ou por meio de um sistema recursivo.

No caso de veículos elétricos, é importante ressaltar que o modelo pode se tornar muito mais complexo, devido a alterações no torque de carga (T_L) e mesmo no momento de inércia do sistema (J_L) causado por variações na distribuição de peso sobre as rodas, alterações do atrito entre o piso e a roda, dentre outros.

2.5.2 Método determinístico de identificação paramétrica

Além do modelo matemático de descrição do sistema há diversos métodos de se estimar uma planta baseado em suas entradas e saída. Os métodos clássicos são determinísticos e baseiam-se na identificação paramétrica a partir dos comportamentos de regime transitório e permanente

e são métodos normalmente baseados com análise gráfica como é o caso de (ZIEGLER; NICHOLS, 1942), (SUNDARESAN; PRASAD; KRISHNASWAMY, 1978), Smith apontado por (COELHO; COELHO, 2004) e (HÄGGLUND; ÅSTRÖM, 1991).

Há ainda métodos de determinação de processos por meio da resposta em frequência, por meio do diagrama de Bode, no entanto esta técnica exige que as entradas sejam senoidais em diferentes frequências e que a medição da magnitude resultante e o deslocamento de fase seja exata, além da necessidade de que o sistema seja estável.

Existem diversos métodos determinísticos com desempenhos semelhantes, os quais tradicionalmente são baseados na resposta que determinado sistema possui quanto à entrada de grau, desta forma, com base em pontos específicos de do gráfico obtidos faz-se a identificação paramétrica do sistema.

Para modelos de primeira ordem busca-se uma função de transferência segundo o modelo apresentado pela equação 2.24.

$$G_p(s) = \frac{K_p}{\tau s + 1} e^{-\theta s} \quad (2.24)$$

O primeiro método usado neste trabalho foi apresentado por Ziegler/Nichols e utiliza-se de uma reta de auxílio traçada para determinação do transiente, na qual o cruzamento desta com o eixo das abscissas representa o valor do θ (tempo de atraso) e o cruzamento da mesma com a ordenada de valor igual ao tempo de estabilização do sistema permite a determinação do valor da constante τ (tempo de acomodação menos o tempo de atraso).

$$\tau = t_s - t_d \quad (2.25)$$

O método de Hägglund difere-se do método de Ziegler/Nichols apenas pela substituição do cálculo do valor da constante τ a qual agora recebe o valor da diferença entre o tempo de 63,2% do valor de acomodação menos o tempo de atraso.

$$\tau = t_{0,632K_p} - t_d \quad (2.26)$$

Para o método de Smith entretanto, substitui-se a reta de auxílio para utilização de pontos notáveis do gráfico de resposta do sistema, conforme mostram as equações 2.27 e 2.28.

$$\tau = 1,5(t_{0,632K_p} - t_{0,283K_p}) \quad (2.27)$$

$$\theta = t_{0,632K_p} - \tau \quad (2.28)$$

Semelhanamente o método apresentado por Sundaresan/Krishnaswamy utiliza pontos de referência como os tempo de 85,3% e 35,3% do valor de estabilização, conforme mostram as equações 2.29 e 2.30.

$$\tau = 0,67(t_{0,853K_p} - t_{0,353K_p}) \quad (2.29)$$

$$\theta = 1,3t_{0,353K_p} - 0,29t_{0,853K_p} \quad (2.30)$$

Ainda utilizam-se neste trabalho os métodos de segunda ordem, sendo o primeiro deles de índices de desempenho, no qual, por meio da observação destes índices, obtidos pela análise do gráfico de resposta do sistema à uma entrada degrau, faz-se uma dedução comparativa, uma vez que, conhecida a função de segunda ordem de um sistema genérico, descrita por

$$G_p(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (2.31)$$

Deste modo, o valor de ω_n e ξ podem ser determinados segundo as equações de 2.32 a 2.34.

$$y_{pico} = 1 + e^{\frac{-\xi\pi}{\sqrt{1-\xi^2}}} \quad (2.32)$$

$$t_p = \frac{\pi}{\omega_n \sqrt{1-\xi^2}} \quad (2.33)$$

$$t_s = \frac{4}{\xi\omega_n} \quad (2.34)$$

O método de Mollenkamp é determinado com base em três instantes de tempo, sendo estes $t_1 = t_{0,15K_p}$, $t_2 = t_{0,45K_p}$ e $t_3 = t_{0,75K_p}$, com base nas equações de 2.35 a 2.41.

$$x = \frac{t_2 - t_1}{t_3 - t_1} \quad (2.35)$$

$$\xi = \frac{0,0805 - 5,547(0,475 - x)^2}{(x - 0,356)} \quad (2.36)$$

$$\begin{aligned} f_2(\xi) &= 0,708(2,811)^\xi, & \xi < 1 \\ f_2(\xi) &= 2,6\xi - 0,60, & \xi \geq 1 \end{aligned} \quad (2.37)$$

$$\omega_n = \frac{f_2(\xi)}{t_3 - t_1} \quad (2.38)$$

$$f_3(\xi) = 0,922(1,66)^\xi \quad (2.39)$$

$$\theta = t_2 - \frac{f_3(\xi)}{\omega_n} \quad (2.40)$$

$$\tau_{1,2} = \frac{\xi \pm \sqrt{\xi^2 - 1}}{\omega_n} \quad (2.41)$$

Além dos diversos métodos determinísticos de identificação paramétrica apresentados nesta seção existem os métodos heurísticos, que permitem o desenvolvimento de softwares para autoajuste e portanto proporcionando capacidade de identificação em tempo real.

2.5.3 Método heurístico de identificação paramétrica

Como possibilidades encontradas no contexto atual existem métodos heurísticos de identificação paramétrica de um sistema, como é o caso da aplicação do método ARMA (*autoregressive moving average*), redes neurais (NARENDRA; FITCH, 1981; NELLES, 2013; PHAM; LIU, 1995), algoritmos genéticos (KRISTINSSON; DUMONT, 1992; PEREIRA; PINTO, 2005), dentre outros tipos de aprendizagem de máquina.

Entretanto escolheu-se, o uso de técnicas baseadas em aproximação por mínimos quadrados, uma vez que teoricamente, a utilização destes permite uma melhor aproximação que métodos gráficos, uma vez que não é necessário análises gráficas, maior facilidade de aplicação que por resposta em frequência. Além de possibilidade de utilização recursiva para um possível aplicação em controle adaptativo.

O controlador adaptativo do tipo auto ajustável é composto por uma malha de ajuste de parâmetros, de forma a compensar variações ocorridas nos parâmetros do sistema ou no ambiente, o que é uma vantagem que este tipo de controle apresenta sobre outros. Neste caso, as variações na dinâmica do sistema são acompanhadas pelo identificador adaptando o controle a cada período de amostragem (COELHO; COELHO, 2004).

O método heurístico de identificação utilizado neste trabalho baseia-se em médias móveis autorregressivas chamado de ARMA. O modelo ARMA portanto é composto por uma parte de modelo autorregressivo (AR) e de médias móveis (MA).

Suponha que se queira encontrar um sistema linear

$$\mathbf{y} = \mathbf{k}\mathbf{x} + \epsilon \quad (2.42)$$

onde $\mathbf{x}$ é o vetor composto pelos estados do sistema (entradas), $\mathbf{y}$ os estados resultantes (saídas), $\mathbf{k}$ um vetor de constantes ótimas para a identificação do sistema linear e ϵ seja o erro de aproximação. Esta parte compõe o modelo autorregressivo do sistema.

Considerando, entretanto que o processo seja realizado com uma única média móvel, pode-se reescrever a equação 2.42 tal que

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_N \end{bmatrix}, \mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}, \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_N \end{bmatrix}, \quad (2.43)$$

no qual N é o número de termos da média móvel.

A somatória do erro quadrático, por sua vez é dada por

$$\boldsymbol{\epsilon}^T \boldsymbol{\epsilon} = \sum_{i=1}^N \epsilon_i^2. \quad (2.44)$$

De modo semelhante utilizando a equação 2.42, tem-se que

$$\boldsymbol{\epsilon}^T \boldsymbol{\epsilon} = [\mathbf{y} - \mathbf{k}\mathbf{x}]^T [\mathbf{y} - \mathbf{k}\mathbf{x}], \quad (2.45)$$

logo

$$\frac{d(\boldsymbol{\epsilon}^T \boldsymbol{\epsilon})}{dk} = -2\mathbf{x}^T \mathbf{y} + 2\mathbf{k}\mathbf{x}^T \mathbf{x} = 0, \quad (2.46)$$

assim

$$\mathbf{k} = \frac{\mathbf{x}^T \mathbf{y}}{\mathbf{x}^T \mathbf{x}}. \quad (2.47)$$

Considerando portanto a equação de estados apresentada em 2.12 e o erro de identificação apresentado na equação 2.42, têm-se

$$\begin{aligned} \mathbf{x}[1] &= \boldsymbol{\Phi}(T)\mathbf{x}[0] + \boldsymbol{\Gamma}(T)u[0] + \epsilon[1] \\ \mathbf{x}[2] &= \boldsymbol{\Phi}(T)\mathbf{x}[1] + \boldsymbol{\Gamma}(T)u[1] + \epsilon[2] \\ &\vdots \\ \mathbf{x}[N] &= \boldsymbol{\Phi}(T)\mathbf{x}[N-1] + \boldsymbol{\Gamma}(T)u[N-1] + \epsilon[N] \end{aligned}, \quad (2.48)$$

logo, pode-se reescrever a equação 2.48 como

$$\begin{bmatrix} \mathbf{x}(1) \\ \mathbf{x}(2) \\ \vdots \\ \mathbf{x}(N) \end{bmatrix} = \begin{bmatrix} \mathbf{x}(0) & u(0) \\ \mathbf{x}(1) & u(1) \\ \vdots & \vdots \\ \mathbf{x}(N-1) & u(N-1) \end{bmatrix} \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{b}_1 \end{bmatrix} + \begin{bmatrix} \epsilon(1) \\ \epsilon(2) \\ \vdots \\ \epsilon(N) \end{bmatrix}. \quad (2.49)$$

Ou seja,

$$\mathbf{x}(N) = \mathbf{F}(N)\psi + \epsilon(N). \quad (2.50)$$

Obtendo-se a equação 2.50 e comparando com a equação 2.42, pode-se dizer que, conforme a equação 2.47

$$\psi = [\mathbf{F}^T(N)\mathbf{F}(N)]^{-1} \mathbf{F}^T(N)\mathbf{x}(N). \quad (2.51)$$

É importante ressaltar que o método de identificação por mínimos quadrados retorna a melhor estimativa baseado nos dados recebidos. No entanto, isto não significa que o resultado é exato e útil. Por exemplo, caso um sistema linear de segundo grau seja identificado como de primeiro grau, este melhor resultado ainda será errado, portanto é necessário uma verificação dos resultados obtidos antes da aplicação final (PHILLIPS; NAGLE, 2007).

2.6 Projeto do controlador proporcional integral derivativo

Primeiramente para o controle de um sistema deve-se determinar se o mesmo é controlável. A controlabilidade de um sistema pode ser determinada por meio a matriz

$$\begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \quad (2.52)$$

de dimensão igual do tamanho do vetor $\mathbf{u}$. Caso a matriz controlabilidade seja linearmente independente, então o sistema é completamente controlável.

Assim, como apresentado na Seção 2.3 há uma dualidade entre o sistema contínuo no tempo e o discreto, esta dualidade pode ser obtida por meio da transformada z logo, é possível que o projeto dos controles sejam todos realizados no tempo contínuo e posteriormente transformados para o tempo discreto, dependendo do período de amostragem do sinal. Em (DORF; BISHOP, 2011) define-se portanto a transformada z de uma função $f(t)$ como

$$Z\{f(t)\} = \sum_{k=0}^{\infty} f(kT)z^{-k} \quad (2.53)$$

Como é possível observar na equação 2.53 o período de amostragem é muito importante para o projeto do controle, pois a alteração neste faz com que o controlador projetado não tenha o mesmo efeito sobre o sistema.

Portanto duas são as vantagens do projeto em tempo contínuo e justificativas para utilização do mesmo neste trabalho. Primeiramente é a facilidade da aplicação, uma vez que os métodos são tradicionalmente projetados para o tempo contínuo, a segunda e talvez mais importante, é a facilidade de se reprojeter o sistema caso haja uma alteração no período de amostragem.

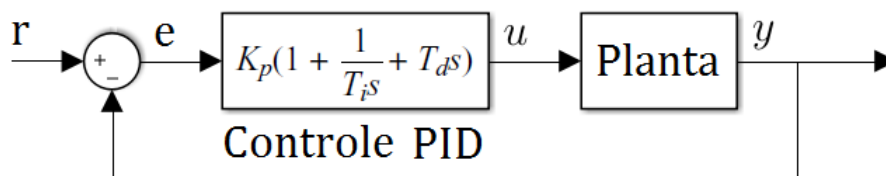
Diversas são as técnicas de controle em malha fechada sobre um sistema, dentre as quais pode-se citar compensadores em atraso e avanço de fase, controles proporcionais, entretanto uma técnica que tem se mostrado bastante robusta para diversas aplicações é o controle PID (proporcional integral derivativo), sendo esta reponsável por 95% dos controladores nas indústrias (LE; NGUYEN; NGUYEN, 2018).

Uma das grandes vantagens deste projeto é a possibilidade de ajuste no ambiente de utilização, ainda, por causa da vasta aplicação do mesmo diversas regras de ajuste desta estratégia de controle foram desenvolvidas na literatura. Um bom ajuste do controlador PID possibilita também a rejeição de distúrbios em um sistema incerto, como proposto por (CHU; CHU, 2018).

2.6.1 Método de Ziegler-Nichols

Os métodos de Ziegler-Nichols apresentam uma aplicabilidade em casos no qual o modelo matemático da planta é complicado e não pode ser obtido facilmente e portanto o ajuste do controlador PID deve ser feito experimentalmente (OGATA, 2010). A Figura 6 apresenta um diagrama de blocos do modelo de controlador PID em malha fechada.

Figura 6 – Diagrama de blocos do controlador PID em malha fechada atuante sobre uma planta qualquer.

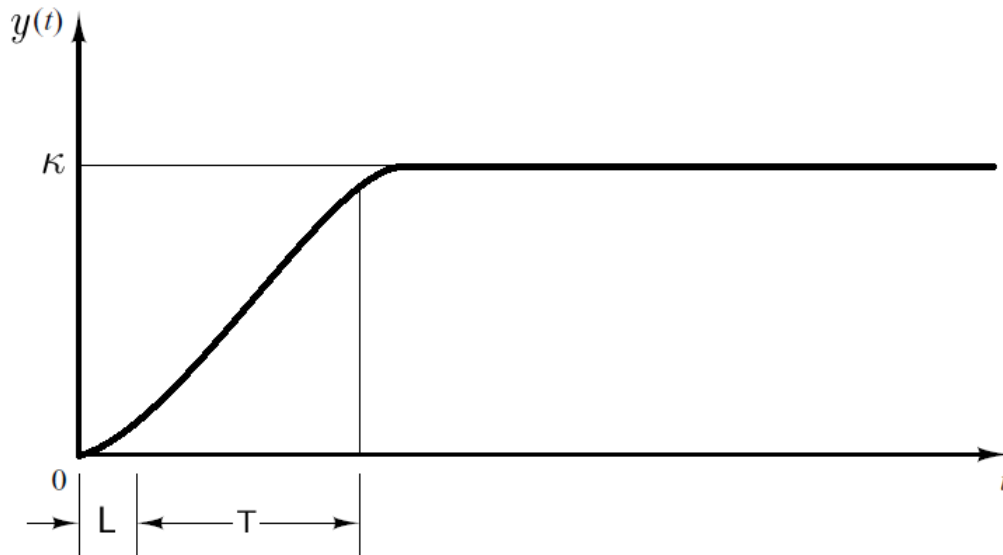


Fonte: Próprio autor.

Como pode ser observado, a entrada do sistema neste caso de malha fechada é a referência (r) a qual deseja-se obter à saída do sistema, esta saída portanto é subtraída do estado atual de saída do sistema (y), resultando em um erro (e). Este erro serve de entrada para o controlador PID que gera um sinal de controle (u) para a planta de modo a aproximar a saída do sistema para o valor de referência.

O primeiro método consiste na obtenção da resposta ao degrau unitário na planta. Caso a saída do sistema apresente uma curva com formato em S, como mostra a Figura 7, significa que a técnica pode ser utilizada para ajuste do PID.

Figura 7 – Exemplo de resposta à entrada degrau para aplicação do primeiro método de Ziegler-Nichols.



Fonte: Próprio autor.

Após a obtenção dos valores de L (tempo de atraso) e T (constante de tempo), podem-se calcular os parâmetros do controlador PID, conforme mostra a Tabela 1. É importante frisar que este método apenas tem aplicabilidade para sistemas que possuem um formato curvo na região de transição (OGATA, 2011), ou seja, sistemas que possuem um tempo de atraso grande.

Tabela 1 – Valores de K_p , T_I e T_d para controladores P, PI e PID, conforme a primeira regra de Ziegler-Nichols, baseado na resposta ao degrau.

Tipo de Controlador	K_P	T_I	T_d
P	$\frac{T}{L}$	∞	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2 L$	$0,5 L$

Fonte: Próprio autor.

O segundo método de Ziegler-Nichols, consiste em calibrar $T_I = \infty$ e $T_D = 0$ e au-

mentar o valor de K_P até atingir um valor crítico de estabilidade que constitui uma oscilação permanente do sistema $K_P = K_{cr}$, os valores dos parâmetros de ajuste do controlador para este caso são dados pela Tabela 2, onde P_{cr} é o período de oscilação do sistema.

Tabela 2 – Valores de K_p , T_I e T_d para controladores P, PI e PID, conforme a segunda regra de Ziegler-Nichols, baseado na constante crítica do controle proporcional.

Tipo de Controlador	K_P	T_I	T_d
P	$\frac{1}{2} K_{cr}$	∞	0
PI	$0.45 K_{cr}$	$\frac{1}{1.2} P_{cr}$	0
PID	$0.6 K_{cr}$	$\frac{1}{2} P_{cr}$	$\frac{1}{8} P_{cr}$

Fonte: Próprio autor.

Entretanto como esta técnica de ajuste dos parâmetros necessita levar o sistema ao limiar crítico de estabilidade, decidiu-se que, pela utilização da mesma é potencialmente prejudicial ao sistema e portanto não é usado neste trabalho.

2.6.2 Alocação de polos

Uma terceira técnica de ajuste do controlador PID consiste no método de alocação de polos, este método possui a capacidade de determinação do posicionamento dos polos conforme o interesse do projetista e, portanto, permite a determinação do desempenho do sistema.

A determinação de uma matriz de ganho de modo a se obter a alocação dos polos desejados pode ser feita por meio da fórmula de Ackermann, para tal modo, considere

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \quad (2.54)$$

o controle de realimentação é dado por

$$u = -K\mathbf{x} \quad (2.55)$$

assumindo que o sistema seja completamente controlável e os polos desejados sejam $\mathbf{s} = [\mu_1 \mu_2 \cdots \mu_n]$. Pode-se reescrever a equação 2.54 como

$$\dot{\mathbf{x}} = -(\mathbf{A} - \mathbf{B}K)\mathbf{x}. \quad (2.56)$$

Definindo portanto

$$\tilde{\mathbf{A}} = \mathbf{A} - \mathbf{B}K \quad (2.57)$$

a equação característica desejada é da forma

$$|s\mathbf{I} - \tilde{\mathbf{A}}| = (s - \mu_1)(s - \mu_2)\dots(s - \mu_n). \quad (2.58)$$

Pelo teorema de Cayley-Hamilton $\tilde{\mathbf{A}}$ satisfaz a sua própria equação característica (OGATA, 2010), logo

$$\Phi(\tilde{\mathbf{A}}) = \tilde{\mathbf{A}}^n + \alpha_1\tilde{\mathbf{A}}^{n-1} + \dots + \alpha_{n-1}\tilde{\mathbf{A}} + \alpha_n\mathbf{I} = 0 \quad (2.59)$$

Substituindo a equação 2.57 em 2.59 obtém-se

$$\Phi(\mathbf{A}) = \begin{bmatrix} \mathbf{B} & \mathbf{A}\mathbf{B} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \begin{bmatrix} \alpha_n K + \dots \alpha_1 K \tilde{\mathbf{A}}^{n-1} + K \tilde{\mathbf{A}}^n \\ \vdots \\ \alpha_1 K + K \tilde{\mathbf{A}} \\ K \end{bmatrix}. \quad (2.60)$$

uma vez que

$$K = \begin{bmatrix} 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} \alpha_n K + \dots \alpha_1 K \tilde{\mathbf{A}}^{n-1} + K \tilde{\mathbf{A}}^n \\ \vdots \\ \alpha_1 K + K \tilde{\mathbf{A}} \\ K \end{bmatrix}, \quad (2.61)$$

então, reescrevendo a equação 2.60 tem-se que

$$K = \begin{bmatrix} 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} \mathbf{B} & \mathbf{A}\mathbf{B} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \Phi(\mathbf{A}). \quad (2.62)$$

Partindo da técnica de alocação de polos para cálculo dos parâmetros do controlador, (GUO; MA; QIAO, 2006) propõe um método de obtenção dos parâmetros de um PID por alocação de polos com a aplicação do método independente para os três estados (proporcional, integral e derivativo). Desta forma ao menos um dos três deve conter os polos desejados.

De modo semelhante, (SUJITJORN; WIBOONJAROEN, 2011), baseado em (GUO; MA; QIAO, 2006) indica o seguinte processo para projeto de um PID em sistemas de múltiplas entradas:

- sabendo que

$$\mathbf{e}_n = \begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix}^T \quad (2.63)$$

e

$$\mathbf{w}_c = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \cdots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix}, \quad (2.64)$$

calcula-se a matriz de transformação de ordem igual a planta do sistema usando $\mathbf{T} = [q_1 q_1 \mathbf{A} \cdots q_1 \mathbf{A}^{n-1}]^T$, onde $q_1 = \mathbf{e}_n^T \mathbf{w}_c^{-1}$;

- calculam-se as matrizes $\mathbf{A}_c$ e $\mathbf{B}_c$ usando $\mathbf{A}_c = \mathbf{TAT}^{-1}$ e $\mathbf{B}_c = \mathbf{TB}$ da forma de Frobenius

$$\xi' = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -\alpha_0 & -\alpha_1 & \cdots & \cdots & -\alpha_{n-1} \end{bmatrix} \xi + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} u; \quad (2.65)$$

- alocam-se os polos de ordem n, adicionando um polo negativo com uma constante de tempo rápida (grande magnitude);
- determina-se o polinômio característico $\Delta_d(s)$ de ordem n ou n+1 correspondendo ao passo 3;
- calcula-se o ganho das matrizes para o PID usando as equações 2.66 ou 2.67:

- para $n = 2$,

$$\begin{aligned} \mathbf{K}_p &= \begin{bmatrix} \alpha_0 & \alpha_1 \end{bmatrix} \mathbf{T}, \\ \mathbf{K}_I &= \begin{bmatrix} -\alpha_0 & -\alpha_1 \end{bmatrix} \mathbf{T}, \\ \mathbf{K}_d &= \begin{bmatrix} -\alpha_2 & 0 \end{bmatrix} \mathbf{T}; \end{aligned} \quad (2.66)$$

– para $n \geq 3$,

$$\begin{aligned} \mathbf{K}_p &= \begin{bmatrix} \alpha_0 & \alpha_1 & \cdots & \alpha_{n-1} \end{bmatrix} \mathbf{T}, \\ \mathbf{K}_I &= \begin{bmatrix} -\alpha_0 & -\alpha_1 & -2\alpha_2 & \cdots & -2\alpha_{n-1} \end{bmatrix} \mathbf{T}, \\ \mathbf{K}_d &= \begin{bmatrix} \alpha_2 & \cdots & \alpha_{n-1} & -\alpha_n & 0 \end{bmatrix} \mathbf{T}. \end{aligned} \quad (2.67)$$

Entretanto, o *MatLab* possibilita uma maior facilidade na sintonia do PID, uma vez que pode-se traçar o lugar das raízes do sistema em malha fechada e então, baseado nos requerimentos de projeto ajustar os dois zeros do controlador de forma a se obter tais características desejadas. Neste trabalho a sintonia do PID foi realizada por meio da alocação de polos e ajuste manual a partir dos resultados obtidos de forma a se respeitar tanto o tempo de estabilização e máximo valor de ultrapassagem desejados, como a máxima tensão fornecida pela fonte.

2.6.3 PID discreto

Uma vez determinado os parâmetros para o controlador PID contínuo no tempo, pode-se através da transformada Z determinar o controle digital do mesmo, para tal, de acordo com (PHILLIPS; NAGLE, 2007), considerando a técnica de integração trapezoidal tem-se a equação 2.68.

$$V_{in} = K_p + K_I \frac{T}{2} \frac{z+1}{z-1} + K_d \frac{z-1}{Tz} \quad (2.68)$$

Logo, caso haja uma alteração no período de atuação do controle é possível corrigir os parâmetros rapidamente por meio da equação 2.68.

2.7 Projeto do observador de estados

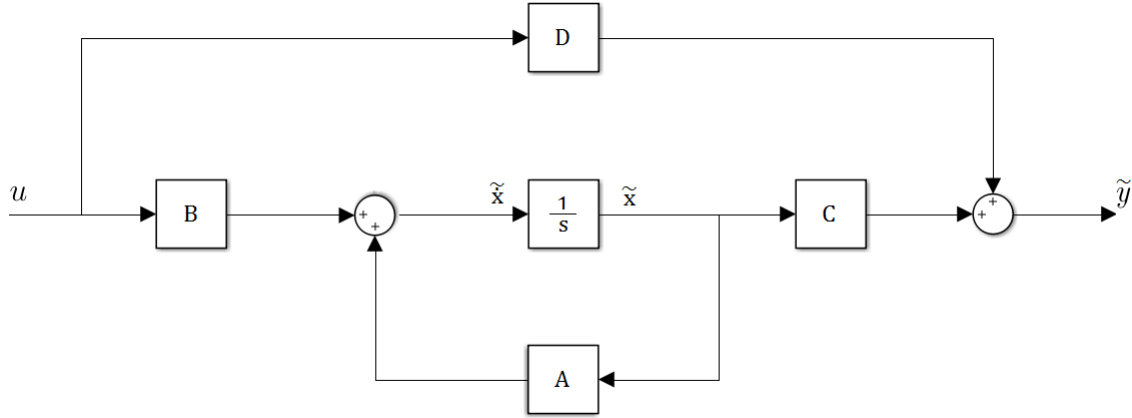
De modo semelhante a controlabilidade, para que se possa observar um estado do sistema deve-se determinar se este é observável. Logo para tal análise é necessário obter a observabilidade do sistema através da matriz

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \quad (2.69)$$

de dimensão igual do tamanho do vetor $\mathbf{x}$. De modo semelhante também, o sistema é observável caso a matriz observabilidade seja linearmente independente.

Em um primeiro momento pode-se imaginar que a observação dos estados pode ser obtida pela simples aplicação do modelo da planta, no qual, conhecendo as entradas do sistema, pode-se observar a respectiva saída, conforme apresenta na Figura 8.

Figura 8 – Diagrama de blocos de um observador de estados em malha aberta baseado no modelo da planta.



Fonte: Próprio autor.

No entanto, esta é uma visão bastante otimista do processo, uma vez que o modelo teórico da planta jamais será a reprodução perfeita do sistema na prática, logo esta aplicação pode levar às mesmas inconsistências que, por exemplo, um sistema de controle em malha aberta.

Para correção deste fator, deve-se utilizar um sistema em malha fechada de correção do erro de observação gerado pelo modelo teórico. Este erro de observação é definido como sendo a diferença entre o estado real do sistema e o valor observado deste estado, ou seja

$$\epsilon = x - \tilde{x}. \quad (2.70)$$

Logo para correção deste erro de observação é necessário criar uma realimentação do erro de saída do processo, conforme a Figura 9. Assim, podem-se determinar as seguintes equações de estados para o observador, sabendo-se que $D = 0$

$$\dot{\tilde{x}} = A\tilde{x} + Bu[k] + L(y - C\tilde{x}) = (A - LC) + Bu + Ly. \quad (2.71)$$

Conhecendo as equações de estado do sistema 2.6, 2.7 e a equação do erro 2.70, tem-se

$$\dot{x} - \dot{\tilde{x}} = Ax - A\tilde{x} - L(Cx - C\tilde{x}) = (A - LC)(x - \tilde{x}) \quad (2.72)$$

portanto,

$$\mathbf{x}[k+1] = \Phi \mathbf{x}[k] + \Gamma \mathbf{u}[k] + \mathbf{w}(t) \quad (2.75)$$

$$y[k] = \mathbf{C} \mathbf{x}[k] + \mathbf{D} \mathbf{u}[k] + \nu(t) \quad (2.76)$$

Portanto os estados seguintes não são apenas dados pelos estados atuais, entradas e suas respectivas matrizes de transmissão de estados e matriz de entrada, mas adiciona-se o ruído da planta a este cálculo, de forma semelhante a saída não é mais dada pela matriz dos sensores e de alimentação direta, mas adiciona-se o ruído de medição.

O filtro de Kalman atua observando os estados do sistema, mas também filtra o ruído de medição na saída do sistema. Para tal operação, considera-se que o ruído seja ortogonal ao sinal, como por exemplo o ruído branco (KALMAN, 1960). Deste modo, o filtro de Kalman é projetado com um passo de predição, seguido de um passo de correção (APRÁIZ; DIEGO; BARROS, 2018). Assim faz-se a predição do próximo estado, como em qualquer observador, conforme a equação 2.77.

$$\tilde{\mathbf{x}}[k] = \Phi \tilde{\mathbf{x}}[k-1] + \Gamma u[k-1] \quad (2.77)$$

Em seguida calcula-se da predição do erro de covariância

$$\tilde{P}[k] = \Phi \tilde{P}[k-1] \Phi^T + Q \quad (2.78)$$

onde Q é a covariância do ruído da planta $Q = E[\mathbf{w}\mathbf{w}^t]$.

Após isto, diferente de um observador comum onde o ganho (L) é ajustado a um vetor constante conforme o projeto, o passo de correção do filtro de Kalman atualiza os parâmetros do ganho (K), conforme a equação 2.79.

$$K[k] = \tilde{P}[k] \mathbf{C}^T (\mathbf{C} \tilde{P}[k] \mathbf{C}^T + R)^{-1} \quad (2.79)$$

No qual R é a covariância do ruído de medição $R = E[\nu \nu^t]$. Assim, o valor predito é corrigido segundo a equação 2.80.

$$\bar{\mathbf{x}}[k] = \tilde{\mathbf{x}}[k] + K[k](y[k] - \mathbf{C} \tilde{\mathbf{x}}[k]) \quad (2.80)$$

Finalmente a covariância do erro de observação é também corrigida

$$P[k] = (\mathbf{I} - K[k] \mathbf{C}) \tilde{P}[k]. \quad (2.81)$$

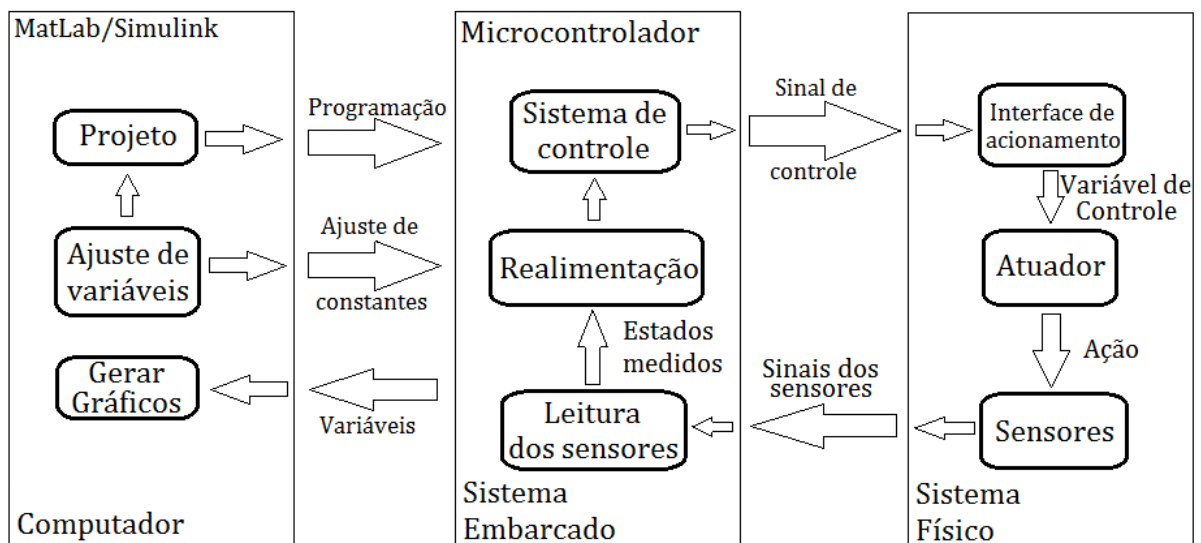
2.9 Sistema em *Hardware-in-the-Loop*

A utilização da simulação para o desenvolvimento de um projeto antes da real aplicação apresenta uma grande vantagem, uma vez que este procedimento possibilita que possíveis falhas possam ser descobertas e corrigidas ainda na fase de projeto, impedindo que equipamentos possam ser danificados durante o processo.

Entretanto a pura simulação consiste em um ambiente controlado adicionando simplificações que são sempre assumidas para facilidade de modelagem, este processo normalmente faz com que diferenças entre o sistema real e o simulado sejam criadas (MAGALLAN; ANGELO; GARCIA, 2011). No entanto o código e execução de um sistema completamente real pode se tornar caro e lento, então o sistema em tempo real de simulação com HIL (*hardware-in-the-loop*) surge como uma solução alternativa de baixo custo (RAJAMANI, 2011).

A Figura 10 apresenta um diagrama de blocos de exemplo de como funciona um sistema em *hardware-in-the-loop* que se caracteriza como uma solução híbrida entre a simulação e o teste em sistema real.

Figura 10 – Diagrama de blocos de um sistema genérico em *hardware-in-the-loop*.



Fonte: Próprio autor.

Assim, conforme a Figura 10, o projeto é realizado no *Simulink* que, por meio do *code generator*, realiza a programação do microcontrolador. Enquanto o sistema embarcado está ativamente controlando o sistema físico, ele permanece em comunicação com o computador, permitindo que ajustes de variáveis e visualização de gráficos em tempo real sejam possíveis. Entretanto o microcontrolador realiza todo o processamento de controle com total independência do computador, fazendo a leitura dos sensores e processando a malha de controle conforme projetado.

Neste trabalho, é utilizado o ambiente de desenvolvimento *Simulink* como um modo de se projetar e programar um controlador embarcado, devido a facilidade de se trabalhar com entradas, saídas e matrizes que o diagrama de blocos que este aplicativo possibilita. O *embedded coder* do *Simulink* realiza a programação do código automaticamente gerado e otimizado diretamente do ambiente de desenvolvimento.

Esta ferramenta é também muito útil para projetos de sistemas de controle, pois propicia a observação das entradas e saídas do sistema embarcado, permitindo o ajuste em tempo real de parâmetros e ganhos enquanto o sistema está ativo.

2.10 Conclusão do capítulo

Apresentam-se neste capítulo quais são as equações elétrica e de movimento que descrevem o sistema, bem como o respectivo espaço de estados para que seja possível desenvolver o sistema de controle. Assim, também justifica-se a necessidade de se discretizar o espaço de estados para o desenvolvimento do sistema com utilização de um microcontrolador.

Discutiu-se também os diversos métodos teóricos ou práticos, determinísticos e estocásticos, para identificação paramétrica de um sistema os quais são usados mais adiante neste trabalho. Além disto, definiu-se a utilização do controlador PID devido a sua facilidade de readequação ao se alterar a frequência de amostragem, mas também que há a necessidade de adição de um observador de estados de preferência um filtro de Kalman para observação dos estados medidos ou não.

Por fim, ressaltou-se a necessidade de se utilizar um sistema em HIL para desenvolvimento de estudos como este, pois a grande quantidade de testes levantados e ajustes a serem feitos durante otimização de parâmetros tornaria o processo de programação de códigos e análise resultados muito lento.

A seguir, no Capítulo 3 é descrito o sistema mecânico e elétrico, tais como relação de engrenagens, modelagem do motor, circuito ponte-H, sensor utilizado, descrição do sistema embarcado. Além da descrição do sistema em *hardware-in-the-loop* utilizado, técnicas de leitura do sensor, otimização do controlador e sistemas de acionamento do motor, proteções e filtro de Kalman.

3 Descrição do sistema

Este capítulo descreve o sistema mecânico desenvolvido para testes e simulações do controlador, um sistema em *hardware-in-the-loop* que possibilita a programação do controlador embarcado e visualização dos resultados, desenvolvimento dos drivers de acionamento e suas respectivas interfaces, apresentação do projeto de controle, observador e filtro de Kalman.

3.1 Projeto mecânico motor-roda

O sistema desenvolvido para execução dos testes e estudo do controle consiste em um sistema motor-roda, no qual um único motor aciona uma roda, em concordância com a arquitetura proposta de veículo elétrico de tração independente.

Para tal, foi desenvolvido sobre uma superfície plana um suporte para que a roda seja suspensa e possa rodar livremente apenas com a tração do motor, de mesma forma o sistema de coroa, engrenagem e pinhão possibilita que o motor freie a roda, permitindo que o sistema tenha controle completo sobre a rotação da roda, ao contrário de sistemas de controle parcial, nos quais há apenas tração ou frenagem. A Figura 11 apresenta o sistema aplicado.

O eixo do motor possui um pinhão de 11 dentes, enquanto a roda possui uma coroa de 55 dentes ligada ao eixo do motor, logo a vantagem mecânica gerada pelo conjunto é dada por

$$MA = \frac{T_{motor}}{T_{roda}} = \frac{\omega_{roda}}{\omega_{motor}} = \frac{55}{11} = 5, \quad (3.1)$$

ou seja, o eixo do motor roda 5 vezes mais rápido que a roda, entretanto o torque da roda é 5 vezes maior. Deste modo, sabendo-se que a velocidade de rotação nominal do motor é 2650 RPM, pode se dizer que a velocidade de rotação nominal da roda presa ao sistema é de

$$n_{NomRoda} = 530 \text{ RPM}. \quad (3.2)$$

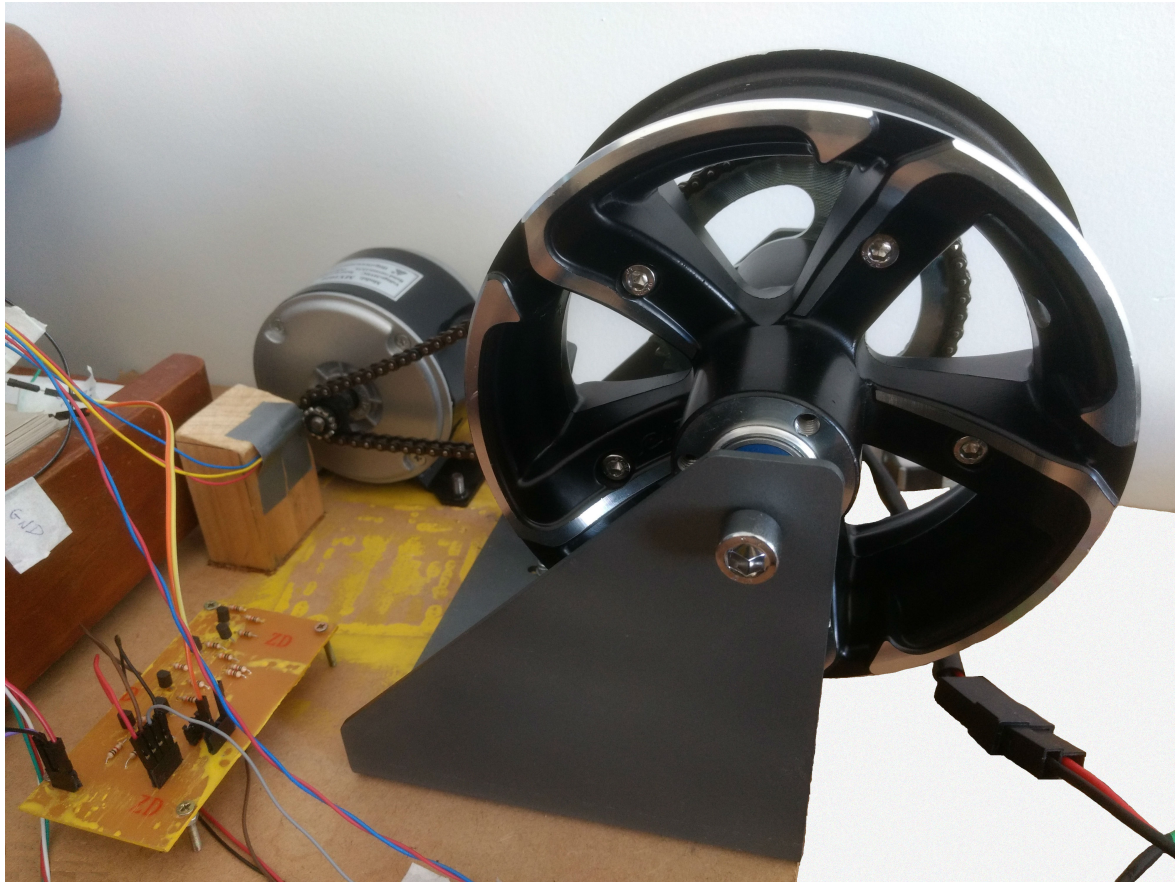
Como a roda possui um diâmetro de 18,0 cm, ou seja, uma circunferência de aproximadamente 56,55 cm, então a velocidade linear máxima nominal do sistema é

$$v_{vel} \approx 4,99 \text{ m/s} \approx 17,98 \text{ km/h}. \quad (3.3)$$

De modo semelhante, sabendo-se que a potência nominal do motor é de 250 W, o torque nominal do motor durante a máxima rotação nominal do sistema é dado por

$$T_{motor} = \frac{60}{2\pi} \frac{P}{\omega_{motor}} = 0,90 \text{ N.m}. \quad (3.4)$$

Figura 11 – Roda suspensa acoplada ao eixo do motor por meio de um conjunto coroa-corrente-pinhão, para realização testes de controle de velocidade e torque.



Fonte: Próprio autor.

Assim o torque nominal no eixo da roda é dado por 4,5 Nm, ou seja, a força de tração máxima da roda no solo durante a velocidade máxima de rotação é

$$F_{roda} = \frac{4,5}{0,09} = 50 \text{ N.} \quad (3.5)$$

3.2 Descrição do sistema elétrico

O sistema estudado é composto por um motor em corrente contínua modelo MY1016 da marca *Unitemotor*. Este motor possui uma tensão nominal de 24 V, corrente de 13,7 A, velocidade máxima de 2650 RPM e potência nominal de 250 W.

A tensão de alimentação é provida por duas baterias de 12 V e 7,0 Ah conectadas em série que alimentam a ponte-H que controla o motor. Esta por sua vez é controlada por um driver que recebe os sinais de controle do microcontrolador.

A velocidade do motor é obtida por meio do sensor de efeito Hall u1881 e seis pares de ímãs acoplados ao seu eixo, com polaridades alternadas, de modo que com a rotação deste

produzem-se seis sinais de borda de subida e decida por revolução. O sinal resultante do sensor é enviado a uma placa de interface responsável por adaptação das amplitudes dos sinais para comunicação com o microcontrolador embarcado.

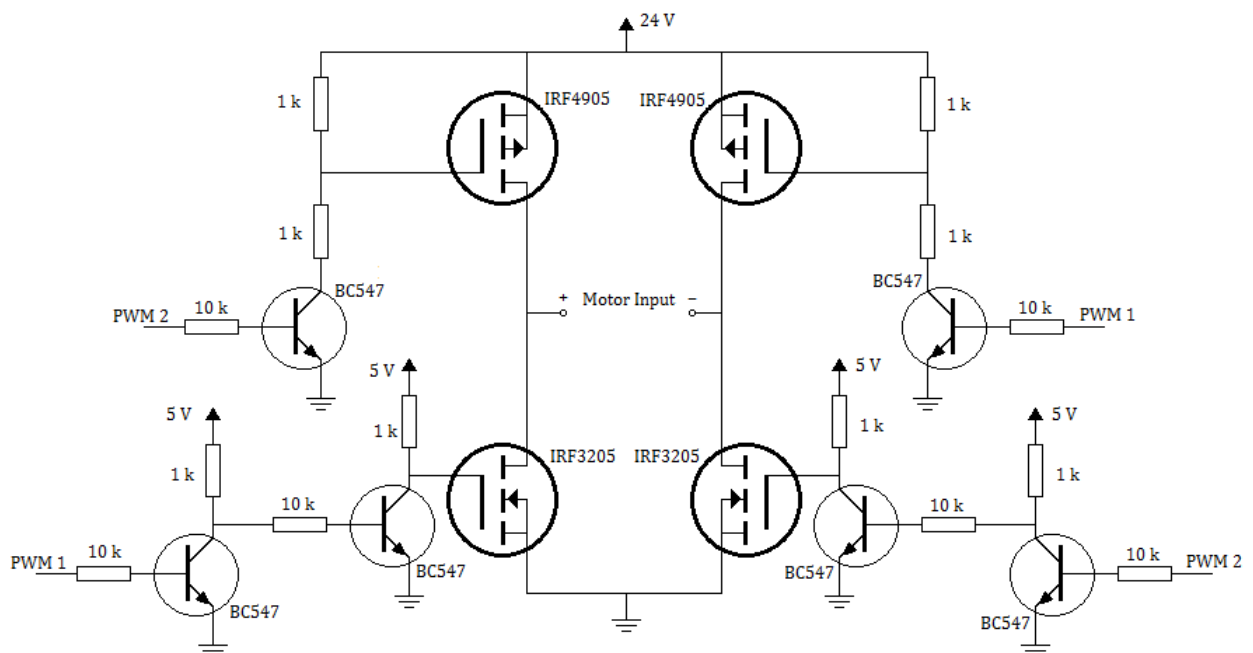
Todos os sinais de controle e sensores são enviados e lidos por um microcontrolador *BeagleBone Black* córtex A8, que possui até 1 GHz de frequência de operação. Este processador é programado neste trabalho por meio do *software Matlab/Simulink code generator* com auxílio do ambiente de diagrama de blocos do *Simulink*.

3.2.1 Driver ponte-H do motor

A ponte-H é controlada por quatro sinais de PWM os quais trabalham em pares controlando a direção com que o motor gira e sua respectiva velocidade de rotação, dependendo do par ativo e do ciclo ativo do sinal de PWM. É de extrema importância que apenas um dos pares de entradas de PWM sejam ativados por vez, pois o acionamento dos dois resulta em um curto circuito da fonte de alimentação.

O motor é acionado por meio desta ponte-H projetada com dois MOSFETs canal-p (IRF4905) e dois canal-n (IRF3205), assim, uma vez que os transistores de efeito de campo usados saturam apenas acima de 5 V e os sinais de PWM gerados pelo microcontrolador são de 3,3 V a ponte-H é acionada por um conjunto de transistores bipolares de junção do tipo NPN (BC547), tratada neste trabalho como driver do motor. A Figura 12 apresenta o esquema elétrico da ponte-H e seu respectivo circuito de interface de acionamento.

Figura 12 – Esquema elétrico de ponte H e seus circuitos de acionamento.



Fonte: Próprio autor.

Portanto, quando se deseja ativar a rotação do motor no sentido horário, utiliza-se o par PWM 1, e quando se deseja frear este motor ou inverter o sentido de rotação ativa-se o par PWM 2. Enquanto o ciclo ativo destes pares controlam o valor da tensão média de alimentação sobre o motor, atuando como entrada no espaço de estados do sistema, ou seja, com o aumento do ciclo ativo há um aumento no valor médio da tensão sobre o enrolamento do motor e vice-versa.

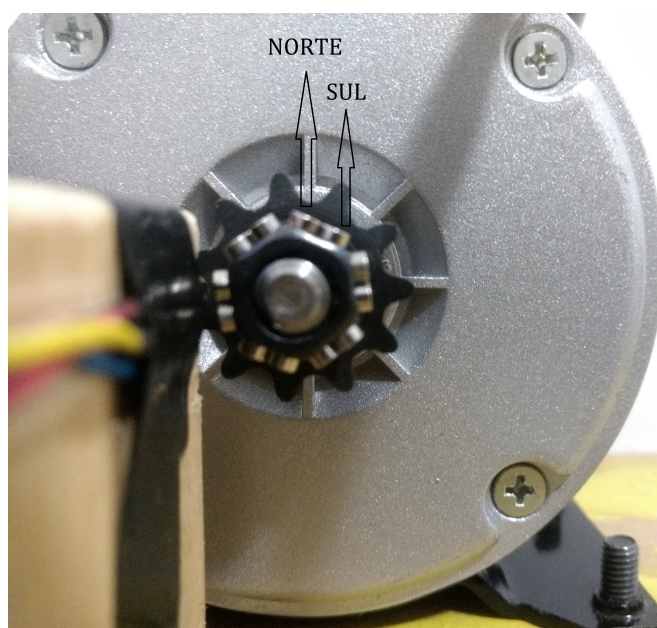
A tensão aplicada sobre o motor não é contínua e livre de ruído, pois a tensão média sobre este é gerada pela comutação dos transistores de efeito de campo. Assim, o sinal de alimentação do motor pode ser considerado um sinal ruidoso com valor médio igual a u (sinal de controle). No entanto, decidiu-se utilizar este circuito por se tratar de um estudo para aplicação com baixo custo, embora a ponte-H seja considerada como a pior maneira de acionamento do motor. Uma vez validado o modelo com a utilização da ponte-H o sistema torna-se válido para qualquer conversor.

3.2.2 Sensor de velocidade do motor

Para a medição da velocidade, foi desenvolvido um método de baixo custo não invasivo, no qual a partir da rotação do motor consegue-se determinar a velocidade de rotação do eixo do motor. Este sistema, é montado a partir de um sensor de efeito Hall fixo e ímãs acoplados ao redor do eixo.

O sensor de efeito Hall foi posicionado a uma distância fixa do eixo do motor, enquanto os ímãs foram posicionados ao entorno deste, com seus respectivos pares em antiparalelo, conforme mostrado na Figura 13.

Figura 13 – Posicionamento do sensor de efeito Hall e ímãs ao redor do eixo do motor elétrico em corrente contínua.



Fonte: Próprio autor.

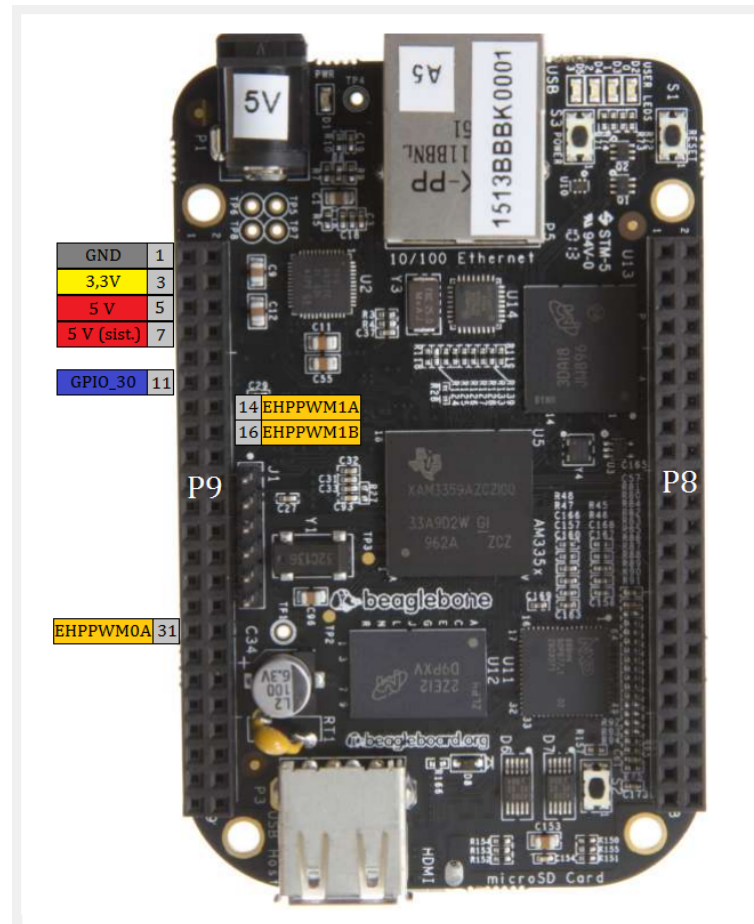
O sensor de efeito Hall u1881 é do tipo *Schmitt trigger*, portanto quando o polo norte é aproximado do sensor o pino referente a saída deste apresenta valor igual ao lógico 1 e este continua igual a 1 até que um polo sul aproxima-se do mesmo. Deste modo, a cada um sexto de revolução há uma transição de borda de subida e uma de descida, no qual, o microcontrolador, por sua vez, lê o intervalo de tempo entre duas bordas de subida e calcula a velocidade de acordo com a equação 3.6.

$$\omega = \frac{\pi}{3\Delta t} \text{ rad/s} \therefore n = \frac{10}{\Delta t} \text{ RPM} \quad (3.6)$$

3.3 Sistema embarcado

O microcontrolador embarcado utilizado para controle deste sistema é o *BeagleBone Black*. Este microcontrolador de córtex A8 possui até 1 GHz de processamento e, portanto, possui uma boa capacidade de comunicação com o *MatLab* em tempo real, possibilitando a aplicação do sistema em *hardware-in-the-loop*. A Figura 14 mostra a pinagem do microcontrolador.

Figura 14 – Ilustração do microcontrolador embarcado *BeagleBone Black* com os pinos usados e suas funções.



Fonte: Adaptado de (COLEY, 2013).

Os pinos em laranja são correspondentes as saídas PWM do microcontrolador e, portanto, existem 3 pinos de PWMs diferentes oferecidos pela placa de desenvolvimento, sendo tais pinos enumerados como 0, 1 e 2. As saídas A e B de cada PWM correspondem ao par complementar de cada porta PWM, ou seja se o pino EHRPWM1A (P9_14) está em nível lógico alto, o pino EHRPWM1B (P9_16) encontra-se em nível lógico baixo. Foram utilizados dois pinos de PWM P9_14 (EHRPWM1A) e P9_31 (EHRPWM0A) para acionamento da ponte-H, uma vez que tais pinos apresentam total independência de controle.

Tais pinos são chamados EHRPWMx, pois possuem tecnologia de alta resolução melhorada (*Enhanced High-Resolution*) e, portanto, podem operar de 0 a 400 MHz. Escolheu-se trabalhar com o valor de 20 kHz de frequência, uma vez que o período máximo do controlador é de 100 μ s, portando o PWM pode operar duas vezes mais rápido que o controlador, além de reduzir o ruído do equipamento na faixa audível. Por motivo de facilitar a realimentação e projeto do sistema de controle, esses pinos operam na polaridade positiva, ou seja o ciclo ativo do PWM é diretamente proporcional ao sinal de entrada deste, e não inversamente proporcional como seria a polaridade negativa.

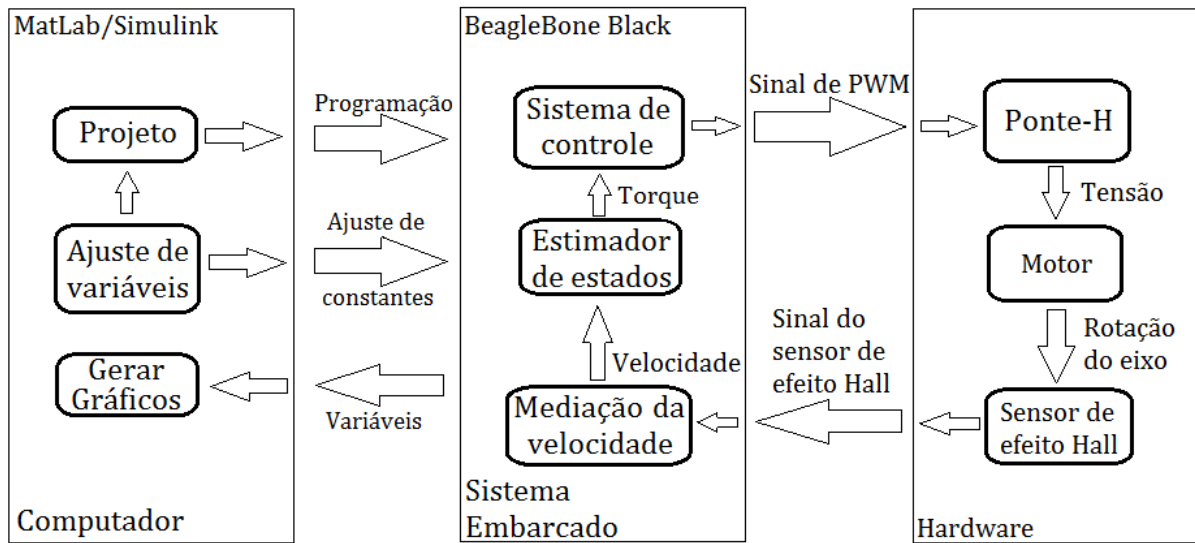
Além disto utilizou-se o pino P_11 (GPIO_30) para leitura do sensor, os quais são entradas digitais de amplitude máxima de 3,3 V. Para tal aplicação, uma vez que os sensores de efeito Hall possuem uma saída de 5 V, aplicou-se um divisor de tensão para redução do nível da tensão para um nível adequado ao equipamento. Este pino é amostrado em uma frequência de 10 kHz, ou seja, um período de amostragem de 100 μ s.

Utilizam-se ainda os pinos de alimentação 5 V, 3,3 V e GND para alimentação dos circuitos de interface, aproveitando-se dos reguladores de tensão internos oferecidos pela placa *BeagleBone Black*.

3.4 Sistema de *Hardware-in-the-Loop*

Para a aplicação do sistema em *hardware-in-the-loop* pesquisou-se primeiramente um microcontrolador que pudesse ser programado diretamente pelo *software MatLab/Simulink*. Para tal, escolheu-se trabalhar com o *beaglebone black*, devido a prévia experiência com este dispositivo.

Como mostra a Figura 15, o projeto do controlador criado no ambiente do *Simulink* é transferido ao sistema embarcado através do *BeagleBone Black*, mas também é possível que os valores de constantes, como ganho e referência, sejam alterados em tempo real, enquanto o sistema está ativo. Portanto tais ajustes podem ser realizados mais rapidamente do que em caso de implementação física comum.

Figura 15 – Diagrama de blocos do funcionamento do sistema em *hardware-in-the-loop*.

Fonte: Próprio autor.

Este sistema permite a prototipagem rápida, possibilitando a visualização de gráficos de velocidade, sinal de controle, observação de estados recebendo todos os dados em tempo real do *BeagleBone*. O microcontrolador, por sua vez, é responsável pela leitura e processamento da velocidade, por meio dos sinais gerados pelo sensor de efeito Hall, além da observação do torque e geração dos sinais de PWM para a interface de acionamento do motor.

3.4.1 Configuração do *MatLab*

Após a devida instalação dos *drivers* da placa *BeagleBone Black*, conforme indicado pelo manual do fabricante, no *MatLab* faz-se a instalação dos pacotes de suporte para o *BeagleBone Black* por meio da opção *Get Hardware Support Packages* nas aba *Add-Ons*. Os pacotes necessários para execução tanto dos projetos apresentados neste trabalho como do *hardware-in-the-loop* são:

- *ARM Cortex-A*:
 - *Generate code optimized for ARM Cortex-A processors*;
- *BeagleBone Black*:
 - *Interact with BeagleBone Black from MATLAB*;
 - *Generate code optimized for BeagleBone Black*.

Com a placa conectada ao computador, fazendo a instalação dos pacotes de suporte seleciona-se a opção que o microcontrolador está acoplado ao computador (*I have a BeagleBone*

Black hardware up and running) e em seguida seleciona-se a opção *Yes*, para que o *Simulink* seja configurado para utilização da placa. Conforme orientação da própria documentação do *MatLab Add Support for BeagleBone Black Hardware*.

3.4.2 Configuração do *Simulink*

Para configurar o *Simulink* deve-se abrir a interface de parâmetros de configuração do modelo, através da opção *Model Configuration Parameters* na aba *Simulation*. Em seguida, na caixa de diálogo seleciona-se a *Hardware Implementation* para as configurações da comunicação com a placa que deseja-se comunicar.

Como pode-se observar na Figura 16 com a seleção da placa *BeagleBone Black* na opção *Hardware board* as configurações da placa (*Hardware board settings*) são ajustadas automaticamente. Contudo para processos que demandam alta taxa de processamento, uma vez que o sistema em *hardware-in-the-loop* também demanda tempo de processamento na comunicação entre o computador e a placa, pode ocorrer atrasos na resposta do microcontrolador.

Figura 16 – Interface de configuração dos parâmetros para comunicação com o *hardware*.

Hardware board: BeagleBone Black

Code Generation system target file: [ert.tlc](#)

Device vendor: ARM Compatible Device type: ARM Cortex

► Device details

Hardware board settings

Operating system options

Base rate task priority:

Target Hardware Resources

Groups	Device Address: 192.168.7.2
Board Parameters	Username: root
Build options	Password:
Clocking	
External mode	

Fonte: Próprio autor.

Portanto para execução de sistemas de controle como o apresentado neste trabalho, no qual é indispensável a sincronia temporal do sistema, deve-se executar a comunicação dos dados em segundo plano. Para isso seleciona-se a opção *Run external mode in a background thread* (rodar o modo externo em segundo plano), na opção *External mode* do bloco *Target Hardware Resources*.

De modo semelhante, como mostra a Figura 17, os parâmetros da geração de código são também automaticamente preenchidos. Portanto a linguagem da programação é C e o arquivo gerado é do tipo *ert.tlc*. Caso deseje-se apenas gerar o código de programação sem que o código seja compilado e descarregado no microcontrolador basta selecionar a opção *Generate code only* (apenas gerar o código) na seção *Build process*, entretanto para o processo em HIL deve-se manter todo o processo pelo próprio *MatLab*.

Figura 17 – Interface de configuração dos parâmetros para geração de código.

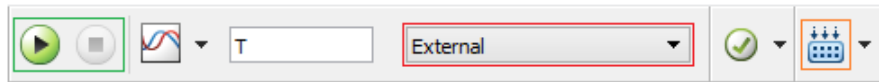
The image shows the 'Target Hardware Resources' configuration window in MATLAB/Simulink. It is divided into four main sections: 'Target selection', 'Build process', 'Toolchain settings', and 'Code generation objectives'. In 'Target selection', 'System target file' is 'ert.tlc' with a 'Browse...' button, 'Language' is 'C', and 'Description' is 'Embedded Coder'. In 'Build process', 'Generate code only' and 'Package code and artifacts' are unchecked, and 'Zip file name' is empty. In 'Toolchain settings', 'Toolchain' is 'Linaro Toolchain v4.8' and 'Build configuration' is 'Faster Builds', with a 'Show settings' link. In 'Code generation objectives', 'Prioritized objectives' is 'Unspecified' with a 'Set Objectives...' button, and 'Check model before generating code' is 'Off' with a 'Check Model...' button.

Fonte: Próprio autor.

Após a realização das configurações e o projeto do sistema que deseja-se executar em *hardware-in-the-loop* feito com auxílio do *Simulink*, deve-se alterar a opção de simulação normal para o modo externo (*External*), como mostra a Figura 18.

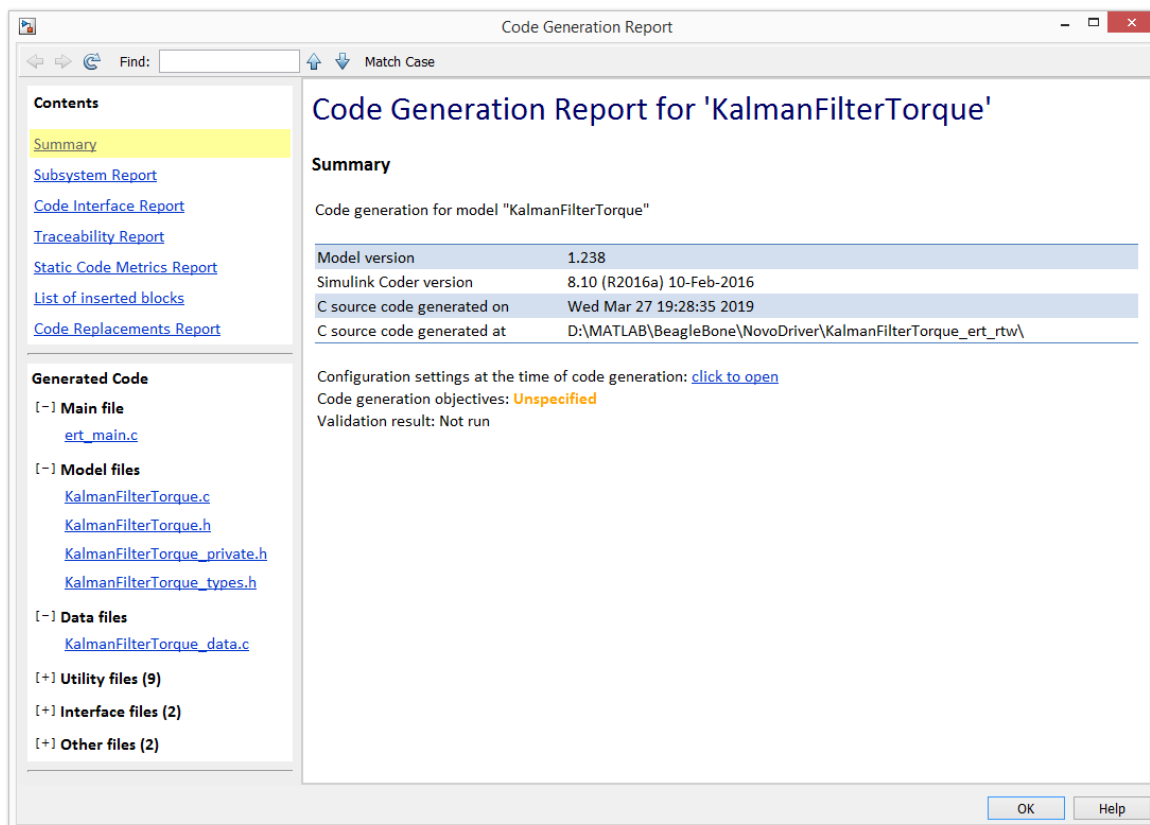
Portanto, a opção para transferir o código para a placa (*Deploy to Hardware*), destacada em laranja, serve para que o microcontrolador seja programado segundo o projeto realizado em diagrama de blocos do *Simulink*, mas não é usado quando deseja-se submeter o sistema a um processo em *hardware-in-the-loop*.

Figura 18 – Ilustração para execução do código em HIL.



Fonte: Próprio autor.

Para a simulação em HIL após selecionar a opção *External* destacada em vermelho, basta rodar o programa com no botão *Run*, destacado em verde na Figura 18. Em seguida, o projeto criado por meio do diagrama de blocos é convertido em código C e, em seguida, gera-se um relatório que pode ser observado na Figura 19.

Figura 19 – Ilustração do relatório gerado a partir do gerador de código do *Simulink*.

Fonte: Próprio autor.

O código em C é internamente compilado e transferido ao microcontrolador sem a necessidade de aplicativo ou compilador externo específico para este. Desta forma, inicia-se automaticamente o processo em hardware-in-the-loop, tornando o ajuste variáveis, observação de gráficos, dentre outros, realizáveis no próprio ambiente de desenvolvimento do *Simulink*, tornando a aplicação mais rápida que por exemplo a programação em baixo nível em *software* específico.

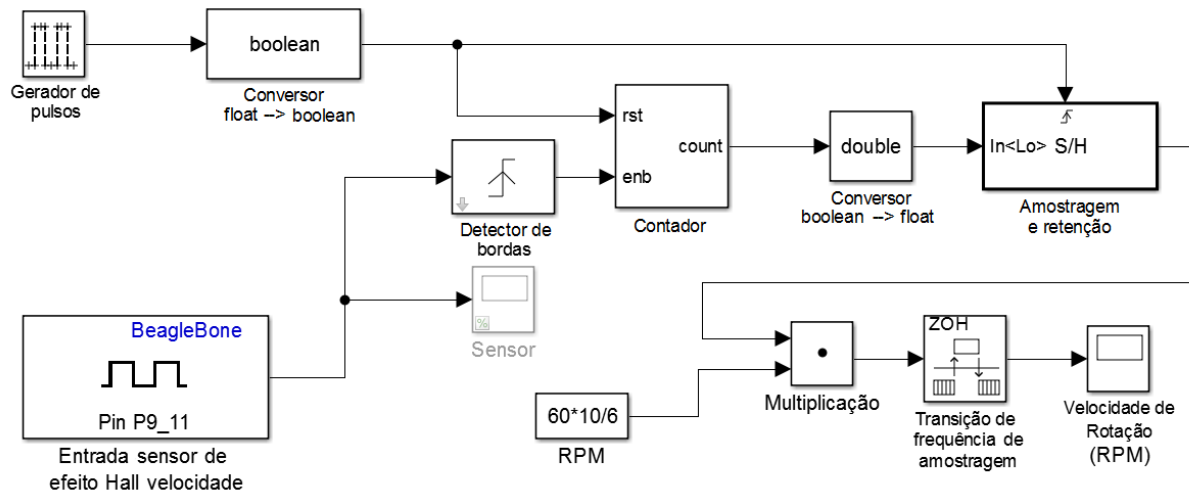
3.5 Leitura do sensor de velocidade

Dois métodos de leitura da velocidade foram estudados para aplicação da realimentação do controle, os quais estão descritos nas Seções 3.5.1 e 3.5.2.

3.5.1 Método da velocidade média

O primeiro método se dá segundo a média da medidas a cada período de tempo. O diagrama de blocos utilizado para o desenvolvimento e execução do código de leitura de velocidade para o rotação do motor deste método é dado pelo sistema apresentado na Figura 20.

Figura 20 – Diagrama de blocos para determinação do valor da velocidade média de rotação do motor a cada 100 ms.



Fonte: Próprio autor.

O *BeagleBone Black* faz a leitura do sinal de efeito Hall por meio do pino P9_11 a cada 0.1 ms (10 kHz), que é a frequência máxima captada pelo circuito integrado U1881. Este sinal pode ser visto no bloco de visualização de sinais chamado *Sensor* em tempo real e corresponde ao sinal de entrada do detector de bordas de subida que alimenta um contador determinando o número de bordas de subida (mudanças de estágio) percebidas durante o período, para determinação da velocidade.

O contador é zerado a cada 100 ms sendo esta, portanto, a taxa de atualização da velocidade. O número de transições de subida a cada 100 ms é amostrado e retido pelo bloco *Amostragem e retenção* que por sua vez tem esse resultado multiplicado pelas devidas constantes para saída em RPM. Ou seja, o valor do sinal é multiplicado por 600, para transformar o sinal obtido de 0.1s para minutos e em seguida dividido por 6, uma vez que ocorrem 6 bordas de subida para cada volta (6 pares de ímãs com polos norte e sul alternados).

Os resultados das velocidades também podem ser observados em tempo real nos respectivos blocos de visualização, entretanto para melhor análise ao final de cada simulação é gerado uma sequência de arquivos *.mat* para armazenamento dos dados obtidos. Assim estes resultados podem ser importados pelo MatLab, conforme o código exemplo da Figura 21.

Figura 21 – Código para captação dos dados do Simulink.

```

1 string1 = 'DataTest\Resultados_';
2 string3 = '.mat';
3 archiveNumber = 0;
4 Exception = 0;
5 string4 = 'Estrutura_';
6
7 while(Exception == 0)
8     try
9         string2 = num2str(archiveNumber);
10        stringLoad = strcat(string1, string2, string3);
11        stringVariable = strcat(string4, string2);
12        data.(stringVariable) = load(stringLoad);
13        archiveNumber = archiveNumber+1;
14    catch
15        archiveNumber = archiveNumber-1;
16        Exception = 1;
17    end
18 end
19
20 clear string1 string3 Exception stringLoad string2
21
22 %% Criando vetores para os dados
23
24 string1 = 'SpeedRPM_';
25 tempo = [];
26 velocidade = [];
27 for i=0:archiveNumber
28     string2 = num2str(i);
29     string = strcat(string1, string2);
30     stringVariable = strcat(string4, string2);
31     tempo = vertcat(tempo, data.(stringVariable).(string).time);
32     velocidade = vertcat(velocidade, data.(stringVariable).(string).signals.values);
33 end
34
35 clear string1 string2 string tempoFinal pontos string4 stringVariable ...
36 archiveNumber i data
37
38 %% Estruturando os dados
39
40 figure(1)
41 plot(tempo, velocidade)
42 grid on

```

Fonte: Próprio autor.

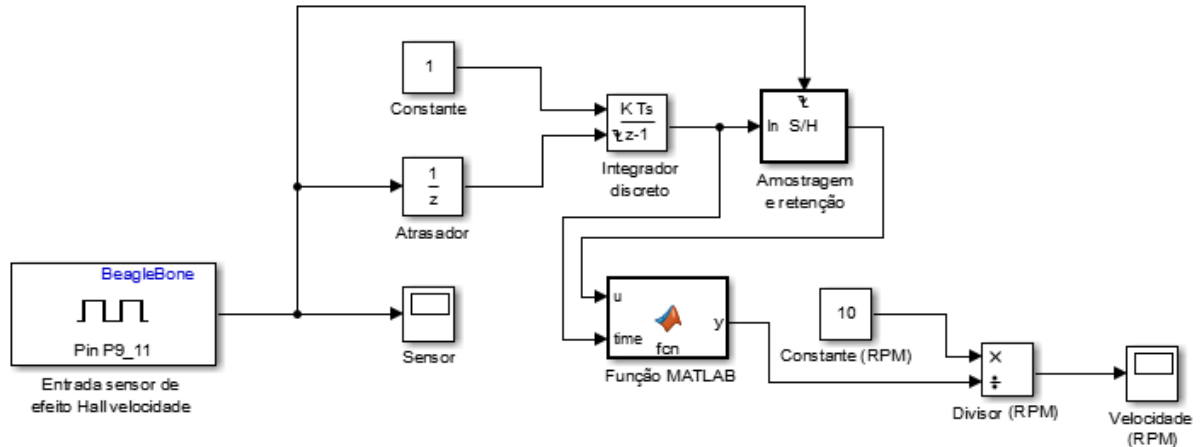
Os gráficos e análises obtidas com a utilização deste modelo serão apresentados e discutidos no Capítulo 5.

3.5.2 Método da velocidade instantânea

Com a análise dos resultados de velocidade obtidos pela leitura do sensor no método apresentado na Seção 3.5.1, notou-se que o sistema apresentava um tempo de resposta alto e que pode-

ria afetar a atuação do controle além de também gerar um erro de quantização fixo, sendo este muito alto para baixas velocidades. Por tal motivo alterou-se o sistema conforme o diagrama de blocos da Figura 22.

Figura 22 – Diagrama de blocos para determinação do valor da velocidade instantânea de rotação do motor pelo segundo método.



Fonte: Próprio autor.

Assim como o primeiro método apresentado, a leitura do sinal do sensor feita pelo *BeagleBone Black* é realizada através do pino p9_11 e este sinal pode ser observado pelo bloco de visualização chamado *Sensor*. A leitura de velocidade neste caso funciona com um integrador discreto que faz a contagem do tempo em milissegundos, assim a cada borda de descida o bloco *Amostragem e retenção* amostra o valor do tempo contado e reinicia a contagem do tempo zerando o integrador.

Este sinal é entrada no bloco *Função MATLAB* que possui o código apresentado na Figura 23.

Figura 23 – Função no *MATLAB* para correção da velocidade medida.

```

1 function y = fcn(u, time)
2 %%codegen
3
4 if u==0
5 y=inf;
6 elseif time > u
7 y = time;
8 else
9 y = u;
10 end

```

Fonte: Próprio autor.

Portanto, a entrada u representa o último valor de tempo medido pelo método apresentado e a entrada $time$ representa o valor do tempo atual que ainda está em contagem no

integrador. A saída y representa o valor do tempo que será considerada pelo sistema como a velocidade efetivamente medida.

Assim, como define o código apresentado na Figura 23, para o caso de u ser igual a zero, significa que ainda não teve nenhuma borda de descida no sistema, e portanto, a velocidade de rotação é igual a zero, logo para correção do cálculo da velocidade define-se $y = \infty$.

Seguindo o programa, caso a entrada $time$ seja maior que a u significa que a velocidade atual do sistema é necessariamente menor que a anterior amostrada, e portanto, a velocidade y começa a ser corrigida a cada iteração em busca da amostragem correta, ou seja, a variação de tempo considerada é $time$. Por fim, caso nenhum dos casos anteriores ocorram, a velocidade de saída é calculada através do tempo amostrado na entrada u .

Por conseguinte a velocidade efetivamente medida pelo segundo método é inversamente proporcional ao valor tempo decorrido entre duas borda de descida, transformado em RPM, conforme apresenta a equação 3.7.

$$n = \frac{10}{y} \quad (3.7)$$

3.6 Modelagem da planta

Esta seção apresenta a comparação da obtenção da modelagem da planta por dois métodos. O primeiro é obtido segundo a identificação dos parâmetros do motor e a partir do circuito elétrico análogo a este sistema, obter as matrizes de espaços de estado correspondentes ao respectivo modelo. Em seguida faz-se a modelagem do mesmo mediante métodos de identificação paramétrica, no qual pela aplicação de um degrau na entrada do sistema e observação da saída do mesmo, determinam-se os polos e zeros que compõem o sistema.

3.6.1 Identificação dos parâmetros do motor

Os parâmetros do motor usado não são providos pelo fabricante, portanto foi necessário obter estes parâmetros por meio de um procedimento experimental. Conforme o modelo matemático apresentado na seção 2.2, os parâmetros que necessitam ser determinados são R_a , L_a , K_e , B_m , K_t e J_m .

Os valores da resistência e indutância da armadura do motor são dois parâmetros que podem ser medidos, com o auxílio de um multímetro, diretamente no conector de entrada do motor. Para isto foi utilizado um multímetro *Minipa* modelo *MX-1001*, nos quais os valores obtidos estão contidos na Tabela 5.

Substituindo a equação 2.2 em 2.1 é possível determinar o valor da constante de força eletromotriz, dado os valores contantes de $v(t)$, $i_a(t)$, R_a e $\omega(t)$. Com o uso do valor obtido de R_a , foi realizado um conjunto de experimentos para 5 valores contantes de $v(t)$, iguais a 5; 7,5;

10; 12,5 e 15 V. Os valores medidos de corrente e velocidade de rotação do eixo do motor são apresentados na Tabela 3.

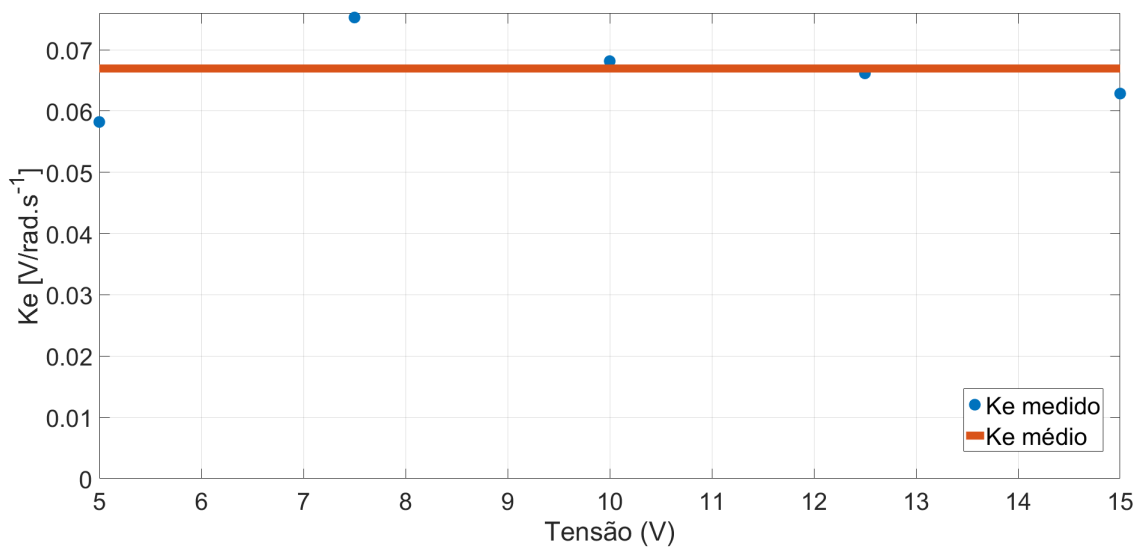
Tabela 3 – Valores de tensão, corrente e velocidade de rotação obtidos experimentalmente e valores de K_e respectivamente calculados.

Tensão [V]	Corrente [A]	Velocidade [rad/s]	K_e [V/rad.s ⁻¹]
5,00	0,56	69,0313	0,0582
7,50	0,65	84,5088	0,0753
10,00	0,71	128,4911	0,0681
12,50	0,80	167,7610	0,0662
15,00	0,86	214,6755	0,0629

Fonte: Próprio autor.

Os valores de K_e encontrados para experimento são mostrados na Figura 24, juntamente com uma linha que representa o valor da constante K_e .

Figura 24 – Valores da constante K_e e valor médio obtidos experimentalmente em função da tensão.



Fonte: Próprio autor.

Dados os valores experimentalmente obtidos para os valores das tensões aplicadas ao motor, determinou-se o valor de K_e por meio do valor médio, sendo este apresentado na Tabela 5.

Considerando-se que a constante de inércia do motor é desprezível em comparação com a da roda o valor de J_m pode ser determinado pela equação da constante de inércia da roda, aproximando esta para um cilindro maciço formado pelo centro da roda, uma vez que a casca possui uma massa muito pequena, assim

$$I_{roda} = \frac{1}{2}mr^2, \quad (3.8)$$

onde a massa m é aproximadamente 2 kg e o raio é 5,0 cm, logo, conhecendo o valor da vantagem mecânica do sistema, calcula-se

$$J_m = \frac{I_{roda}}{5} = 0,0005 \text{ Kg.m}^2. \quad (3.9)$$

Os valores dos parâmetros B_m e K_t estão relacionados à equação 2.3, portanto, utilizando-se os parâmetros de corrente e velocidade de rotação do experimento apresentados na Tabela 3 calcula-se a relação entre B_m e K_t segundo a equação 3.10.

$$\frac{K_t}{B_m} = \frac{I}{\omega} \quad (3.10)$$

Assim, conhecendo a relação entre as duas variáveis e sabendo que K_t apresenta valor aproximado de K_e , através de um processo recursivo, otimizou-se K_t para o valor apresentado na Tabela 5 e foram obtidos, portanto, os valores experimentais de B_m , conforme mostra a Tabela 4.

Tabela 4 – Valores de corrente e velocidade de rotação obtidos experimentalmente para determinação do valor de B_m .

Corrente [A]	Velocidade [rad/s]	B_m [kg.m ² /s]
0,56	69,0313	$0,7060e^{-3}$
0,65	84,5088	$0,6694e^{-3}$
0,71	128,4911	$0,4809e^{-3}$
0,80	167,7610	$0,4150e^{-3}$
0,86	214,6755	$0,3487e^{-3}$

Fonte: Próprio autor.

Desta forma, também determina-se o valor da constante B_m por meio da obtenção do valor médio dos valores experimentais obtidos. A Tabela 5 apresenta um resumo dos parâmetros determinados.

Tabela 5 – Parâmetros que compõem o modelo matemático da planta do sistema determinados experimentalmente.

R_a [Ω]	L_a [H]	K_e [V/rad.s ⁻¹]	K_t [N.m.rad/s.A]	B_m [Kg.m ² /s]	J_m [Kg.m ²]
1,752	$344,6e^{-6}$	0,0669	0,0870	$0,5679e^{-3}$	0,0005

Fonte: Próprio autor.

3.6.2 Identificação do sistema completo

Devido a proposta deste trabalho de se desenvolver um sistema de controle robusto e de baixo custo o driver do motor acaba por afetar a operação deste de maneira ideal, por este motivo foi também desenvolvido a modelagem do sistema como um todo na intenção de se minimizar as discrepâncias causadas pelo efeito supracitado.

Portanto, utilizando-se do mesmo equacionamento e processo da Seção 3.6.1, porém desta vez com o sistema completo montado e utilização do próprio sensor de velocidade para leitura da mesma.

Considerando-se também que os valores da resistência e indutância da armadura, respectivamente, R_a e L_a foram medidos corretamente e não dependem da alimentação do sistema, têm-se os mesmos valores para o modelo novo, bem como o valor da inércia da roda. Entretanto, o experimento da tensão, corrente e velocidade de rotação constantes foi refeito de modo a se obter os valores da Tabela 6.

Tabela 6 – Valores de tensão, corrente e velocidade de rotação para determinação das constantes do sistema.

Tensão [V]	Corrente [A]	Velocidade [rad/s]
6,8	0,614	157
6,9	0,617	164
7,0	0,620	171
7,1	0,623	194
7,2	0,626	238

Fonte: Próprio autor.

Os valores de tensão foram escolhidos dentro de uma faixa estreita, pois esta faixa apresenta maior linearidade entre a entrada e a saída do mesmo, uma vez que um pouco abaixo há baixa ou nenhuma rotação do motor e acima há alta velocidade de rotação, mas baixa variação da rotação. Como resultado obteve-se a Tabela 7.

Tabela 7 – Valores das constantes de força eletromotriz, de torque eletromagnético e de viscosidade experimentalmente obtidos.

K_e [V/rad.s ⁻¹]	K_t [Nm/A]	B_m [Kgm ² /s]
23,6.10 ⁻³	42,4.10 ⁻³	0,15379.10 ⁻³

Fonte: Próprio autor.

Resultando na matriz de estados:

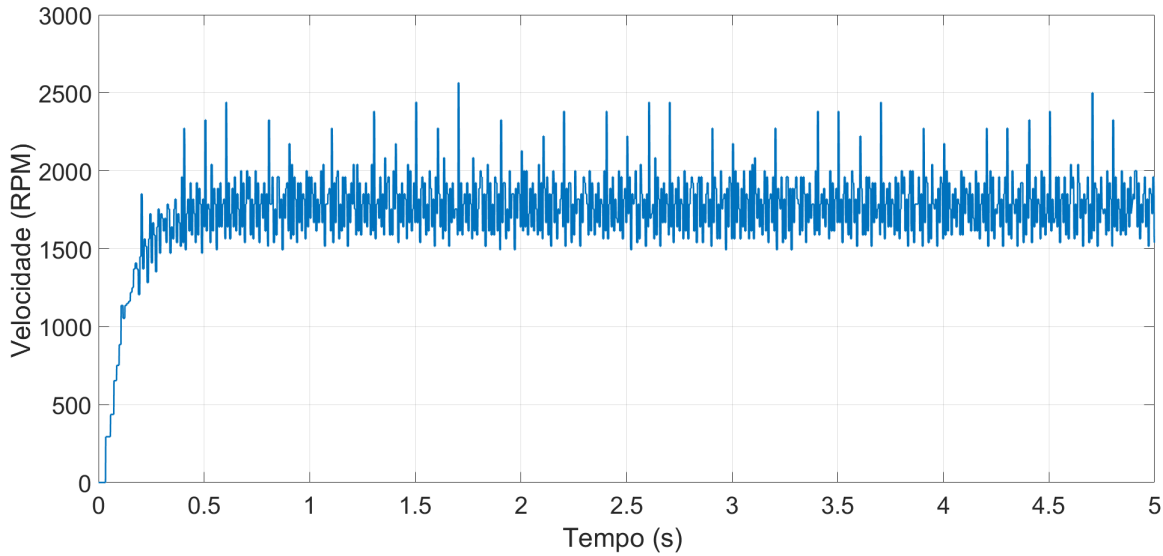
$$\begin{bmatrix} \dot{\omega}(t) \\ \dot{T}_m(t) \end{bmatrix} = \begin{bmatrix} -0,3076 & -2,8978 \\ 2000 & -13741 \end{bmatrix} \begin{bmatrix} \omega(t) \\ T_m(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 123,0299 \end{bmatrix} \begin{bmatrix} v(t) \end{bmatrix} \quad (3.11)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \omega(t) \\ T_m(t) \end{bmatrix}. \quad (3.12)$$

3.6.3 Identificação paramétrica por meio da entrada degrau

A determinação da função de transferência do sistema pode ser obtida por meio da entrada degrau e observação da respectiva saída. Para este procedimento utilizou-se uma entrada degrau de 12 V e obteve-se a curva apresentada na Figura 25, na qual aplicou-se o segundo método de leitura da velocidade.

Figura 25 – Curva da velocidade em função do tempo para a entrada degrau.



Fonte: Próprio autor.

Para a determinação do modelo utilizaram-se diversos métodos de identificação paramétrica os quais são apresentados e discutidos no Capítulo 4.

3.7 Projeto do controlador PID

É feita uma comparação entre o controle de velocidade realimentado diretamente e o obtido através de um observador de estados, de forma a se testar diversas modelagens para o mesmo sistema determinadas através de diferentes métodos.

O controle projetado de torque, por sua vez utiliza como realimentação a saída do observador de estados filtrada, primeiramente porque o sensor utilizado produz uma grande quantidade de ruído devido ao método de leitura adotado. Mas também porque não há presença de nenhum sensor para a medição do valor do torque e então é necessário a utilização de um observador de estados.

Portanto, para o controle de torque, foi primeiramente observado o valor deste estado por meio de um do filtro de Kalman que funciona como uma realimentação para este controle PID de torque. O valor do estado obtido é subtraído do sinal de referência (torque desejado) de forma a se obter o erro do controle, que funciona como sinal de entrada do controlador PID. O sinal de saída altera o valor da tensão média de alimentação do motor que é a única variável controlada do sistema.

Para facilitar o desenvolvimento de projetos de controle PID discreto o *MatLab* possui, no ambiente *Simulink*, um bloco para configuração e ajuste deste tipo de controlador. A Figura 26 apresenta a interface de ajuste dos parâmetros deste controlador.

Portanto na seção *Controller* seleciona-se a opção do controlador a ser usado (PID, PI, PD, P, I) e também o domínio discreto no tempo na opção *Discrete-time*, conforme observa-se na Figura 26. Na seção de configurações para o tempo discreto *Discrete-time settings* seleciona-se os métodos de integração e filtragem dentre as opções de método retangular com início no valor medido, retangular com fim no valor medido e método trapezoidal. Por fim ainda na seção *Discrete-time settings* faz-se o ajuste do período de amostragem.

Na aba principal (*Main*) ajusta-se portanto os parâmetros do controlador (K_p , K_i e K_d) conforme mostra a Figura 26 e mais o coeficiente do filtro derivativo (N). Deste modo observa-se pela formula do compensador que quando se altera o período de amostragem do sistema não é necessário fazer o reajuste dos parâmetros do controlador PID, basta ajustar o período de amostragem para a nova frequência e o bloco realiza internamente todos os ajustes necessários.

Figura 26 – Diagrama de blocos do controlador PID embarcado, com realimentação, normalização e limitador para geração do sinal de controle.

Block Parameters: Discrete PID Controller

PID Controller
This block implements continuous- and discrete-time PID control algorithms and includes advanced features such as anti-windup, external reset, and signal tracking. You can tune the PID gains automatically using the 'Tune...' button (requires Simulink Control Design).

Controller: **PID** Form: **Parallel**

Time domain:
☐ Continuous-time
☒ Discrete-time

Discrete-time settings
 Integrator method: **Forward Euler**
 Filter method: **Forward Euler**
 Sample time (-1 for inherited): **Ts**

Main | PID Advanced | Data Types | State Attributes

Controller parameters

Source: **internal** [Compensator formula](#)

Proportional (P): **Kp**

Integral (I): **Ki**

Derivative (D): **Kd**

☒ Use filtered derivative

Filter coefficient (N): **N**

Tune...

$$P + I \cdot T_s \frac{1}{z-1} + D \frac{N}{1 + N \cdot T_s \frac{1}{z-1}}$$

Initial conditions

Source: **internal**

Integrator: **0**

Filter: **0**

External reset: **none**

☐ Ignore reset when linearizing
☒ Enable zero-crossing detection

OK **Cancel** **Help** **Apply**

Fonte: Próprio autor.

3.7.1 Ajuste de parâmetros do controlador PID

O projeto do controlador PID foi desenvolvido em etapas subsequentes, no qual primeiramente tentou-se estimar os parâmetros de controle (K_P ; T_I ; T_D) segundo os métodos propostos por Ziegler/Nichols, no entanto, como apresentado na seção 2.6.1, devido ao sistema apresentar um tempo de atraso muito pequeno o método se tornou inviável.

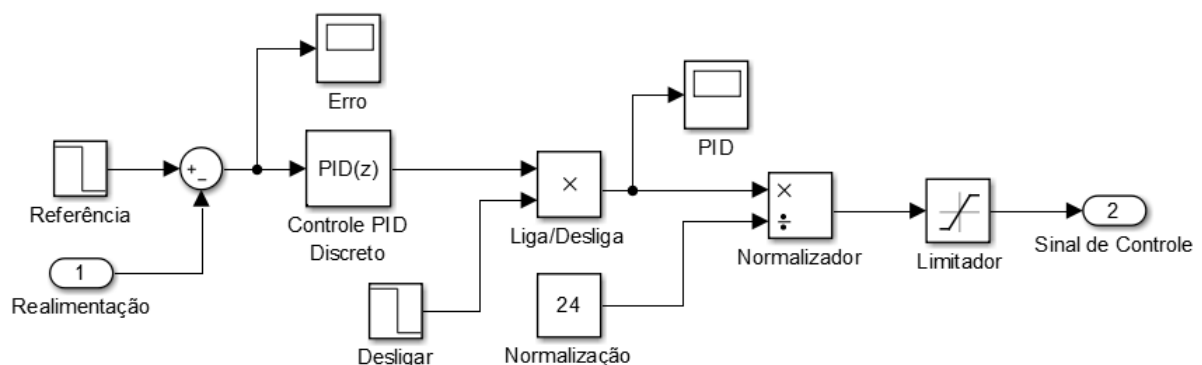
Portanto obteve-se o controlador estimado por meio da utilização do comando *pidtune* e ajuste manual dos polos e zeros do sistema de forma a se atingir os parâmetros desejados de tempo de estabilização e máximo valor de ultrapassagem. Nesta etapa, utilizou-se como modelo tanto o sistema identificado pelo método ARMA quanto o modelo matemático para controle de velocidade. Além do controle de torque do sistema pelo modelo matemático. Os resultados são apresentados e discutidos no Capítulo 5.

3.7.2 Sistema de acionamento

O valor da tensão de controle é normalizado e o resultado é convertido em uma variação do ciclo ativo do sinal de PWM. A saída do PWM controla a ponte-H que aciona o motor através da alteração do valor da tensão média sobre a armadura do motor, delimitando também sua direção, calculada conforme a equação do PID apresentada na equação 2.68, sendo este valor limitado em $\pm 24V$ fornecido pelas baterias.

O sinal de PID pode, portanto, ativar o controle para os dois sentidos de rotação, no qual o sentido horário é dado por valores positivos do sinal de controle e o anti-horário por negativos. O projeto do diagrama de blocos deste sistema é apresentado na Figura 27.

Figura 27 – Diagrama de blocos do controlador PID embarcado, com realimentação, normalização e limitador para geração do sinal de controle.



Fonte: Próprio autor.

Como pode ser observado na Figura 27 o sinal do erro pode ser visualizado no bloco de visualização *Erro* e serve de entrada para o controlador discreto PID, que por sua vez pode ser observado no bloco de visualização *PID*.

Este sinal possui uma chave temporal para desligamento do sistema em um tempo específico realizado pelos blocos *Desligar* e *Liga/Desliga*. Tal conjunto de blocos que alterna entre o sistema ligado e desligado serve para realização de testes, no qual deseja-se zerar as saídas do sistema no final dos testes, para que o motor seja desligado.

Por fim o sinal de controle é normalizado dividindo-se o sinal de controle pelo módulo do valor máximo de alimentação do sistema (24 V), assim, uma vez que a saída do pino de PWM do microcontrolador assume somente valores menores 1, o sinal é saturado de -1 a +1 pelo bloco *Limitador*. O sinal resultante é o sinal de controle normalizado do sistema.

3.7.3 Sistema de controle de direção

Como o sinal de controle normalizado do sistema é um sinal que varia entre -1 e 1, mas o ciclo ativo relativo a saída PWM do pino do *BeagleBone Black* só pode variar entre 0 e 1, então este sinal é dividido em dois PWMs, sendo um de ativação chamado *Horário* e o outro *Anti-horário*.

O projeto do diagrama de blocos relativo ao tratamento deste sinal pode ser observado na Figura 28.

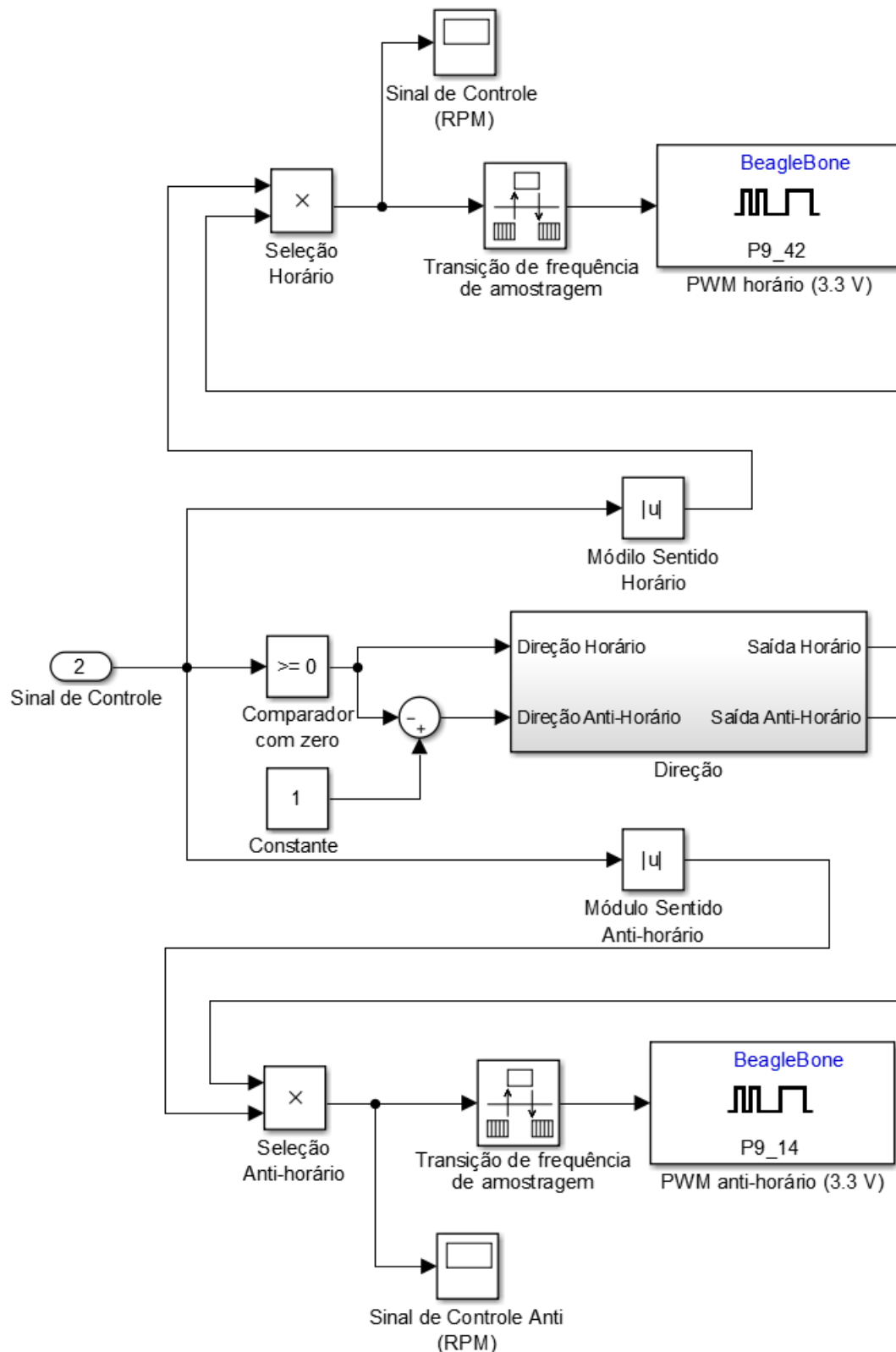
Como observa-se na Figura 28, o sinal de controle resultante do controlador PID é dividido em módulo, representado pelos blocos *Módulo Sentido Horário* e *Módulo Sentido Anti-horário*, os quais poderiam ser resumidos a um único bloco, entretanto por motivo de simplificação de compreensão este bloco é dividido em dois.

Tal sinal de controle ainda é submetido a um comparador com zero, que irá reconhecer se o sinal é positivo, assumindo em sua saída o valor 1 ou negativo igual a 0, deste modo o sinal da entrada *Direção Horário* é considerado o sentido convencional e portanto recebe 1 caso o sinal seja positivo e 0 caso o sinal seja negativo. De modo oposto, a entrada *Direção Anti-Horário* recebe 1 caso a entrada seja negativa e 0 caso esta seja positiva. Desta forma os sinais de saída (*Saída Horário* e *Saída Anti-Horário*) jamais serão iguais a 1 ao mesmo tempo, evitando que a ponte-H possa ser danificada.

O sinal resultante da saída do bloco *Direção* portanto serve como uma seleção de qual PWM será ativado, enquanto o sinal de módulo identifica o percentual do ciclo ativo do sinal de PWM desta saída, portanto os respectivos sinais podem ser observados nos blocos de visualização *Sinal de Controle (RPM)* e *Sinal de Controle Anti (RPM)*.

Por fim existe a geração interna no processador do sinal PWM e a saída externa no respectivo pino com o sinal de amplitude constante com valor igual 3,3 V, e largura de pulso variável com ciclo ativo proporcional a percentual do sinal de controle.

Figura 28 – Diagrama de blocos do sistema de controle de direção a partir do sinal de controle gerado pelo PID.

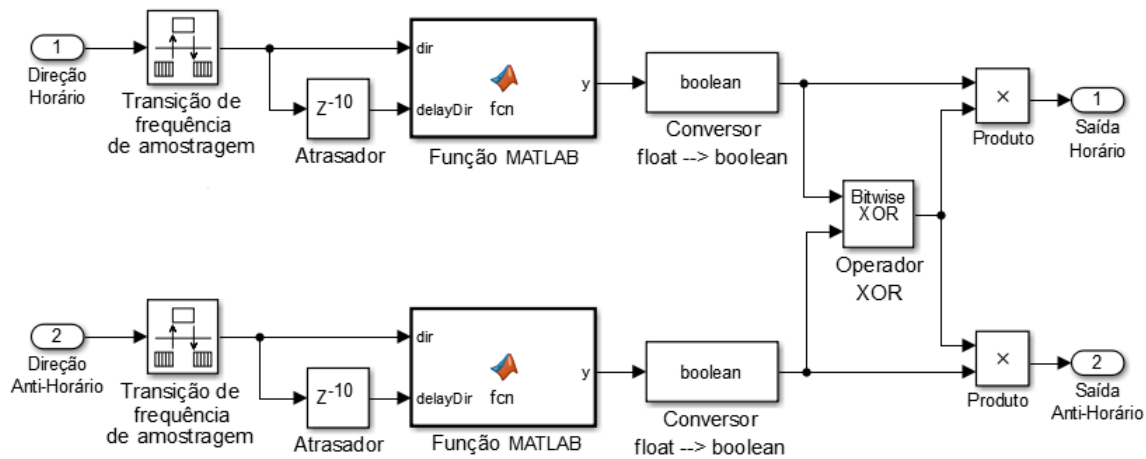


Fonte: Próprio autor.

3.7.4 Sistema de proteção

O bloco *Direção* serve como redundância da proteção da ponte-H contra curto-circuito, sendo que seu diagrama de blocos pode ser observado na Figura 29.

Figura 29 – Diagrama de blocos do sistema de acionamento dos pinos *PWM horário* e *PWM anti-horário* com proteção contra curto-circuito.



Fonte: Próprio autor.

No início do sistema apresentado os sinais de entrada são re-amostrados com frequência de $100 \mu s$, em seguida este sinal passa por um conjunto de blocos formado pelo *Atrasador* e *Função MATLAB* em ambos os braços. Este conjunto serve para que o sinal resultante *y* tenha um atraso de 1 ms apenas para a borda de subida. O código da função pode ser observado na Figura 29.

Figura 30 – Função no *MATLAB* mostrada na Figura 29 para proteção contra curto-circuito durante a mudança de sentido.

```

1 function y = fcn(dir, delayDir)
2 %#codegen
3
4 if delayDir == 0 && dir == 1
5     y = delayDir;
6 else
7     y = dir;
8 end

```

Fonte: Próprio autor.

Uma vez que o sistema real possui um tempo de comutação diferente de zero, caso a mudança de sentido seja simultânea por alguns instantes de tempo os dois braços apresentarão um curto circuito, podendo vir a danificar as baterias ou mesmo os transistores do driver do motor. Assim, este sistema funciona de modo a impedir que os dois braços da ponte-H sejam acionados ao mesmo tempo.

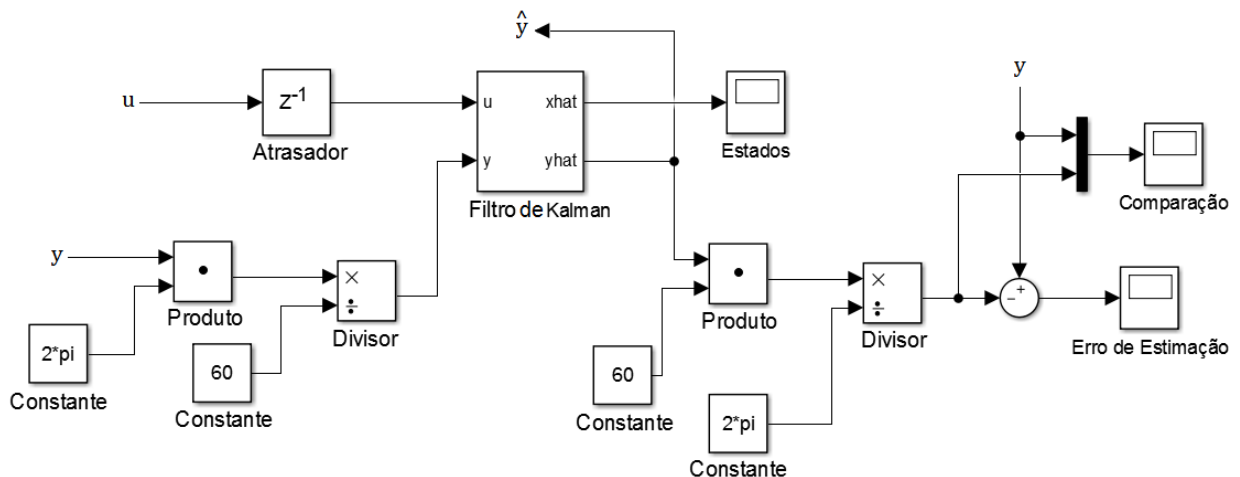
Em seguida o sinal passa por operador XOR para garantir que as saídas jamais sejam simultaneamente iguais a 1. Ou seja, caso as duas saídas seja iguais a 1 o operador XOR automaticamente zera ambas as saídas, protegendo o circuito elétrico de qualquer risco.

3.8 Aplicação do filtro de Kalman

Por causa da grande quantidade de ruído gerado pelo sensor de velocidade, a simples utilização do observador de estados pode resultar em baixa eficiência e alta vibração, o que gera desgaste físico das peças mecânicas do sistema.

Como solução deste problema, resolveu-se acrescentar um filtro de Kalman (descrito na Seção 2.8) para observação dos estados. A Figura 31 apresenta um diagrama de blocos para aplicação do filtro de Kalman ao sistema de controle de velocidade.

Figura 31 – Aplicação do filtro de Kalman para observação dos estados e realimentação do valor da velocidade observado.



Fonte: Próprio autor.

Nota-se na Figura 31 que como o valor da velocidade medida (y) é dado em RPM é necessário fazer uma transformação de RPM para rad/s para entrada no bloco *Filtro de Kalman* e o processo inverso é feito à saída do bloco para comparação de resultados.

A entrada u corresponde ao sinal de controle na entrada da planta, enquanto y é a saída medida do sistema. As saídas $\hat{x}$ e $\hat{y}$ são, respectivamente, os estados e a saída do sistema observados pelo filtro. É possível visualizar, portanto, a velocidade observada ($\hat{y}$) e a comparação desta com o sinal de saída, além do erro de observação resultante em seus respectivos blocos de visualização de sinais (*Estados*, *Comparação* e *Erro de Estimação*).

A Figura 31 apresenta a utilização do mesmo para utilização em um controle de velocidade, uma vez que a saída do sistema é obtida pelo único sensor presente no mesmo. Contudo

através de uma pequena mudança na aplicação do mesmo é possível utilizá-lo para o controle de torque, realimentando o torque observado obtido na saída x_{hat} .

Note que as saídas medida e observada são em rad/s e, portanto, devem ser transformadas para RPM para aplicação no sistema. Além da entrada u do sistema ter um atraso de uma amostra, uma vez que o observador utiliza o valor atual com relação aos estados e entradas anteriores. No controle de torque a realimentação é feita com o torque, que portanto deve ser obtida mediante a saída dos estados observados pelo filtro de Kalman. Com o intuito de facilitar a aplicação deste filtro o *MatLab* provê um bloco pronto chamado *Kalman Filter*, a Figura 32 apresenta a interface de ajustes destes parâmetros.

Portanto como observa-se na Figura 32 que é possível ajustar o bloco para operação no domínio do tempo contínuo ou discreto, por meio da aba de seleção *Time domain*, enquanto na caixa de dialogo inferior *Sample time* foi ajustada a frequência de amostragem de 0,1 ms.

O modelo do sistema utilizado é configurado na seção *System Model*, o qual pode ser alternado entre espaços de estados linear invariantes no tempo (*LTI State-Space Variable*), entradas externas (*input port*), usado em sistemas como controle adaptativo e matrizes de estado individuais (*Individual A,B,C,D matrices*). Estas matrizes do modelo do sistema são expressas pela modelagem discreta do mesmo para a mesma frequência de amostragem do filtro de Kalman.

Por fim ajusta-se as características de ruído na seção *Noise Characteristics*, no qual os parâmetros a serem ajustados são a variância da medida do processo, dada por Q , e a covariância do ruído da medida, dada por R , conforme apresentado na Seção 2.8. Os respectivos valores apresentados na Figura 32 são estimados mediante processo iterativo.

Figura 32 – Interface do filtro de Kalman com os respectivos parâmetros.

Block Parameters: Kalman Filter

Kalman Filter
Estimate the states of a discrete-time or continuous-time linear system. Time-varying systems are supported.

Filter Settings
Time domain: **Discrete-Time**
☐ Use the current measurement $y[n]$ to improve $\hat{x}[n]$

Model Parameters **Options**

System Model
Model source: **Individual A, B, C, D matrices**
A: **estadosDiscretos.A** B: **estadosDiscretos.B**
C: **estadosDiscretos.C** D: **estadosDiscretos.D**

Initial Estimates
Source: **Dialog**
Initial states $x[0]$: **0**

Noise Characteristics
☐ Use G and H matrices (default $G=I$ and $H=0$)
Q: **0.01** ☒ Time-invariant Q
R: **100** ☒ Time-invariant R
N: **0** ☒ Time-invariant N

Sample time (-1 for inherited): **0.0001**

OK **Cancel** **Help** **Apply**

Fonte: Próprio autor.

3.9 Conclusão do capítulo

Apresentou-se neste capítulo a relação entre velocidade e torque do motor e na roda, dadas as dimensões físicas do sistema e conjunto de engrenagens. Além de que o sistema de ponte-H para acionamento do motor gera uma grande quantidade de ruído sobre o motor, mas em contrapartida possui fácil montagem e se apresenta como um bom teste para o sistema de controle, justamente devido às condições não ideais.

Observou-se também que o sensor de efeito Hall apresenta vantagens quanto ao tamanho e posicionamento, mas que uma grande quantidade de ruído é produzida. Bem como há dois tipos de leitura do sensor de velocidade, sendo o primeiro baseado na média e o segundo na velocidade instantânea.

Conclui-se também nesta seção que a utilização do *MatLab* para desenvolvimento do *hardware-in-the-loop* se mostrou bastante proveitosa para o ajuste rápido de parâmetros e que para tal a escolha de um microcontrolador como o *BeagleBone Black* é uma boa opção, pela sua grande capacidade de processamento, quantidade de portas de entrada e saída, além de pinos PWM, portas analógicas dentre outros.

Foi notado também que é possível fazer a identificação paramétrica de um sistema tanto parte a parte, com o auxílio de uma bancada, como do sistema como um todo e que produzem resultados diferentes, que são estudados no Capítulo 5.

O ajuste dos parâmetros do PID deve ser feito mediante o lugar das raízes segundo os parâmetros de desempenho desejados uma vez que para sistemas com baixo tempo de atraso os métodos de Ziegler/Nichols não funcionam, além disto conclui-se que os sistemas de proteção contra curto-circuito por *software* são eficientes quando possíveis erros de execução do projeto.

Por fim, conclui-se que a aplicação de uma tecnologia como do filtro de Kalman é simples, uma vez que, conhecida a planta, este possui poucos parâmetros necessários para ajuste e devido a sua natureza matemática a simples utilização do bloco *Kalman filter* no *Matlab* se mostra possível.

No capítulo seguinte é apresentada a discussão sobre os diversos métodos de identificação paramétrica e seus respectivos desempenhos.

4 Identificação paramétrica do sistema

Este capítulo apresenta um estudo sobre identificação paramétrica de sistemas, no qual há uma curva resultante para cada modelagem estimada, seu respectivo gráfico de erro, tabelas comparativas e análise dos respectivos resultados.

4.1 Modelagem matemática com análise parcial

A modelagem matemática do sistema, conforme discutido na Seção 2.5 foi determinada assim como apresentado na Seção 3.6.1. Deste modo, o espaço de estados obtido é dado por

$$\begin{bmatrix} \dot{\omega} \\ \dot{T}_m \end{bmatrix} = \begin{bmatrix} -1,1357 & 2000 \\ -16,9090 & -5083,6 \end{bmatrix} \begin{bmatrix} \omega \\ T_m \end{bmatrix} + \begin{bmatrix} 0 \\ 252,5652 \end{bmatrix} \begin{bmatrix} v(t) \\ T_L(t) \end{bmatrix} \quad (4.1)$$

Baseado na modelagem determinada na equação 4.1 pode-se também reescrever o modelo discreto desde que se conheça o período de amostragem que será trabalhado. Para este trabalho é usado o comando *c2d* do *Matlab*, com amostrador ZOH (*zero order holder*), por causa do conversor analógico-digital utilizado para aplicação prática.

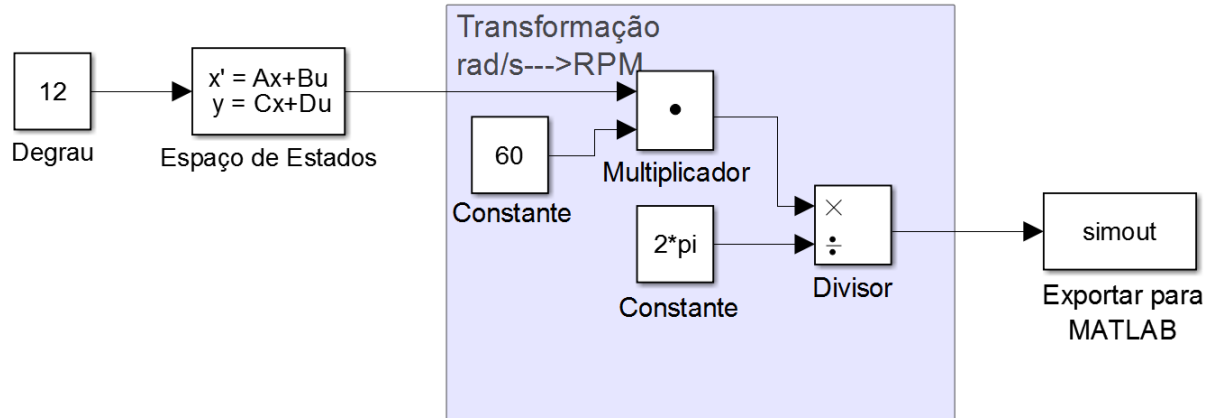
É importante que o amostrador da identificação paramétrica esteja de acordo com a aplicação real, pois ainda que haja outros amostradores para aplicações de controle, como por exemplo o amostrador de primeira ordem, a utilização de um diferente do hardware criará uma divergência entre o estado estimado e o real.

Portanto para a configuração descrita acima, utilizou-se o diagrama de blocos apresentado na Figura 33, a qual dado um degrau de valor igual a 12 V aplicado a entrada do sistema representado pelo espaço de estados obtido pela modelagem matemática, tem-se a velocidade em rad/s, a qual após transformada em RPM de modo a obter o resultado mostrado na Figura 34, no qual a velocidade em RPM é amostrada a cada 0,0001 s, ou seja, na mesma frequência que o sensor de velocidade.

Como o modelo teórico é calculado segundo o sistema internacional de unidades, é necessário fazer uma transformação para determinação da saída em RPM, como mostra a Figura 33.

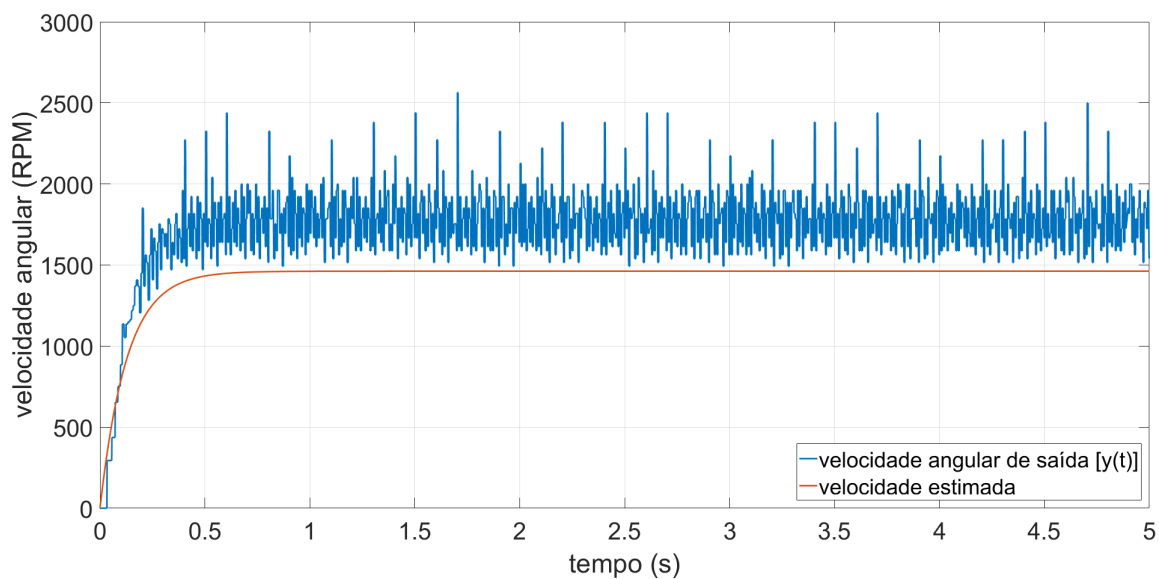
Como resultado, apresentado na Figura 34, nota-se um erro de regime permanente, este erro possui uma diferença de aproximadamente 14,29% na velocidade medida. Isto ocorre devido a diferença entre o modelo ideal e o real, além de imprecisões em medidas e aproximações. A perfeita identificação paramétrica é impossível, pois para toda medida sempre há imprecisão e ruído, entretanto esse erro, que pode ser observado na Figura 35, pode ser corrigido com um sistema de controle em malha fechada.

Figura 33 – Diagrama de blocos para simulação da resposta do espaço de estados, com conversão de rad/s para RPM.



Fonte: Próprio autor.

Figura 34 – Velocidade angular medida e estimada em função do tempo para uma entrada de grau com valor igual a 12 V.

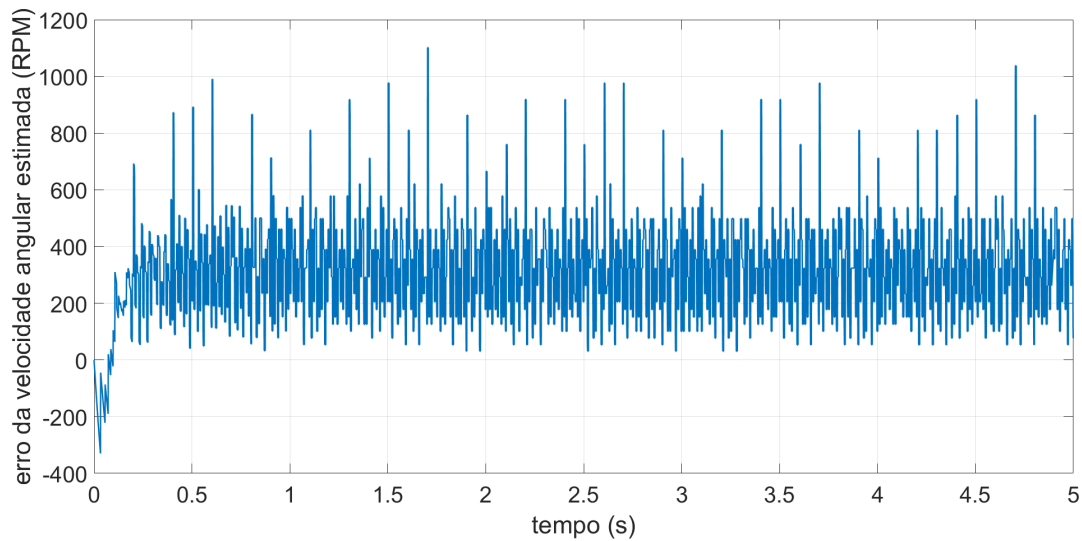


Fonte: Próprio autor.

O erro ITSE (*Integral of the Time multiplied by the Square Error*) para este modelo é de

$$ITSE_{SS} = 1,7004 \cdot 10^6. \quad (4.2)$$

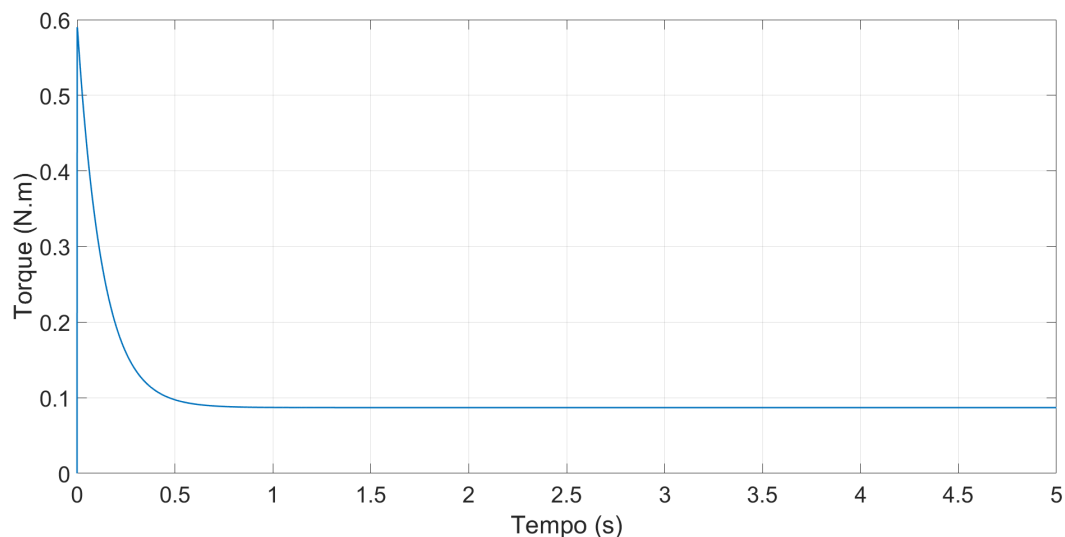
Figura 35 – Erro entre velocidade angular medida e estimada em função do tempo para uma entrada degrau com valor igual a 12 V.



Fonte: Próprio autor.

Por fim, é importante ressaltar que este modelo é o único que possibilita a observação do torque gerado pelo motor sem a necessidade da utilização de um dinamômetro ou torquímetro. Portanto para o presente sistema tem-se a curva de torque estimada, conforme a Figura 36.

Figura 36 – Torque estimado pelo observador de estados por meio do modelo matemático da planta em função do tempo para uma entrada degrau com valor igual a 12 V.



Fonte: Próprio autor.

Nota-se portanto um alto crescimento do torque gerado para aceleração do eixo e uma diminuição deste torque quando o eixo adquire maior velocidade de rotação.

4.2 Métodos experimentais determinísticos de primeira ordem

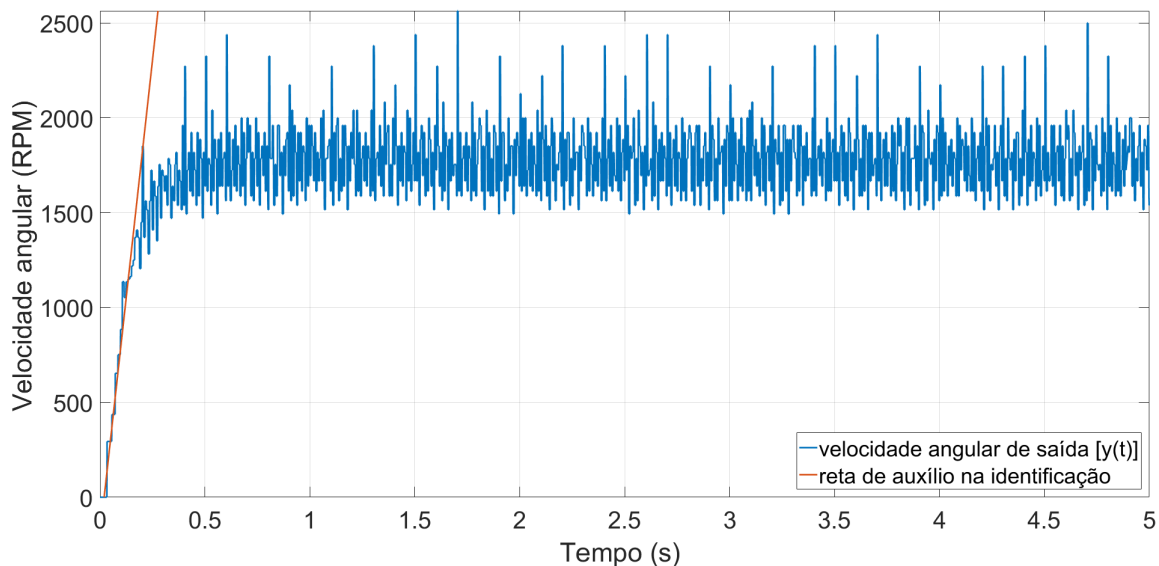
Esta seção apresenta os métodos experimentais determinísticos de identificação paramétrica do sistema de tração de uma roda independente. Dentre os métodos clássicos abordados existem os de modelagem de primeira ordem discutidos na Seção 4.2 e os de segunda ordem, Seção 4.4.

Os métodos de identificação de sistemas, baseados na resposta ao degrau, por modelagem de primeira ordem discutidos nesta seção são os apresentados por (ZIEGLER; NICHOLS, 1942) Seção 4.2.1, (SUNDARESAN; PRASAD; KRISHNASWAMY, 1978) Seção 4.2.4, Smith apontado por (COELHO; COELHO, 2004) Seção 4.2.3 e (HÄGGLUND; ÅSTRÖM, 1991) Seção 4.2.2.

4.2.1 Método de Ziegler/Nichols

Para a modelagem do processo pelo método de Ziegler/Nichols é necessário primeiramente traçar uma reta de auxílio. Esta reta foi traçada visualmente de forma a coincidir com a inclinação do gráfico na parte do transiente da resposta ao degrau, como mostra a Figura 37.

Figura 37 – Reta de auxílio para Método de Ziegler/Nichols sobre o gráfico de velocidade em função do tempo para uma entrada degrau de valor igual a 12 V.



Fonte: Próprio autor.

A expressão matemática da reta é dada por $f_o(i) = 10000t - 200$, deste modo, o modelo do processo de identificação paramétrica se dá conforme abordado na Seção 2.5.2, como mostra o código do *MatLab* apresentado na Figura 38.

Obteve-se portanto o modelo é representado pela função de transferência da equação 4.3.

$$G_p = \frac{K_p e^{-\theta s}}{\tau s + 1} = \frac{1754 e^{-0,0133s}}{0,1169s + 1}. \quad (4.3)$$

Figura 38 – Código para identificação paramétrica segundo o método de Ziegler/Nichols.

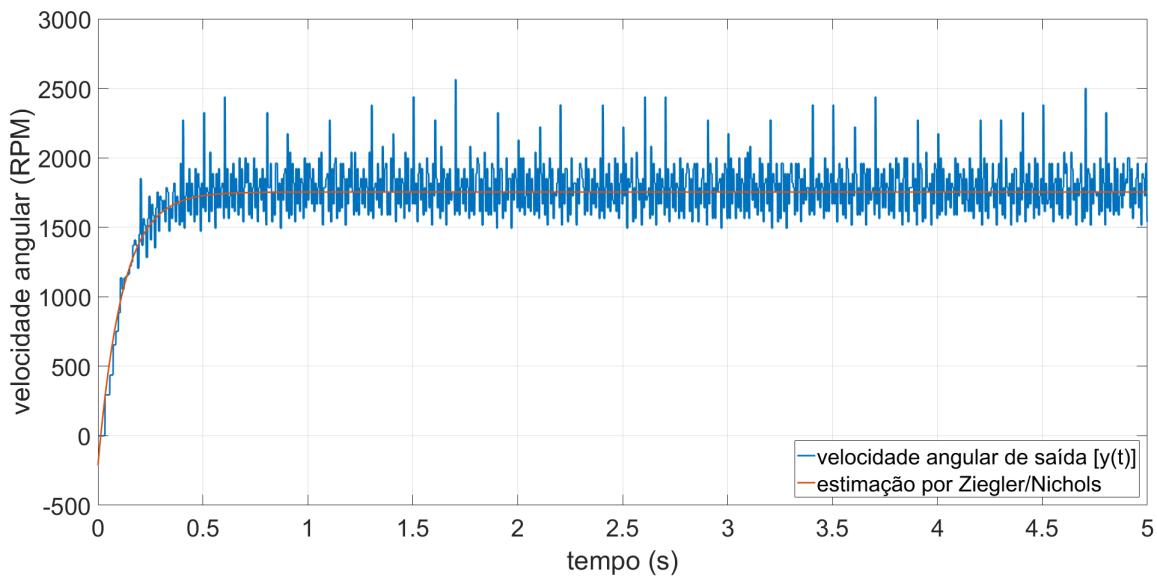
```

1 t1ZN = b/a;
2 thetaZN = t1ZN;
3 t2ZN = (Yss+b)/a;
4 tauZN = t2ZN-t1ZN;
5 KpZN = (Yss-0)/(1-0);
6 estimadoZN = zeros(pontos,1);
7
8 for i = 1:pontos
9     estimadoZN(i) = KpZN*(1-exp(-(tempo(i)-thetaZN)/tauZN));
10 end
11 createfigure2(tempo, saida, tempo, estimadoZN);
12 grid on
13 xlabel('tempo_(s)')
14 ylabel('velocidade_angular_(RPM)')
15 legend('velocidade_angular_de_saida_[y(t)]', 'estimação_por_Ziegler/Nichols', ...
16 'Location', 'SouthEast')
17
18 try
19     for i=1:total
20         erro(i) = saida(i)-estimadoZN(i);
21     end
22 end
23 itseZN = tempo.*(erro.^2);
24 ITSEZN = sum(itseZN)*0.001
25
26 createfigure(tempo, erro)
27 grid on
28 xlabel('tempo_(s)')
29 ylabel('erro_da_velocidade_angular_estimada_(RPM)')
```

Fonte: Próprio autor.

Submetendo-se assim o sistema identificado à mesma entrada degrau obteve-se como resultado a curva mostrada na Figura 39.

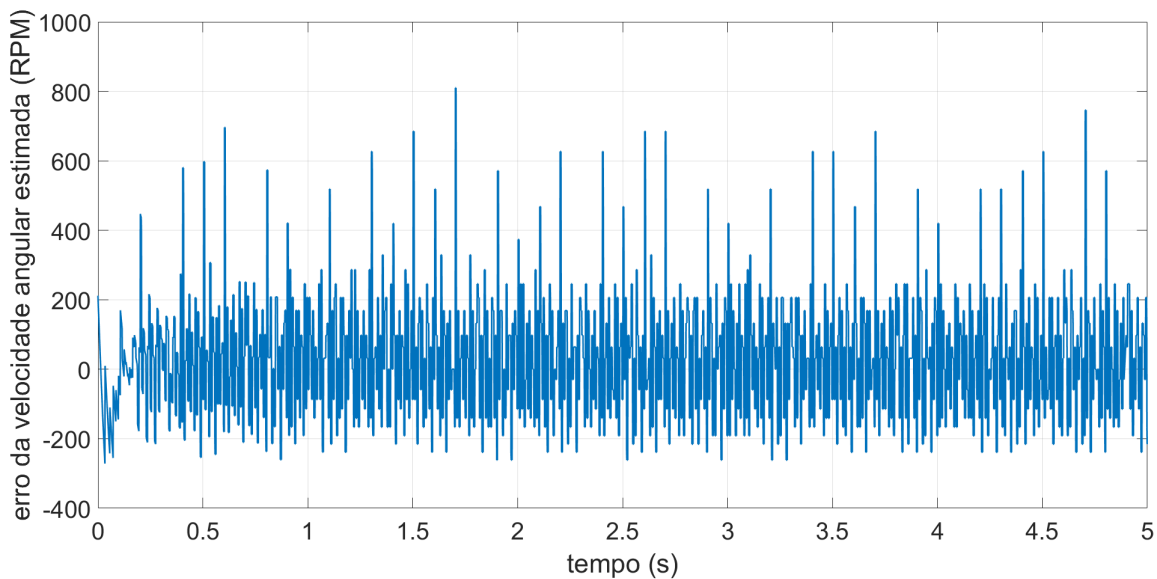
Figura 39 – Velocidade medida e estimada pelo método de Ziegler/Nichols, em função do tempo.



Fonte: Próprio autor.

Nota-se que diferente do modelo por espaço de estados onde o sinal apresenta um erro de regime permanente, este modelo não possui um erro de estado estacionário. Tal método dispõe simplicidade de aplicação, entretanto necessita de análise gráfica para o mesmo permitindo que erros de leitura alterem o resultado, mas permite a correção de erros causados por ruído que são nítidos visualmente. O gráfico do erro em função do tempo é dado pela Figura 40.

Figura 40 – Erro entre a velocidade medida e a estimada pelo método de Ziegler/Nichols, em função do tempo.



Fonte: Próprio autor.

O gráfico do erro apresentado na Figura 40 permite observar que o ruído de medição possui um *offset* próximo de zero, devido a própria natureza de um modelo de primeira ordem e, também, há uma correção do erro de regime permanente. Como resultado portanto tem-se um método de identificação paramétrica aplicável a sistemas compostos por motor-roda, onde o índice de erro ITSE é dado pela equação 4.4.

$$ITSE_{ZN} = 4,1266.10^5 \quad (4.4)$$

4.2.2 Método de Hägglund

Para a modelagem pelo método de Hägglund também utilizou-se a reta da Figura 37, conforme a metodologia apontada na Seção 2.5.2, assim, o sistema foi identificado segundo o código apresentado na Figura 41.

Figura 41 – Código para identificação paramétrica segundo o método de Hägglund.

```

1 %% Hägglund
2
3 t1HAG = b/a;
4 thetaHAG = t1HAG;
5 t2HAG = 0.125; %vizualmente no gráfico 0.632*y(inf)
6 tauHAG = t2HAG-t1HAG;
7 KpHAG = Yss;
8 estimadoHAG = zeros(pontos,1);
9
10 for i = 1:pontos
11     estimadoHAG(i) = KpHAG*(1-exp(-(tempo(i)-thetaHAG)/tauHAG));
12 end
13 createfigure2(tempo, saida, tempo, estimadoHAG);
14 grid on
15 xlabel('tempo_(s)')
16 ylabel('velocidade_angular_(RPM)')
17 legend('velocidade_angular_de_saida_[y(t)]','estimação_por_Hägglund', ...
18 'Location','SouthEast')
19
20 try
21     for i=1:total
22         erro(i) = saida(i)-estimadoHAG(i);
23     end
24 end
25 itseHAG = tempo.*(erro.^2);
26 ITSEHAG = sum(itseHAG)*0.001
27
28 createfigure(tempo, erro)
29 grid on
30 xlabel('tempo_(s)')
31 ylabel('erro_da_velocidade_angular_estimada_(RPM)')

```

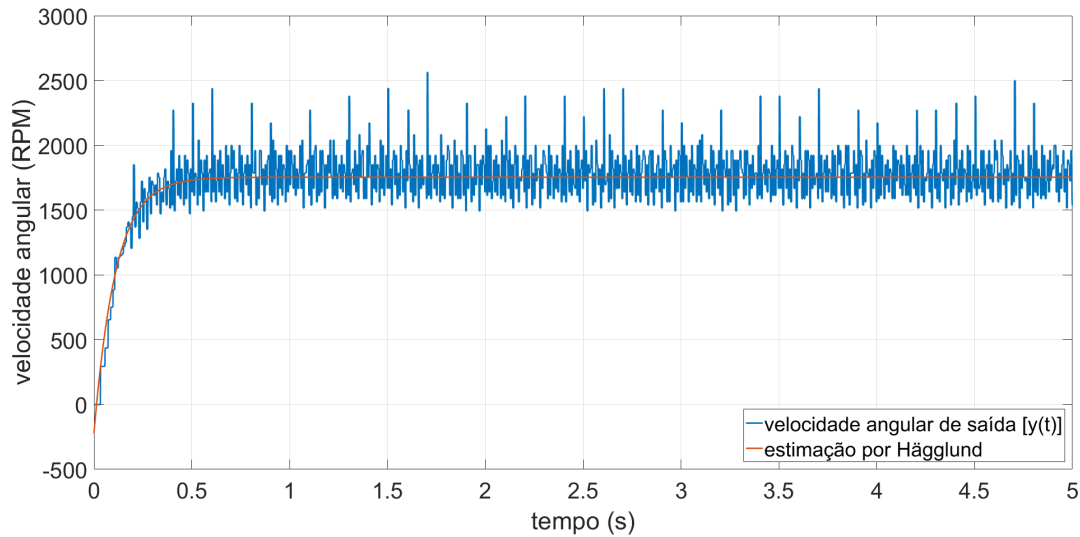
Fonte: Próprio autor.

Como resultado obteve-se o sistema representado pela função de transferência apresentada na equação 4.5.

$$G_p = \frac{1754e^{-0,0133s}}{0,1117s + 1} \quad (4.5)$$

Portanto, a resposta do modelo identificado pelo método de Hägglund quando submetido a mesma entrada degrau é mostrado na Figura 42.

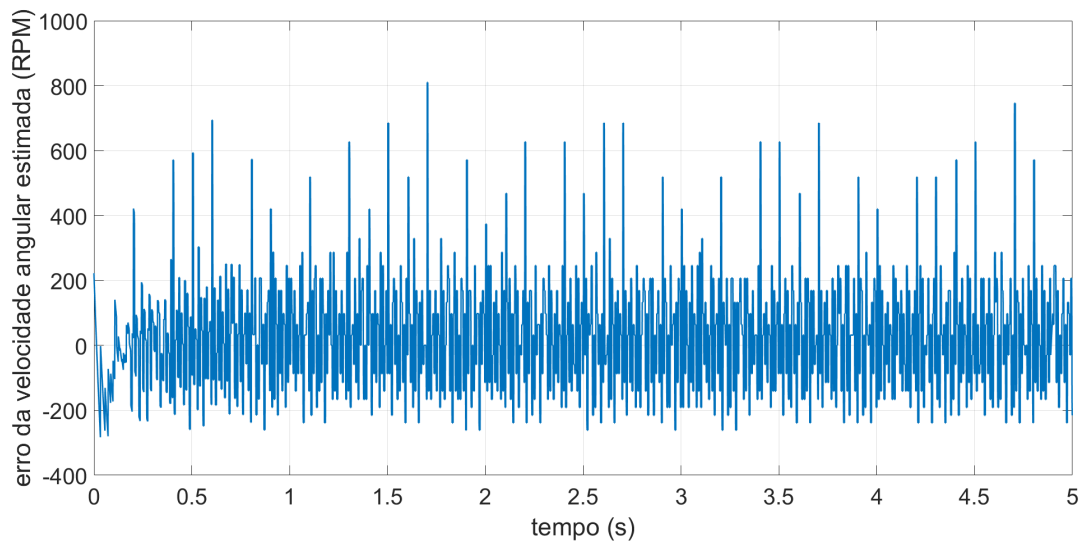
Figura 42 – Velocidade medida e estimada pelo método de Hägglund, em função do tempo.



Fonte: Próprio autor.

O gráfico do erro em função do tempo é dado pela Figura 43. Onde nota-se também a remoção do erro de estado estacionário.

Figura 43 – Erro entre a velocidade medida e a estimada pelo método de Hägglund, em função do tempo.



Fonte: Próprio autor.

Onde o índice de erro ITSE é dado pela equação 4.6.

$$ITSE_{HAG} = 4,1261 \cdot 10^5 \quad (4.6)$$

4.2.3 Método de Smith

O método de Smith para modelagem de um processo, diferente dos métodos anteriores, não necessita de uma reta de auxílio para determinação dos coeficientes e, portanto, diminui os possíveis erros gerados nesta etapa. O código do método utilizado é apresentado na Figura 44.

Como resultado obteve-se a função de transferência da equação 4.7.

$$G_p = \frac{1754e^{-0,0455s}}{0,0107s + 1} \quad (4.7)$$

Figura 44 – Código para identificação paramétrica segundo o método de Smith.

```

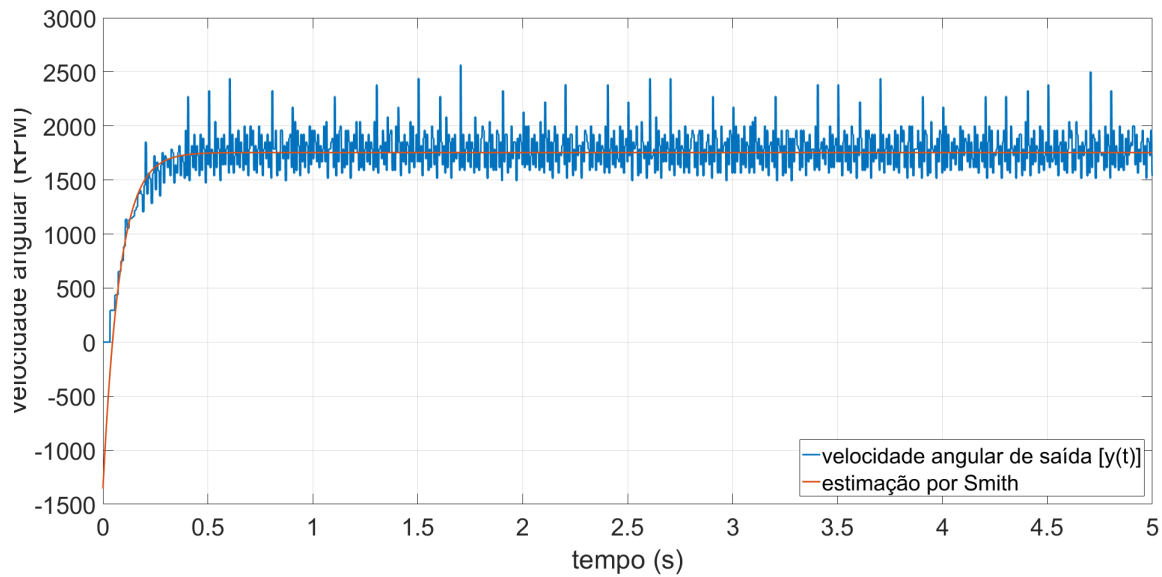
1 %% Smith
2
3 t1Smith = 0.072; %vizualmente no gráfico 0.283*y(inf)
4 t2Smith = 0.125; %vizualmente no gráfico 0.632*y(inf)
5 KpSmith = Yss;
6 tauSmith = 1.5*(t2Smith-t1Smith);
7 thetaSmith = t2Smith-tauSmith;
8
9 estimadoSmith = zeros(pontos,1);
10
11 for i = 1:pontos
12     estimadoSmith(i) = KpSmith*(1-exp(-(tempo(i)-thetaSmith)/tauSmith));
13 end
14 createfigure2(tempo, saida, tempo, estimadoSmith);
15 grid on
16 xlabel('tempo_(s)')
17 ylabel('velocidade_angular_(RPM)')
18 legend('velocidade_angular_de_saida_[y(t)]','estimação_por_Smith', ...
19 'Location','SouthEast')
20
21 try
22     for i=1:total
23         erro(i) = saida(i)-estimadoSmith(i);
24     end
25 end
26 itseSmith = tempo.*(erro.^2);
27 ITSESmith = sum(itseSmith)*0.001
28
29 createfigure(tempo, erro)
30 grid on
31 xlabel('tempo_(s)')
32 ylabel('erro_da_velocidade_angular_estimada_(RPM)')

```

Fonte: Próprio autor.

No qual a resposta deste sistema à entrada degrau é dada pelo gráfico da Figura 45.

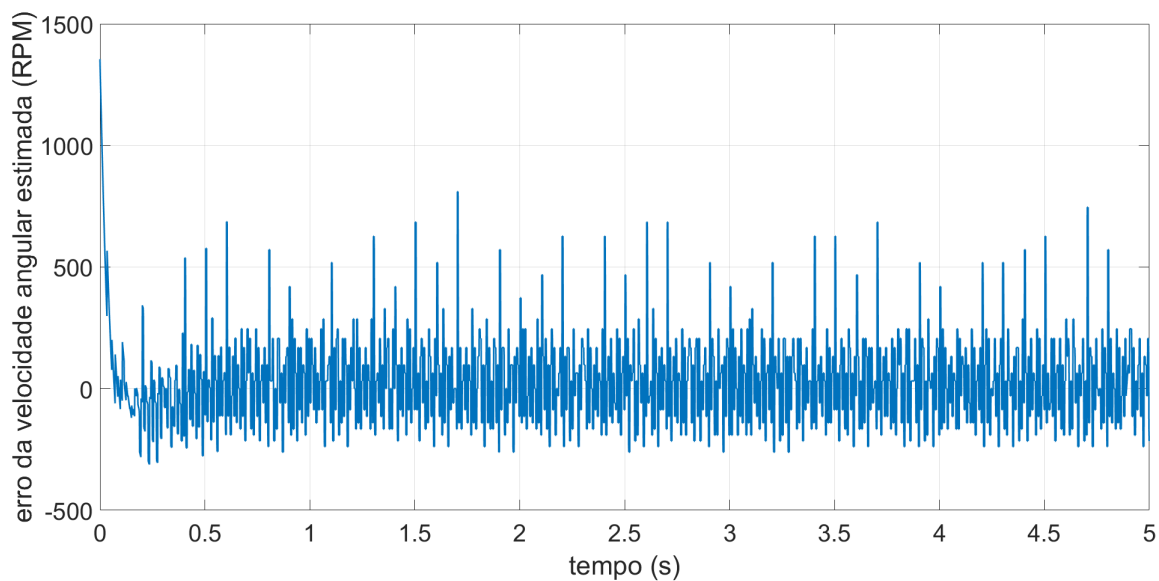
Figura 45 – Velocidade medida e estimada pelo método de Smith, em função do tempo.



Fonte: Próprio autor

Nota-se entretanto, pela observação do gráfico da Figura 45, que assim como nos métodos anteriores não há um erro de regime permanente da resposta do sistema em função do tempo, portanto, mesmo sem o auxílio da reta o sistema mantém o mesmo desempenho que os anteriores para o estado estacionário. O gráfico do erro em função do tempo é apresentado na Figura 46, onde pode-se observar mais claramente as divergências entre o sinal medido e o estimado.

Figura 46 – Erro entre a velocidade medida e a estimada pelo método de Smith, em função do tempo.



Fonte: Próprio autor.

Onde o índice de erro ITSE é dado pela equação 4.8.

$$ITSE_{Smith} = 4,1261.10^5. \quad (4.8)$$

4.2.4 Método de Sundaresan/Krishnaswamy

A identificação paramétrica de um processo pelo método de Sundaresan/Krishnaswamy é muito similar ao método de Smith, ou seja, não necessita de uma reta de auxílio para determinação dos coeficientes, dependendo apenas de pontos relacionados ao valor do regime permanente, portanto, diminui os possíveis erros causados pela inexatidão na geração da reta, mas exige leituras mais exatas dos pontos notáveis para a modelagem. O código do método utilizado é apresentado Figura 47.

Figura 47 – Código para identificação paramétrica segundo o método de Sundaresan/Krishnaswamy.

```

1 %% Sundaresan/Krishnaswamy
2
3 t1SK = 0.073; %vizualmente no gráfico 0.353*y(inf)
4 t2SK = 0.208; %vizualmente no gráfico 0.853*y(inf)
5 KpSK = Yss;
6 tauSK = 0.67*(t2SK-t1SK);
7 thetaSK = 1.3*t1SK-0.29*t2SK;
8
9 estimadoSK = zeros(pontos,1);
10
11 for i = 1:pontos
12     estimadoSK(i) = KpSK*(1-exp(-(tempo(i)-thetaSK)/tauSK));
13 end
14 createfigure2(tempo, saida, tempo, estimadoSK);
15 grid on
16 xlabel('tempo_(s)')
17 ylabel('velocidade_angular_(RPM)')
18 legend('velocidade_angular_de_saida_[y(t)]','estimação_por_Sundaresan/Krishnaswamy', ...
19 'Location','SouthEast')
20
21 try
22     for i=1:total
23         erro(i) = saida(i)-estimadoSK(i);
24     end
25 end
26 itseSK = tempo.*(erro.^2);
27 ITSESK = sum(itseSK)*0.001
28
29 createfigure(tempo, erro)
30 grid on
31 xlabel('tempo_(s)')
32 ylabel('erro_da_velocidade_angular_estimada_(RPM)')

```

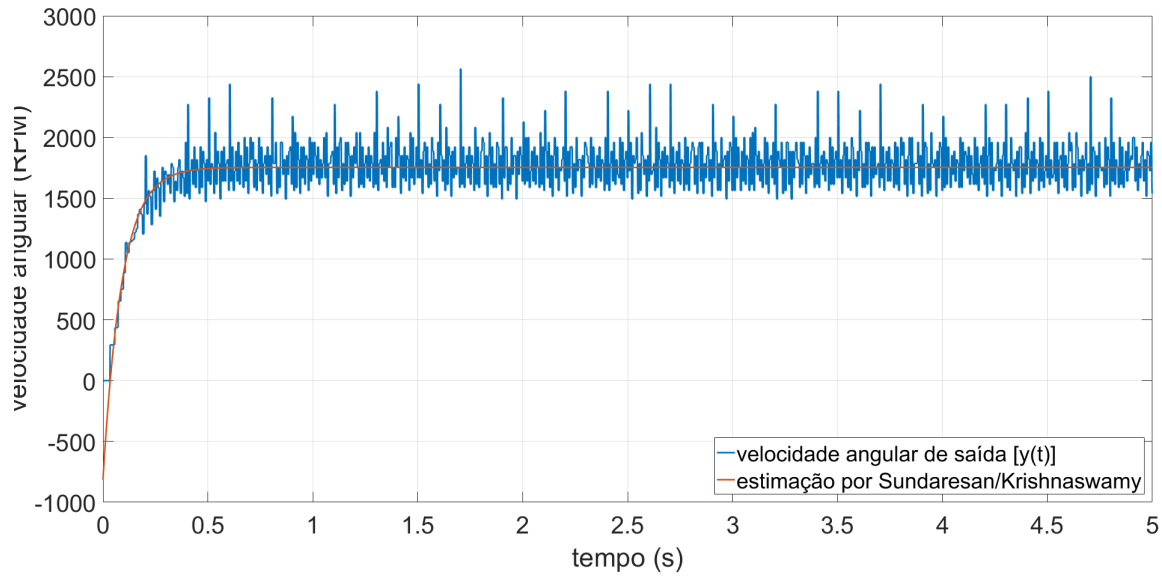
Fonte: Próprio autor.

Como resposta obteve-se a função de transferência modelada, conforme mostra a equação 4.9.

$$G_p = \frac{1754e^{-0,0346s}}{0,0905s + 1} \quad (4.9)$$

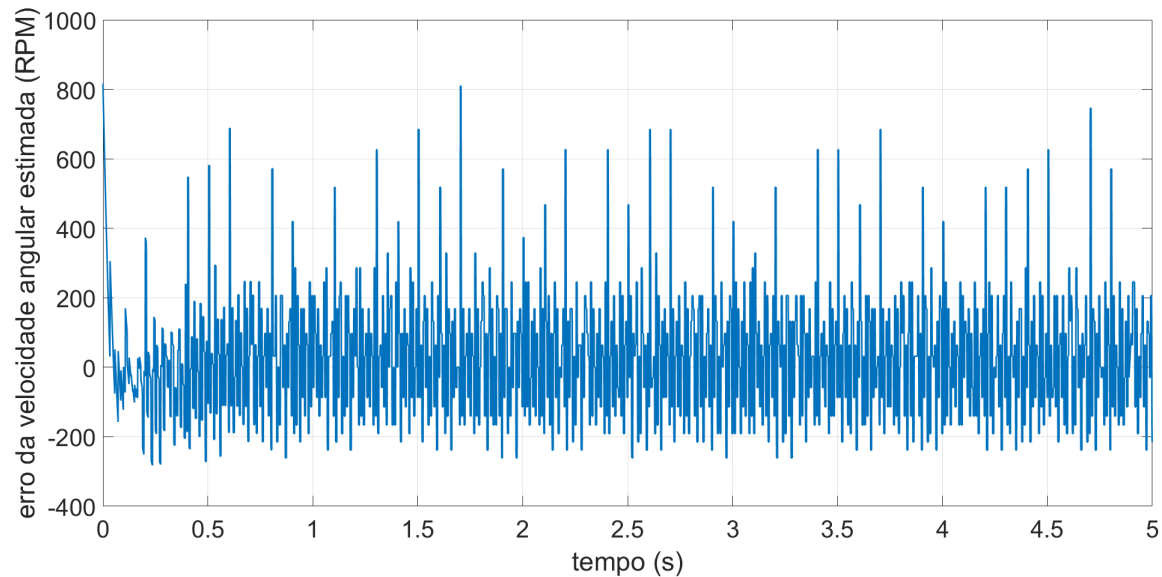
O sinal de saída estimado pelo método de Sundaresan/Krishnaswamy com uma entrada degrau é dado pela Figura 48.

Figura 48 – Velocidade medida e estimada pelo método de Sundaresan/Krishnaswamy, em função do tempo.



Fonte: Próprio autor.

Figura 49 – Erro entre a velocidade medida e a estimada pelo método de Sundaresan/Krishnaswamy, em função do tempo.



Fonte: Próprio autor.

Este resultado, assim como os métodos anteriores apresenta uma resposta ao degrau semelhante o sinal obtido pela leitura do sensor, lembrando que este último possui um ruído adicional ao seu valor real, entretanto também não há erro de estado estacionário neste método. O gráfico do erro em função do tempo é apresentado na Figura 49.

Nota-se assim que o sinal do erro se dá basicamente pelo transiente e ruído de medição, onde o índice ITSE é dado pela equação 4.10.

$$ITSE_{SK} = 4,1258.10^5. \quad (4.10)$$

4.3 Análise dos resultados dos métodos de primeira ordem

A Tabela 8 resume os resultados obtidos em função de cada método.

Tabela 8 – Funções de transferência e ITSE para cada método de identificação paramétrica de primeira ordem.

	Função de transferência	ITSE
Ziegler/Nichols	$\frac{1754e^{-0,0133s}}{0,1169s + 1}$	$4,1266.10^5$
Hägglund	$\frac{1754e^{-0,0133s}}{0,1117s + 1}$	$4,1261.10^5$
Smith	$\frac{1754e^{-0,0455s}}{0,0107s + 1}$	$4,1298.10^5$
Sundaresan/Krishnaswamy	$\frac{1754e^{-0,0346s}}{0,0905s + 1}$	$4,1258.10^5$

Fonte: Próprio autor.

Nota-se, portanto, que todos os métodos de identificação paramétrica removem o erro de regime permanente, com ganhos idênticos entre todos os métodos. Os tempos de atraso são também iguais para os métodos de Ziegler/Nichols e Hägglund, devido ao uso da reta de auxílio, mas diferem dos métodos de Smith e Sundaresan/Krishnaswamy, devido a utilização de suas próprias técnicas de determinação do atraso. Todos os portanto diferem no valor do parâmetro τ .

Dentre os métodos discutidos, o de Sundaresan/Krishnaswamy possui o melhor resultado e devido a ausência de uma reta de auxílio torna o processo de identificação mais rápido de ser executado, se mostrando o melhor resultado, entretanto todos métodos se mostraram bastante semelhantes em suas respostas.

4.4 Métodos experimentais determinísticos de segunda ordem

São tratados nesta seção os métodos gráficos de identificação de sistemas de segunda ordem, baseados na resposta ao degrau por meio dos índices de desempenho apresentados na Seção 4.4.1 e o por (MOLLENKAMP, 1988) na Seção 4.4.2.

4.4.1 Método por índices de desempenho

O primeiro método de modelagem de um processo de segunda ordem é realizado por meio da análise de índices de desempenho. Este é um processo reverso ao tradicional projeto de controle por índices de desempenho, ou seja, por meio de índices específicos de desempenho conhecidos através da análise gráfica, deduz-se os zeros e polos no lugar das raízes e, por consequência, a função de transferência do sistema. O código para execução deste método é apresentado na Figura 50.

Figura 50 – Código para identificação paramétrica segundo o método de índices de desempenho.

```

1 %% Método por índices de desempenho
2 Yss = 1754;
3 tp = 0.451;
4
5 xi = sqrt((log(Yss-1))^2/(pi^2+(log(Yss-1))^2));
6 B = sqrt(1-xi^2);
7 fi = atan(xi/B);
8 omegan = pi/(tp*B);
9
10 estimadoMID = zeros(pontos,1);
11
12 for i = 1:pontos
13     estimadoMID(i) = Yss-(Yss/B)* ...
14     exp(-xi*omegan*tempo(i))*cos(omegan*B*tempo(i)-fi);
15 end
16 createfigure2(tempo, saida, tempo, estimadoMID);
17 grid on
18 xlabel('tempo_(s)')
19 ylabel('velocidade_angular_(RPM)')
20 legend('velocidade_angular_de_saida_[y(t)]','estimação_por_índices_de_desempenho', ...
21 'Location','SouthEast')
22
23 try
24     for i=1:total
25         erro(i) = saida(i)-estimadoMID(i);
26     end
27 end
28 itseMID = tempo.*(erro.^2);
29 ITSEMID = sum(itseMID)*0.001
30
31 createfigure(tempo, erro)
32 grid on
33 xlabel('tempo_(s)')
34 ylabel('erro_da_velocidade_angular_estimada_(RPM)')

```

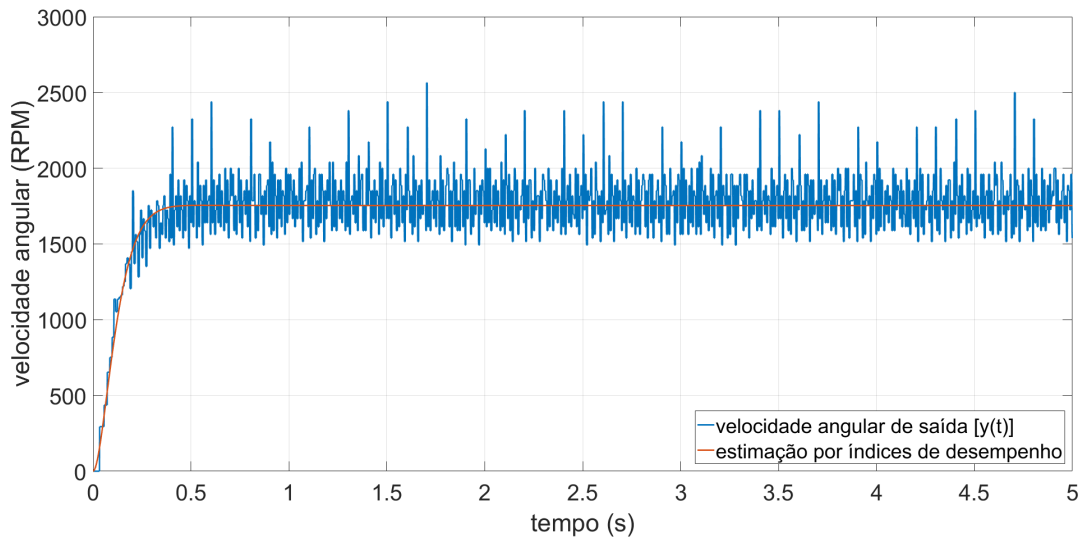
Fonte: Próprio autor.

Define-se a função de transferência modelada conforme a equação 4.11.

$$G_p = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} = \frac{216,2602}{s^2 + 13,5555s + 216,2602} \quad (4.11)$$

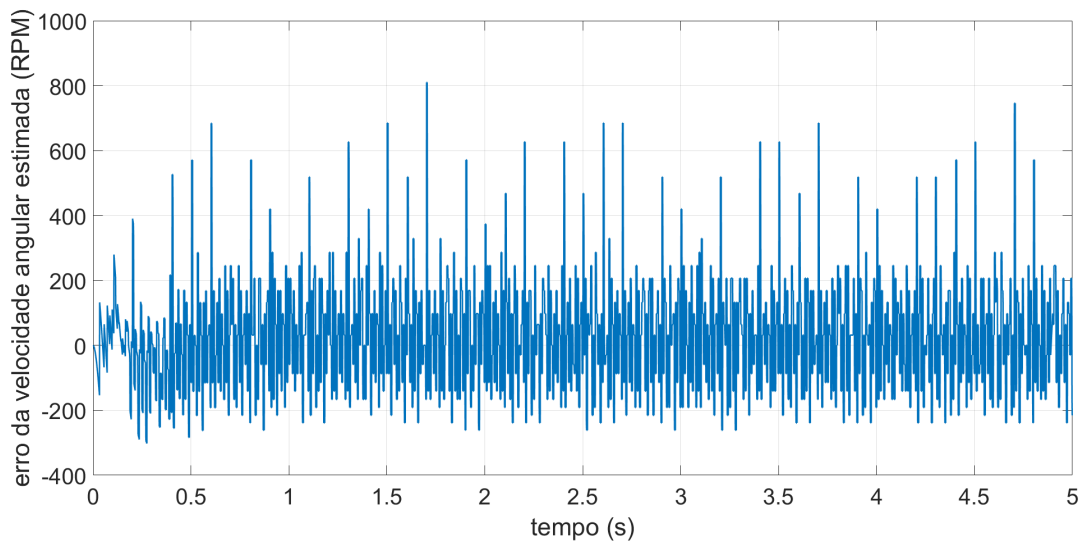
Desta forma, obteve-se a resposta do sistema identificado pela função de transferência da equação 4.11 ao degrau, resultando no sinal da Figura 51.

Figura 51 – Velocidade medida e estimada pelo método de índices de desempenho, em função do tempo.



Fonte: Próprio autor.

Figura 52 – Erro entre a velocidade medida e a estimada pelo método de índices de desempenho, em função do tempo.



Fonte: Próprio autor.

O gráfico do erro em função do tempo é apresentado na Figura 52. Nota-se o desempenho aparente do método de identificação de segundo grau é semelhante, mas no transiente do sinal de erro percebe-se uma curva em S característica de um sistema de segundo grau em comparação.

O índice de erro ITSE é apresentado na equação 4.12.

$$ITSE_{MID} = 4,1263 \cdot 10^5. \quad (4.12)$$

4.4.2 Método de Mollenkamp

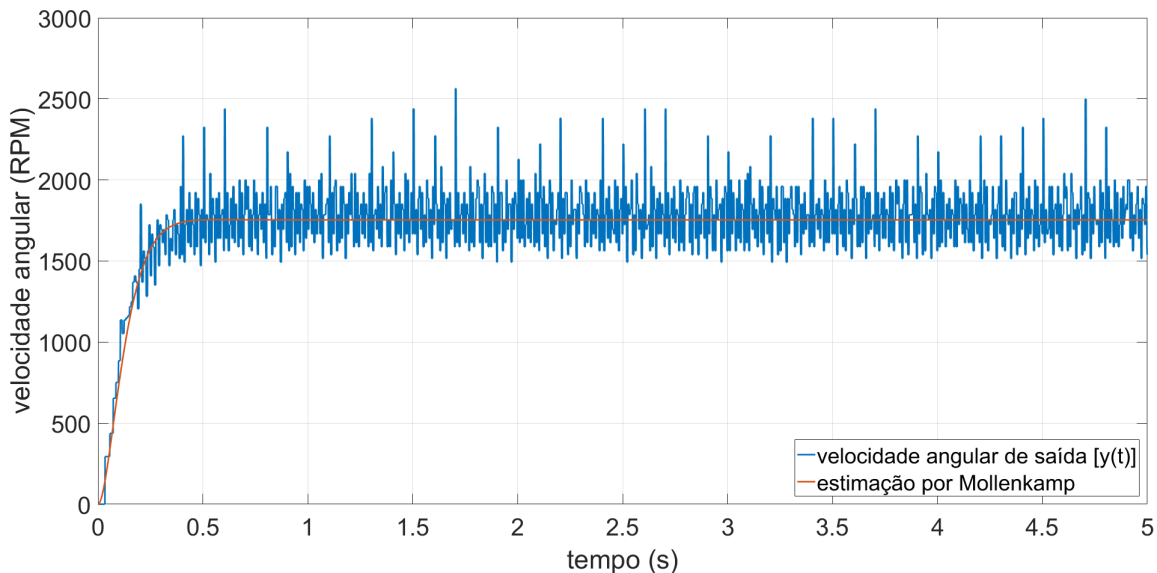
O método de Mollenkamp, conforme apresentado na Seção 2.5.2 foi executado conforme o código apresentado na Figura 54.

Como resultado da aplicação deste metodologia obteve-se a função de transferência conforme a equação 4.13.

$$G_p = \frac{193,3841 \cdot e^{-0,0398}}{s^2 + 12,8185s + 193,3841} \quad (4.13)$$

Submetendo-se o modelo do sistema obtido à entrada degrau, obteve-se o sinal da Figura 53.

Figura 53 – Velocidade medida e estimada pelo método de Mollenkamp, em função do tempo.



Fonte: Próprio autor.

Figura 54 – Código para identificação paramétrica segundo o método de Mollenkamp.

```

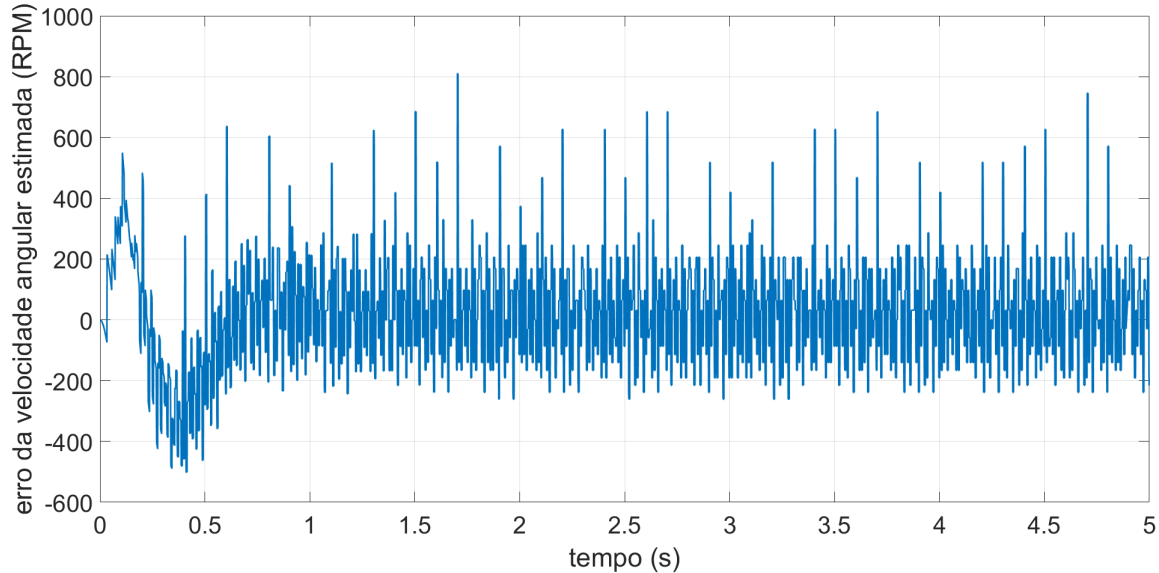
1 t1Mollenkamp = 0.034; % 15% do máximo
2 t2Mollenkamp = 0.066; % 45% do máximo
3 t3Mollenkamp = 0.166; % 75% do máximo
4 x = (t2Mollenkamp-t1Mollenkamp)/(t3Mollenkamp-t1Mollenkamp);
5 xiMollenkamp = (0.0805-5.547*(0.475-x)^2)/(x-0.356);
6 if (xi < 1)
7     f2 = 0.708*2.811^xi;
8 else
9     f2 = 2.6*xi-0.60;
10 end
11 omegan = f2/(t3Mollenkamp-t1Mollenkamp); f3 = 0.922*1.66^xi;
12 thetaMollenkamp = t2Mollenkamp-(f3/omegan);
13 tauMollenkamp1 = (xi+sqrt(xi^2-1))/omegan;
14 tauMollenkamp2 = (xi-sqrt(xi^2-1))/omegan;
15 if xi<1
16     disp('fator_de_amortecimento') xi
17     disp('frequência_natural') omegan
18     disp('atraso_de_transporte') thetaMollenkamp
19     sys = tf([Yss*omegan^2],[1 2*xi*omegan omegan^2]);
20     tempoTeste = 0:0.0001:5;
21     [estimadoMollenkamp, tempoMollenkamp] = step(sys, tempoTeste);
22     createfigure2(tempo, saida, tempoMollenkamp, estimadoMollenkamp); grid on
23     xlabel('tempo_(s)') ylabel('velocidade_angular_(RPM)')
24     legend('velocidade_angular_de_saidá_[y(t)]','estimação_por_Mollenkamp', ...
25     'Location','SouthEast')
26     try
27         for i=1:total
28             for j=1:lenght(tempoMollenkamp)
29                 if (tempo(i) == tempoMollenkamp(j))
30                     erro(i) = velocidade(i)-estimadoMollenkamp(j);
31                 end
32             end
33         end
34     end
35     itseMollenkamp = tempo.*(erro.^2);
36     ITSEMollenkamp = sum(itseMollenkamp)*0.001
37     createfigure(tempo, erro) grid on
38     xlabel('tempo_(s)')
39     ylabel('erro_da_velocidade_angular_estimada_(RPM)')
40 else
41     disp('constante_de_tempo_1') tauMollenkamp1
42     disp('constante_de_tempo_2') tauMollenkamp2
43     disp('atraso_de_transporte') thetaMollenkamp
44     sys = zpk([],[-1/tauMollenkamp1 -1/tauMollenkamp2],[Yss]);
45     s = zpk('s')
46     d = exp(-thetaMollenkamp*s)*sys
47     [estimadoMollenkamp, tempoMollenkamp] = step(sys, tempo);
48     createfigure2(tempo, saida, tempoMollenkamp, estimadoMollenkamp); grid on
49     xlabel('tempo_(s)') ylabel('velocidade_angular_(RPM)')
50     legend('velocidade_angular_de_saidá_[y(t)]','estimação_por_Mollenkamp', ...
51     'Location','SouthEast')
52     try
53         for i=1:total
54             erro(i) = saida(i)-estimadoMollenkamp(i);
55         end
56     end
57     itseMollenkamp = tempo.*(erro.^2);
58     ITSEMollenkamp = sum(itseMollenkamp)*0.001
59     createfigure(tempo, erro)
60     grid on
61     xlabel('tempo_(s)')
62     ylabel('erro_da_velocidade_angular_estimada_(RPM)')
63 end

```

Fonte: Próprio autor.

O gráfico do erro em função do tempo é apresentado na Figura 55.

Figura 55 – Erro entre a velocidade medida e a estimada pelo método de Mollenkamp, em função do tempo.



Fonte: Próprio autor.

Nota-se pela Figura 55 que o erro tanto de transiente como de estado estacionário foi , resultando em um índice de erro ITSE dado pela equação 4.14.

$$ITSE_{Mollenkamp} = 4,2043.10^5. \quad (4.14)$$

4.4.3 Análise dos resultados dos métodos de segunda ordem

A Tabela 9 apresenta um resumo dos resultados da Seção 4.4.

Tabela 9 – Funções de transferência e ITSE para cada método de identificação paramétrica de segunda ordem.

	Função de transferência	ITSE
Índices de desempenho	216,2602	4,1263.10 ⁵
	$s^2 + 13,5555s + 216,2602$	
Mollenkamp	$193,3841.e^{-0,0398}$	4,2043.10 ⁵
	$s^2 + 12,8185s + 193,3841$	

Fonte: Próprio autor.

Dentre os métodos determinísticos de segunda ordem nota-se que ambos os métodos apresentam um resultado semelhante aos de primeira ordem, isso se deve ao fato de que ainda

que o esquema elétrico do motor de corrente contínua a ser de segunda ordem, devido ao baixo valor da indutância intrínseca a este, pode-se fielmente aproximar-se para um modelo de primeira ordem.

Portanto, conclui-se nesta seção que ainda que usualmente se adote um modelo de primeiro grau para descrição e modelagem de um motor elétrico, este pode ser também modelado como de segundo grau sem alteração no desempenho.

4.5 Método de mínimos quadrados

O método apresentado nesta seção é baseado na identificação paramétrica por mínimos quadrados. Este método é baseado em média-móvel autorregressiva de segunda ordem (ARMA), pois sabe-se que o sistema matemático estudado é de segunda ordem. O método ARMA de identificação do sistema pode ser obtido como mostra a Figura 56.

Figura 56 – Código para identificação paramétrica segundo o método dos mínimos quadrados.

```

1 %% Identificação de sistemas por mínimos quadrados ARMA 2º grau
2
3 yARMA2 = zeros(pontos-2,1);
4 xARMA2a = zeros(pontos-2,1);
5 xARMA2b = zeros(pontos-2,1);
6 uARMA2a = zeros(pontos-2,1);
7 estimadoARMA2 = zeros(pontos,1);
8
9 for i=1:(pontos-2)
10     yARMA2(i) = saida(i+2);
11     xARMA2a(i) = saida(i+1);
12     xARMA2b(i) = saida(i);
13     uARMA2a(i) = entrada(i+1);
14 end
15
16 FARMA2 = [xARMA2a xARMA2b uARMA2a];
17
18 thetaARMA2 = (FARMA2'*FARMA2)\(FARMA2'*yARMA2);
19
20 estimadoARMA2(1) = 0;
21 estimadoARMA2(2) = thetaARMA2(1)*estimadoARMA2(1)+thetaARMA2(3)*uARMA2a(1);
22 for i=3:(pontos)
23     estimadoARMA2(i) = thetaARMA2(1)*estimadoARMA2(i-1)+ ...
24         thetaARMA2(2)*estimadoARMA2(i-2)+thetaARMA2(3)*entrada(i-1);
25 end
26
27 createfigure2(tempo,saida,tempo,estimadoARMA2);
28 grid on
29 xlabel('tempo_(s)')
30 ylabel('velocidade_angular_(rad/s)')
31 legend('velocidade_angular_de_saida_[y(t)]','estimação_por_ARMA_de_2º_grau', ...
32 'Location','SouthEast')

```

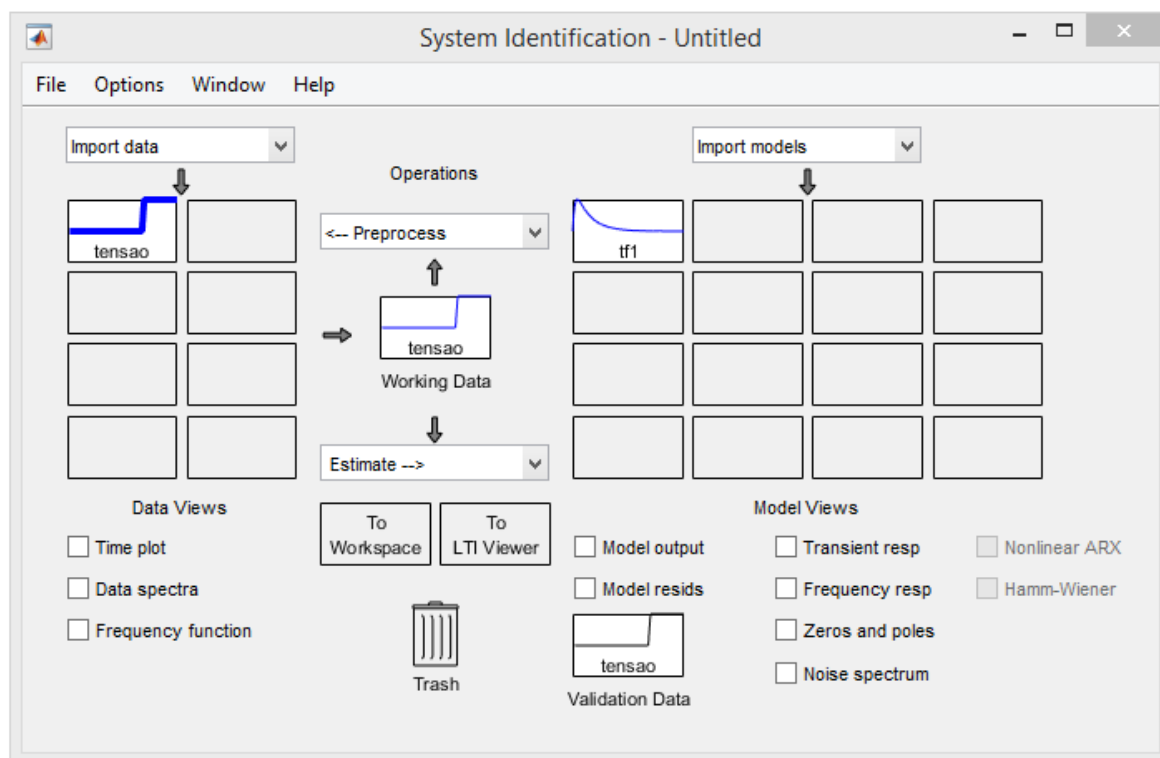
Fonte: Próprio autor.

Desta forma um novo vetor *thetaARMA* encontrado, apresentado no formado $[a_1; a_2; b_1; b_2]$ (PHILLIPS; NAGLE, 2007), i.e., a função de transferência é escrita conforme a equação 4.15.

$$G_p = \frac{4698000}{s^2 + 419,2s + 31560} \quad (4.15)$$

Outra forma de se fazer a identificação paramétrica do sistema pelo método de mínimos quadrados se dá pela utilização do *APP System Identification* do próprio *Matlab*, o qual abre a interface gráfica *System Identification* (identificação do sistema) mostrada na Figura 57.

Figura 57 – Interface do *APP System Identification* com seus respectivos parâmetros.



Fonte: Próprio autor.

Na aba *import data* (importar dados) foi selecionada a opção *time domain data* (dados no domínio do tempo), no qual abre a segunda interface gráfica (*Import Data*) apresentada na Figura 58. Nesta nova janela coloca-se o sinal de tensão como *input* (entrada) e o sinal de velocidade como *output* (saída) e seleciona-se o botão *import* (importar), que adiciona o sinal à janela *System Identification*.

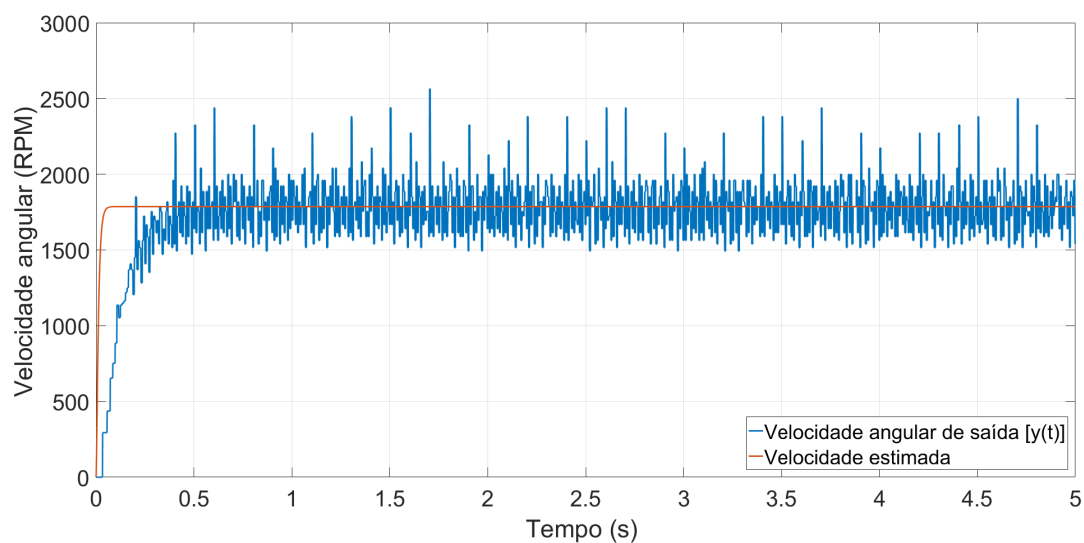
Em seguida, na aba *Estimate-->* (estimar) seleciona-se a opção *Transfer Function Models* que abre a interface chamada *Transfer Functions* também apresentada na Figura 58. Nesta janela escolhe-se o número de polos e zeros quadros *Number of poles* e *Number of zeros*, respectivamente.

Figura 58 – Interface do APP *System Identification* com seus respectivos parâmetros.

Fonte: Próprio autor.

Após selecionar o botão *Estimate* um processo recursivo não linear baseada em ARMA faz a identificação paramétrica do sistema segundo o modelo, resultando na mesma função de transferência apresentada na equação 4.15. A Figura 59 apresenta a resposta deste sistema estimado pelo método ARMA de segunda ordem em comparação com a saída em função do tempo.

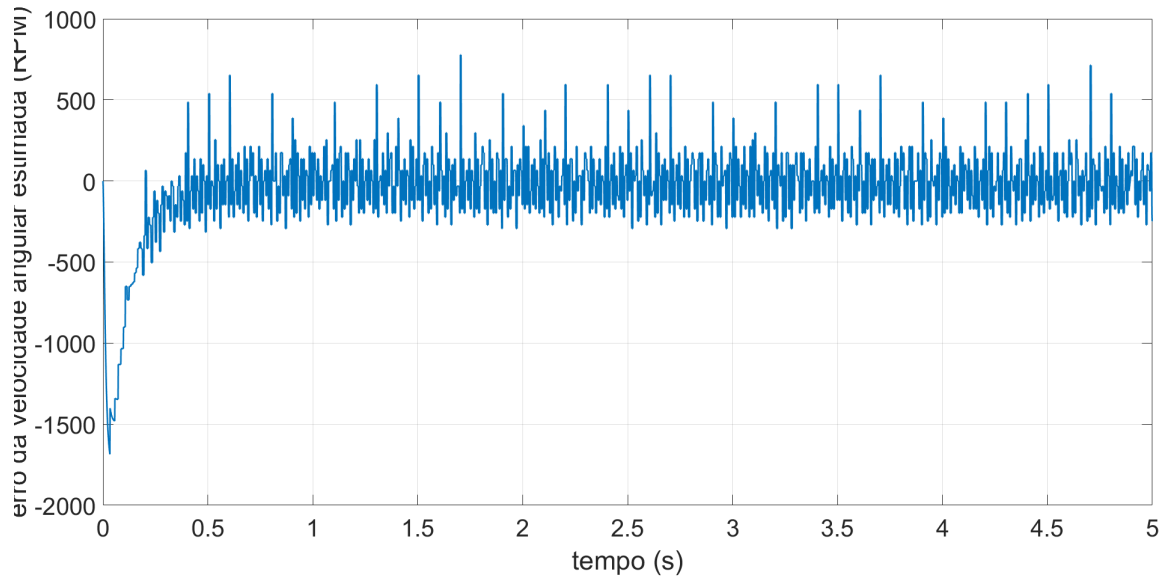
Figura 59 – Velocidade medida e estimada pelo método ARMA de segunda ordem, em função do tempo.



Fonte: Próprio autor.

O gráfico do erro em função do tempo é apresentado na Figura 60, onde nota-se que não há um erro estacionário, mas o erro de transiente cresce.

Figura 60 – Erro entre a velocidade medida e a estimada pelo método ARMA de segunda ordem, em função do tempo.



Fonte: Próprio autor.

Como resultado obteve-se o menor índice de erro ITSE, conforme mostra a equação 4.16.

$$ITSE_{ARMA2} = 4,1474 \cdot 10^5. \quad (4.16)$$

4.6 Conclusão do capítulo

Conclui-se neste capítulo que o método ARMA de 2ª ordem possui um desempenho muito parecido com o de 1ª ordem e portanto a utilização de ambos os modelos apresentam desempenhos semelhantes, por este motivo o modelo matemático utilizado é composto por todos os parâmetros do motor elétrico.

Além disto a técnica ARMA permite a aplicação em tempo real, justamente pela possibilidade de utilização como um sistema completamente automatizado. Tornando possível a adaptação do sistema de controle como por exemplo em um controle adaptativo. Comprovou-se também que o modelo matemático, obtido de maneira separada, com uma análise de componente a componente, possui um erro de ordem acima de qualquer outro método e portanto poderá afetar o desempenho do sistema de controle.

Entretanto para o modelo matemático remodelado após a montagem completa do sistema obtém-se um resultado melhor, se tornando uma ótima técnica de identificação paramétrica, ainda que esta seja bastante trabalhosa custando maior tempo de análise que por exemplo uma técnica completamente computacional, uma vez que esta é a única técnica que permite a construção de um observador de estados para estimar o torque do motor sem a necessidade de um sensor específico para isso.

A Tabela 10 apresenta um resumo de todos os métodos apresentados neste capítulo.

Tabela 10 – Funções de transferência e ITSE para os métodos de identificação paramétrica de primeira e segunda ordem gráficos, determinísticos, através do modelo matemático e ARMA.

	Função de transferência	ITSE
Modelo Matemático	-	$1,7004 \cdot 10^6$
Ziegler/Nichols	$\frac{1754e^{-0,0133s}}{0,1169s + 1}$	$4,1266 \cdot 10^5$
Hägglund	$\frac{1754e^{-0,0133s}}{0,1117s + 1}$	$4,1261 \cdot 10^5$
Smith	$\frac{1754e^{-0,0455s}}{0,0107s + 1}$	$4,1261 \cdot 10^5$
Sundaresan/Krishnaswamy	$\frac{1754e^{-0,0346s}}{0,0905s + 1}$	$4,1258 \cdot 10^5$
Índices de desempenho	$\frac{216,2602}{s^2 + 13,5555s + 216,2602}$	$4,1263 \cdot 10^5$
Mollenkamp	$\frac{193,3841 \cdot e^{-0,0398s}}{s^2 + 12,8185s + 193,3841}$	$4,2043 \cdot 10^5$
ARMA 2ª ordem	$\frac{4698000}{s^2 + 419,2s + 31560}$	$4,1474 \cdot 10^5$

Fonte: Próprio autor.

A seguir, o Capítulo 5 apresenta os resultados referentes ao método de leitura de velocidade, bem como os do controle de velocidade, resultados experimentais para a remodelagem do sistema e aplicação do filtro de Kalman para observar o torque do sistema.

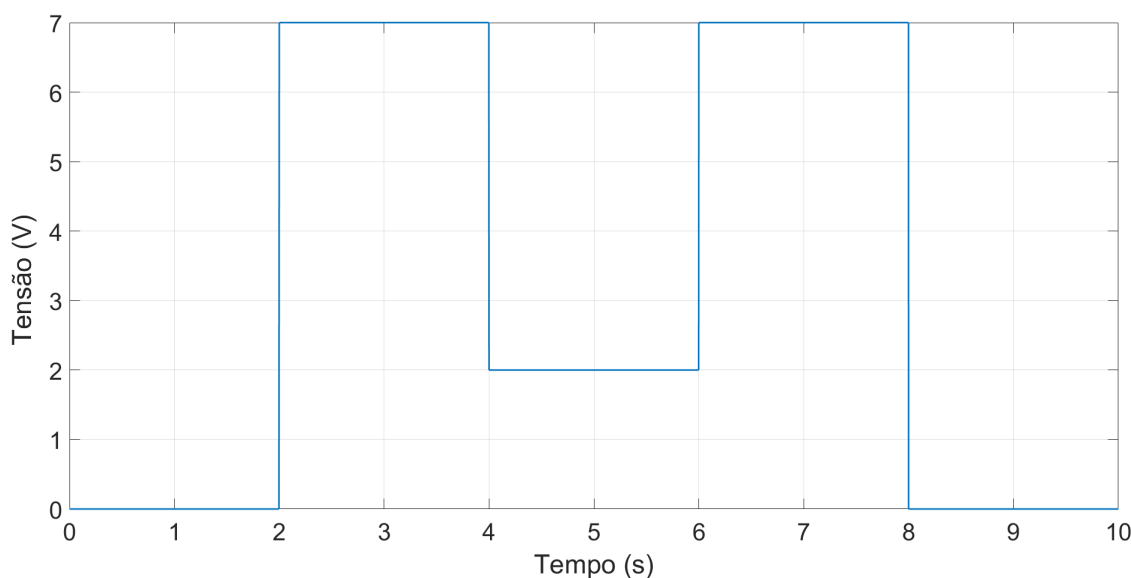
5 Resultados

Neste capítulo são apresentados os resultados dos métodos de medição de velocidade, aplicação do controle PID com e sem o filtro de Kalman, além dos resultados parciais do controle de torque.

5.1 Medição de velocidade

Para análise dos dois métodos de leitura da velocidade propostos, foi gerado um sinal de entrada para alimentação do sistema, conforme mostra a Figura 61.

Figura 61 – Valor do sinal de controle não normalizado em função do tempo.

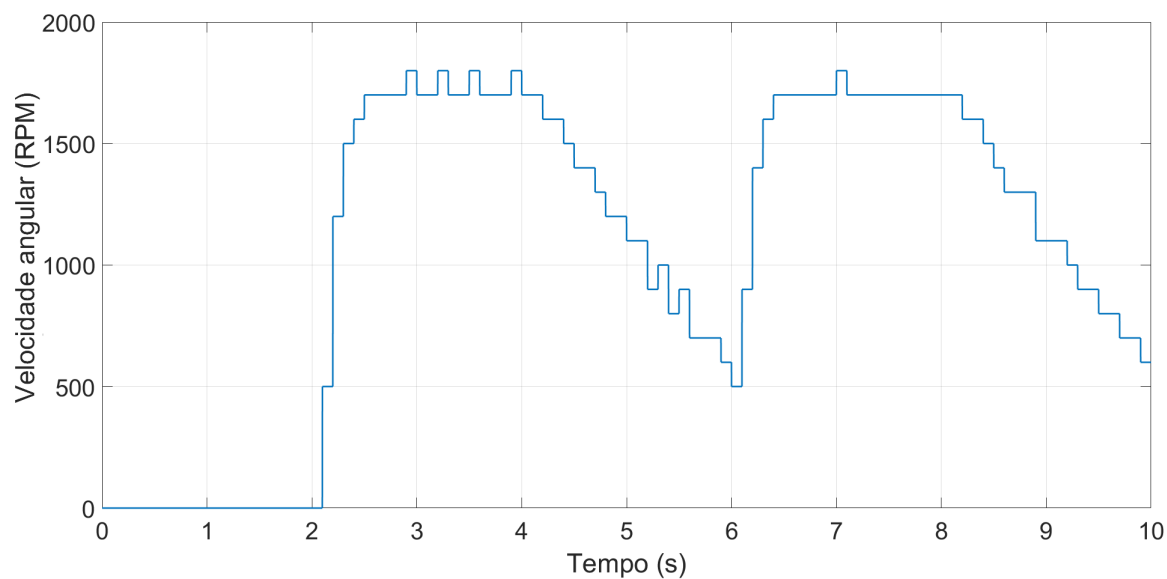


Fonte: Próprio autor.

Este sinal, gerado com uma soma de ondas quadradas serve de entrada para os dois métodos de medição da velocidade e possibilita a análise de velocidade de resposta da leitura de cada sistema, além da observação da exatidão e precisão. Como resultado para a entrada descrita, obteve-se o gráfico da Figura 62 para o método de leitura baseado na velocidade média, Seção 3.5.1.

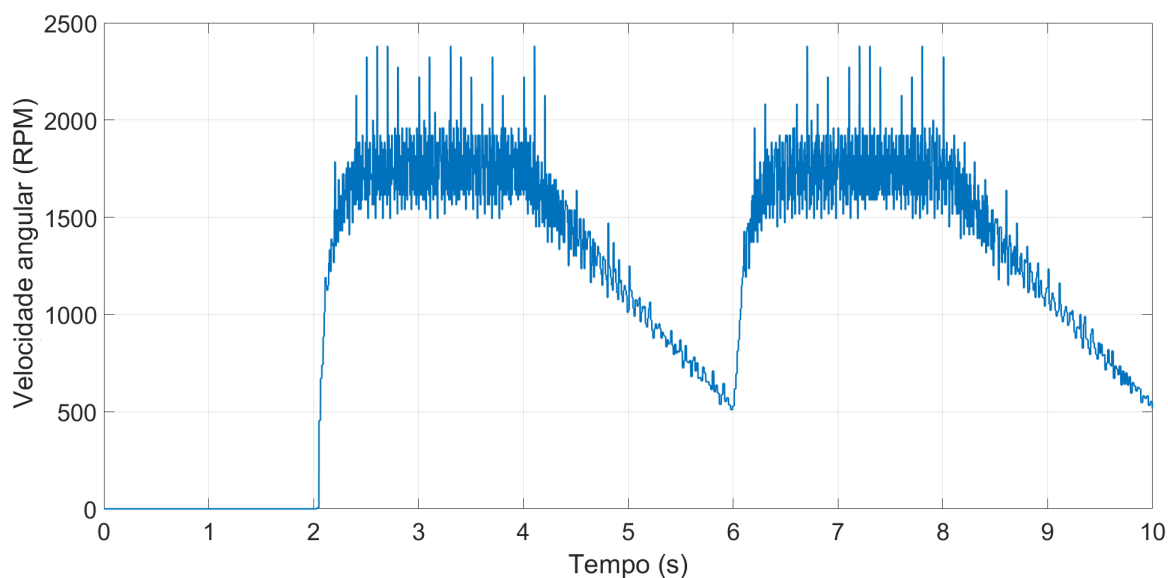
Nota-se que este método apresenta um erro de quantização elevado e baixo ruído de leitura. A Figura 63 apresenta o gráfico do sinal resultante pela execução do método baseado na velocidade instantânea, Seção 3.5.2.

Figura 62 – Velocidade de rotação do motor obtida pela leitura do sensor pelo método da velocidade média, em função do tempo.



Fonte: Próprio autor.

Figura 63 – Velocidade de rotação do motor obtida pela leitura do sensor pelo método da velocidade instantânea, em função do tempo.

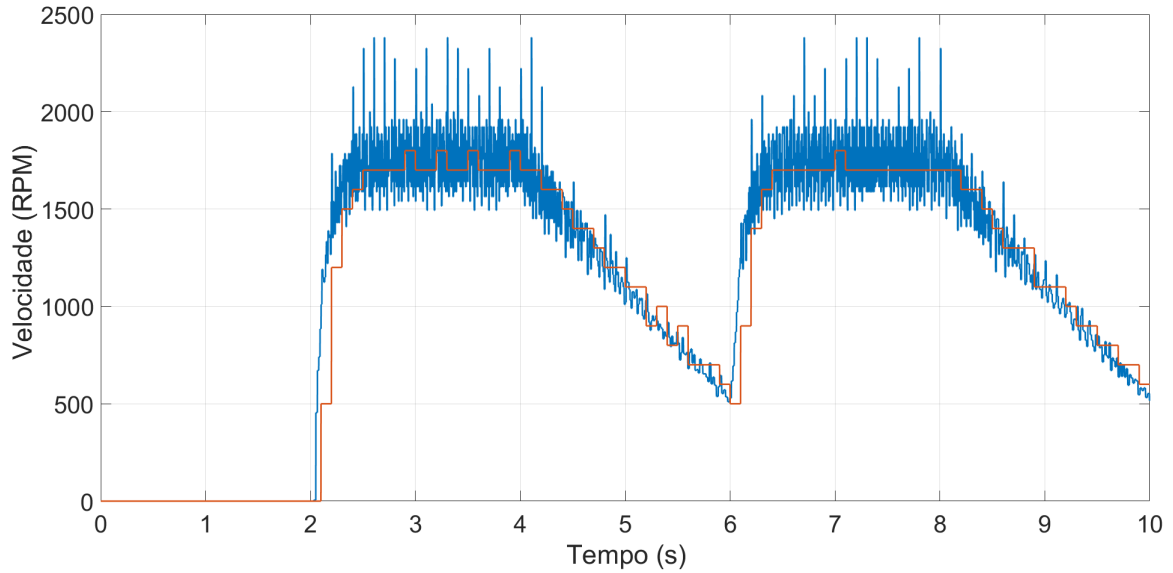


Fonte: Próprio autor.

Percebe-se que o método da velocidade instantânea apresenta um ruído no sinal de medida muito alto, entretanto com observação da Figura 64, nota-se que este sinal também apresenta uma resposta significativamente mais rápida que o primeiro método, se adaptando

melhor às transições do sistema. Em contrapartida o método da velocidade média apresenta maior precisão para velocidades estacionárias.

Figura 64 – Comparação entre as velocidades obtidas pela leitura do sensor pelo primeiro e segundo método, em função do tempo.



Fonte: Próprio autor.

Portanto, conclui-se com esta análise que dada a possibilidade de utilização de filtros e observadores que permitem uma diminuição dos efeitos dos ruídos, escolheu-se trabalhar com o sistema apresentado pelo segundo método, justamente por este apresentar uma resposta mais rápida.

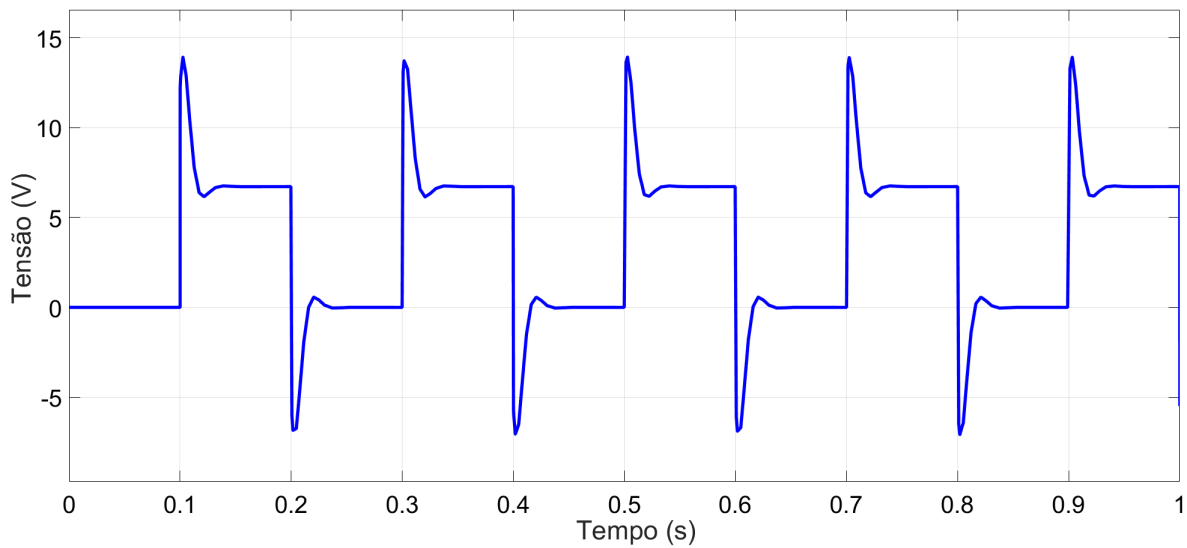
5.2 Controle PID

O projeto dos parâmetros do controle PID do sistema estimado pelo método ARMA é dado partindo do modelo do sistema apresentado na equação 4.15. Em seguida, com o auxílio do comando *pidtune* determinam-se os parâmetros da equação 5.1, como $K_p = 0,0122$, $K_i = 1,43$, $K_d = 1,84 \cdot 10^{-5}$, $T_f = 5,7 \cdot 10^{-5}$.

$$V_{in} = K_p + K_I \frac{1}{s} + K_d \frac{s}{T_f s + 1} \quad (5.1)$$

Com os parâmetros identificados foi simulado o sistema para uma entrada de referência de uma onda quadrada com 1000 RPM de pico a pico e 0,2 s de período. Como resultado obteve-se o gráfico do sinal de controle da Figura 65.

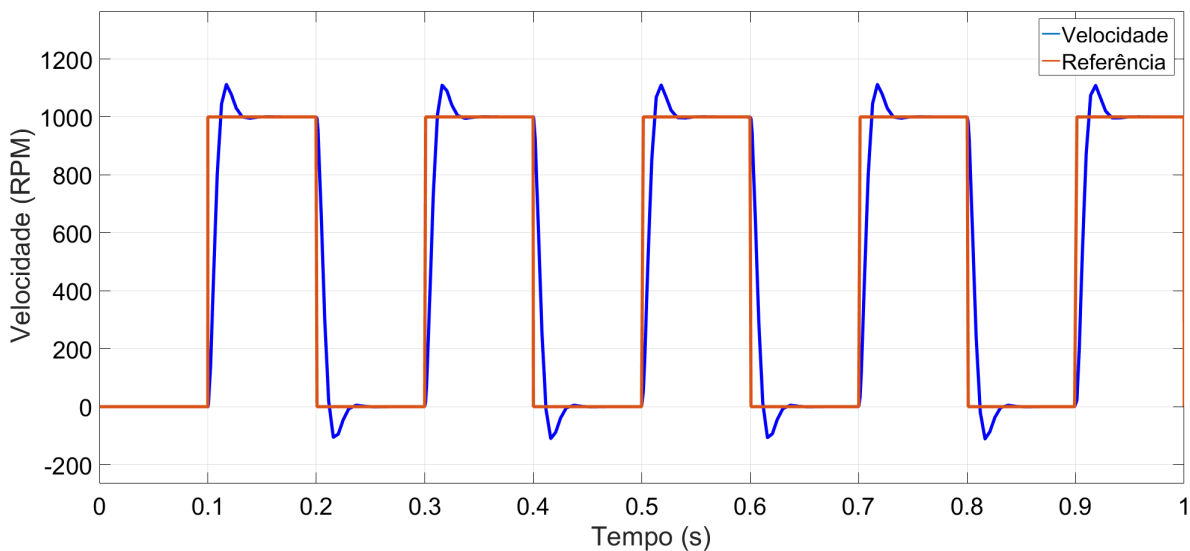
Figura 65 – Resultado simulado da tensão do controlador PID para modelo obtido pela identificação paramétrica pelo método ARMA em função do tempo.



Fonte: Próprio autor.

Nota-se que o sinal de controle gerado pelo controlador tem seu comportamento dentro da margem de trabalho do controlador (± 24 V) e portanto pode ser aplicado ao sistema. Como resultado na saída obteve-se o seguinte valor de velocidade de saída.

Figura 66 – Resultado simulado da velocidade de rotação do motor para modelo obtido pela identificação paramétrica pelo método ARMA em função do tempo.

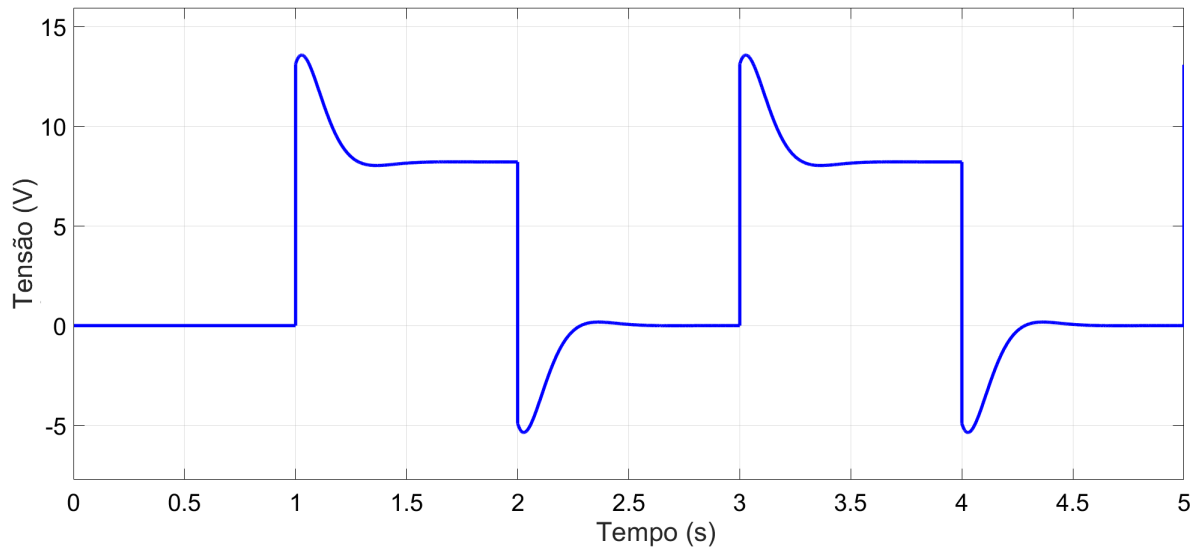


Fonte: Próprio autor.

Como pode-se observar na Figura 66 o sistema atua em menos de 0,1 s para a simulação. O mesmo procedimento foi realizado para o sistema obtido pela modelagem matemática, no

qual os parâmetros do controlador são dados por $K_P = 0,125$; $K_I = 1,9$; $K_D = -0,00073$ e $T_f = 0,0369$. Os resultados obtidos são mostrados nas Figuras 67 e 68.

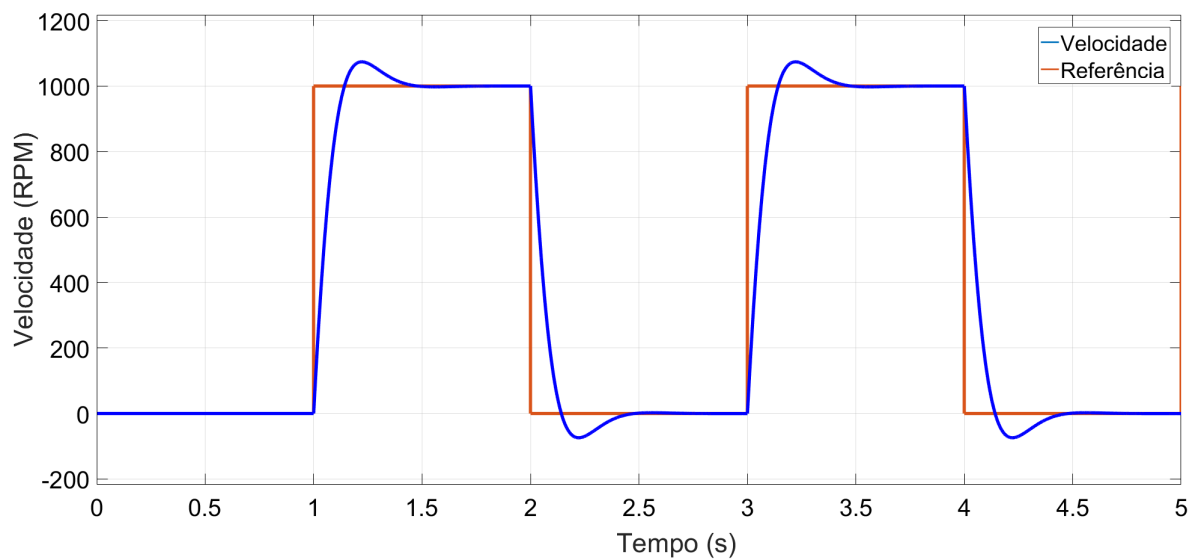
Figura 67 – Resultado simulado da tensão do controlador PID segundo o modelo matemático do sistema, em função do tempo.



Fonte: Próprio autor.

Este sinal de controle gerado também possui uma faixa de trabalho condizente com a faixa do sistema e portanto também pode ser aplicado ao controlador real. Como resultado na saída obteve-se o valor de velocidade de saída apresentado na Figura 68.

Figura 68 – Resultado simulado da velocidade de rotação do motor segundo o modelo matemático do sistema, em função do tempo.



Fonte: Próprio autor.

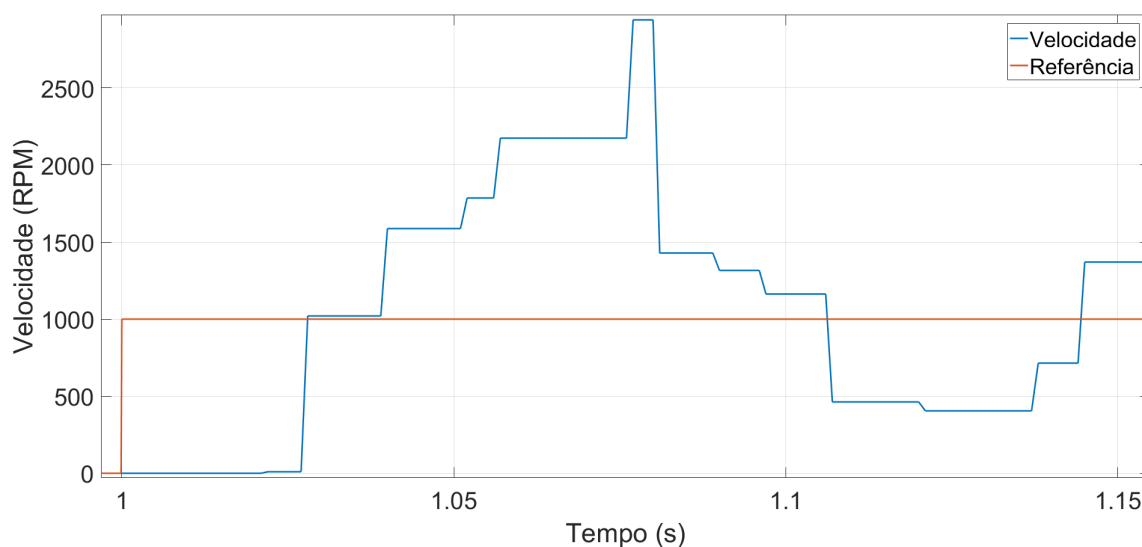
A saída possui um tempo de estabilização 10 vezes maior que o controle da Figura 65 e 66, entretanto, o modelo ARMA apresentou um erro muito maior no transiente que o sistema real, portanto os dois sistemas devem ser considerados como possíveis soluções para o sistema. Por fim, para o sistema discreto, fez-se a alteração do controlador PID contínuo para o discreto, resultando mesmos gráficos das Figuras 67 e 68, uma vez que o bloco de controle PID discreto (*Discrete PID Controller*) apresentado na Figura 26 utiliza os mesmos parâmetros K_p , K_I e K_d fazendo a devida correção no controlador baseado no período de amostragem utilizado, conforme discutido na equação 2.68.

Portanto para uma frequência de amostragem ao menos cem vezes maior que o tempo de estabilização, o resultado do sistema discreto é semelhante ao caso contínuo no tempo.

5.2.1 Aplicação ao sistema físico

Ainda que os sistemas apresentados possuam estabilidade na simulação, para o sistema real, devido à alta taxa de ruído do sensor de velocidade ambos sistemas não são suficientes para o controle do motor, nem para controle de velocidade, nem para o controle de torque, como mostram as Figuras de 69 a 74.

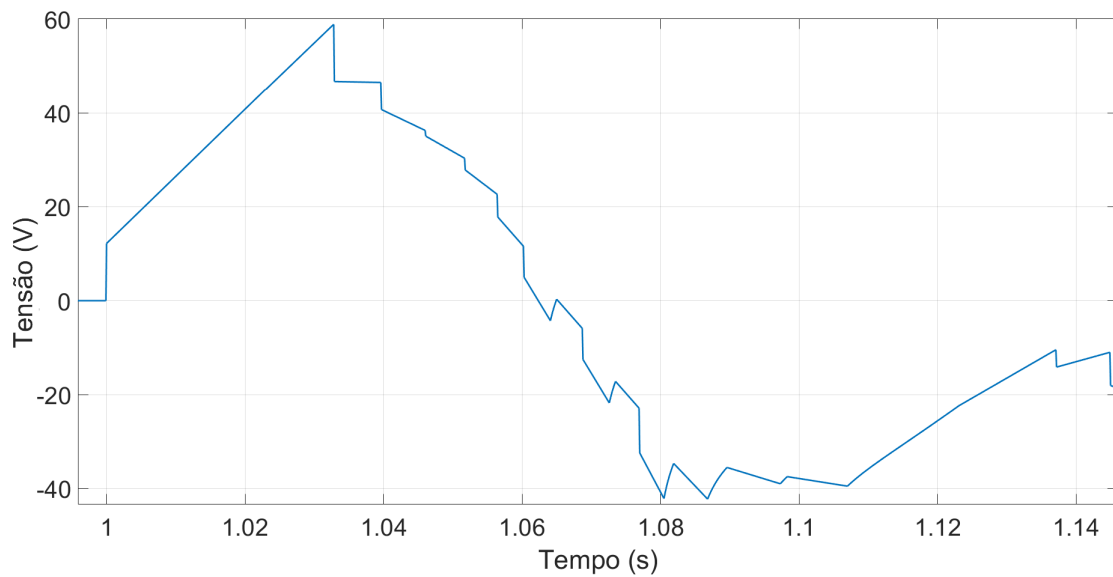
Figura 69 – Velocidade de rotação do motor em função do tempo, para o controle PID ajustado para o modelo identificado pelo método ARMA.



Fonte: Próprio autor.

Para um sinal de referência de 1000 RPM o sistema tem uma variação de mais de 1000 RPM em menos de 50 ms, portanto o experimento foi encerrado em curto prazo de tempo para evitar danificar o sistema. O sinal de controle por ser observado na Figura 70.

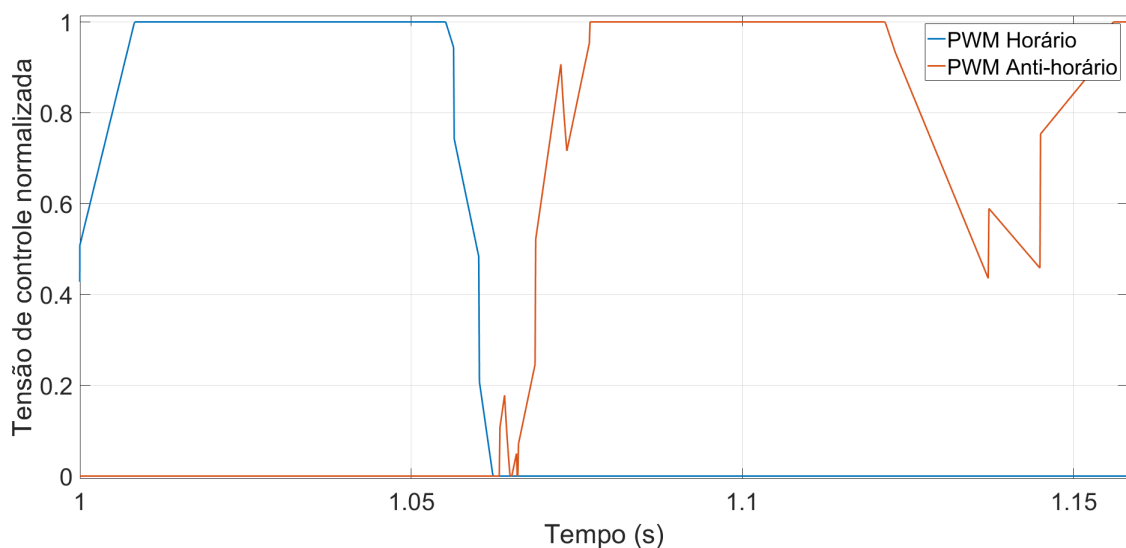
Figura 70 – Valor do sinal de controle em função do tempo, para o controlador PID ajustado para o modelo identificado pelo método ARMA.



Fonte: Próprio autor.

Nota-se que o sinal de controle aumenta rapidamente para 60 V, sendo saturado pela fonte por cerca de 50 ms e então em menos de 20 ms o sistema é saturado novamente polaridade reversa, o que pode levar a queima dos MOSFETs, devido ao pico de corrente.

Figura 71 – Sinal de controle do ciclo ativo do PWM em função do tempo, para o PID ajustado para o modelo identificado pelo método ARMA.

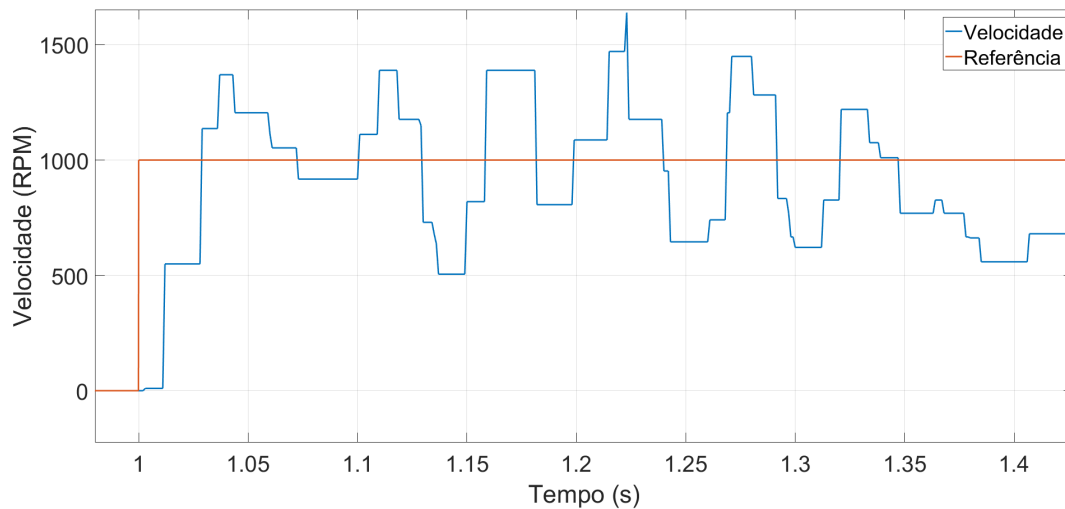


Fonte: Próprio autor.

O gráfico da Figura 71 mostra claramente a rápida reversão de tensão gerada pelo controlador, no qual o PWM horário em azul representa o percentual do ciclo ativo dos sinais

de controle com valores positivos e em laranja o PWM anti-horário para valores negativos. Observa-se também o saturação destes sinais tanto para o sentido horário quanto anti-horário. De modo semelhante é mostrado na Figura 72 a saída resultante para o controlador PID modelado matematicamente.

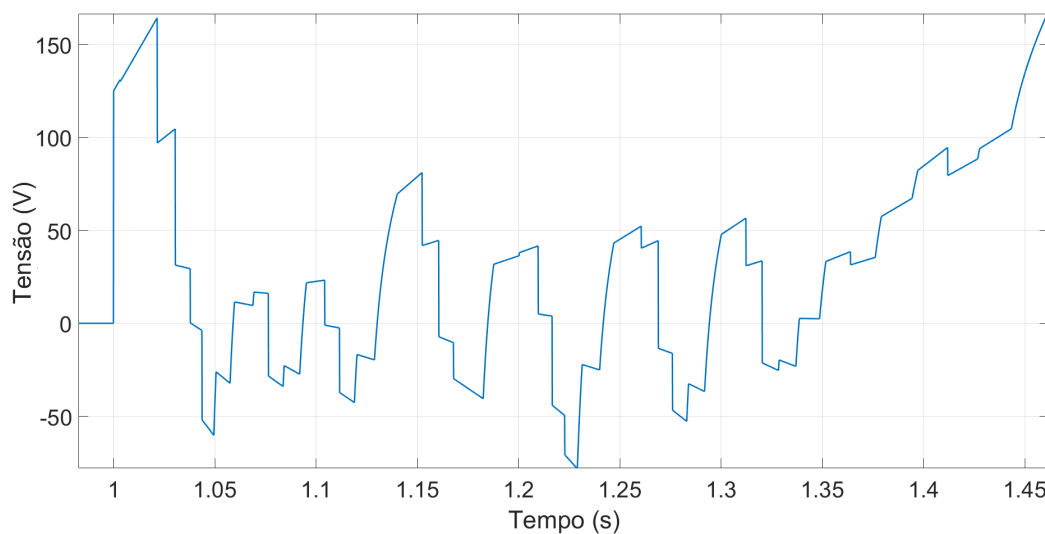
Figura 72 – Velocidade de rotação do motor em função do tempo, para o controle PID ajustado para o modelo matemático.



Fonte: Próprio autor.

O valor do sinal de controle pode ser observado na Figura 73, nota-se que a tensão oscila entre a saturação positiva e negativa da fonte, o que sobrecarrega tanto os componentes elétricos como mecânicos do sistema.

Figura 73 – Valor do sinal de controle em função do tempo, para o controlador PID ajustado para o modelo matemático.

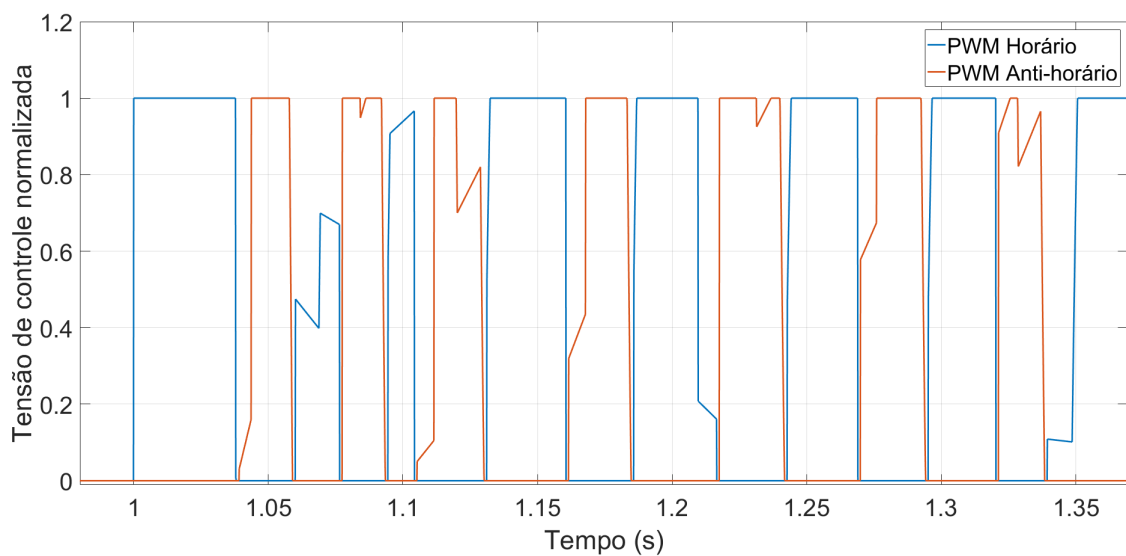


Fonte: Próprio autor.

O controle de velocidade para o sinal de referência igual a 1000 RPM mostra que assim como para o modelo ARMA a presença do ruído de medição influencia na estabilidade do sistema.

Nota-se que assim como para o controlador pelo modelo ARMA, o controle atua acionando o lado positivo e negativo do sistema, o que é muito prejudicial para o sistema, portanto este experimento também foi terminado em um curto período de tempo para evitar dano a ponte-H.

Figura 74 – Sinal de controle do ciclo ativo do PWM em função do tempo, para o controle PID ajustado para o modelo matemático.



Fonte: Próprio autor.

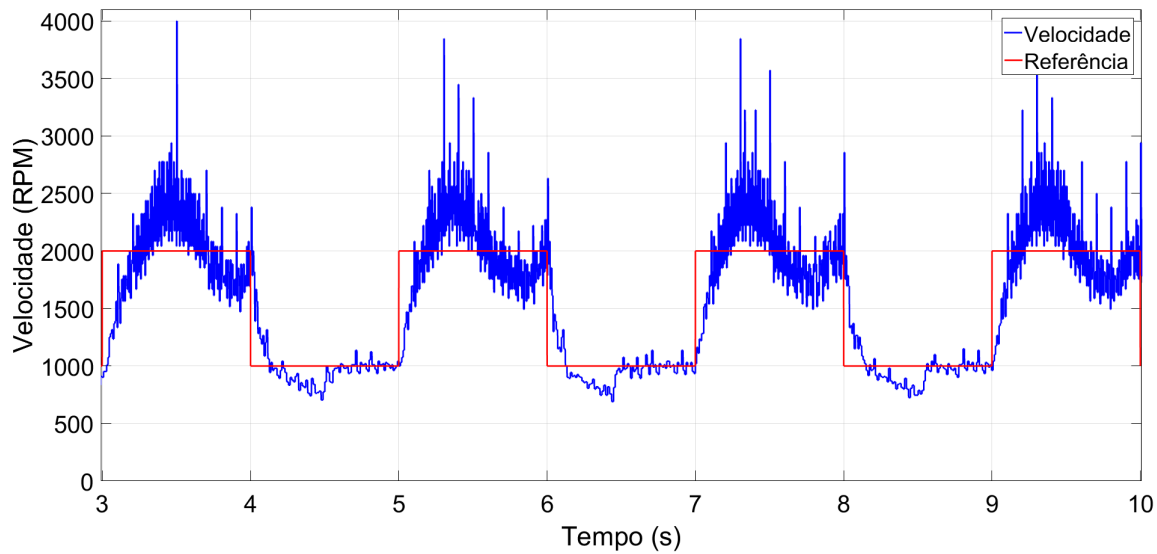
A Figura 74 apresenta de maneira nítida a transição entre o acionamento do *PWM horário* e *PWM anti-horário* o que é extremamente prejudicial ao sistema. Portanto para diminuição do problema causado pelo ruído exagerado apresentado neste sistema, propôs-se a utilização de um filtro de Kalman, conforme mostra na Seção 5.3.

5.2.2 Controle PID para a frequência de amostragem reduzida

Ao se reduzir a frequência de amostragem da realimentação do sistema em dez vezes (1 kHz) obteve-se uma melhora na estabilidade do sistema. As Figuras de 75 a 79 mostram a resposta para o sistema em malha fechada sem a utilização do filtro de Kalman e ajustado para o modelo matemático, quando o sinal de referência desejado é uma onda quadrada de valor máximo igual a 2000 RPM e mínimo de 1000 RPM e período igual a 2 s.

O ensaio começa em $t=3$ s e termina em $t=10$ s. Nota-se que o controle apresenta maior tempo de estabilização para maior velocidade, assim como também apresenta maior ruído, devido à natureza do sensor de velocidade.

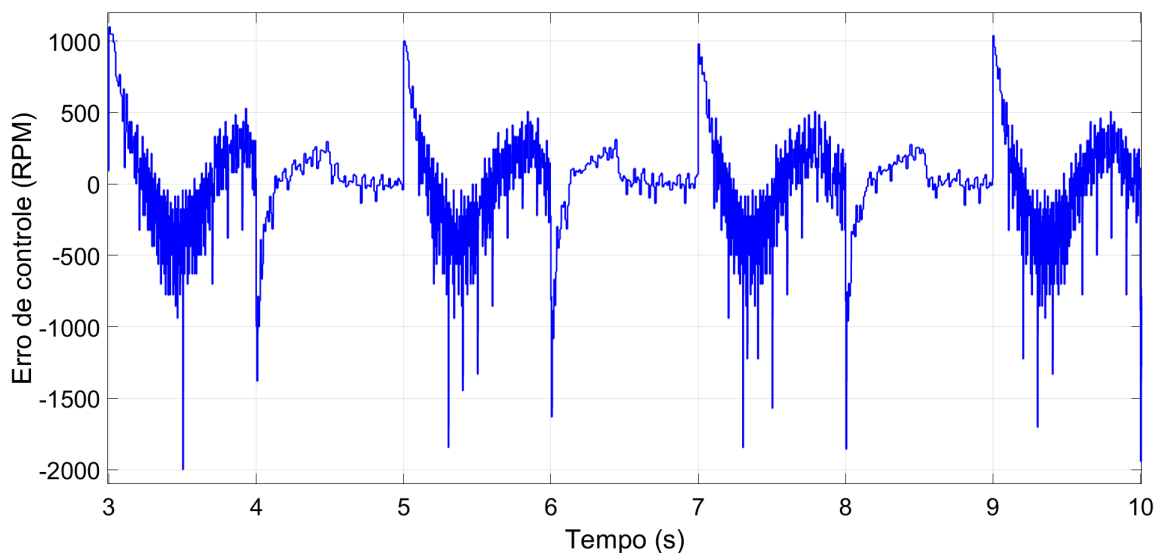
Figura 75 – Velocidade em função do tempo para um valor de frequência de amostragem igual a 1 kHz.



Fonte: Próprio autor.

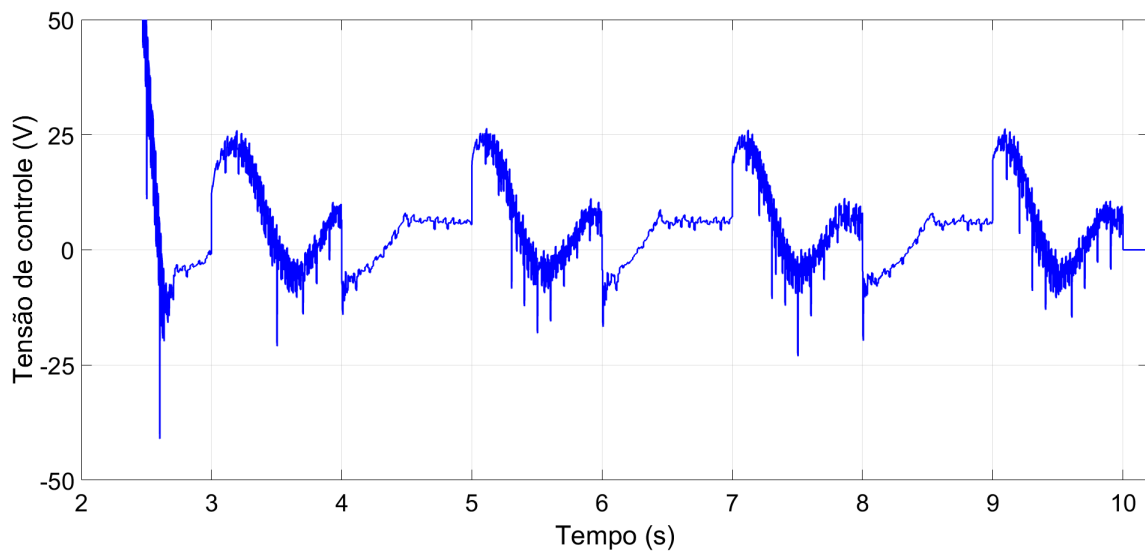
Na Figura 76 pode-se observar o sinal de erro do sistema, no qual torna-se mais nítida a oscilação do sistema ser maior para maiores velocidades, enquanto o sistema estabiliza rapidamente para a menor velocidade.

Figura 76 – Sinal de erro entre a velocidade requerida e a medida, em função do tempo, um valor de frequência de amostragem igual a 1 kHz.



Fonte: Próprio autor.

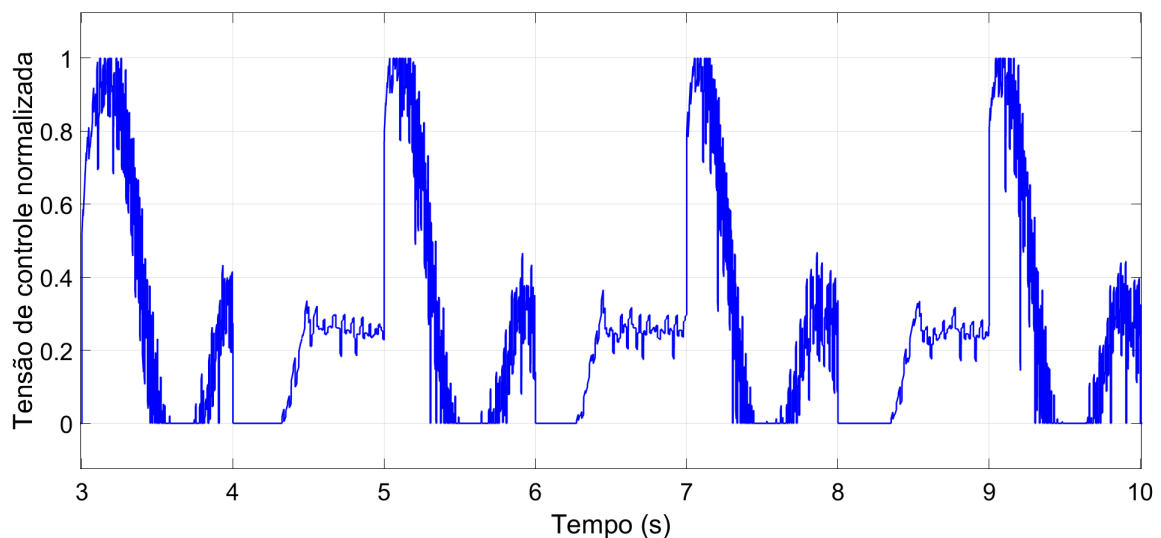
Figura 77 – Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz.



Fonte: Próprio autor.

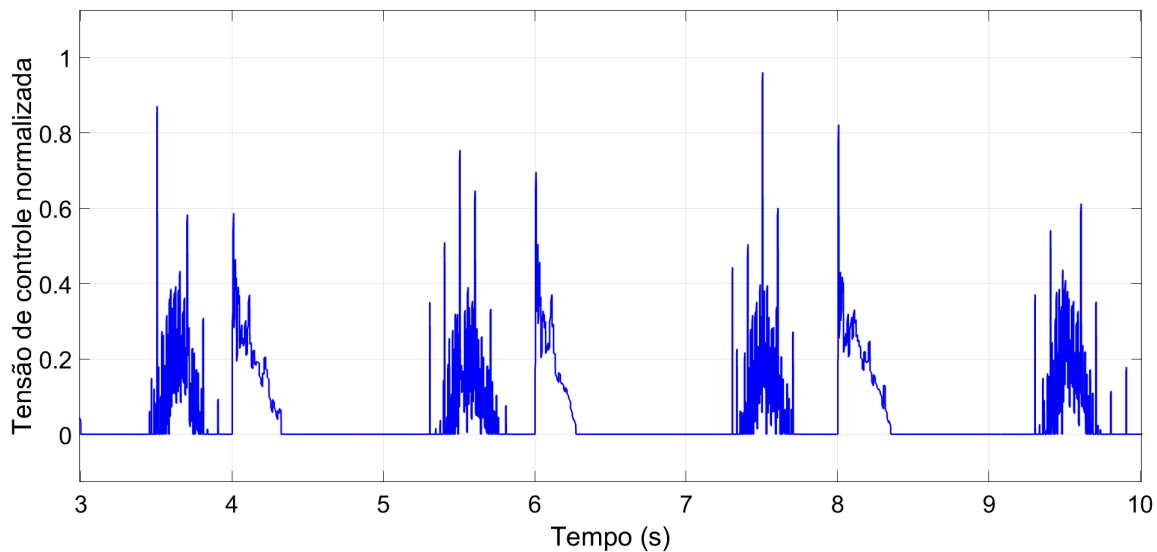
Ainda que o sistema seja estável nota-se no sinal de PID que o ruído da leitura da velocidade contamina a determinação do erro real, alterando também o erro compensado, entretanto, diferente do caso anterior, apresentado na Figura 74 em que o sinal de controle saturava seguidamente em mais ou menos 24 V, este só é saturado nos momentos de transição. A porcentagem do ciclo ativo do PWM horário e anti-horário podem ser vistos, respectivamente, nas Figuras 78 e 79.

Figura 78 – Sinal de controle do ciclo ativo do PWM horário em função do tempo, um valor de frequência de amostragem igual a 1 kHz.



Fonte: Próprio autor.

Figura 79 – Sinal de controle do ciclo ativo do PWM anti-horário em função do tempo, um valor de frequência de amostragem igual a 1 kHz.



Fonte: Próprio autor.

Nota-se portanto que diferente dos resultados com maior frequência de amostragem o sistema agora possui uma menor tensão média dos sinais de cada PWM, no qual não há saturação em ambos os lados seguidamente, proporcionando maior segurança para utilização do sistema. Contudo, com intuito de minimizar o ruído no sinal de controle gerado pelo ruído de leitura, na próxima seção será apresentado os resultados com a utilização do filtro de Kalman.

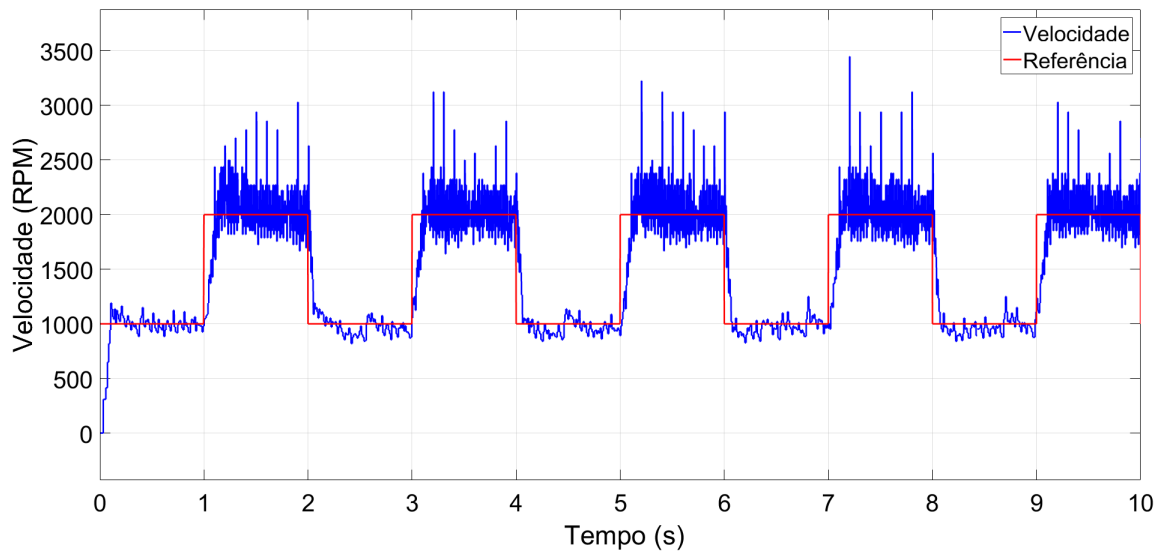
5.2.3 Controle PID para o sistema remodelado

A partir dos resultados apresentados anteriormente, notou-se que a divergência entre o modelo matemático e a resposta do sistema acarreta em problemas de estabilização, portanto, notou-se a necessidade de uma nova modelagem do sistema matemático que apresenta-se maior aproximação do o modelo real.

Para a determinação da nova modelagem do sistema utilizou-se o método apresentado na Seção 3.6.2, resultando nos parâmetros apresentados na Tabela 7 e o espaço de estados descrito pelas equações 3.11 e 3.12.

Após a remodelagem do sistema obteve-se o novo controlador PID com parâmetros $K_p = 0,3159$, $K_I = 0,7797$, $K_d = -0,0004$ e filtro $N = 551,6778$. Desta forma, obteve-se como resultado os gráficos das Figuras de 80 a 83.

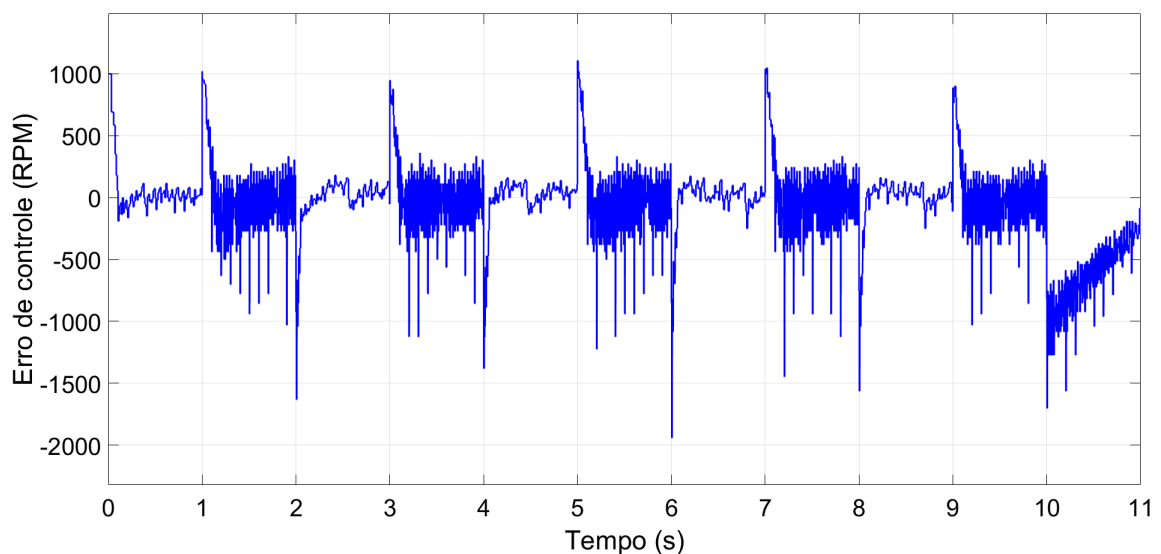
Figura 80 – Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado.



Fonte: Próprio autor.

Observa-se portanto que com a remodelagem do sistema como um todo há uma maior similaridade entre a planta real e o sistema identificado, havendo maior estabilidade nas velocidades desejadas e menor tempo de estabilização, tornando possível o ajuste para menor tempo de estabilização. A Figura 81 apresenta o erro de controle do sistema.

Figura 81 – Sinal de erro entre a velocidade requerida e a medida à saída do sistema, em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado..

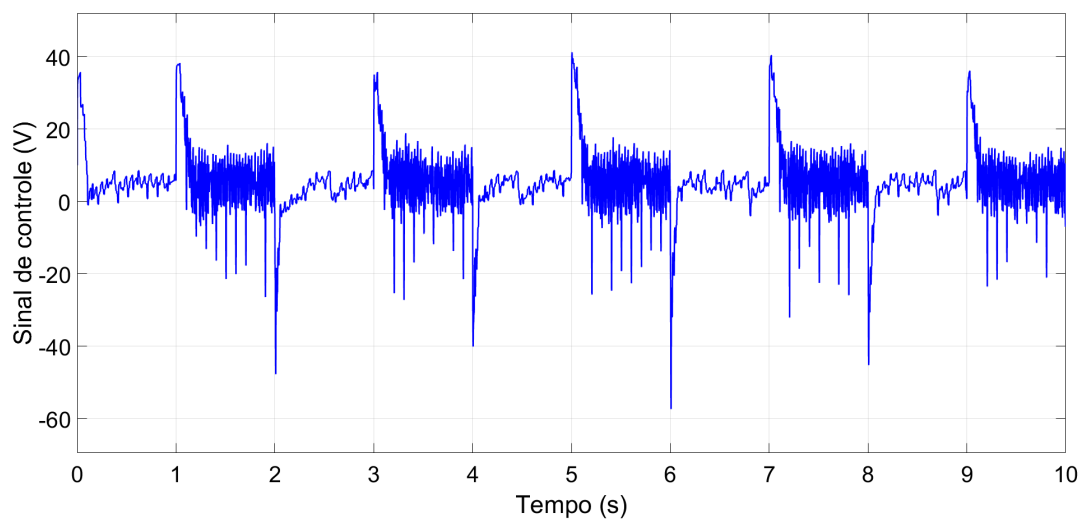


Fonte: Próprio autor.

Nota-se que não há erro de regime permanente do controlador, entretanto, devido a ausência de filtragem e alto nível de ruído gerado pelo sensor, este ruído continua presente durante toda atuação do controlador.

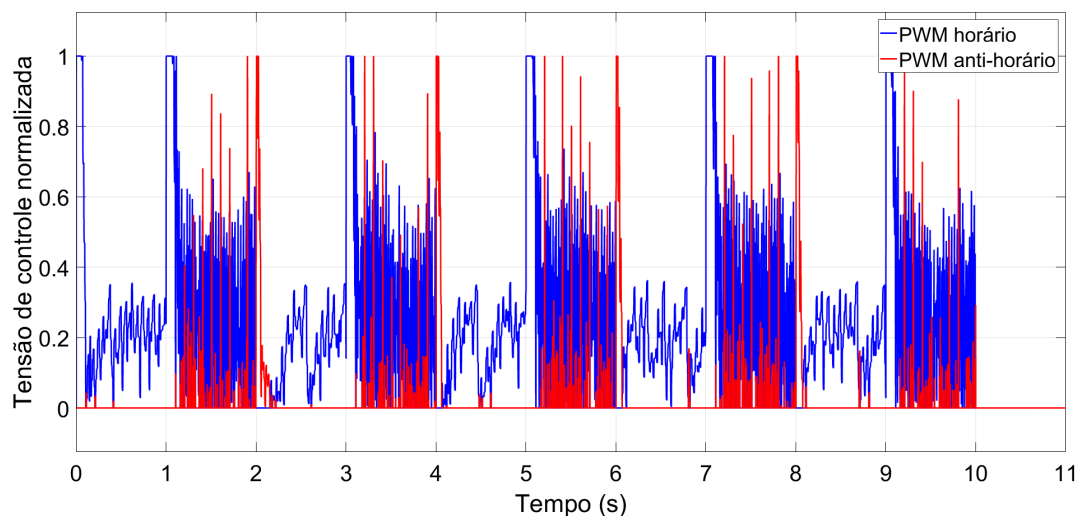
A Figura 82 permite a observação do sinal de controle do sistema, ou seja, a tensão que o controlador PID calcula para aplicação nos enrolamentos do motor. Nota-se portanto que o alto nível de ruído do sensor de velocidade se transmite à atuação do controlador, dissipando-se constante energia no sistema, tornando-o pouco eficiente energeticamente.

Figura 82 – Sinal de controle gerado pelo PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado.



Fonte: Próprio autor.

Figura 83 – Sinal de controle do ciclo ativo do PWM em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado.



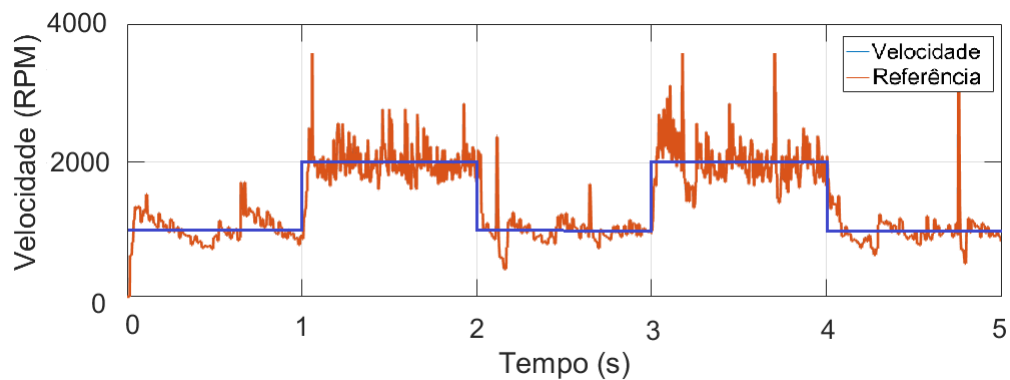
Fonte: Próprio autor.

Observando-se Figura 83 nota-se a constante atuação do controlador nos dois braços da ponte-H o que gera outro problema de geração de altos picos de corrente, devido ao cruzamento por zero. Com a adição do filtro de Kalman e redução do ruído deseja-se minimizar este efeito, tornando o sistema mais seguro e eficiente energeticamente.

5.3 Controle PID com filtro de Kalman

Com a implementação do filtro de Kalman conforme apresentado na Seção 3.8, foi utilizado o modelo matemático do sistema para ajuste dos parâmetros do PID e filtro de Kalman, resultando no gráfico apresentado na Figura 84.

Figura 84 – Velocidade de rotação do motor em função do tempo, com aplicação do filtro de Kalman.

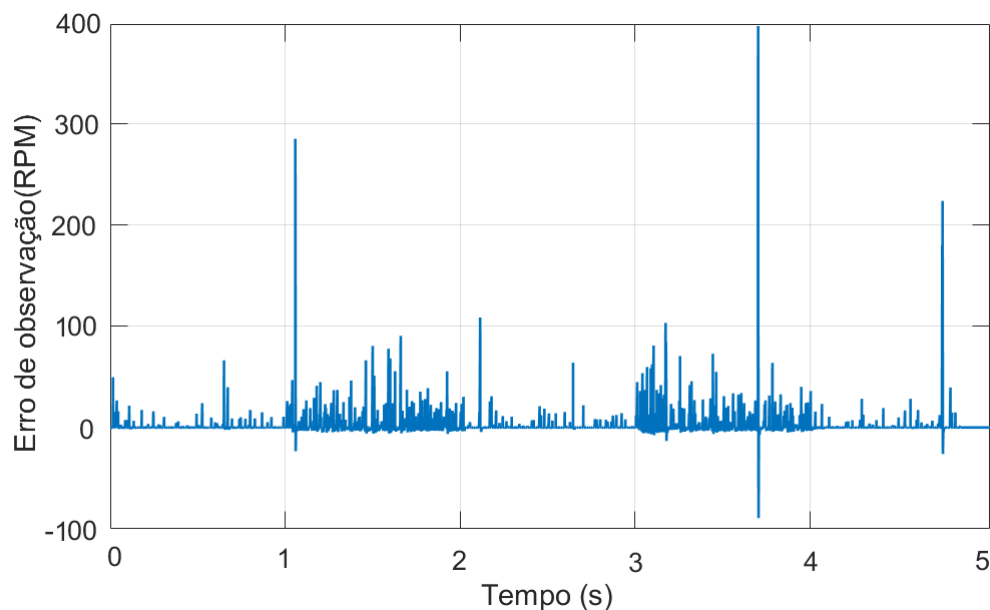


Fonte: Próprio autor.

Sabendo que o sinal de referência é uma onda quadrada que varia de 1000 RPM a 2000 RPM com período de 2 s nota-se primeiro que a adição do filtro de Kalman torna possível a estabilidade do controlador. Nota-se também que o controlador se torna mais estável para maiores velocidades.

Observa-se na Figura 85 que o filtro de Kalman atua diminuindo o ruído do sistema, entretanto sem apresentar atraso significativo.

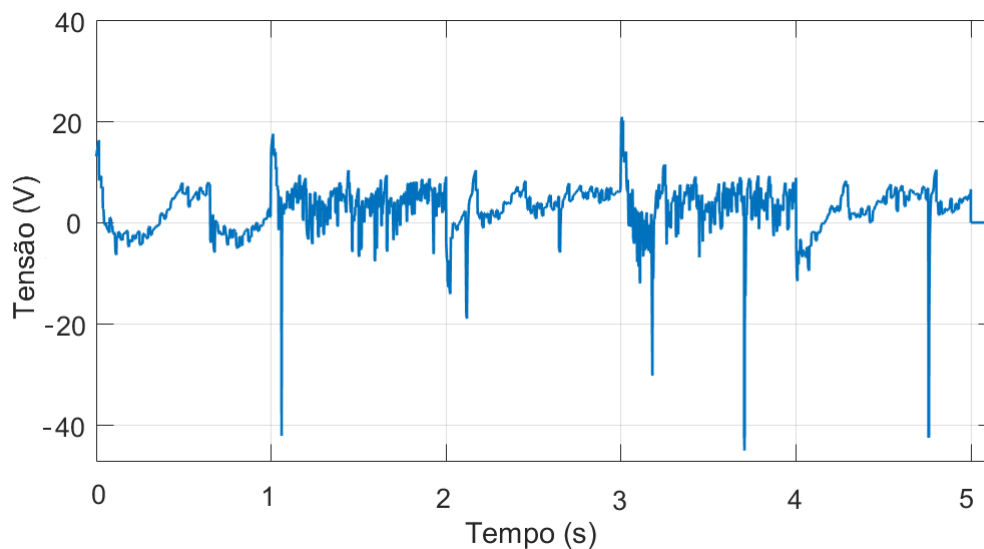
Figura 85 – Erro entre a velocidade medida e real, em função do tempo, com aplicação do filtro de Kalman.



Fonte: Próprio autor.

Diferente dos controladores sem filtro de Kalman, percebe-se que o controlador durante os 5 segundos de atuação é saturado apenas quatro vezes, como mostra a Figura 86, pontos nos quais o módulo do valor da tensão ultrapassa 24 V, gerando assim, menor desgaste para o sistema.

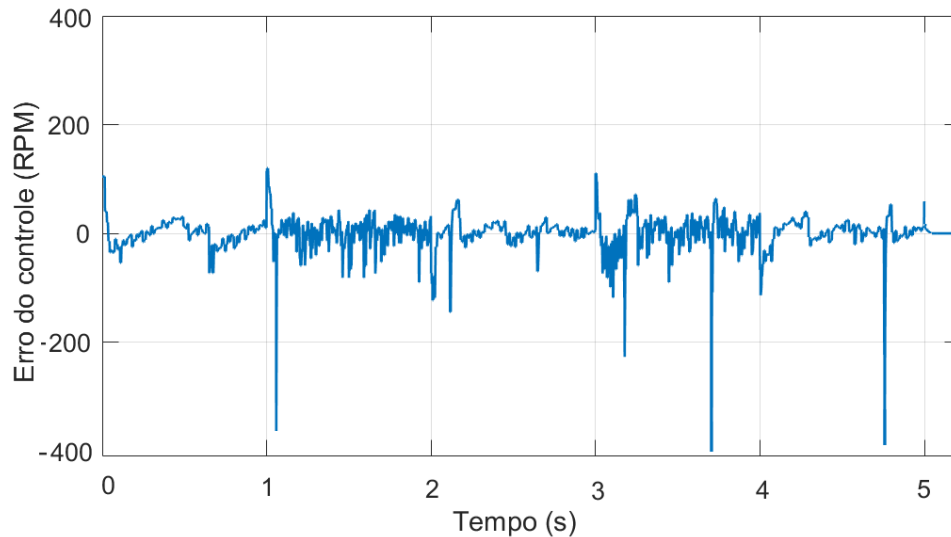
Figura 86 – Sinal de controle em função do tempo, com aplicação do filtro de Kalman.



Fonte: Próprio autor.

A Figura 87 realça o resultado da oscilação do sistema ser maior para menores rotações (1000 RPM) e mais exata para velocidades maiores. Entretanto também nota-se que em ambos os casos há estabilidade do sistema.

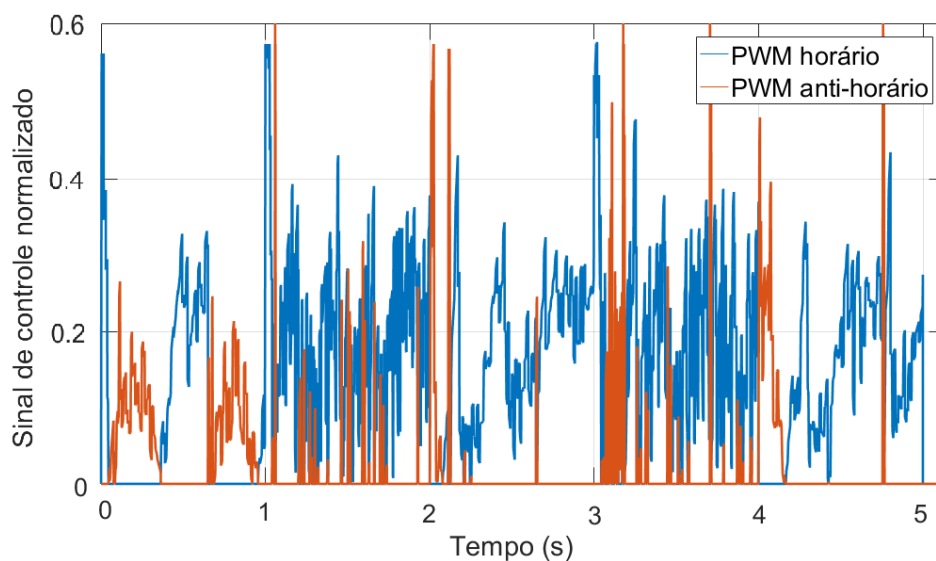
Figura 87 – Erro entre a velocidade requerida e observada, em função do tempo, com aplicação do filtro de Kalman.



Fonte: Próprio autor.

Nota-se, na Figura 88, que com a aplicação do filtro os PWMs trabalham com menos que 50% da capacidade máxima (valores menores que 0,5) durante a maior parte do tempo, tornando o sistema muito mais seguro, estável e energeticamente eficiente.

Figura 88 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, com aplicação do filtro de Kalman.

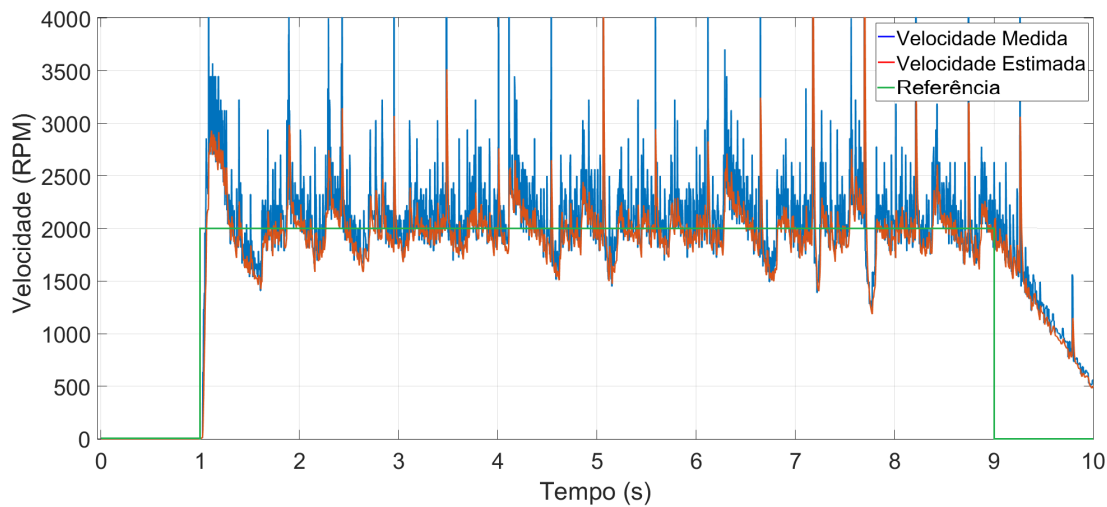


Fonte: Próprio autor.

O mesmo controlador foi testado para a execução aplicação de um degrau unitário obtendo-se os seguintes resultados.

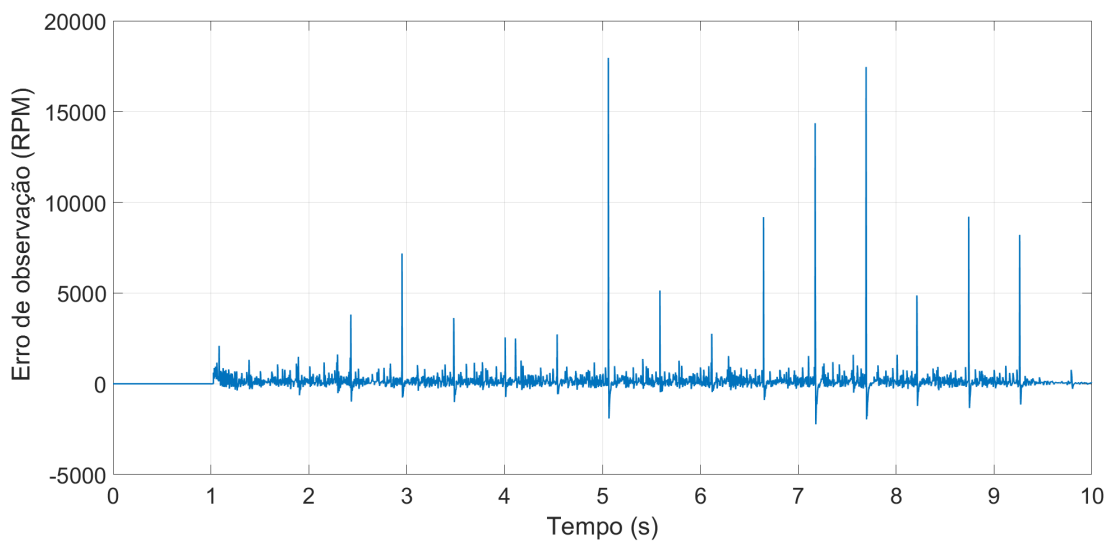
Neste caso (Figura 89) o sinal de referência é um degrau que inicia em 1 s e tem amplitude de 2000 RPM, no qual o sinal em azul é dado pela velocidade lida pelo sensor de velocidade, enquanto a onda em laranja é a velocidade estimada à saída do filtro de Kalman. Nota-se que mais uma vez o sistema foi estabilizado em torno da região desejada, mas que o ruído do sensor não permite a perfeita adaptação do controle.

Figura 89 – Velocidade de rotação do motor em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.



Fonte: Próprio autor.

Figura 90 – Erro entre a velocidade medida e estimada, em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.

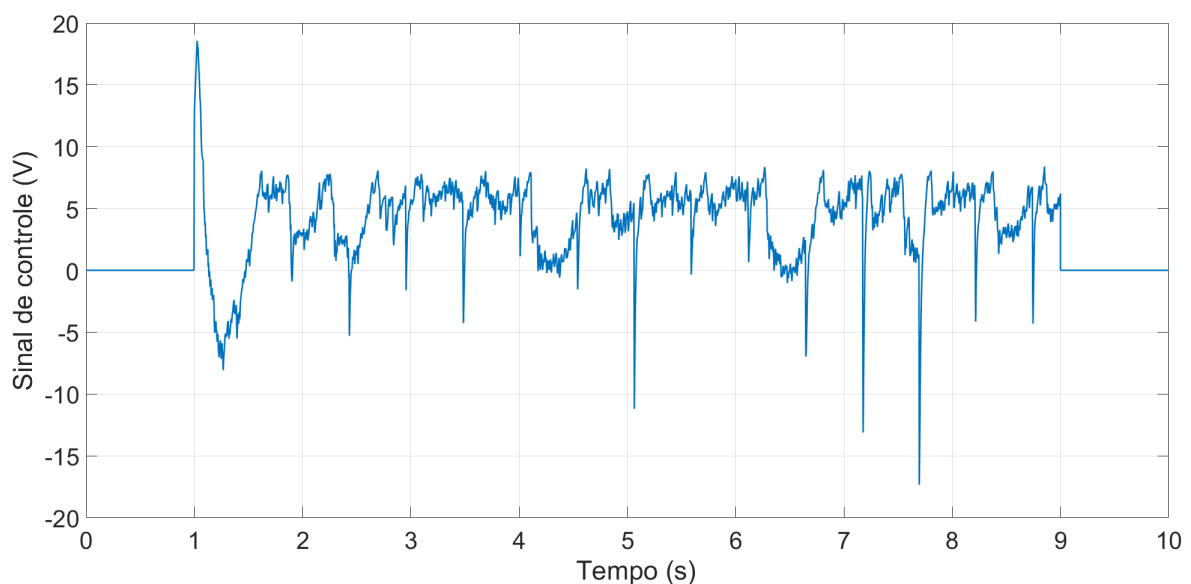


Fonte: Próprio autor.

Como mostra a Figura 90 o ruído de leitura é mais uma vez reduzido pelo filtro de Kalman, assim como discutido na Figura 85. Ou seja, os picos causados pela leitura ruidosa do sensor de velocidade é removido do sinal de realimentação do controlador.

O sistema de controle, mostrado na Figura 91, por sua vez não apresenta nenhum momento de saturação, uma vez que em comparação com a entrada de onda quadrada este sistema exige muito menos do controlador. Entretanto este teste permite a observação da estabilidade do controle por longo período de atuação, resultando em menor nível de tensão na ativação do atuador o que comprova a utilidade do filtro para maior eficiência do sistema.

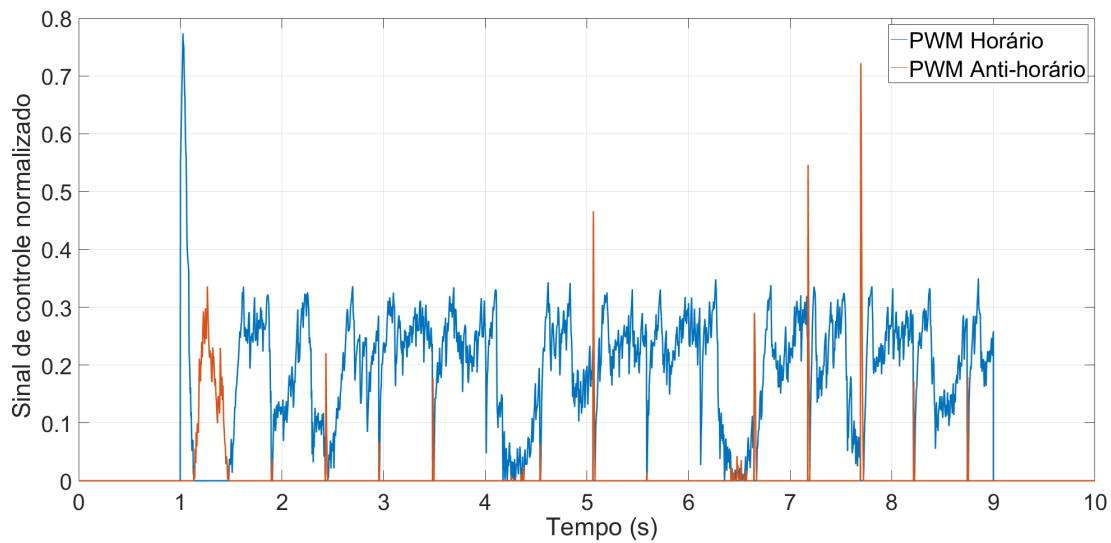
Figura 91 – Sinal de controle em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.



Fonte: Próprio autor.

O filtro gráfico do sinal de controle do PWM da Figura 92 deixa bastante evidente a melhora da atuação do controlador, no qual não apresenta nenhuma saturação e apenas possui alguns valores de inversão do PWM indesejados em momentos esparsos, resultado em um movimento muito mais fluido do acionamento do motor, quando comparado com o sistema sem filtro de Kalman.

Figura 92 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, com aplicação do filtro de Kalman, para uma entrada degrau.

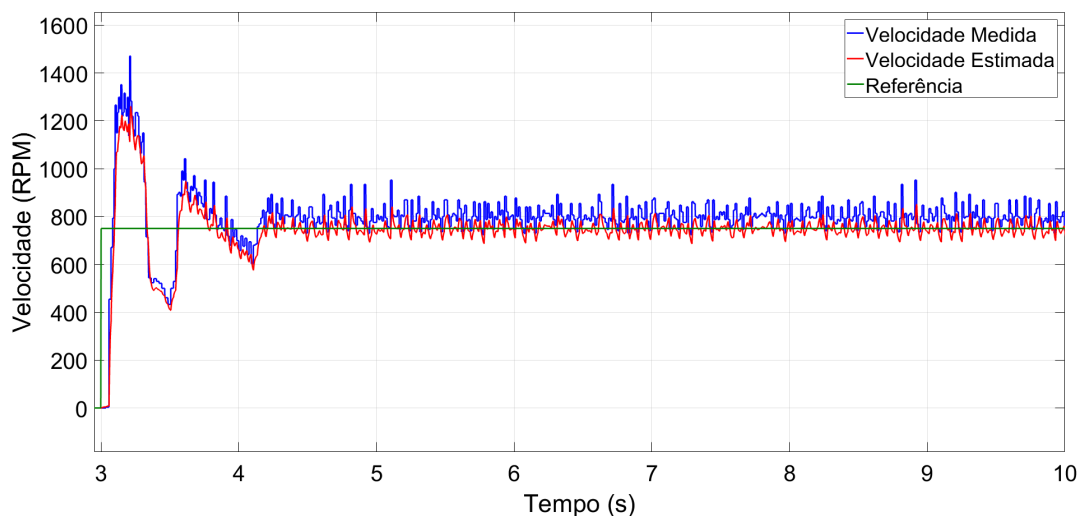


Fonte: Próprio autor.

5.3.1 Controle com filtro de Kalman e redução da frequência de amostragem.

Assim como nos testes sem filtro de Kalman apresentados na Seção 5.2.2 aumentou-se o período de amostragem do sistema de 0,1 ms para 1 ms, sendo possível obter um resultado ainda melhor que o apresentado na Seção 5.3. Primeiramente para o experimento um degrau de referência com nível em 750 RPM obtiveram-se os resultados das Figuras 93 a 97.

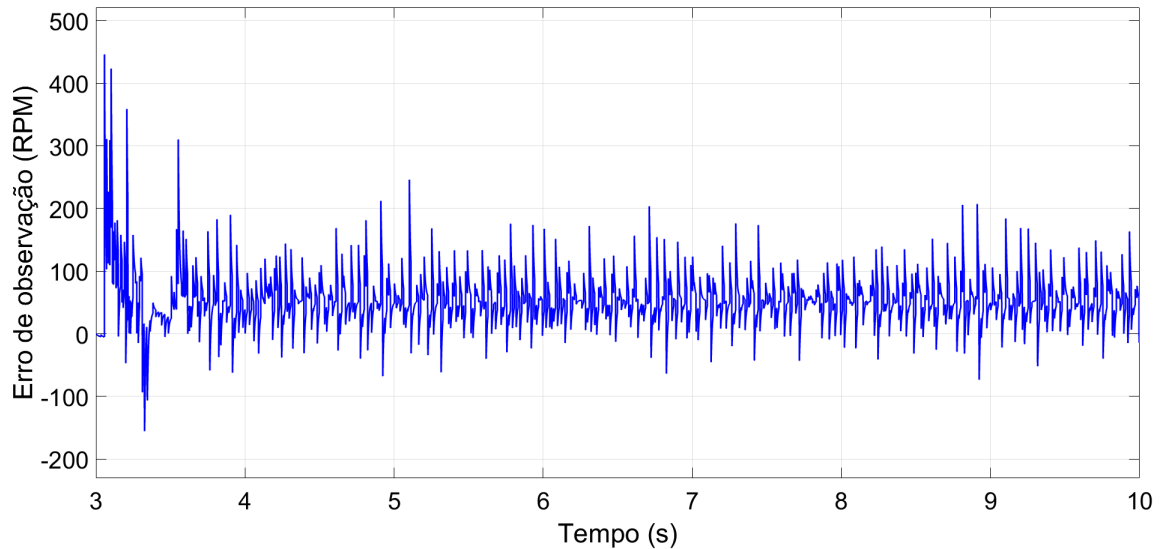
Figura 93 – Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman.



Fonte: Próprio autor.

Nota-se que com a redução do valor da frequência de amostragem há uma melhora significativa do desempenho do controlador, como pode ser vista na Figura 93, no qual o experimento se dá início aos $t=3$ s e termina em $t=10$ s.

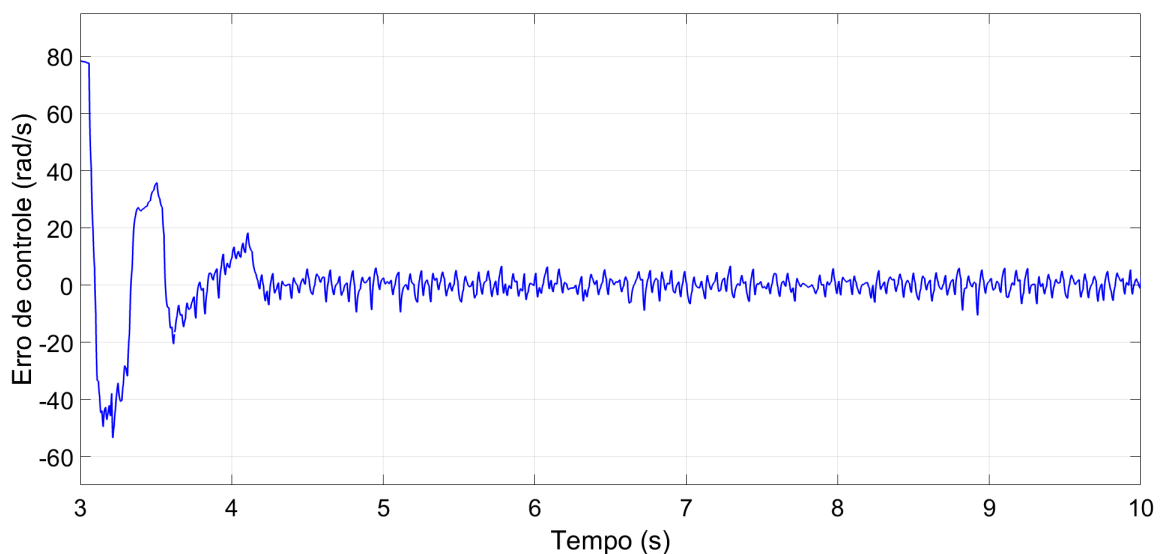
Figura 94 – Erro entre o valor da velocidade medida e o estimado pelo filtro de Kalman em função do tempo, um valor de frequência de amostragem igual a 1 kHz.



Fonte: Próprio autor.

Quando comparado este sistema sem a aplicação do filtro de Kalman com o da Figura 94 nota-se que há uma redução significativa do ruído (em torno de 100 RPM no erro de leitura).

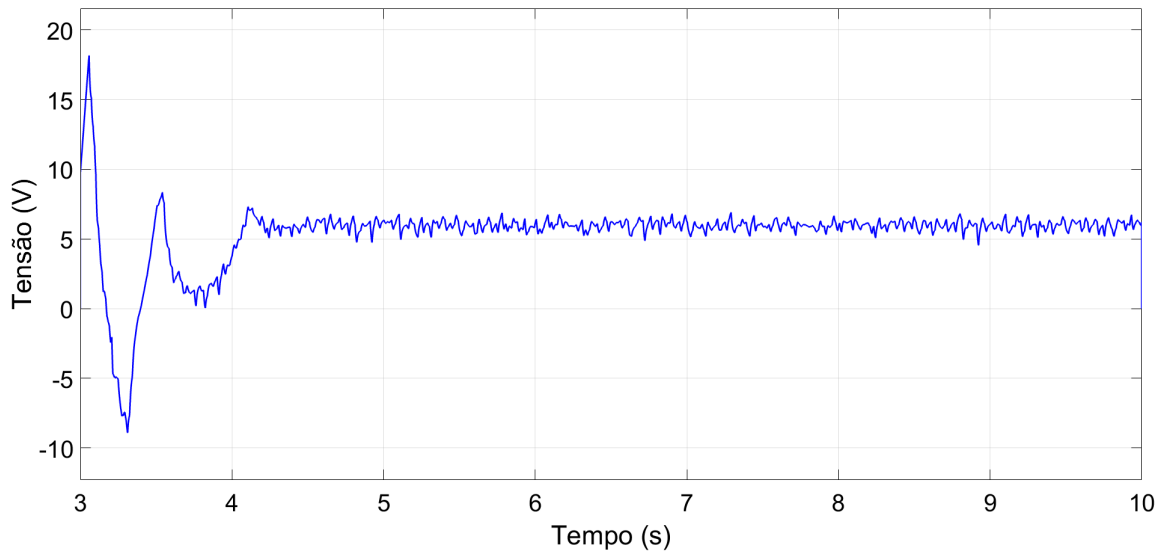
Figura 95 – Erro entre a velocidade requerida e a estimada, em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman.



Fonte: Próprio autor.

Como resultado o erro de estabilização do controlador pode ser visto na Figura 95, onde o tempo de estabilização do sistema é de aproximadamente 1,20 s.

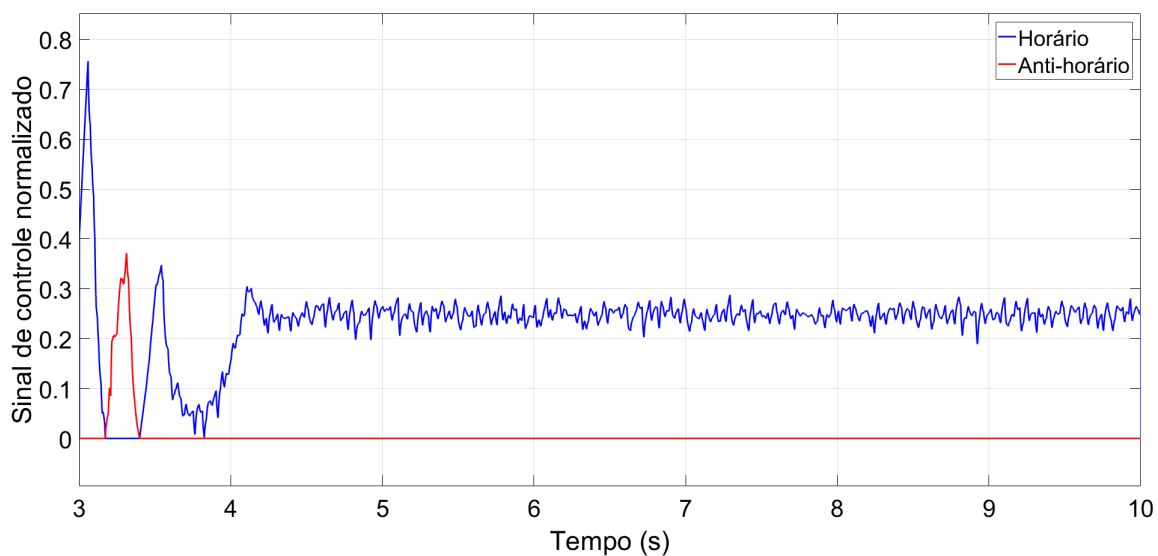
Figura 96 – Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman.



Fonte: Próprio autor.

Como pode-se observar na Figura 96 o sistema ainda apresenta ruído na atuação do controle durante a estabilização do sistema, porém muito menor que sem a aplicação do filtro.

Figura 97 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman.

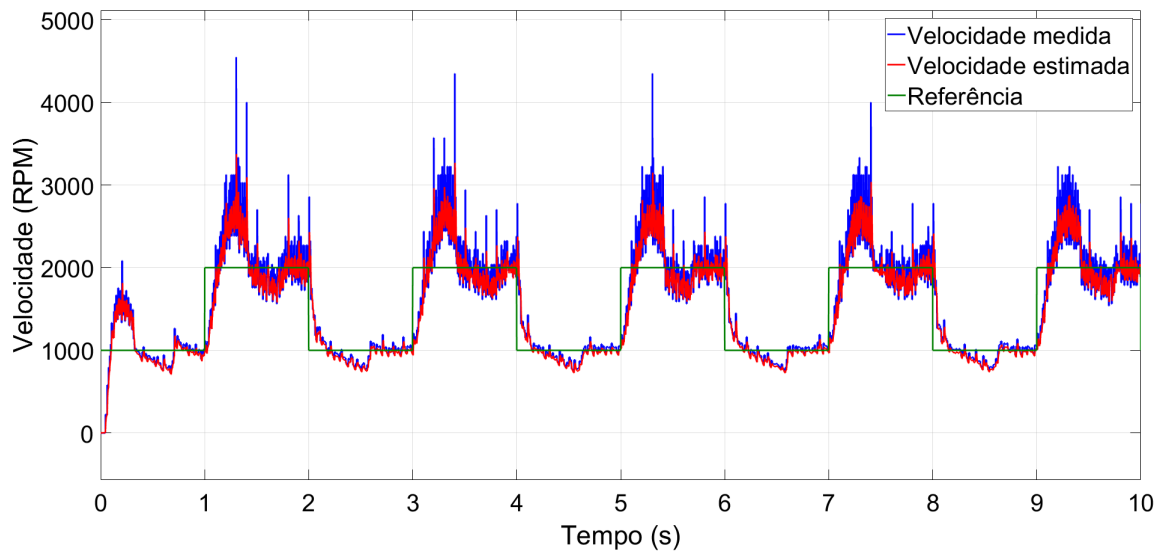


Fonte: Próprio autor.

Por fim percebe-se na Figura 97 que o sistema não apresenta saturação do sinal de controle em nenhum momento. A fim de examinar melhor os resultados do controlador com filtro de Kalman, este foi submetido à uma entrada de referência de uma onda quadrada com níveis em 1000 RPM e 2000 RPM, e período de 2 s. Os resultados podem ser observados nas Figuras de 98 a 102.

Comparando o experimento com aplicação do filtro de Kalman, da Figura 98, com o experimento na ausência deste, da Figura 75, nota-se que há uma estabilização mais rápida para velocidades mais altas, justamente pela diminuição do ruído presente na utilização do filtro.

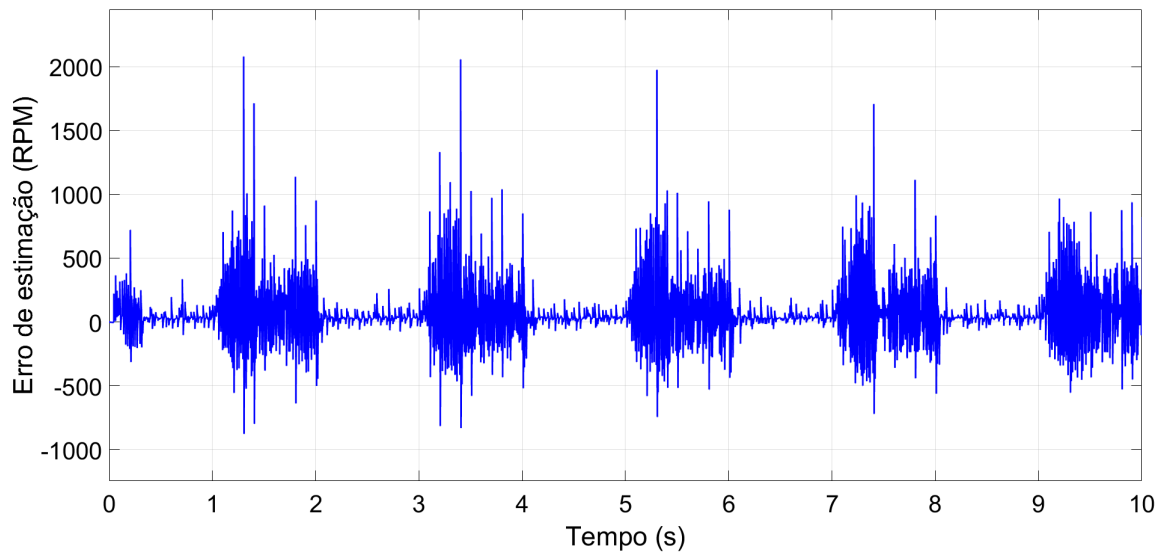
Figura 98 – Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

Quanto ao erro de estimação há uma redução considerável do ruído de leitura da velocidade principalmente para altas velocidades, os quais podem ser vistos nos segundos ímpares da Figura 99.

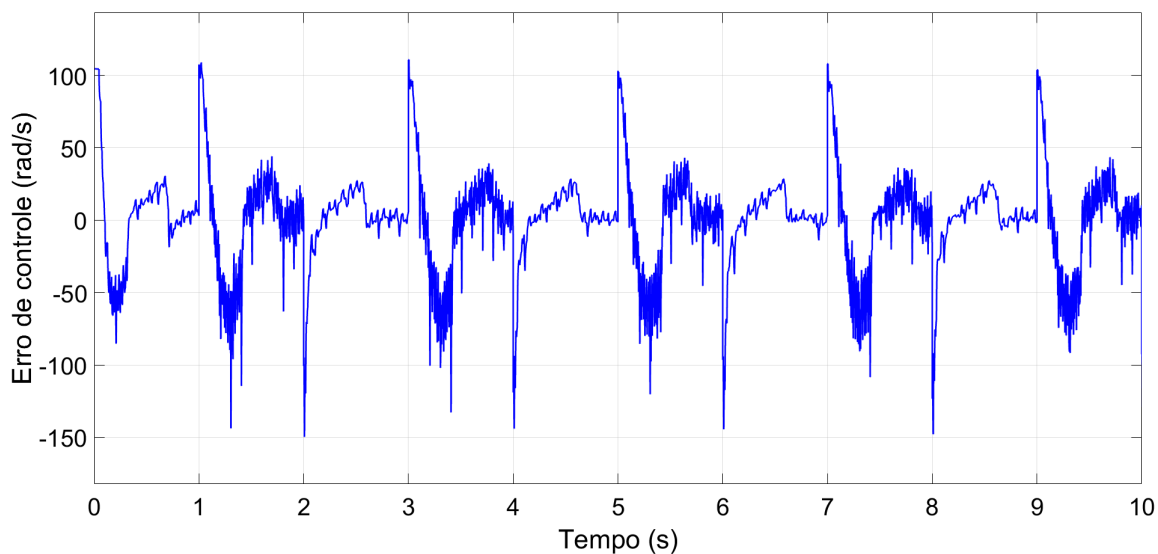
Figura 99 – Erro entre o valor da velocidade medida e o estimado pelo filtro de Kalman em função do tempo, um valor de frequência de amostragem igual a 1 kHz, para uma entrada de onda quadrada.



Fonte: Próprio autor.

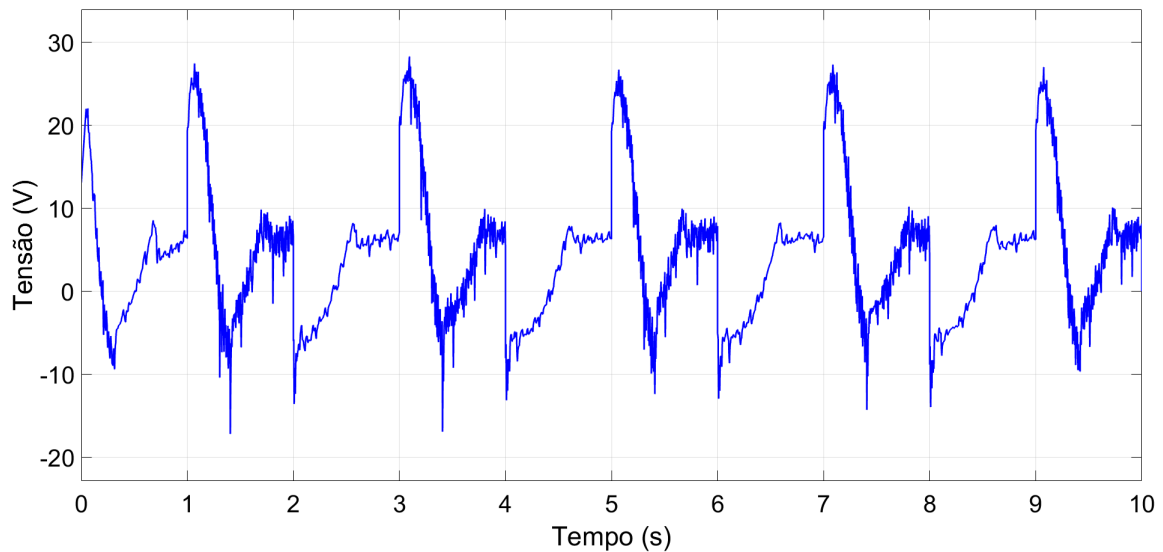
A Figura 100 mostra o gráfico do erro do controlador PID, onde é possível notar a estabilização do sistema em menos de 1 segundo mesmo para velocidades mais altas.

Figura 100 – Erro entre a velocidade requerida e a estimada, em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

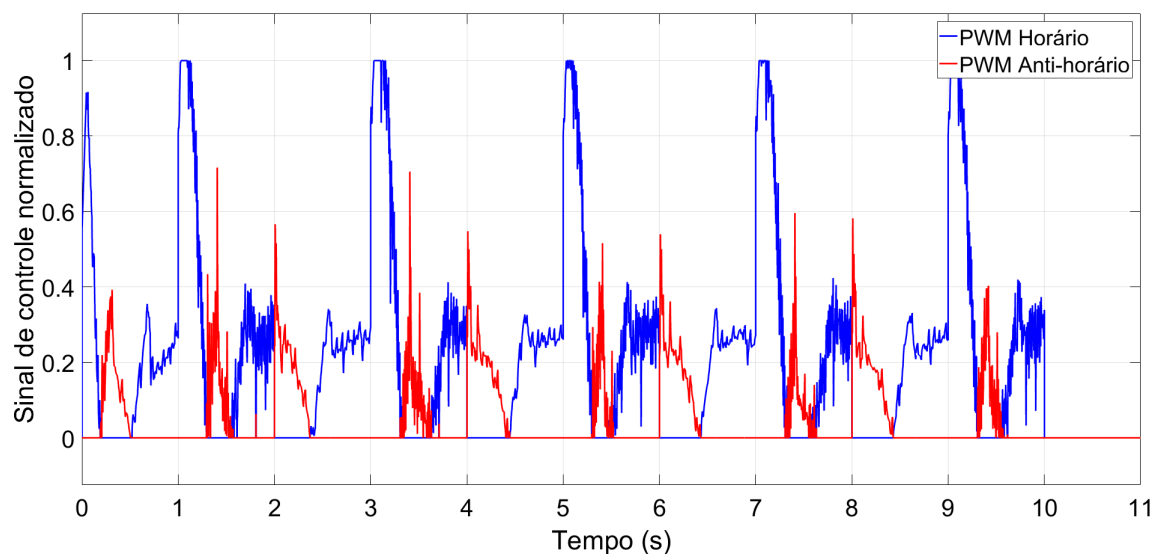
Figura 101 – Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHza, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

Como pode-se observar na Figura 101 o sistema ainda apresenta ruído na atuação do controle durante a estabilização do sistema, porém muito menor que a apresentada na Seção 5.2 (sem o filtro de Kalman), mesmo para aplicação de um sinal degrau na referência.

Figura 102 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, um valor de frequência de amostragem igual a 1 kHz, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

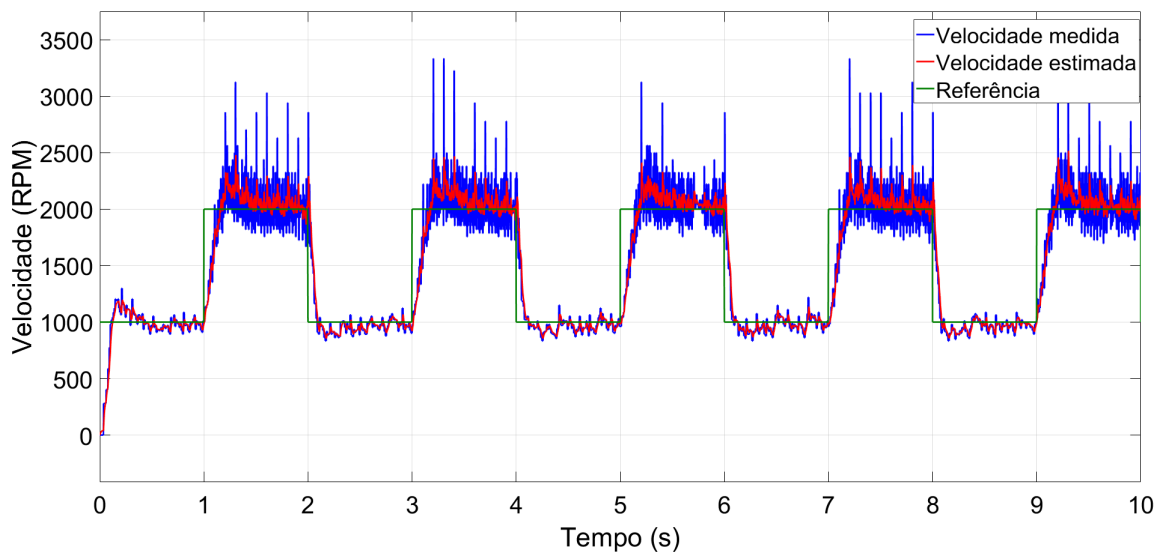
Como resultado, observa-se na Figura 102, que há sinais de controle menos agressivos ao driver ponte-H. Contudo ainda que os resultados não sejam homogêneos como em teoria já apresentam maior similaridade com estes além de um avanço na melhora da eficiência energética.

5.3.2 Controle com filtro de Kalman e redução da frequência de amostragem e remodelagem da planta para o sistema completo

Após a realização da remodelagem do sistema completo e ajuste do novo PID com valores $K_p = 0,3159$, $K_I = 0,7797$, $K_d = -0,0004$ e filtro $N = 551,6778$, e filtro de Kalman, realizou-se o experimento da onda quadrada, obtendo-se os resultados apresentados nas Figuras de 103 a 107.

Nota-se portanto na Figura 103 que o sistema possui um tempo de estabilização menor, redução da oscilação em estado estacionário e menor índice de ruído na medida que todos os testes anteriores.

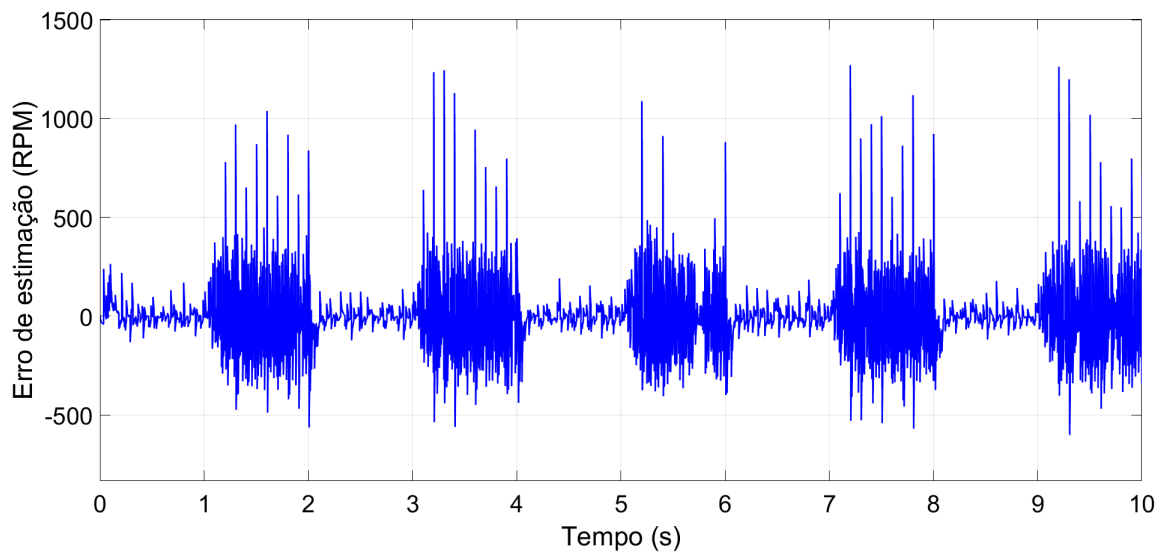
Figura 103 – Velocidade de rotação do motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

A Figura 104 apresenta, por sua vez a diferença entre o valor da velocidade medida e o obtido através do filtro de Kalman, observa-se portanto, que a capacidade do sistema em reduzir o ruído foi significativamente aumentada (cerca de duas vezes), sem que haja adição de erro no regime estacionário.

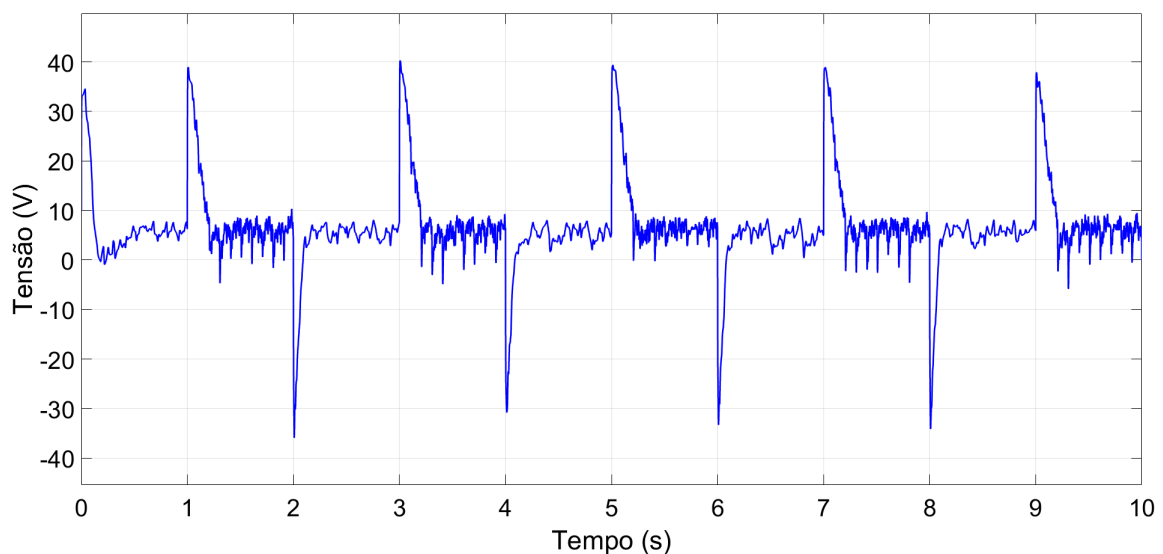
Figura 104 – Erro entre o valor da velocidade medida e o estimado pelo filtro de Kalman em função do tempo, um valor de frequência de amostragem igual a 1 kHz, para uma entrada de onda quadrada.



Fonte: Próprio autor.

Como resultado da redução do erro de regime estacionário, nota-se na Figura 105 que o sinal de controle possui um índice de ruído também muito menor que os sistemas anteriores, tornando possível o controle de forma muito mais eficiente energeticamente.

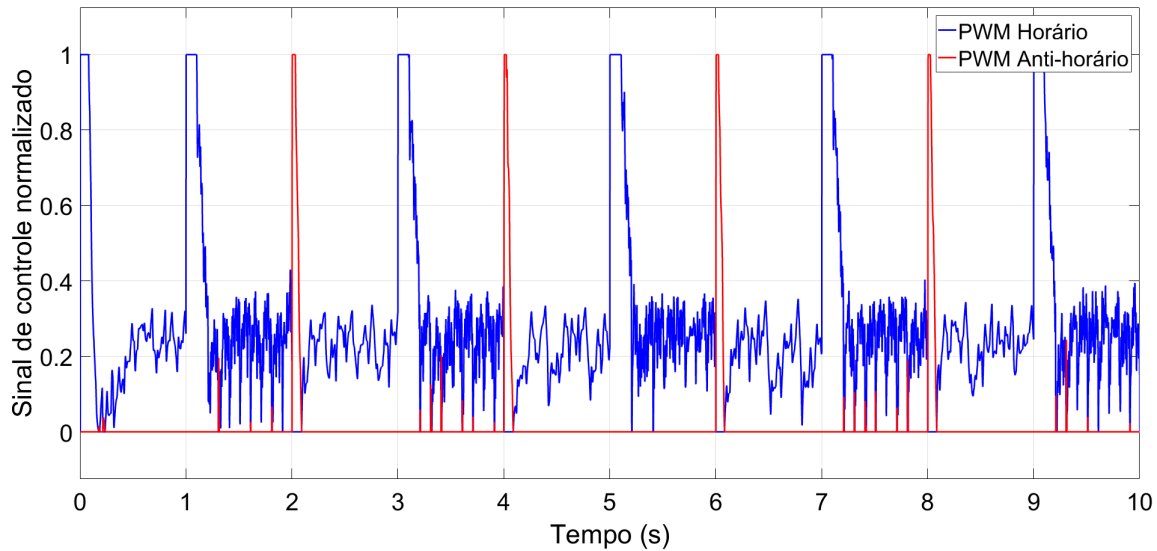
Figura 105 – Sinal de controle PID em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

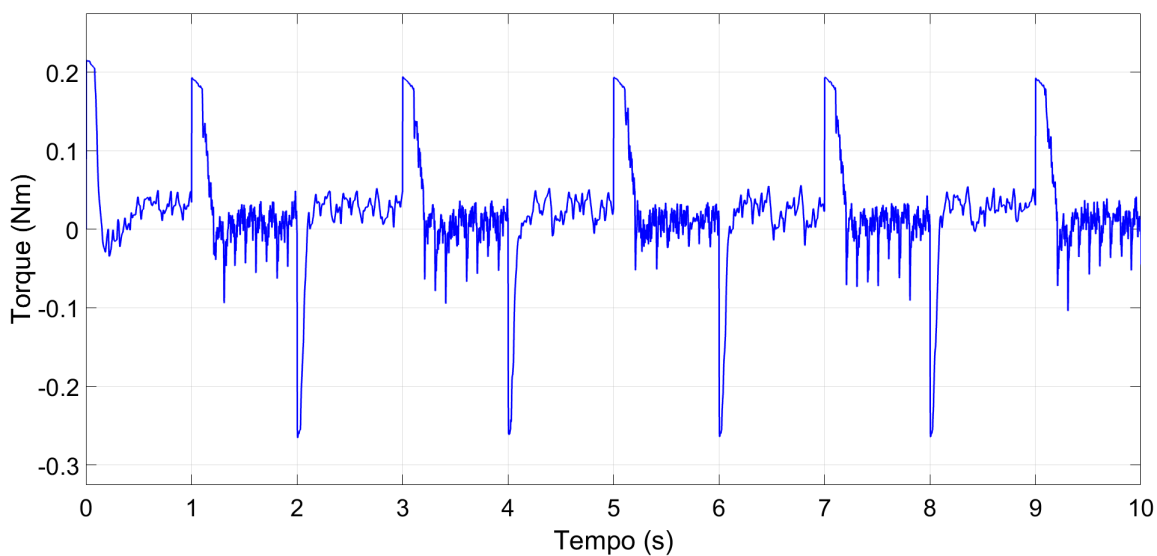
Como resultado observa-se na Figura 106 um acionamento pouco ruidoso dos pinos de PWM, quando comparados a resultados anteriores e portanto considera-se, dado o sistema apresentado neste trabalho, um sistema de controle otimizado.

Figura 106 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

Figura 107 – Sinal de Torque gerado pelo motor em função do tempo, um valor de frequência de amostragem igual a 1 kHz e modelo matemático remodelado, com aplicação do filtro de Kalman, para uma entrada de onda quadrada.



Fonte: Próprio autor.

Por fim, utiliza-se o observador de estados para estimar o torque gerado a cada instante do teste apresentado, como resultado dá-se a Figura 107. Percebe-se portanto que o acionamento possui picos de torque em situações de mudança de velocidades e este portanto estabiliza-se próximo de zero de forma a manter a rotação constante, conforme esperado.

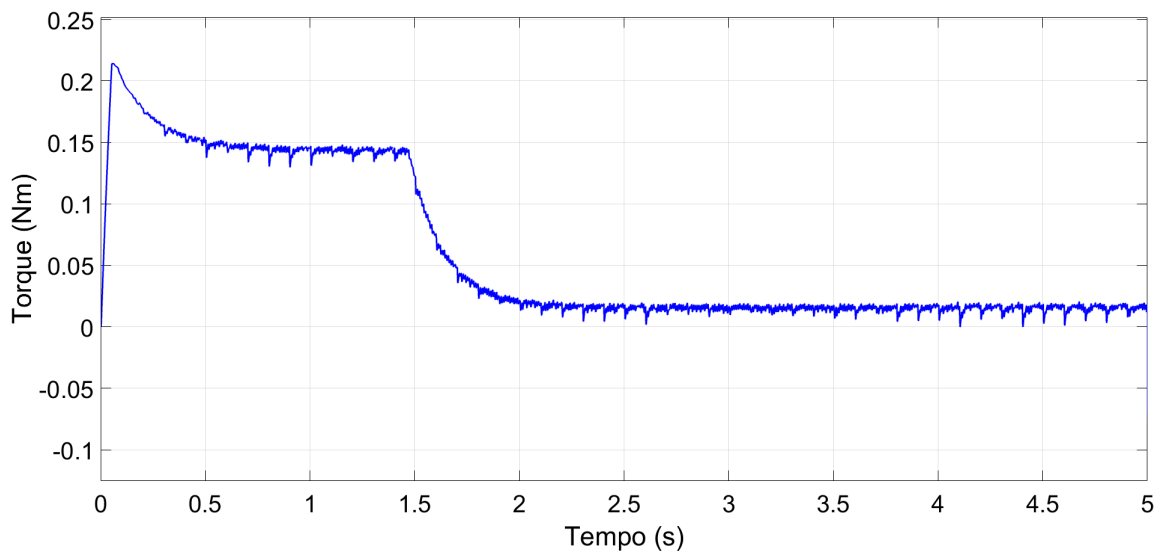
5.3.3 Controle de torque

Devido a presença de ruído gerado pelo sensor de velocidade, o torque, que é a derivada segunda da velocidade, apresenta um ruído ainda maior. Este fator afeta a estabilidade do controle mesmo com a utilização do filtro de Kalman.

Entretanto com a redução da frequência de amostragem do sistema, juntamente com a remodelagem do sistema como um todo, obtiveram-se os seguintes novos parâmetros do controlador PI $K_p = 0,3375$, $K_I = 675,0992$. Resultando em um sistema estável, como mostram os gráficos das Figuras de 108 a 110.

Nota-se na Figura 108 que assim como na Figura 107 o sistema tem um valor de pico de torque para partida do motor e então estabiliza com um valor próximo de zero para estabilização do sistema, devido a baixa carga.

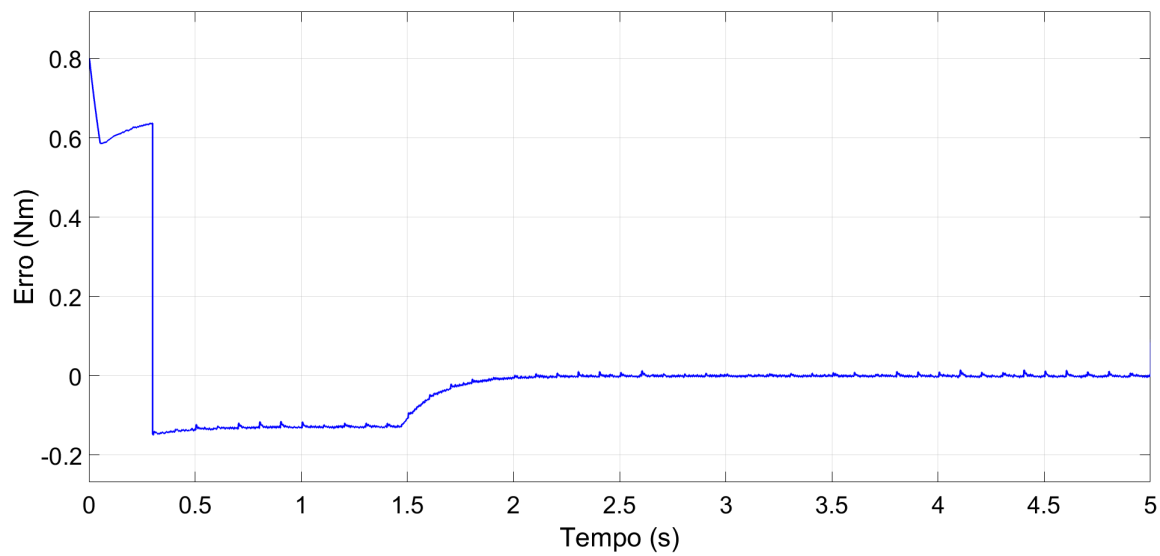
Figura 108 – Sinal do torque estimado com filtro de Kalman em função do tempo.



Fonte: Próprio autor.

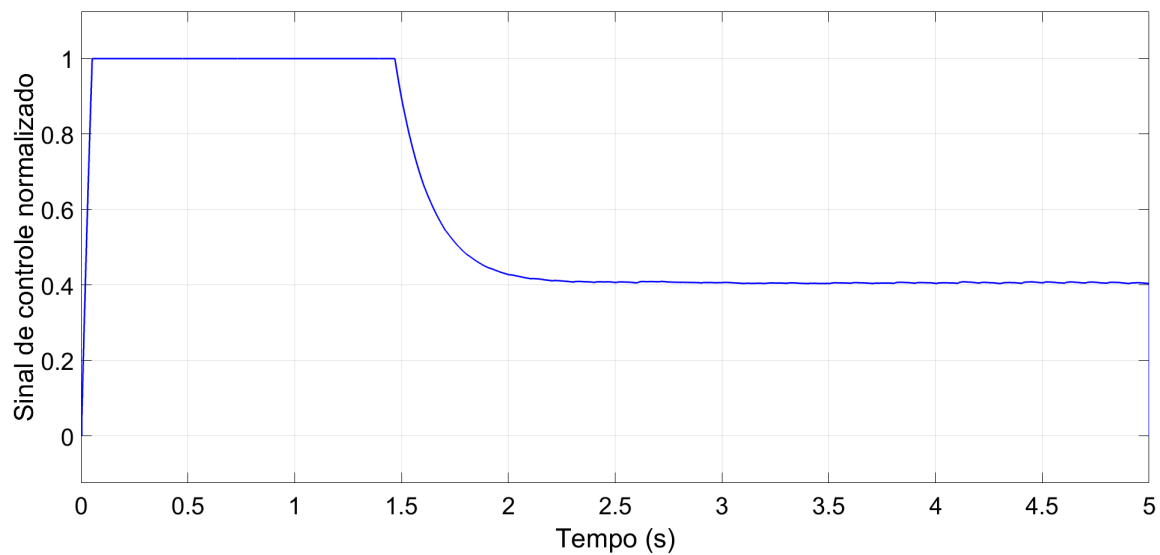
Observando-se o gráfico do erro na Figura 109, conclui-se que é possível estabilizar o controle de torque, o que torna viável a tentativa de execução do controle global-local apresentada no Capítulo 6.

Figura 109 – Erro entre o torque requerido e o estimado pelo filtro de Kalman em função do tempo.



Fonte: Próprio autor.

Figura 110 – Sinal de controle do ciclo ativo do PWM em função do tempo.



Fonte: Próprio autor.

Por fim, nota-se que o acionamento do controle de torque bem aplicado reduz consideravelmente o ruído do sistema de controle.

5.4 Conclusão do capítulo

Conclui-se neste capítulo que ainda que o método de leitura da velocidade baseado na média possua maior precisão de medida no regime estacionário, este possui um atraso de resposta que pode afetar o sistema, enquanto o método baseado na velocidade instantânea ainda que possua menor precisão devido a alta taxa de ruído é mais rápida e pode ser filtrada para melhores leituras.

Também notou-se que a simulação de um sistema por meio da planta do sistema pode obter diferentes resultados que o sistema real, devido a fatores ausentes em um sistema ideal, como ruídos e variáveis externas. Desta forma é importante que os primeiros testes sempre possuam algum tipo de proteção contra imprevistos para não danificar o sistema.

Para sensores ruidosos, o ajuste perfeito do período de amostragem é muito importante, uma vez que uma frequência de amostragem muito baixa resulta em um controle lento e instável, no entanto para uma frequência muito alta o ruído do sensor afeta a estabilidade do sistema, tornando-o novamente instável. Notou-se portanto que a redução do período de amostragem de 0,1 ms para 1 ms tornou o sistema muito mais estável mesmo para o controle com realimentação direta sem observador de estados.

Concluiu-se também que para a modelagem matemática a identificação paramétrica realizada com o equipamento completamente montado possui um resultado mais estável que para a identificação feita por análise de cada parte de modo independente.

Observou-se ainda que a utilização do filtro de Kalman é muito útil para sistemas com muitos ruídos de medição, podendo ser a diferença entre a estabilidade ou não do sistema. Com a utilização do filtro obteve-se, portanto, os melhores resultados além também da possibilidade do controle de torque.

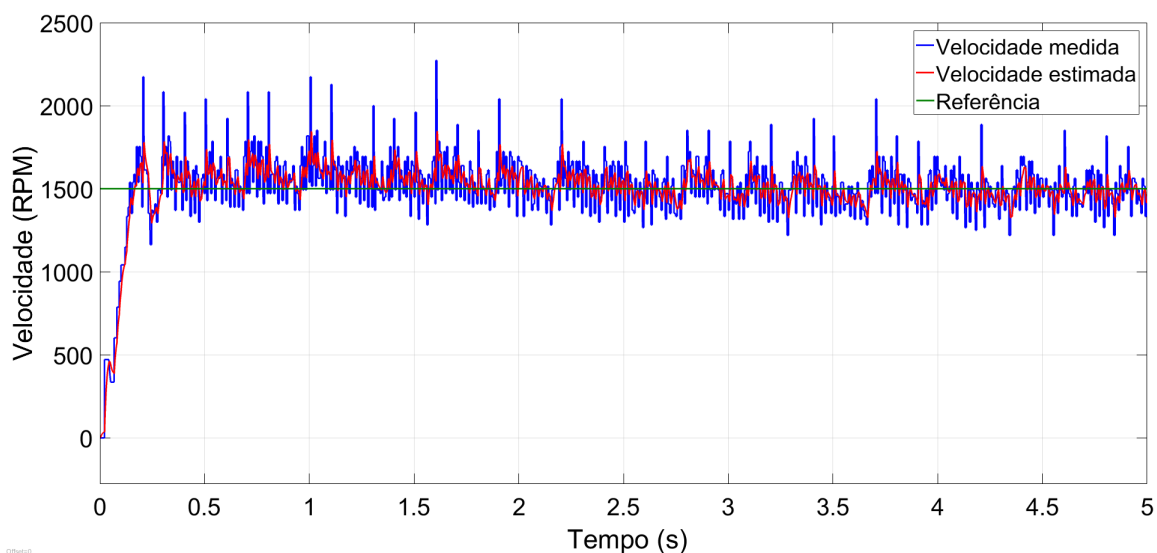
No Capítulo 6, a seguir é analisado o desempenho do controle global-local, no qual o global tende a controlar a velocidade do sistema, enquanto o local controla o torque necessário para a estabilização do global.

6 Controle global-local

Neste capítulo será apresentado um estudo de controle global-local, onde controla-se a velocidade através da realimentação do valor do torque, mostrar-se-á também um estudo deste sistema sob cargas variadas, com a intenção de testar o desempenho do controlador.

Aplicando o sistema da Seção 5.3.3 em um controlador global-local, no qual o controle de torque local PI é controlado por um PID global que define o torque de referência necessário baseado na velocidade desejada do sistema global. O controle global simulado neste caso é representado por um PID de parâmetros $K_p = 0,0002$, $K_I = 0,0002$, $K_d = 0,0001$ e filtro $N = 100$ com velocidade desejada igual a 1500 RPM. Resultando nos gráficos das Figuras de 111 a 115.

Figura 111 – Sinal de velocidade medido e estimado pelo filtro de Kalman, em função do tempo.

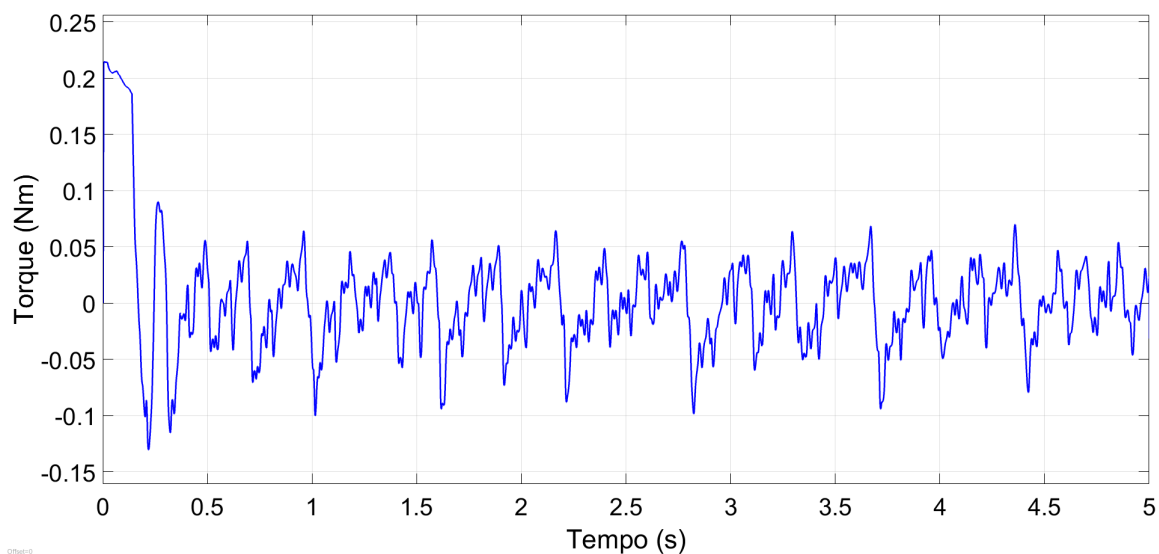


Fonte: Próprio autor.

Nota-se na Figura 111 que o sistema foi capaz de atingir estabilidade com uma oscilação, devida ao ajuste do PID global não ser ótimo.

Observando-se a Figura 112 nota-se que originalmente há uma oscilação na velocidade porque o controle global está com um tempo de estabilização alto, fazendo com que o sistema não estabilize conforme desejado. Entretanto o controlador local tem uma estabilização muito melhor, como pode-se observar na Figura 113.

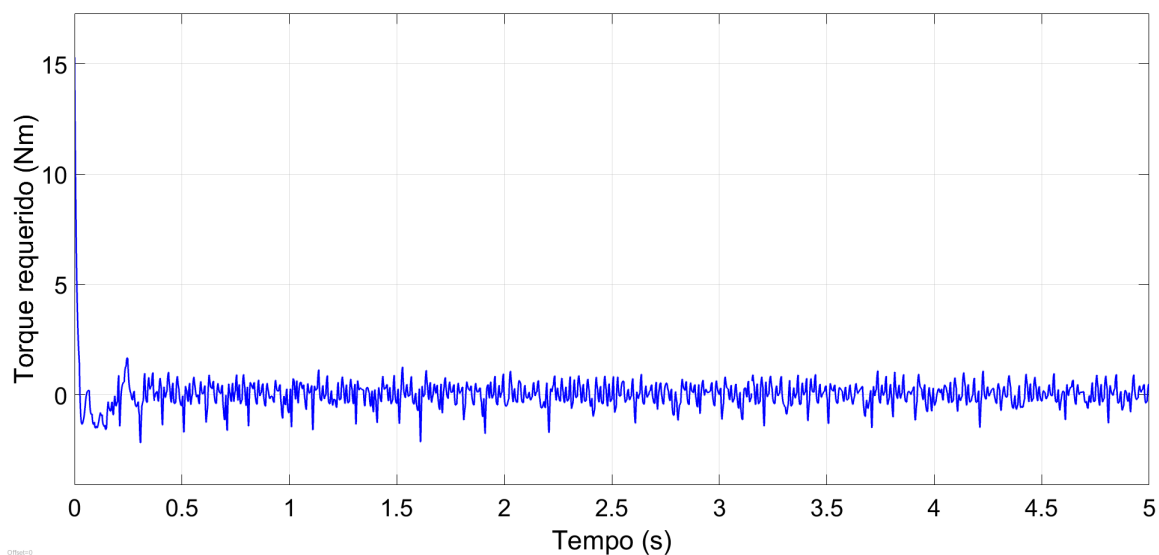
Figura 112 – Sinal do torque estimado pelo filtro de Kalman em função do tempo.



Fonte: Próprio autor.

Neste caso, há apenas a oscilação da corrente de acoplamento entre o pinhão e a coroa, o que pode ser otimizado por melhoras mecânicas, como por exemplo uso de engrenagens.

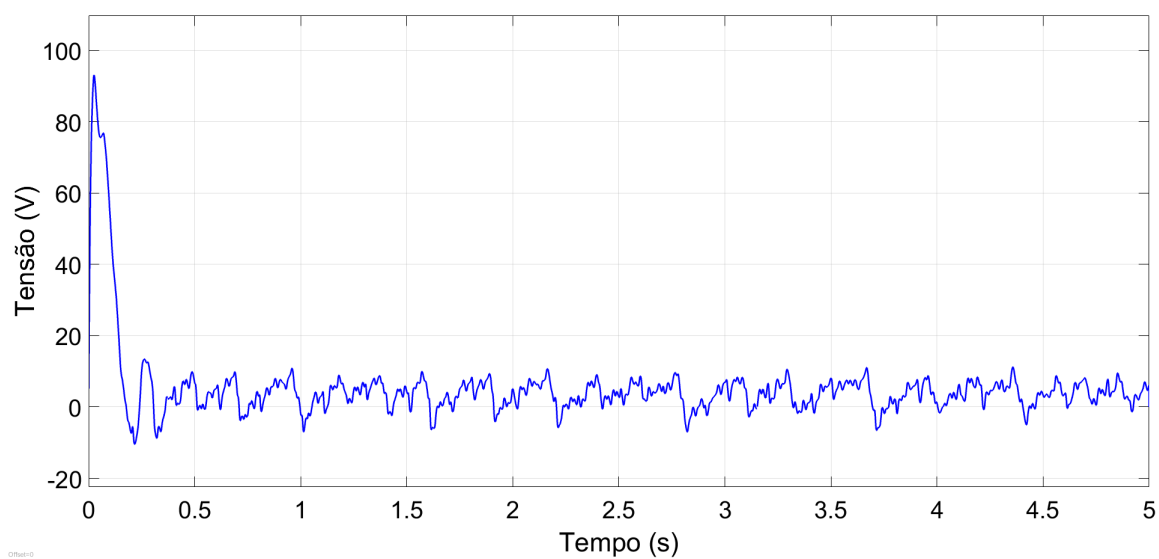
Figura 113 – Erro entre o torque requerido e estimado pelo filtro de Kalman, em função do tempo.



Fonte: Próprio autor.

Nota-se na Figura 114 a validação do controle de torque, uma vez que, ainda que com a presença de oscilação no regime permanente o valor da tensão não é saturado a não ser na partida do motor.

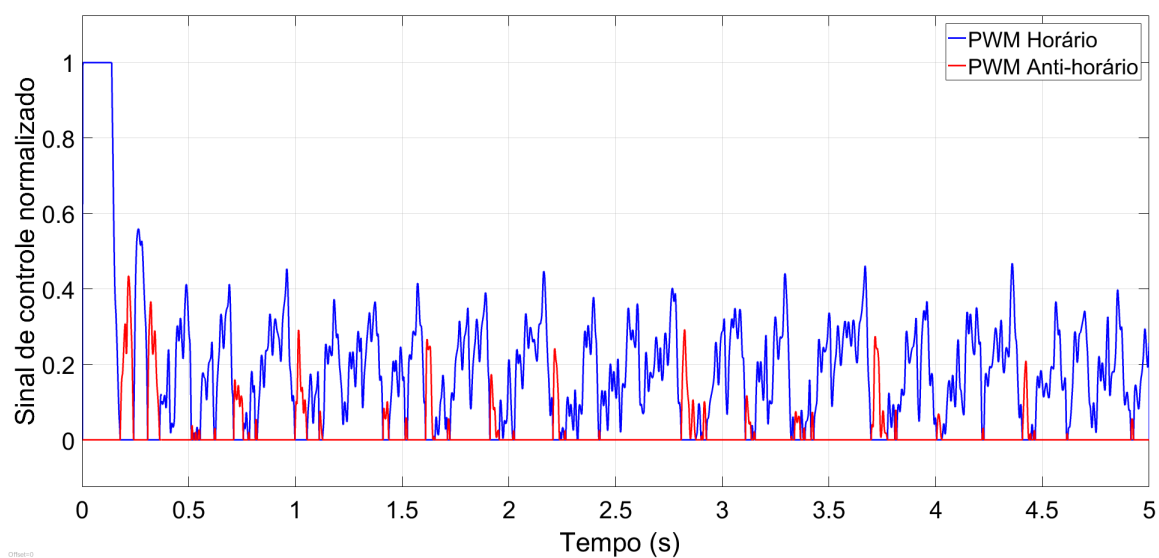
Figura 114 – Sinal de controle em função do tempo.



Fonte: Próprio autor.

Resultando-se em uma distribuição do acionamento dos PWMs como mostra a Figura 115.

Figura 115 – Sinal de controle do ciclo ativo dos PWMs em função do tempo.



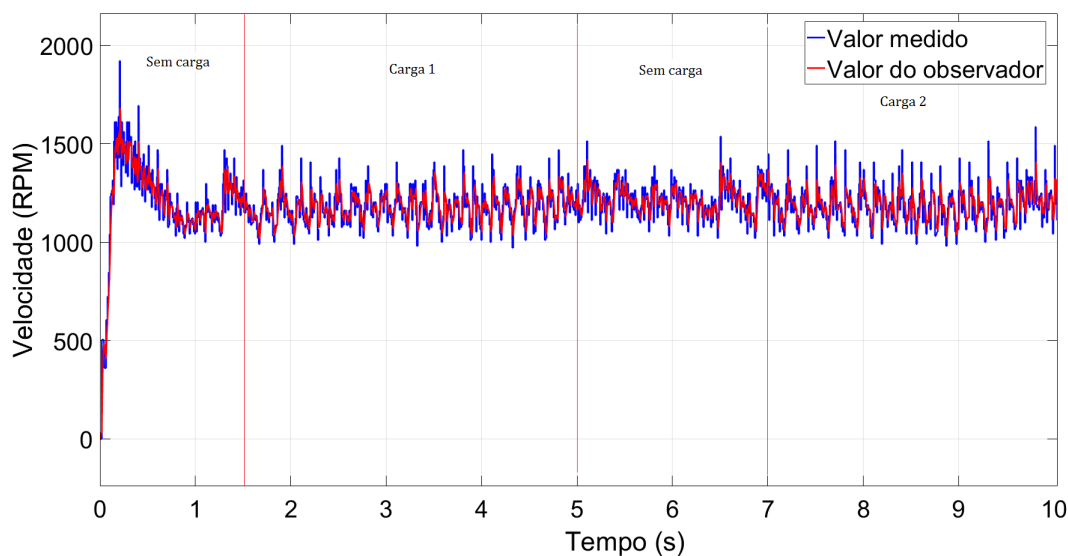
Fonte: Próprio autor.

6.1 Atuação do controle sob cargas variadas

De forma a se analisar melhor os efeitos do controlador de torque sob cargas variadas montou-se um sistema de alavanca sobre a roda adicionando-se carga sobre esta. Com a observação da resposta do sistema a esta variação de carga gerou-se os gráficos das Figuras 116 a 120.

As cargas foram adicionadas por meio de uma alavanca de modo a frear a roda, entretanto, devido a dificuldade da obtenção de equipamentos para medição do atrito gerado por esta carga, apenas mediu-se as cargas, sendo estas respectivamente de 10 kg e 15 kg.

Figura 116 – Velocidade de rotação do motor em RPM, em função do tempo, com alteração de cargas.

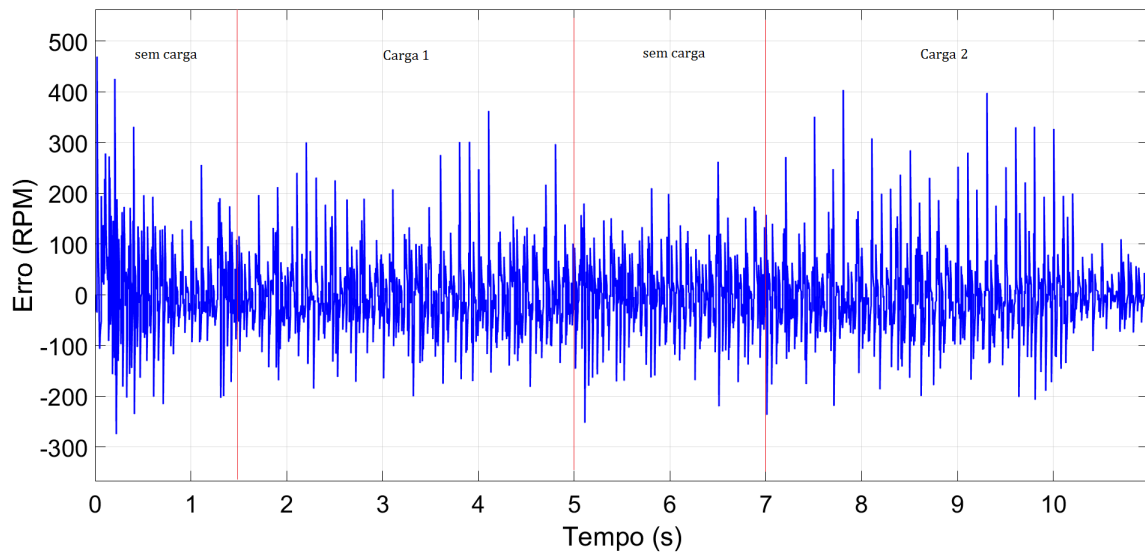


Fonte: Próprio autor.

A Figura 116 apresenta a resposta da velocidade lida pelo sensor e observada pelo filtro de Kalman em consequência da variação de carga sobre a roda. O sistema é iniciado sem adição de carga, com alvo de 1250 RPM, após 1,5 s adiciona-se a carga mais leve que é mantida por 3,5 s. Nota-se durante este período não há transição visível de adaptação do controlador, onde a única diferença se dá na redução da oscilação mecânica, causada pelo conjunto corrente-pinhão.

No tempo igual 5 s, retira-se toda a carga por 2 s, onde nota-se novamente a rápida adaptação do controlador para o controle da velocidade. Por fim, adiciona-se a carga mais pesada repetindo o desempenho obtido até o momento. O experimento é finalizado no tempo igual a 10 s.

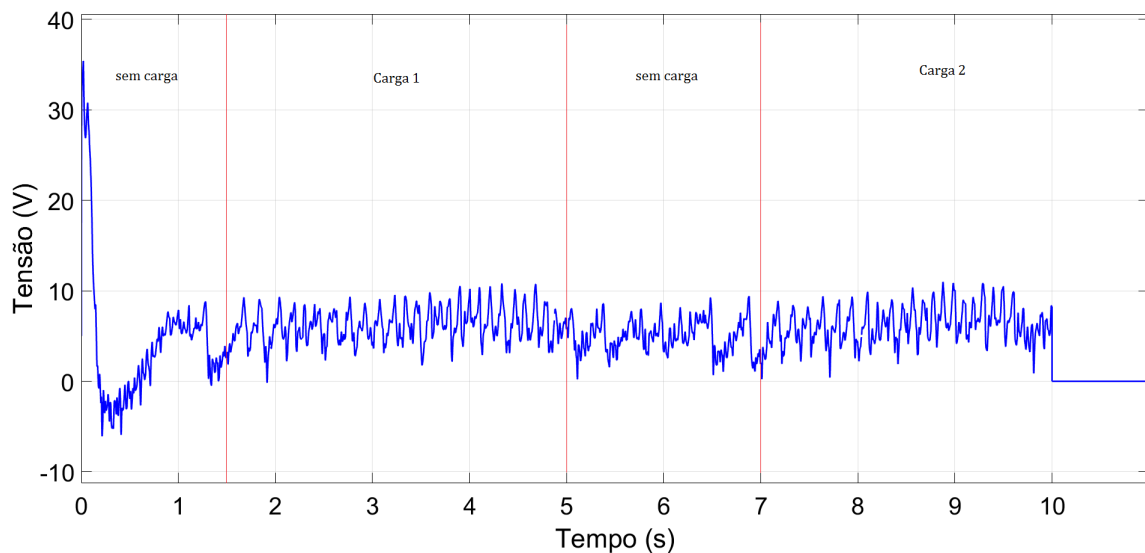
Figura 117 – Sinal do ruído de medição eliminado pelo filtro de Kalman em RPM, em função do tempo, com alteração de cargas.



Fonte: Próprio autor.

A Figura 117 apresenta o sinal de ruído de medição da velocidade eliminado pelo filtro de Kalman, onde é possível notar que, como esperado, a velocidade estimada está em concordância com o valor medido, uma vez que não possui *offset* no sinal nem mesmo nas transições da carga adicionada.

Figura 118 – Tensão de alimentação em V, em função do tempo, com alteração de cargas.

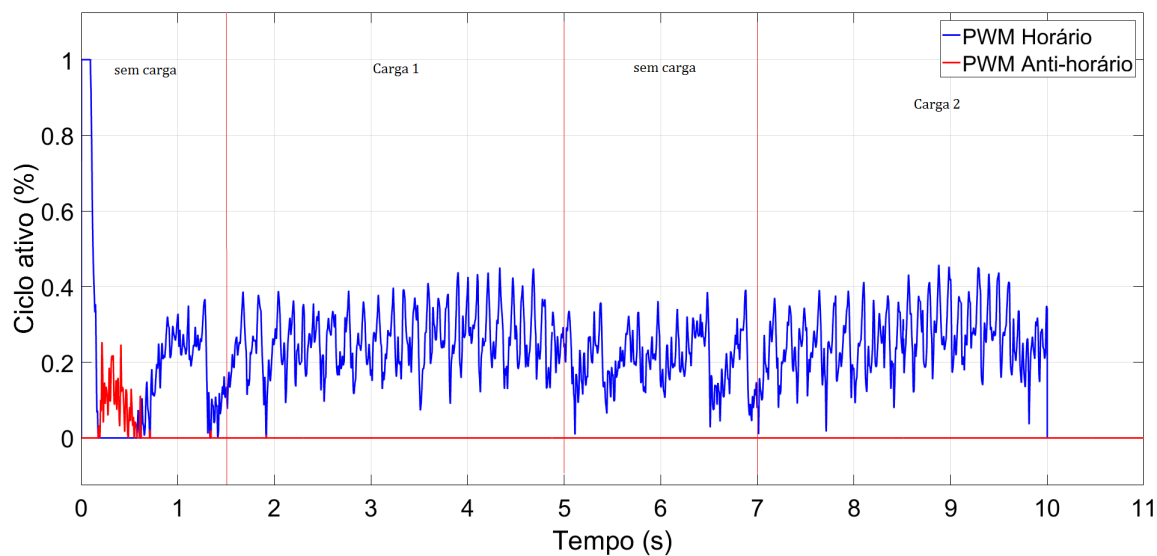


Fonte: Próprio autor.

Nota-se que na Figura 118 que o valor da tensão média de alimentação altera quando há variação na carga sobre a roda, fazendo com que o torque gerado seja alterado também de

forma a manter a velocidade constante conforme desejado.

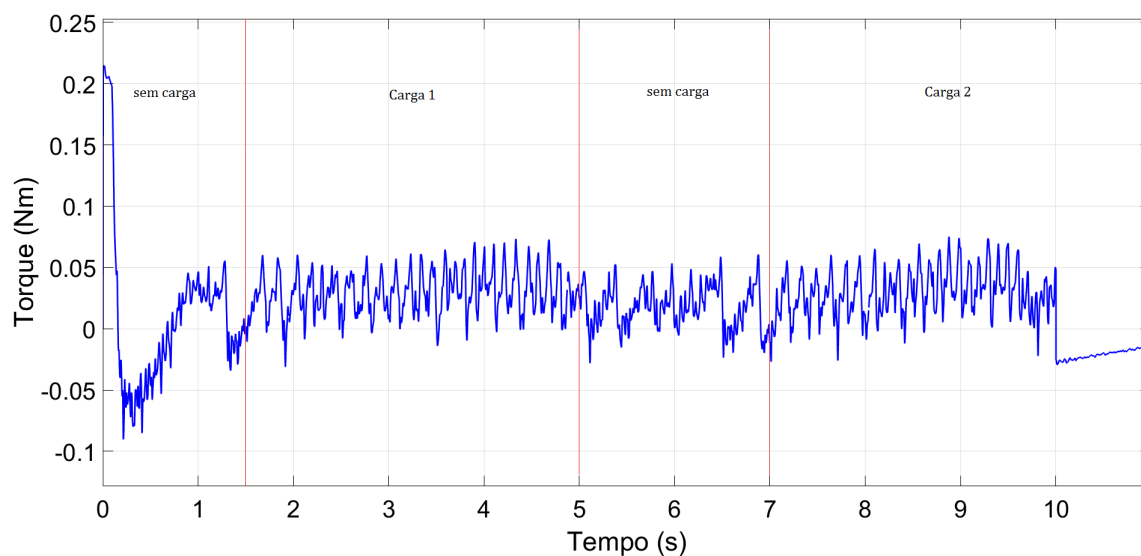
Figura 119 – Sinal de PWM para acionamento do controlador, em função do tempo, com alteração de cargas.



Fonte: Próprio autor.

Com o sistema proposto nesta seção percebe-se na Figura 119 que o ruído já não chega a acionar o sentido inverso do acionador do motor e portanto se mostra mais consistente para aplicação em um sistema real, no qual haverá menor desgaste do equipamento.

Figura 120 – Sinal de controle do ciclo ativo dos PWMs em função do tempo, com alteração de cargas.



Fonte: Próprio autor.

Por fim, a Figura 120 apresenta o sinal do torque gerado para tal aplicação do controle, percebe-se assim a alteração de torque gerado para as diferentes cargas, de forma a se manter a velocidade assim como a desejada.

6.2 Conclusão do capítulo

É possível utilizar o controle de torque para atuar na velocidade de giro do motor, com isso podem-se limitar os valores de velocidade de rotação e escorregamento, mantendo-se desta forma o veículo a uma velocidade constante.

O Capítulo seguinte apresenta as conclusões finais do trabalho.

7 Conclusões

O estudo de novas arquiteturas de tração para veículos são cada vez mais relevantes para o mercado, uma vez que veículos elétricos estão cada vez mais populares o desenvolvimento de tecnologias para melhor desempenho, segurança, economia dentre outros são de interesse universal. Dentre as estratégias de tração que têm sido levantadas para tais melhoras a de motores independentes têm se mostrado a mais promissora. Há diversos métodos que podem ser utilizados para levantamento do modelo matemático de um sistema, notou-se contudo que os métodos gráficos apresentam resultados semelhantes entre si.

A utilização do modelo matemático do motor apresenta um resultado muito próximo dos métodos gráficos de identificação paramétrica do sistema, entretanto este é o único modelo que possibilita a observação de estados que não possuem sensores e, portanto, deve ser considerado para sistemas de baixo custo e não invasivos. Contudo percebeu-se que a identificação paramétrica do sistema como um todo é mais eficiente e apresenta resultados mais estáveis para quando o sistema está montado por completo do que se cada parte for modelada separadamente.

Observou-se também que sistema em *hardware-in-the-loop* possibilitado pelo *MatLab*, aliado ao ambiente de desenvolvimento *Simulink* com programação por diagrama de blocos, permitiu um ajuste rápido de parâmetros e observação de sinais e variáveis, reduzindo tempo de pesquisa no desenvolvimento de códigos de baixo nível.

Os ajustes de parâmetros do controlador PID podem ser rapidamente alterados com a utilização do sistema HIL, sem necessidade de *softwares* específicos tornando o processo de estudo e testes muito mais rápido e proveitoso, além proteções criadas no *Simulink*, por meio de diagrama de blocos para evitar danos ao equipamento por erro de projeto.

Os dois métodos de leitura de velocidade apresentaram vantagens e desvantagens, no qual o primeiro, baseado na média apresentava maior precisão nas medidas, entretanto o tempo de resposta é maior. Contudo o segundo método possui um tempo de resposta menor e maior amplitude de ruído, prejudicando a medida exata da velocidade, podendo este ser filtrado de forma a se obter uma medida rápida e mais precisa.

Notou-se que o ajuste do valor da frequência de amostragem é muito importante para a estabilidade do sistema, uma vez que ruídos estocásticos podem ter maior efeito sobre a estabilidade com alta frequência, enquanto um longo período de amostragem torna a resposta lenta e possivelmente instável.

A aplicação do filtro de Kalman pode ser fator decisivo para estabilidade de um sistema no qual a saída possui uma potência do ruído maior que 10% do valor do sinal, uma vez que este pode ser calculado para reduzir o ruído, tendo um impacto pequeno sobre o sistema. Além de possibilitar a observação de estados sem sensores específicos, o que permite a aplicação do controle de torque e de velocidade controlador por torque.

Destaca-se que é possível controlar a velocidade de um motor c.c. qualquer apenas com um sensor de baixo custo, por meio da identificação do sistema e aplicação do filtro de Kalman com um controlador em malha fechada.

7.1 Sugestões para trabalhos futuros

Além dos experimentos descritos, sugere-se para trabalhos futuros a aplicação do sistema proposto de controle de velocidade por torque em um robô móvel com tração diferencial. Pode também tracionar o mesmos sobre superfícies escorregadias, de forma a observar o controle de torque do sistema.

O desenvolvimento de um sistema melhor de potência e acionamento do motor poderá reduzir os ruídos apresentados neste trabalho, o que acarretaria em melhores resultados e eficiência do controlador. A adição de um barramento capacitivo para o acionamento do motor poderia melhorar a capacidade do circuito em prover picos de corrente.

A adição de fontes separadas para cada circuito de acionamento do motor, incluindo uma fonte específica para o sensor de velocidade e para alimentação do microcontrolador podem reduzir consideravelmente o ruído na leitura da velocidade e melhorar o desempenho geral do sistema.

Como discutido no decorrer do Capítulo 5 o filtro de Kalman possui uma função importante não só de estimar o torque atual, mas como de redução do ruído de medição. A aplicação de uma técnica de otimização dos parâmetros de inicialização ($P[0]$, Q e R) podem melhorar o desempenho do observador e produzir resultados ainda melhores.

Sugere-se ainda para trabalhos futuros a utilização do PID com técnica *backward Euler* ou *Tustin*. Assim como a utilização do modelo ARMAX (*Autoregressive–moving-average model with exogenous inputs*) para identificação do sistema pode produzir resultados melhores que o método ARMA utilizado.

Referências

- ALI, D. et al. Experimental testing of observers comprising discrete kalman filter and high-gain observers. In: IEEE. *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*. [S.l.], 2019. p. 1–5.
- APRÁIZ, M. de; DIEGO, R.; BARROS, J. An extended kalman filter approach for accurate instantaneous dynamic phasor estimation. *Energies*, Multidisciplinary Digital Publishing Institute, v. 11, n. 11, p. 2918, 2018.
- CHU, M.; CHU, J. Graphical robust pid tuning based on uncertain systems for disturbance rejection satisfying multiple objectives. *International Journal of Control, Automation and Systems*, Springer, v. 16, n. 5, p. 2033–2042, 2018.
- COELHO, A. A. R.; COELHO, L. dos S. *Identificação de sistemas dinâmicos lineares*. [S.l.: s.n.], 2004.
- COLEY, G. Beaglebone black system reference manual. *Texas Instruments, Dallas*, 2013.
- DORF, R. C.; BISHOP, R. H. *Modern control systems*. [S.l.]: Pearson, 2011.
- GUO, G.; MA, Z.; QIAO, J. State-pid feedback control with application to a robot vibration absorber. *International Journal of Modelling, Identification and Control*, Inderscience Publishers, v. 1, n. 1, p. 38–43, 2006.
- HÄGGLUND, T.; ÅSTRÖM, K. J. Industrial adaptive controllers based on frequency response techniques. *Automatica*, v. 27, n. 4, p. 599–609, 1991.
- HEIKKINEN, J.; SERYKH, E. V.; MINAV, T. A control strategy for an autonomous differential drive mobile robot. In: IEEE. *Control in Technical Systems (CTS), 2017 IEEE II International Conference on*. [S.l.], 2017. p. 83–86.
- KALMAN, R. E. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, American Society of Mechanical Engineers, v. 82, n. 1, p. 35–45, 1960.
- KARA, T.; EKER, I. Nonlinear modeling and identification of a dc motor for bidirectional operation with real time experiments. *Energy Conversion and Management*, Elsevier, v. 45, n. 7-8, p. 1087–1106, 2004.
- KRISTINSSON, K.; DUMONT, G. A. System identification and control using genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, IEEE, v. 22, n. 5, p. 1033–1046, 1992.
- KUNTANAPREEDA, S. Traction control of electric vehicles using sliding-mode controller with tractive force observer. *International Journal of Vehicular Technology*, Hindawi, v. 2014, 2014.
- LE, H. T. T.; NGUYEN, N. H.; NGUYEN, P. D. Pid adaptive tuning with the principle of receding horizon. In: SPRINGER. *International Conference on Engineering Research and Applications*. [S.l.], 2018. p. 497–503.
- LI, B. et al. An optimal torque distribution control strategy for four-independent wheel drive electric vehicles. *Vehicle System Dynamics*, Taylor & Francis, v. 53, n. 8, p. 1172–1189, 2015.

- LJUNG, L. Perspectives on system identification. *Annual Reviews in Control*, Elsevier, v. 34, n. 1, p. 1–12, 2010.
- MAGALLAN, G. A.; ANGELO, C. H. D.; GARCIA, G. O. Maximization of the traction forces in a 2wd electric vehicle. *IEEE Transactions on Vehicular Technology*, IEEE, v. 60, n. 2, p. 369–380, 2011.
- MAUER, G. F. A fuzzy logic controller for an abs braking system. *IEEE Transactions on Fuzzy Systems*, IEEE, v. 3, n. 4, p. 381–388, 1995.
- MOLLENKAMP, R. A. Controle automático de processos. *EBRAS Editora Brasileira–SMAR*, 1988.
- MUTOH, N. Driving and braking torque distribution methods for front-and rear-wheel-independent drive-type electric vehicles on roads with low friction coefficient. *IEEE Transactions on Industrial Electronics*, IEEE, v. 59, n. 10, p. 3919–3933, 2012.
- NARENDRA, P. M.; FITCH, R. C. Real-time adaptive contrast enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3, n. 6, p. 655–661, nov. 1981.
- NELLES, O. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. [S.l.]: Springer Science & Business Media, 2013.
- OGATA, K. *Modern Control Engineering*. [S.l.]: New Jersey: prentice Hall, 2010.
- OGATA, K. Engenharia de controle moderno. *KATSUHIKO Ogata, 5th Ed. 801p*, 2011.
- PEREIRA, D. S.; PINTO, J. O. Genetic algorithm based system identification and pid tuning for optimum adaptive control. In: *IEEE. Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on*. [S.l.], 2005. p. 801–806.
- PHAM, D.; LIU, X. Neural networks for identification, prediction and control. 1995.
- PHILLIPS, C. L.; NAGLE, H. T. *Digital control system analysis and design*. [S.l.]: Prentice Hall Press, 2007.
- RAJAMANI, R. *Vehicle dynamics and control*. [S.l.]: Springer Science & Business Media, 2011.
- SUJITJORN, S.; WIBOONJAROEN, W. State-pid feedback for pole placement of lti systems. *Mathematical Problems in Engineering*, Hindawi, v. 2011, 2011.
- SUNDARESAN, K.; PRASAD, C. C.; KRISHNASWAMY, P. Evaluating parameters from process transients. *Industrial & Engineering Chemistry Process Design and Development*, ACS Publications, v. 17, n. 3, p. 237–241, 1978.
- WANG, Y.; FUJIMOTO, H.; HARA, S. Driving force distribution and control for ev with four in-wheel motors: A case study of acceleration on split-friction surfaces. *IEEE Transactions on Industrial Electronics*, IEEE, v. 64, n. 4, p. 3380–3388, 2017.
- WILL, A.; HUI, S.; ZAK, S. Sliding mode wheel slip controller for an antilock braking system. *International Journal of Vehicle Design*, Inderscience Publishers, v. 19, n. 4, p. 523–539, 1998.

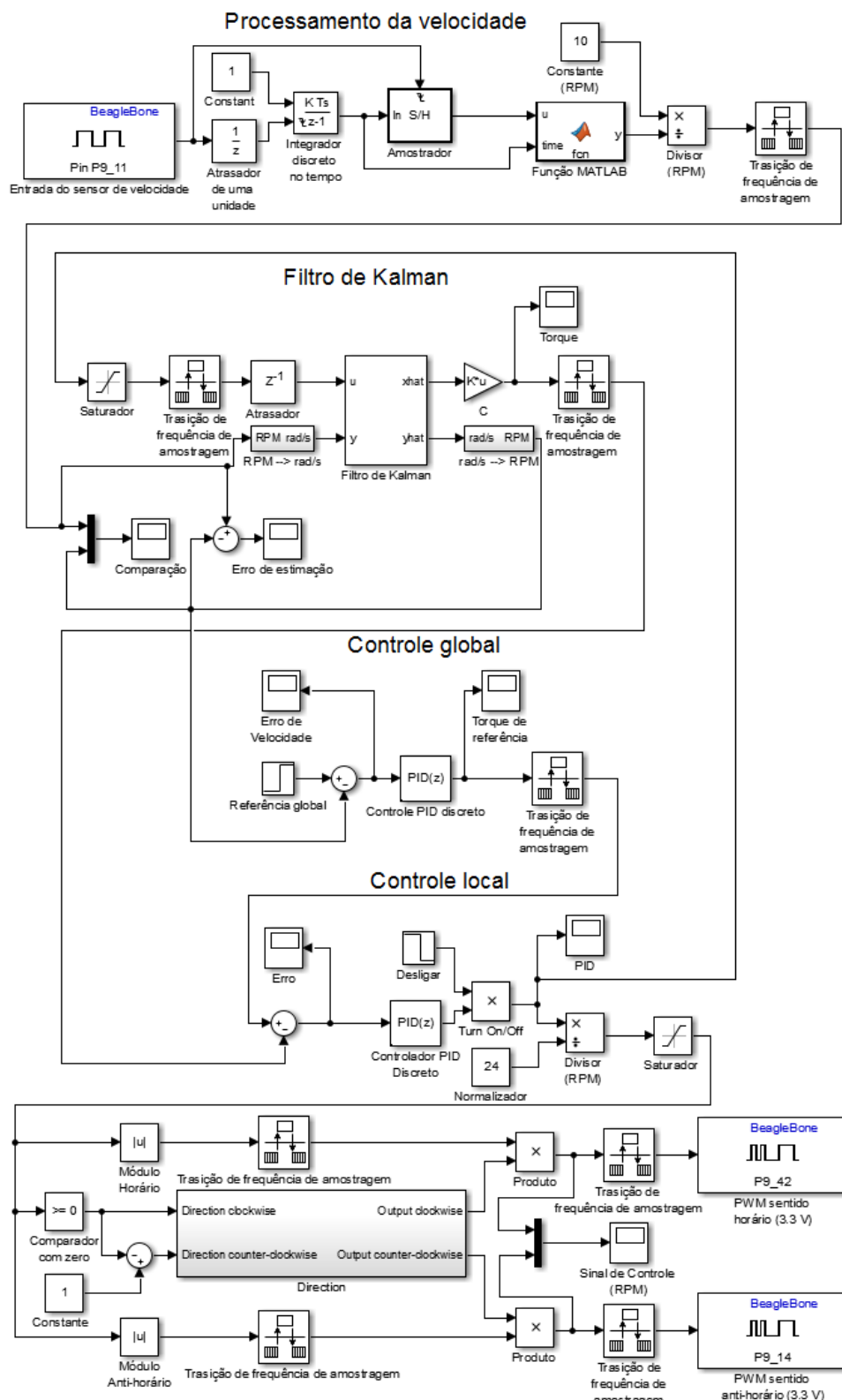
YIN, D.; HORI, Y. Traction control for ev based on maximum transmissible torque estimation. *International Journal of Intelligent Transportation Systems Research*, Springer, v. 8, n. 1, p. 1–9, 2010.

YIN, D.; OH, S.; HORI, Y. A novel traction control for ev based on maximum transmissible torque estimation. *IEEE Transactions on Industrial Electronics*, IEEE, v. 56, n. 6, p. 2086–2094, 2009.

ZIEGLER, J. G.; NICHOLS, N. B. Optimum settings for automatic controllers. *trans. ASME*, v. 64, n. 11, 1942.

APÊNDICE A – Controlador completo

Figura 121 – Projeto do controle de velocidade por torque, com filtro de Kalman.



Fonte: Próprio autor.

ANEXO A – BeagleBone Black Datasheet

REF: BBONEBLK_SRM

BeagleBone Black System
Reference Manual

Rev A5.2

4.2 BeagleBone Black Features and Specification

This section covers the specifications and features of the board and provides a high level description of the major components and interfaces that make up the board.

Table 2 provides a list of the features.

Table 2. BeagleBone Black Features

	Feature
Processor	Sitara AM3359AZCZ100
Graphics Engine	1GHz, 2000 MIPS
SDRAM Memory	SGX530 3D, 20M Polygons/S
Onboard Flash	512MB DDR3L 606MHZ
PMIC	2GB, 8bit Embedded MMC
Debug Support	TPS65217C PMIC regulator and one additional LDO.
Power Source	Optional Onboard 20-pin CTI JTAG, Serial Header
PCB	miniUSB USB or DC Jack
Indicators	5VDC External Via Expansion Header
HS USB 2.0 Client Port	3.4" x 2.1"
HS USB 2.0 Host Port	6 layers
Serial Port	1-Power, 2-Ethernet, 4-User Controllable LEDs
Ethernet	Access to USB0, Client mode via miniUSB
SD/MMC Connector	Access to USB1, Type A Socket, 500mA LS/FS/HS
User Input	UART0 access via 6 pin 3.3V TTL Header. Header is populated
Video Out	10/100, RJ45
Audio	microSD , 3.3V
Expansion Connectors	Reset Button Boot Button Power Button
Weight	16b HDMI, 1280x1024 (MAX) 1024x768, 1280x720, 1440x900 w/EDID Support
Power	Via HDMI Interface, Stereo
	Power 5V, 3.3V , VDD_ADC(1.8V) 3.3V I/O on all signals
	McASP0, SPI1, I2C, GPIO(65), LCD, GPMC, MMC1, MMC2, 7 AIN(1.8V MAX), 4 Timers, 3 Serial Ports, CAN0, EHRPWM(0,2), XDMA Interrupt, Power button, Expansion Board ID (Up to 4 can be stacked)
	1.4 oz (39.68 grams)
	Refer to Section 6.1.7

ANEXO B – Sensor de efeito Hall



US1881 Hall Latch – High Sensitivity

Features and Benefits

- ❑ Wide operating voltage range from 3.5V to 24V
- ❑ High magnetic sensitivity – Multi-purpose
- ❑ CMOS technology
- ❑ Chopper-stabilized amplifier stage
- ❑ Low current consumption
- ❑ Open drain output
- ❑ Thin SOT23 3L and flat TO-92 3L both RoHS Compliant packages

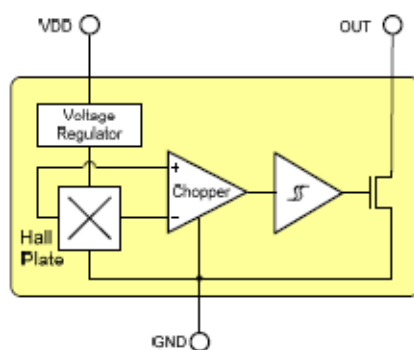
Application Examples

- ❑ Automotive, Consumer and Industrial
- ❑ Solid-state switch
- ❑ Brushless DC motor commutation
- ❑ Speed detection
- ❑ Linear position detection
- ❑ Angular position detection
- ❑ Proximity detection

Ordering Information

Part No.	Temperature Code	Package Code
US1881	E (-40°C to 85°C)	SE (TSOT-3L)
US1881	E (-40°C to 85°C)	UA (TO-92)
US1881	K (-40°C to 125°C)	SE (TSOT-3L)
US1881	K (-40°C to 125°C)	UA (TO-92)
US1881	L (-40°C to 150°C)	SE (TSOT-3L)
US1881	L (-40°C to 150°C)	UA (TO-92)

1 Functional Diagram



2 General Description

The Melexis US1881 is a Hall-effect latch designed in mixed signal CMOS technology.

The device integrates a voltage regulator, Hall sensor with dynamic offset cancellation system, Schmitt trigger and an open-drain output driver, all in a single package.

Thanks to its wide operating voltage range and extended choice of temperature range, it is quite suitable for use in automotive, industrial and consumer applications.

The device is delivered in a Thin Small Outline Transistor (TSOT) for surface mount process and in a Plastic Single In Line (TO-92 flat) for through-hole mount.

Both 3-lead packages are RoHS compliant.