

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”  
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA  
CURSO DE ENGENHARIA ELÉTRICA**

**GIOVANNA FERREIRA CORDEIRO**

**TRABALHO DE GRADUAÇÃO**

**CONTROLE CLÁSSICO DA PLANTA *BALL BALANCER***

**ILHA SOLTEIRA – SP**

**2022**

**GIOVANNA FERREIRA CORDEIRO**

**CONTROLE CLÁSSICO DA PLANTA *BALL BALANCER***

Trabalho de Graduação apresentado à  
Faculdade de Engenharia de Ilha Solteira –  
UNESP como parte dos requisitos para  
obtenção do título de Engenheiro Eletricista.

Orientador: Prof. Dr. Jean Marcos de Souza  
Ribeiro

**ILHA SOLTEIRA – SP**

**2022**

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

C794c Cordeiro, Giovanna Ferreira.  
Controle clássico da planta Ball Balancer / Giovanna Ferreira Cordeiro. -- Ilha Solteira: [s.n.], 2022  
68 f. : il.

Trabalho de conclusão de curso (Graduação em Engenharia Elétrica) -  
Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira, 2022


Orientador: Jean Marcos de Souza Ribeiro  
Inclui bibliografia

1. Controle clássico. 2. Controle PID. 3. Sistema Ball Balancer.

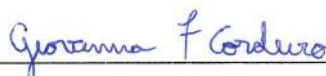
  
Raiane da Silva Santos

## ATA DE DEFESA DE TRABALHO DE GRADUAÇÃO

Aos doze dias do mês de julho do ano de dois mil e vinte e dois, a discente Giovanna Ferreira Cordeiro, matriculada sob o nº162055072, tendo como banca examinadora o seu orientador, o Prof. Dr. Jean Marcos de Souza Ribeiro, o Prof. Dr. Carlos Antonio Alves e o Prof. Dr. Rodrigo Cardim, apresentou o Trabalho de Graduação intitulado "**Controle Clássico da Planta Ball Balancer**" obtendo a nota 9,5 (NOVE E MEIO) e conceito APROVADO.




Prof. Dr. Jean Marcos de Souza Ribeiro  
- orientador -



Giovanna Ferreira Cordeiro  
- discente -



Prof. Dr. Carlos Antonio Alves  
- Membro da Banca -



Prof. Dr. Rodrigo Cardim  
- Membro da Banca -

## RESUMO

Neste trabalho é apresentado um estudo e aplicação experimental de teorias de controle clássico de um sistema instável e não linear. Foi escolhido o sistema *Ball Balancer* como equipamento de implementação. Primeiramente é feita a modelagem matemática, utilizando as leis da física para equacionar as plantas do sistema e suas funções de transferência, obtidas através do sistema linearizado em torno da região de operação. Com isso será feito o cálculo dos controladores necessários, de acordo com a resposta esperada. Também foi elaborado um programa em linguagem MATLAB, que realize os cálculos anteriores de forma interativa com o usuário e execute a simulação em um circuito do SIMULINK. Foi então feita a simulação para cada tipo de controlador, considerando também situações de falha nos motores e possíveis melhorias. Por fim, os resultados serão aplicados no módulo 2DOF *Ball Balancer* da Quanser.

**Palavras-chave:** Controle clássico, *Ball Balancer*, MATLAB

## ABSTRACT

In this work it is presented a study and experimental application of classic control theories in an unstable non linear system. The Ball Balancer system was chosen as implementation equipment. First it is done the mathematical modeling, using the laws of physics to equate the plants of the system and its transfer functions, obtained through the linearized system around the operating region. With that, was made the calculation of the necessary controllers. It will also be elaborated a program in MATLAB language, that performs the previous calculations and runs a SIMULINK circuit. Then, was done the simulation for each kind of controller, also considering engine failure situations. Lastly, the final results will be applied on the Quaser 2DOF *Ball Balancer* module.

**Key-words:** Classic control, *Ball Balancer*, MATLAB

## LISTA DE FIGURAS

|   |    |
|---|----|
| Figura 1 - Visão geral do equipamento .....   | 3  |
| Figura 2 - Posicionamento dos motores .....   | 4  |
| Figura 3 - Visão da mesa pela posição da câmera .....   | 4  |
| Figura 4 - Esquemática do sistema no eixo X.....  | 5  |
| Figura 5 - Forças que agem sobre a bola .....   | 6  |
| Figura 6 – Diagrama de blocos de malha aberta .....   | 9  |
| Figura 7 - Diagrama de blocos do controle em cascata .....  | 10 |
| Figura 8 - Diagrama do controle do motor .....  | 12 |
| Figura 9 - Diagrama do controle da mesa.....  | 13 |
| Figura 10 - Diagrama do controle da mesa com <i>anti-windup</i> .....   | 15 |
| Figura 11 - Esquemático do Simulink .....   | 17 |
| Figura 12 - Posição $X(s)$ , Tensão $V(s)$ e Ângulo $\theta(s)$ para simulação PD inicial com $bsd=1$ .           | 18 |
| Figura 13 - Posição $X(s)$ para simulação PD inicial com $bsd=1$ .....  | 18 |
| Figura 14 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PD inicial com $bsd=0$ ....        | 19 |
| Figura 15 - Posição $X(s)$ para simulação PD inicial com $bsd=0$ .....  | 20 |
| Figura 16 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PD melhorada.....                  | 21 |
| Figura 17 - Posição $X(s)$ para simulação PD melhorada.....   | 21 |
| Figura 18 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PID inicial .....                  | 22 |
| Figura 19 - Posição $X(s)$ para simulação PID inicial .....   | 23 |
| Figura 20 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PID melhorada .....                | 24 |
| Figura 21 - Posição $X(s)$ para simulação PID melhorada .....   | 24 |
| Figura 22 - Esquemático no Simulink com falhas.....   | 26 |
| Figura 23 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PD com 30% de falha....            | 27 |
| Figura 24 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PD com 50% de falha....            | 27 |
| Figura 25 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PD com 70% de falha....            | 28 |
| Figura 26 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PID com 30% de falha ..            | 29 |
| Figura 27 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PID com 50% de falha ..            | 30 |
| Figura 28 - Posição $X(s)$ , tensão $V(s)$ e ângulo $\theta(s)$ para simulação PID com 70% de falha ..            | 30 |
| Figura 29 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PD com entrada quadrada..... | 32 |

|  |    |
|--|----|
| Figura 30 - Posição $Y(s)$ , tensão $V_y(s)$ e ângulo $\theta_y(s)$ para controlador PD com entrada quadrada.....  | 33 |
| Figura 31 - Posição X versus Y para controlador PD com entrada quadrada .....                                      | 34 |
| Figura 32 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PD com entrada senoidal ..... | 34 |
| Figura 33 - Posição $Y(s)$ , tensão $V_y(s)$ e ângulo $\theta_y(s)$ para controlador PD com entrada senoidal ..... | 35 |
| Figura 34 - Posição X versus Y para controlador PD com entrada senoidal.....                                       | 36 |
| Figura 35 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PID com entrada quadrada..... | 37 |
| Figura 36 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PID com entrada quadrada..... | 37 |
| Figura 37 - Posição X versus Y para controlador PID com entrada quadrada.....                                      | 38 |
| Figura 38 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PID com entrada senoidal..... | 39 |
| Figura 39 - Posição $Y(s)$ , tensão $V_y(s)$ e ângulo $\theta_y(s)$ para controlador PID com entrada senoidal..... | 39 |
| Figura 40 - Posição X versus Y para controlador PID com entrada senoidal .....                                     | 40 |
| Figura 41 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PD com falha de 30% .....     | 41 |
| Figura 42 - Posição $Y(s)$ , tensão $V_y(s)$ e ângulo $\theta_y(s)$ para controlador PD com falha de 30% .....     | 41 |
| Figura 43 - Posição x versus y para controlador PD com falha de 30% .....  | 42 |
| Figura 44 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PD com falha de 50% .....     | 42 |
| Figura 45 - Posição $Y(s)$ , tensão $V_y(s)$ e ângulo $\theta_y(s)$ para controlador PD com falha de 50% .....     | 43 |
| Figura 46 - Posição x versus y para controlador PD com falha de 50% .....  | 43 |
| Figura 47 - Posição $X(s)$ , tensão $V_x(s)$ e ângulo $\theta_x(s)$ para controlador PD com falha de 70% .....     | 44 |
| Figura 48 - Posição $Y(s)$ , tensão $V_y(s)$ e ângulo $\theta_y(s)$ para controlador PD com falha de 70% .....     | 45 |
| Figura 49 - Posição x versus y para controlador PD com falha de 70% .....  | 45 |

Figura 50 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PID com falha de 30% ..... 46

Figura 51 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PID com falha de 30% ..... 46

Figura 52 - Posição  $x$  versus  $y$  para controlador PID com falha de 30% ..... 47

Figura 53 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PID com falha de 50% ..... 47

Figura 54 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PID com falha de 50% ..... 48

Figura 55 - Posição  $x$  versus  $y$  para controlador PID com falha de 50% ..... 48

Figura 56 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PID com falha de 70% ..... 49

Figura 57 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PID com falha de 70% ..... 49

Figura 58 - Posição  $x$  versus  $y$  para controlador PID com falha de 70% ..... 50

**LISTA DE TABELAS**

Tabela 1 - Parâmetros do equipamento ..... 7  
Tabela 2 - Efeito dos controladores PID ..... 11

**LISTA DE ABREVIATURAS**

- L – Comprimento da viga  
x – Distância da bola ao ponto zero  
 $\alpha$  – Ângulo de inclinação da barra  
 $\theta$  – Ângulo de rotação do motor  
r – Raio do eixo do motor  
m – Massa da bola  
rb – Raio da bola  
g – Aceleração da gravidade  
Fp – Força peso  
Fa – Força de atrito sobre o ponto de contato  
Fn – Força normal do plano sobre a bola  
Ib – Momento de inércia  
Kbb – Ganho do sistema  
Pbb – Função de transferência do movimento da bola  
Ps – Função de transferência do motor  
V – Tensão de entrada no motor  
X – Posição da bola no eixo x  
Y – Posição da bola no eixo y  
K – Ganho do motor  
 $\tau$  – Constante de tempo do motor  
P – Função de transferência resultante  
Xd – Posição desejada da bola  
X – Posição real da bola  
Cbb – Controlador da mesa  
Cs – Controlador do motor  
 $\theta_d$  – Ângulo de rotação necessário do motor  
 $\theta$  – Ângulo real de rotação do motor  
V – Tensão necessária para causar a rotação do motor  
 $\xi$  – Coeficiente de amortecimento  
 $\omega_n$  – Frequência natural  
po – Localização do polo

$T_p$  – Constante de decaimento do polo

ess – Erro em regime estacionário

$t_p$  – Tempo de pico

PO – Porcentagem de overshoot

$t_s$  – Tempo de assentamento

cts – Porcentagem de tempo de assentamento

$K_{ps}$  – Ganho proporcional do motor

$K_{ds}$  – Ganho derivativo do motor

$K_{pbb}$  – Ganho proporcional da mesa

$K_{dbb}$  – Ganho derivativo da mesa

$K_{ibb}$  – Ganho integrativo da mesa

INT\_MAX – Limite de saturação do integrador

$T_r$  – Constante de tempo do anti-windup

## SUMÁRIO

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>INTRODUÇÃO.....</b>                                     | <b>1</b>  |
| <b>2</b> | <b>EQUIPAMENTO BALL BALANCER 2 DOF .....</b>               | <b>3</b>  |
| <b>3</b> | <b>EQUACIONAMENTO DA PLANTA DO SISTEMA .....</b>           | <b>5</b>  |
| 3.1      | FUNÇÃO DE TRANSFERÊNCIA .....                              | 8         |
| 3.2      | MODELAGEM MATEMÁTICA COM O MOTOR.....                      | 8         |
| <b>4</b> | <b>CONTROLE EM CASCATA DO SISTEMA BALL BALANCER .....</b>  | <b>10</b> |
| 4.1      | RESPOSTA ESPERADA.....                                     | 10        |
| 4.2      | CONTROLADORES PID .....                                    | 11        |
| 4.3      | CONTROLE DO MOTOR.....                                     | 12        |
| 4.4      | CONTROLADOR DA MESA .....                                  | 13        |
| <b>5</b> | <b>PROGRAMA DE CÁLCULO E SIMULAÇÕES .....</b>              | <b>16</b> |
| 5.1      | PROGRAMA DE CÁLCULO.....                                   | 16        |
| 5.2      | SIMULAÇÕES .....   | 16        |
| 5.2.1    | <i>Controlador PD para motor e mesa .....</i>              | <i>17</i> |
| 5.2.2    | <i>Controlador PD para o motor e PID para a mesa .....</i> | <i>22</i> |
| 5.2.3    | <i>Simulação de falhas.....</i>                            | <i>25</i> |
| <b>6</b> | <b>RESULTADOS EXPERIMENTAIS COM O BALL BALANCER .....</b>  | <b>32</b> |
| 6.1      | CONTROLADOR PD PARA ENTRADA QUADRADA .....                 | 32        |
| 6.2      | CONTROLADOR PD PARA ENTRADA SENOIDAL .....                 | 34        |
| 6.3      | CONTROLADOR PID PARA ENTRADA QUADRADA .....                | 36        |
| 6.4      | CONTROLADOR PID PARA ENTRADA SENOIDAL.....                 | 38        |
| 6.5      | RESULTADOS COM FALHAS.....                                 | 40        |
| <b>7</b> | <b>CONCLUSÃO .....</b>                                     | <b>51</b> |
| <b>8</b> | <b>REFERÊNCIAS.....</b>                                    | <b>53</b> |
|          | <b>APÊNDICE A: CÓDIGO PARA CÁLCULO E SIMULAÇÃO.....</b>    | <b>54</b> |

## 1 INTRODUÇÃO

Pode-se definir um sistema de controle como uma interconexão de componentes, formando uma configuração de sistemas que produzirá uma resposta desejada. A teoria de sistemas lineares faz a análise destes sistemas, propondo uma relação de causalidade entre os componentes. Já o controle de um sistema consiste na leitura de uma variável e atuação sobre ela de forma a obter um valor desejado (ASSUNÇÃO; TEIXEIRA, 2013).

Um dos primeiros trabalhos realizados sobre controle foi o de James Watt, no século XVIII, seguido por Minorsky, Nyquist e Hazen com o início da criação das teorias de controle clássico, no início do próximo século. Somente na década de 1940 começaram a ser introduzidos os controladores PID, que foram sendo mais desenvolvidos ao longo dos anos. Na década de 1960 foi introduzida a teoria de controle moderno e a partir de 1980, de controle robusto (OGATA, 2014).

Segundo Knospe (2006) em seu artigo para a revista *IEEE Control Systems Magazine*, o controle PID é até então o tipo de controle mais utilizado, sendo estimado que aproximadamente 90% dos loops utilizam deste tipo de controle de alguma forma. Isso pois ele é flexível e permite a inclusão de recursos adicionais, além de não exigir desenvolvimentos matemáticos muito complexos, por isso sua ampla utilização em cursos de graduação.

Com o avanço da tecnologia e a modernização das indústrias, os sistemas de controle se tornaram indispensáveis, tanto para automatização de processos industriais, quanto para a vida diária da população, como o controle de temperatura de ambientes, o controle de velocidade de um veículo, entre outros.

O sistema *Ball Balancer* (também conhecido como *Ball and Plate System*) tem sido amplamente utilizado como ferramenta educacional e de pesquisa, inclusive pela Faculdade de Engenharia de Ilha Solteira. Ele se mostra uma ferramenta ideal para o ensino experimental de técnicas de controle, por ser um sistema instável e não linear.

O sistema *Ball and Plate* também pode ser visto como uma extensão do sistema conhecido como *Ball and Beam*, possuindo o mesmo conceito, mas com um eixo a mais de liberdade.

Um estudo, utilizando o sistema *Ball and Beam* com controladores robustos, foi feito por Din et. al (2017). Nele são comparadas diferentes formas de controle robusto do tipo *Sliding Mode Control*, mostrando que cada situação possui um controle mais adequado, dependendo dos resultados desejados.

Já para o caso *Ball and Plate*, Bang e Lee (2018) apresentaram um estudo utilizando este mesmo tipo de controlador (*Sliding Mode Control*), mostrando também como construir um equipamento de baixo custo para implementação.

Bang e Lee (2019) também conduziram um estudo mostrando a aplicação de um modelo de controle preditivo (*Model Predictive Control*) em um sistema *Ball and Plate*, utilizando microcontrolador de performance limitada.

O objetivo deste trabalho é a elaboração de um projeto de controle para um sistema *Ball Balancer 2 DOF* (Quanser), realizando simulações computacionais e ensaios de bancada das teorias apresentadas no curso de graduação de engenharia elétrica. Será realizado um estudo do equipamento e seus componentes, para desenvolvimento de controladores do tipo PID (Proporcional - Integrativo - Derivativo).

No Capítulo 2 será feita uma descrição do equipamento *Ball Balancer*, de seus componentes e funcionamento.

No Capítulo 3 serão feitos os cálculos para obter as plantas do sistema, além da modelagem matemática do controle a ser utilizado.

O Capítulo 4 apresentará os conceitos de controle em cascata, um pouco da teoria de controladores PID, além da obtenção do circuito e dos valores necessários no controlador, para obter a resposta dentro dos parâmetros estabelecidos.

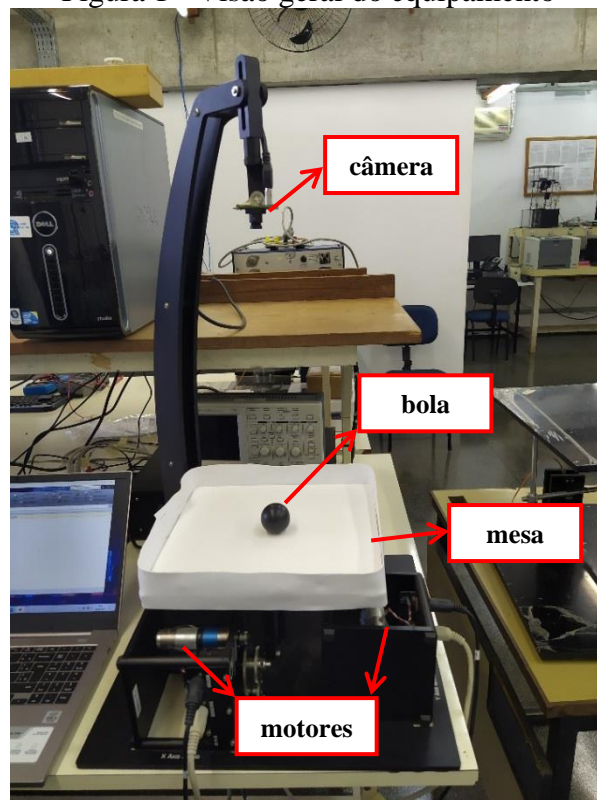
No Capítulo 5 é mostrado o programa do MATLAB, desenvolvido para automatizar os cálculos e realizar as simulações de forma interativa. Em seguida serão vistos os resultados das simulações do controlador no computador, no objetivo de prever problemas na aplicação física e também otimizar ao máximo a resposta.

Por fim, o Capítulo 6 mostra os resultados da aplicação dos controladores no próprio *Ball Balancer*, e o Capítulo 7 expressa as conclusões tiradas dos experimentos.

## 2 EQUIPAMENTO BALL BALANCER 2 DOF

O equipamento *Ball Balancer 2DOF* da Quanser, utilizado neste trabalho, consiste em uma mesa na qual pode ser posicionada uma bola, com dois servomotores que podem controlar o ângulo de inclinação da mesa em dois eixos, permitindo movimento da bola. Os motores são de fabricação também da Quanser, modelo SRV02. Na Figura 1 é apresentada uma foto do equipamento completo no laboratório e na Figura 2 é mostrado como é o posicionamento dos motores.

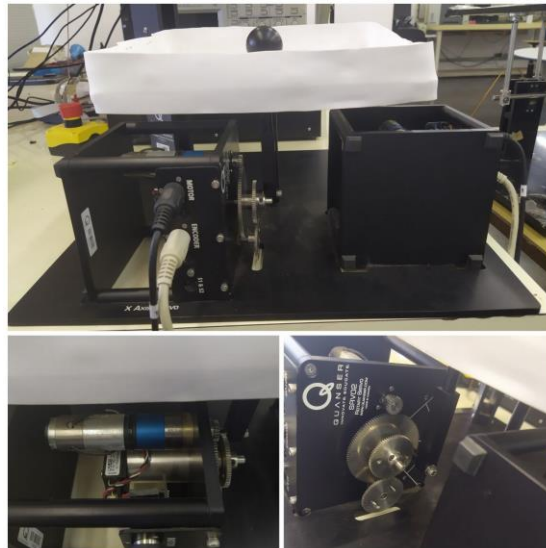
Figura 1 - Visão geral do equipamento



Fonte: Elaborado pelo autor

Esta Figura mostra a posição de cada componente do equipamento, com os motores abaixo da mesa, o objeto posicionado sobre ela e a câmera acima.

Figura 2 - Posicionamento dos motores



Fonte: Elaborado pelo autor

Acima da mesa há uma câmera de alta definição IEEE-1394, utilizada para monitorar a posição da bola. Na Figura 3 é mostrada a visão da mesa pelo ângulo da câmera.

Figura 3 - Visão da mesa pela posição da câmera



Fonte: Elaborado pelo autor

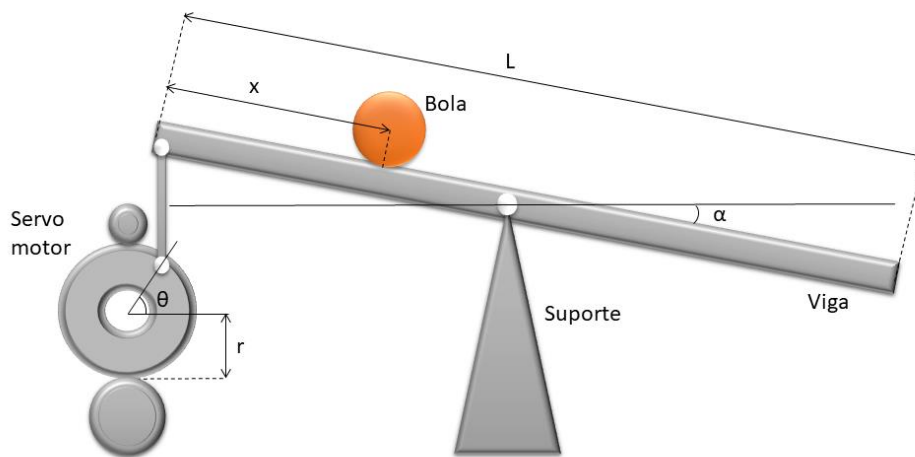
Para configuração e ativação do movimento, ele deve ficar conectado a um computador via cabos. Neste computador deve ser instalado um programa compatível que permita o controle do dispositivo, assim como os arquivos de ativação fornecidos.

### 3 EQUACIONAMENTO DA PLANTA DO SISTEMA

Neste capítulo é demonstrado o equacionamento da planta do sistema *Ball Balancer*, através das leis físicas que regem o movimento do sistema.

O equipamento 2DOF da QUANSER é constituído de dois servomotores, um para cada eixo, dando à base dois graus de liberdade. A análise dos eixos pode ser feita separadamente, já que não são interligados entre si. Na Figura 4 é representada uma ilustração da composição do sistema no eixo x.

Figura 4 - Esquemática do sistema no eixo X



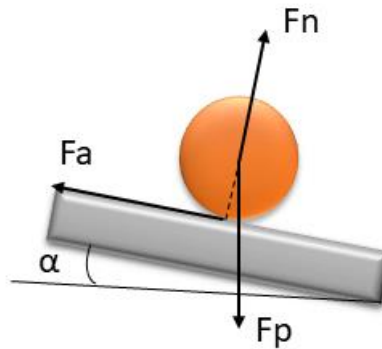
Fonte: Elaborado pelo autor

Na Figura 4,  $L$  representa o comprimento da viga,  $x$  é a distância da bola ao ponto zero, considerado como a borda superior,  $\alpha$  é o ângulo de inclinação da barra,  $\theta$  é o ângulo de rotação do motor e  $r$  é o raio do eixo do motor.

Pode-se considerar esta representação como um sistema viga-bola (*Ball and Beam*).

O modelo matemático da planta é calculado a partir das leis de Newton. Na Figura 5 é ilustrado o conjunto de forças que atuam sobre a bola em um plano inclinado.

Figura 5 - Forças que agem sobre a bola



Fonte: Elaborado pelo autor

Na Figura 5,  $F_p$  representa a força peso,  $F_a$  é a força de atrito sobre o ponto de contato e  $F_n$  é a força normal do plano sobre a bola. A força peso possui duas componentes, uma perpendicular ao plano ( $F_p \cdot \cos(\alpha)$ ), que é cancelada por  $F_n$  e outra paralela ao plano ( $F_p \cdot \sin(\alpha)$ ), que causa o deslocamento da bola para baixo.

Da segunda lei de Newton tem-se a somatória de forças, conforme (1), sendo  $m$  a massa da bola.

$$\sum F = F_p \cdot \sin(\alpha) - F_a = m \cdot \ddot{x}(t) \quad . \quad (1)$$

A força de atrito possui um braço de alavanca de comprimento  $r_b$ , produzindo um torque  $F_a \cdot r_b$ , utilizando a forma angular da segunda lei de Newton,  $F_a$  é descrita de acordo com (2).

$$F_a \cdot r_b = \frac{I_b}{r_b} \cdot \ddot{x}(t) \quad . \quad (2)$$

A força resultante do peso da bola é dada por (3).

$$F_{pr} = m \cdot g \cdot \sin \alpha \quad , \quad (3)$$

sendo  $g$  a aceleração da gravidade de  $9,8 \text{ m/s}^2$ ,  $m$  a massa da esfera,  $I_b$  o momento de inércia e  $r_b$  o raio da bola.

Substituindo (2) e (3) em (1), obtém-se (4).

$$m \cdot \ddot{x}(t) = m \cdot g \cdot \sin \alpha - \frac{I_b}{r_b^2} \cdot \ddot{x}(t) \quad . \quad (4)$$

Isolando a aceleração obtém-se (5).

$$\ddot{x}(t) = \frac{m \cdot g \cdot \text{sen}\alpha \cdot rb^2}{m \cdot rb^2 + Ib} \quad (5)$$

De Halliday e Resnick (2012) tem-se que o momento de inércia de uma casca esférica, como a usada no laboratório, é dado por  $Ib = \frac{2}{3}m \cdot rb^2$ . Substituindo este valor em (5) tem-se (6).

$$\ddot{x}(t) = \frac{3 \cdot g \cdot \text{sen}\alpha}{2} \quad (6)$$

Observa-se em (6), que a aceleração da esfera não depende de seus parâmetros e pode-se fazer a mesma análise para o eixo y.

A Tabela 1 contém os parâmetros do equipamento, fornecidos pelo fabricante.

Tabela 1 - Parâmetros do equipamento

| Parâmetro             | Símbolo | Valor    |
|-----------------------|---------|----------|
| Massa da bola         | m       | 0,003 kg |
| Raio da bola          | rb      | 1,96 cm  |
| Comprimento da mesa   | L       | 27,5 cm  |
| Raio do eixo do motor | r       | 2,54 cm  |

Fonte: Elaborado pelo autor com base em QUANSER (2008)

Considerando que a base seja inclinada de forma que a extremidade seja elevada em uma altura h, os valores do seno dos ângulos são dados por (7) e (8).

$$\text{sen}\alpha = \frac{2 \cdot h}{L} \quad (7)$$

$$\text{sen}\theta = \frac{h}{r} \quad (8)$$

Substituindo (7) em (8) obtém-se (9).

$$\text{sen}\alpha = \frac{2 \cdot \text{sen}\theta \cdot r}{L} \quad (9)$$

Substituindo (9) em (6):

$$\ddot{x}(t) = \frac{3 \cdot g \cdot \text{sen}\theta \cdot r}{2 \cdot L} \quad (10)$$

Utilizando os valores da Tabela 1 em (10) e considerando que para ângulos pequenos  $\text{sen}\theta \approx \theta$ :

$$\ddot{x}(t) = 1,3577 \cdot \theta \quad . \quad (11)$$

Na equação (11) o número 1,3577 representa o ganho  $K_{bb}$  do sistema.

### 3.1 FUNÇÃO DE TRANSFERÊNCIA

A partir das equações de movimento encontradas anteriormente, será calculada a função de transferência,  $P_{bb}(s)$ , do movimento da bola, em relação ao ângulo do servomotor.

Calculando a transformada de Laplace de (11), obtém-se (12).

$$s^2 \cdot X(s) - s \cdot x(0) - \dot{x}(0) = K_{bb} \cdot \theta(s) \quad . \quad (12)$$

Isolando  $X(s)$  e considerando as condições iniciais zero, chega-se em (13).

$$X(s) = \frac{K_{bb} \cdot \theta(s)}{s^2} \quad . \quad (13)$$

Desta forma, a função de transferência da posição da bola,  $x(t)$ , em relação ao ângulo do motor,  $\theta$ , é representada por (14).

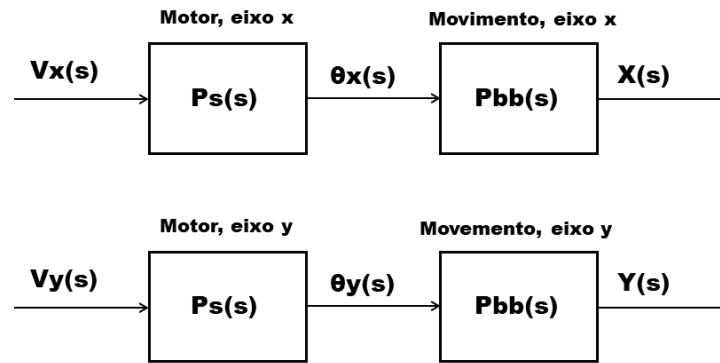
$$P_{bb}(s) = \frac{X(s)}{\theta(s)} = \frac{K_{bb}}{s^2} \quad . \quad (14)$$

### 3.2 MODELAGEM MATEMÁTICA COM O MOTOR

Neste item será demonstrada a modelagem matemática utilizada para o cálculo do controlador do *Ball Balancer*.

Na Figura 6 é ilustrado o diagrama de blocos de malha aberta do sistema.

Figura 6 – Diagrama de blocos de malha aberta



Fonte: Elaborado pelo autor

Para cada eixo há uma tensão de entrada que atua no motor, produzindo uma rotação de ângulo  $\theta$ , que provoca o movimento da bola para a posição (X, Y).

Pode-se assumir que a dinâmica dos dois eixos é igual, já que a mesa é simétrica e possui o mesmo hardware.

Desta forma, as funções de transferência  $P_s(s)$  e  $P_{bb}(s)$  são dadas por (15) e (16), respectivamente.

$$P_s(s) = \frac{\theta}{V} \quad (15)$$

$$P_{bb}(s) = \frac{X}{\theta} \quad (16)$$

De QUANSER (2008), tem-se que a função de transferência do motor utilizado no equipamento é descrita por (17), sendo  $K$  o ganho e  $\tau$  a constante de tempo do motor.

$$P_s(s) = \frac{K}{(\tau \cdot s + 1) \cdot s} \quad (17)$$

No item anterior foi encontrada a função de transferência do movimento da bola,  $P_{bb}(s)$ , dada por (14).

As funções de movimento e do motor estão em série, portanto a função de transferência resultante é dada por (18).

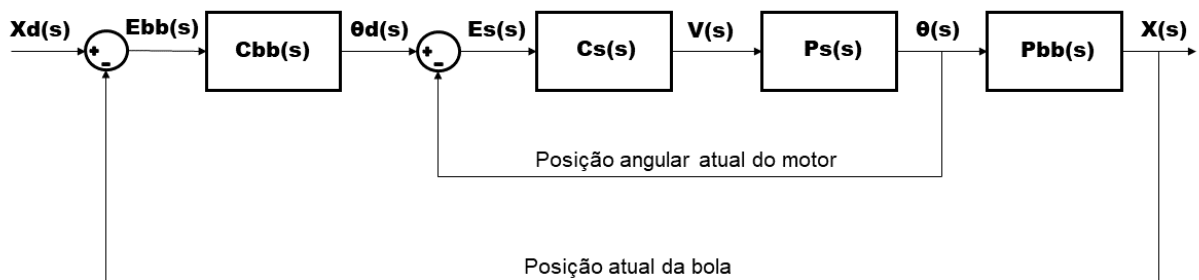
$$P(s) = P_s(s) \cdot P_{bb}(s) = \frac{K \cdot K_{bb}}{(\tau \cdot s + 1) \cdot s^3} \quad (18)$$

#### 4 CONTROLE EM CASCATA DO SISTEMA BALL BALANCER

Neste capítulo é apresentado um sistema de controle em cascata, que consiste em um controlador primário que regula um controlador secundário, ligados em série. No caso estudado, um controlador primário para calcular o ângulo necessário para estabilização da mesa e um controlador secundário para encontrar a tensão de entrada de cada motor (considerando os dois eixos).

Na Figura 7 é ilustrado o diagrama do controle em cascata.

Figura 7 - Diagrama de blocos do controle em cascata



Fonte: Elaborado pelo autor

Na Figura, tem-se  $X_d(s)$  como a posição desejada da bola, esta é subtraída de  $X(s)$ , posição atual da bola, que passando pelo controlador  $C_{bb}(s)$  obtém-se o ângulo de rotação do motor  $\theta_d(s)$  necessário para fazer esta trajetória, este é subtraído do ângulo atual do motor,  $\theta(s)$ , e passa pelo controlador  $C_s(s)$ , que encontra a tensão  $V(s)$  necessária para causar a rotação desejada no motor. Esta tensão é aplicada na planta do motor, encontrada anteriormente, produzindo uma rotação resultante de ângulo  $\theta(s)$ , o qual ao ser aplicado na planta de movimento, produz uma posição final da bola,  $X(s)$ . Este ciclo se repete enquanto a posição da bola não estiver dentro do esperado.

##### 4.1 RESPOSTA ESPERADA

De acordo com o manual do fabricante (QUANSER, 2008) deve-se obedecer aos critérios das equações (19) a (21) para o controlador do motor, sendo  $ess$  o erro em regime estacionário,  $tp$  o tempo de pico e  $PO$  a porcentagem de *overshoot*.

$$ess = 0 \quad , \quad (19)$$

$$tp \leq 0,15s \quad , \quad (20)$$

$$PO \leq 5\% \quad . \quad (21)$$

Para o controlador da mesa, são fornecidos os requisitos das equações de (22) a (25), sendo  $t_s$  o tempo de assentamento e  $cts$  a porcentagem de tempo de assentamento.

$$|ess| \leq 0,001m \quad , \quad (22)$$

$$t_s \leq 2,5s \quad , \quad (23)$$

$$cts \leq 0,04 \quad , \quad (24)$$

$$PO \leq 7,5\% \quad . \quad (25)$$

Ou seja, em um intervalo de até 2,5s, o sinal deve ter uma variação máxima de 4% do valor estacionário, sendo o valor estacionário considerado como até 1mm de diferença da posição final esperada.

## 4.2 CONTROLADORES PID

Os controladores do tipo PID possuem três componentes, proporcional, integrativo e derivativo. Na Tabela 2 são mostrados os efeitos de cada um destes componentes na resposta do sistema.

Tabela 2 - Efeito dos controladores PID

| <b>Componente</b> | <b>Tempo de subida</b> | <b>Overshoot</b> | <b>Tempo de estabelecimento</b> | <b>Erro de estado estacionário</b> |
|-------------------|------------------------|------------------|---------------------------------|------------------------------------|
| Proporcional      | Reduz                  | Aumenta          | -                               | Reduz                              |
| Integral          | Reduz                  | Aumenta          | Aumenta                         | Elimina                            |
| Derivativo        | -                      | Reduz            | Reduz                           | -                                  |

Fonte: Elaborado pelo autor com base em (MUNIZ, 2017)

A equação (26) representa a função de transferência de um controlador PID com suas três componentes.

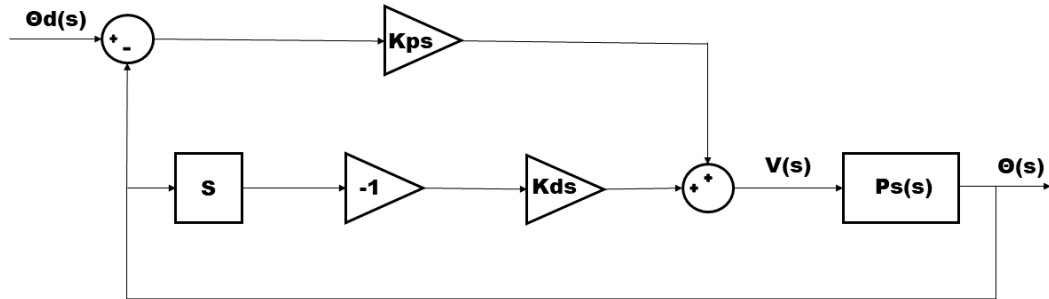
$$G_c(s) = K_p + \frac{K_i}{s} + K_d \cdot s \quad . \quad (26)$$

Sendo  $K_p$  o ganho proporcional,  $K_i$  o ganho da integral e  $K_d$  o ganho da derivada, ao igualar somente  $K_i$  a zero tem-se o controlador PD e ao igualar somente  $K_d$  a zero tem-se o controlador PI.

### 4.3 CONTROLE DO MOTOR

Para o controle da tensão aplicada no motor será utilizado um controlador de estilo PD, adaptado pelo fabricante. Na Figura 8 é representado o diagrama de blocos do controlador PD utilizando amplificadores operacionais.

Figura 8 - Diagrama do controle do motor



Fonte: Elaborado pelo autor com base em QUANSER (2008c)

Da Figura 8, pode ser montada a equação (27), utilizando (17) para  $P_s(s)$ .

$$((\theta_d - \theta) \cdot K_{ps} - \theta \cdot s \cdot K_{ds}) \cdot \frac{K}{\tau \cdot s^2 + 1 \cdot s} = \theta \quad (27)$$

Isolando os ângulos em (27), pode-se obter a função de transferência (28).

$$\frac{\theta(s)}{\theta_d(s)} = \frac{K_{ps} \cdot K}{K_{ps} \cdot K + s \cdot (1 + K_{ds} \cdot K) + \tau \cdot s^2} \quad (28)$$

Segundo os dados do fabricante, as constantes do motor em regime permanente são  $K = 1,76 \frac{rad}{sV}$  e  $\tau = 0,0285s$ .

Em OGATA (2011) encontra-se as equações para a porcentagem de *overshoot* e para o tempo de pico, em função do coeficiente de amortecimento ( $\xi$ ) e da frequência natural do controlador ( $\omega_n$ ). Como são conhecidas a porcentagem de *overshoot* e o tempo de pico, pode-se calcular  $\xi$  e  $\omega_n$ , de acordo com (29) e (30).

$$\xi = \sqrt{\frac{\ln^2(PO/100)}{\pi^2 + \ln^2(PO/100)}} = 0,6901 \quad (29)$$

$$\omega_n = \frac{\pi}{tp\sqrt{1-\xi^2}} = 28,9398 \text{ .} \quad (30)$$

A partir destes valores é possível encontrar os ganhos  $Kps$  e  $Kds$ , através de (31) e (32).

$$Kps = \frac{\omega_n^2 \cdot \tau}{K} = 13,5620 \text{ ,} \quad (31)$$

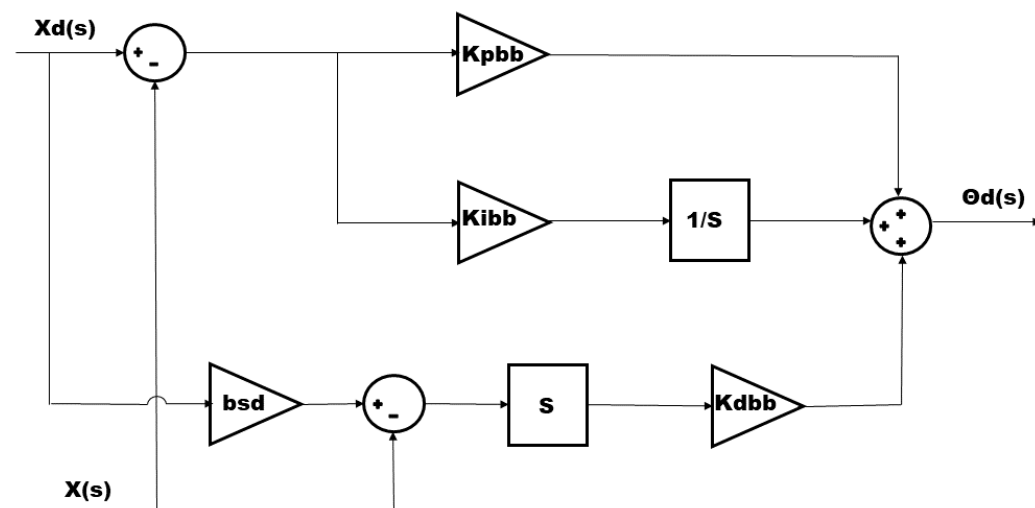
$$Kds = \frac{\tau}{K} \cdot \left(2 \cdot \xi \cdot \omega_n - \frac{1}{\tau}\right) = 0,0786 \text{ .} \quad (32)$$

#### 4.4 CONTROLADOR DA MESA

Analisando a Tabela 2, conclui-se que a melhor opção neste caso é um controlador PID, cujo diagrama com amplificadores operacionais é mostrado na Figura 9.

Foi incluído, como recomendado pelo fabricante, um parâmetro de ajuste na parte derivativa do controlador ( $bsd$ ). Em um controlador PID comum, esta constante seria igual a 1, mas neste trabalho será estudado o efeito causado ao variar este valor, mais adiante nas simulações.

Figura 9 - Diagrama do controle da mesa



Fonte: Elaborado pelo autor com base em QUANSER (2008a)

Vale notar que ao considerar  $bsd=0$ , o diagrama da parte derivativa do controlador se torna semelhante ao controlador utilizado para o motor, que representa uma adaptação do controlador PD.

Para obter o diagrama do controlador PD, basta igualar a constante de integração,  $Kibb$ , a zero.

Do diagrama da Figura 9, obtém-se (33).

$$(Xd - X) \cdot \left[ Kpbb + Kibb \cdot \frac{1}{s} \right] + (Xd \cdot bsd - X) \cdot s \cdot Kdbb = \theta d . \quad (33)$$

Para encontrar a função de transferência de  $X(s)$  em relação a  $Xd(s)$  pode-se considerar que  $\theta d(s) = \theta(s)$ . Considerando (11) e isolando os termos desejados, chega-se à equação (34), somente em função das constantes.

$$\frac{X(s)}{Xd(s)} = \frac{(Kpbb \cdot s + Kibb + Kdbb \cdot bsd \cdot s^2) \cdot Kbb}{Kpbb \cdot s + Kibb + Kdbb \cdot s^2 + s^3} . \quad (34)$$

Para o sistema da terceira ordem, a equação (35) é utilizada para encontrar os parâmetros.

$$(s^2 + 2 \cdot \xi \cdot \omega n \cdot s + \omega n^2) \cdot (s + po) . \quad (35)$$

Com os requisitos de projeto dados anteriormente, encontram-se  $\xi$  e  $\omega n$  e, com estes, as constantes do controlador, sendo  $po$  a localização do polo.

$$\xi = \frac{\sqrt{\ln^2(PO/100)}}{\sqrt{\pi^2 + \ln^2(PO/100)}} = 0,6362 , \quad (36)$$

$$\omega n = -\frac{\ln(cts\sqrt{1-\xi^2})}{ts \cdot \xi} = 2,1870 , \quad (37)$$

$$Kibb = \frac{\omega n^2 \cdot po}{Kbb} , \quad (38)$$

$$Kpbb = \frac{\omega n^2 + 2 \cdot \xi \cdot \omega n \cdot po}{Kbb} , \quad (39)$$

$$Kdbb = \frac{2 \cdot \xi \cdot \omega n + po}{Kbb} . \quad (40)$$

Para implementar um controle PD na mesa, deve ser considerado  $po=0$ , o que torna a constante do integrador ( $Kibb$ ) nula, portanto, só estariam em funcionamento as partes

proporcional e derivativa. Os valores das constantes para controlador PD são dadas pelas equações de (41) a (43).

$$K_{ibb} = 0 \quad , \quad (41)$$

$$K_{pbb} = 3,5228 \quad , \quad (42)$$

$$K_{dbb} = 2,0496 \quad . \quad (43)$$

No caso do controlador PID,  $p_o$  deve ser considerado de acordo com (44), sendo  $T_p$  a constante de decaimento do polo, que será adotada como 1.

$$p_o = \frac{1}{T_p} = 1 \quad . \quad (44)$$

Com o valor de  $p_o$  calculado encontra-se o valor das constantes para o controle PID, como nas equações de (45) a (47).

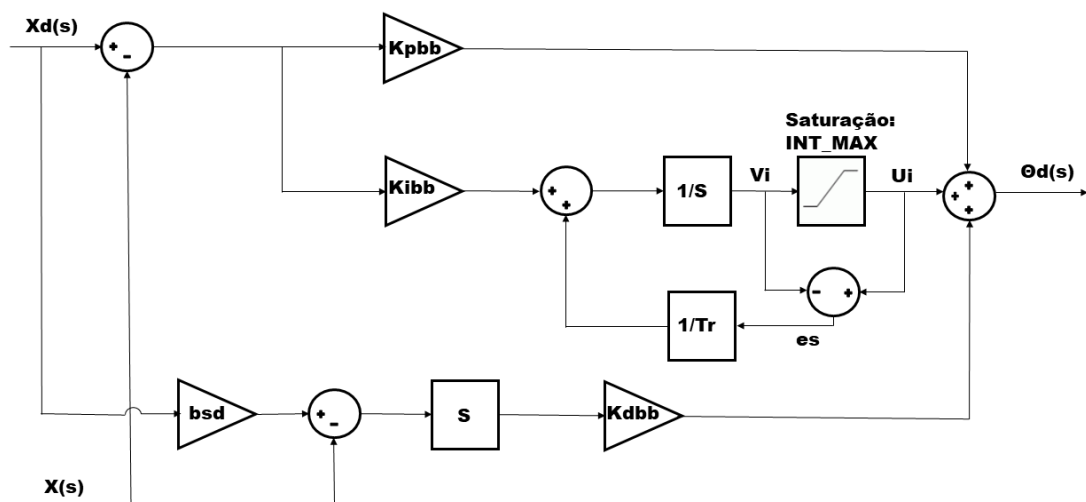
$$K_{ibb} = 3,5228 \quad , \quad (45)$$

$$K_{pbb} = 5,5724 \quad , \quad (46)$$

$$K_{dbb} = 2,7861 \quad . \quad (47)$$

Deve ser acrescentado ao controlador PID, um loop de realimentação *anti-windup*, desta forma, quando houver saturação do integrador o *anti-windup* diminui os valores de entrada do integrador. O novo esquema é apresentado na Figura 10, sendo INT\_MAX o limite do integrador para saturação e  $T_r$  a constante de tempo do *anti-windup*, da qual depende o tempo de reinicialização do integrador.

Figura 10 - Diagrama do controle da mesa com *anti-windup*



## 5 PROGRAMA DE CÁLCULO E SIMULAÇÕES

A fim de facilitar as simulações do controlador, foi desenvolvido um programa em linguagem MATLAB, que execute os cálculos das constantes de controle, de acordo com os parâmetros desejados pelo usuário, e inicie a simulação, plotando os gráficos finais. A seguir é explicado o passo a passo da interação com o programa e na sequência, os resultados das simulações com os controladores calculados.

### 5.1 PROGRAMA DE CÁLCULO

Ao rodar o programa, o usuário deverá fornecer os valores de tempo de pico do motor e máximo *overshoot* do motor, então será feito o cálculo dos ganhos do controlador usado para o motor.

Em seguida devem ser fornecidos, o tempo de estabelecimento da mesa, o máximo *overshoot* da mesa e o tempo de assentamento da mesa. O usuário deve escolher entre o controlador PD, ou PID. Com isso são calculados os ganhos do controlador da mesa e mostrados os resultados.

O usuário então tem a opção de executar ou não a simulação. Caso não queira, o programa é finalizado, mas caso queira simular, será lhe dada a opção de continuar com os valores calculados, ou inserir manualmente novos ganhos. Será perguntado também qual a constante *bsd*.

Antes de simular, o usuário terá a opção de considerar perdas de potência do motor, que deverá ser informada em porcentagem, em caso afirmativo.

Caso tenha sido escolhido o controlador PID, neste nível será perguntado o limite de saturação do integrador e a constante de tempo do *anti-windup*.

Por fim, será aberto o arquivo do SIMULINK, executada a simulação e plotado o gráfico final, com posição do objeto, ângulo do motor e tensão aplicada. Esta parte pode demorar alguns segundos a mais, pelo esforço do programa em iniciar o SIMULINK.

O código completo foi incluído como apêndice deste trabalho. Para utilizá-lo basta salvá-lo como um arquivo script do MATLAB (versão de 2019).

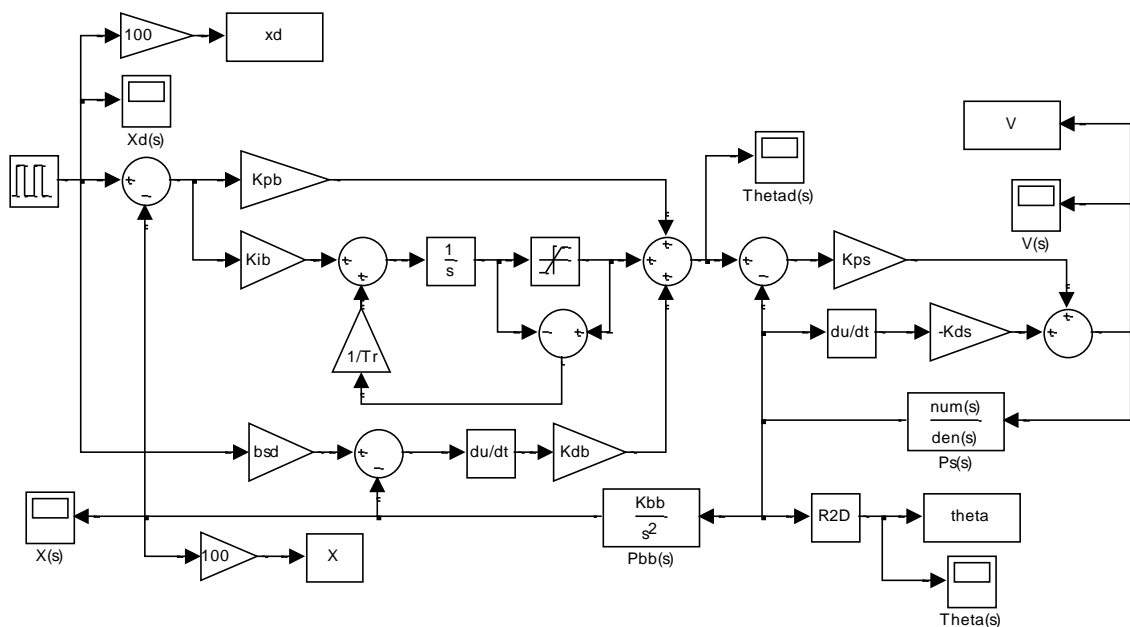
### 5.2 SIMULAÇÕES

A seguir foi iniciada a etapa de simulações dos controladores projetados, para isso foi utilizado o software MATLAB e a ferramenta SIMULINK.

As simulações foram feitas usando como entrada uma onda quadrada de amplitude de 5cm, que é próximo da distância usada no sistema físico. Foi simulado apenas o eixo X, considerando que o eixo Y possui uma resposta semelhante.

O esquemático utilizado no Simulink é representado pela Figura 11. Foi montado o esquema completo com o controlador PD para o motor e controlador PID para a mesa, ambos com as adaptações adicionadas.

Figura 11 - Esquemático do Simulink



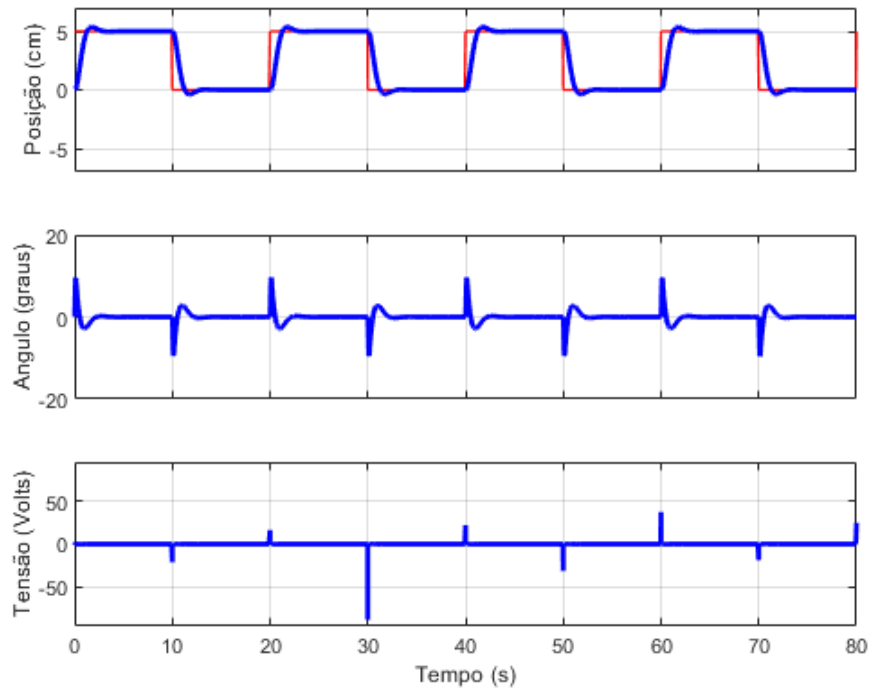
Fonte: Elaborado pelo autor

### 5.2.1 Controlador PD para motor e mesa

As simulações foram iniciadas com os valores calculados nos capítulos anteriores, utilizando um controlador PD para o motor e para a mesa, sendo a simulação inicial representada pela Figura 12, com o ângulo dado em graus, a tensão em volts, a posição em metros e o tempo em segundos. Inicialmente é considerado o controlador PD tradicional.

Foi usado o mesmo esquema da Figura 11, apenas levando a constante de integração  $K_{ib}$  a zero, o que remove a parte da integral, deixando somente um controlador PD.

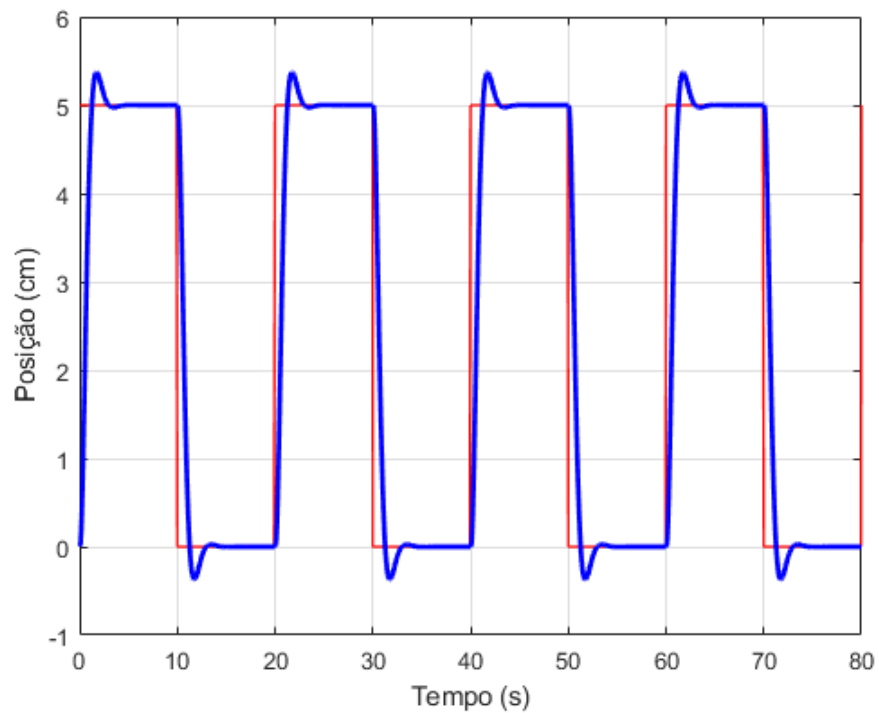
Figura 12 - Posição  $X(s)$ , Tensão  $V(s)$  e Ângulo  $\theta(s)$  para simulação PD inicial com  $bsd=1$



Fonte: Elaborado pelo autor

Foi também plotado na Figura 13, o gráfico da posição pelo tempo, separadamente, para melhor visualização dos parâmetros.

Figura 13 - Posição  $X(s)$  para simulação PD inicial com  $bsd=1$

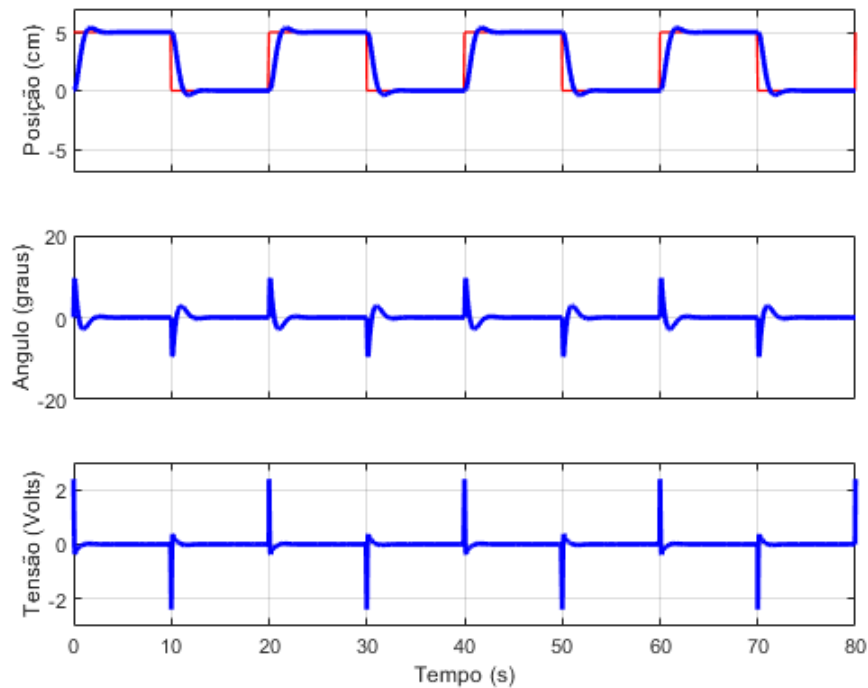


Fonte: Elaborado pelo autor

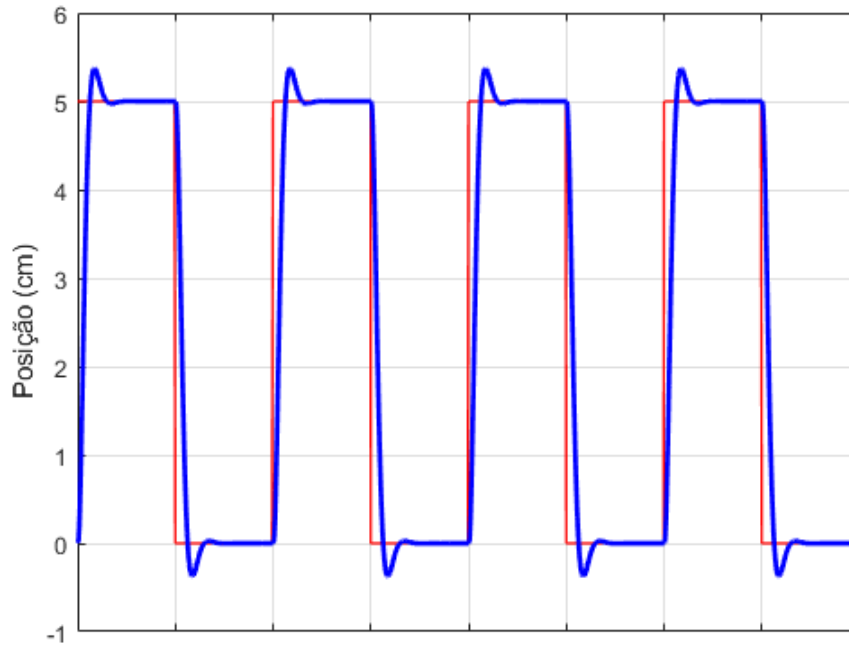
Foi notado durante as simulações, que ao utilizar a constante  $bsd$  como 1, os valores de tensão necessários para o movimento são muito altos (variando de 20V até mais de 80V), mas ao abaixar esta constante, a tensão diminui com ela. Por isso, no intuito de exigir o mínimo esforço do equipamento, será utilizado em diante  $bsd=0$  (redução da tensão em 10 vezes), assim como foi feito para o controlador do motor.

Nas Figura 14 e Figura 15, são mostrados os resultados da simulação com  $bsd$  nulo, na qual pode-se perceber a drástica redução da tensão, sem diferenças perceptíveis na posição, se comparado com a Figura 12.

Figura 14 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PD inicial com  $bsd=0$



Fonte: Elaborado pelo autor

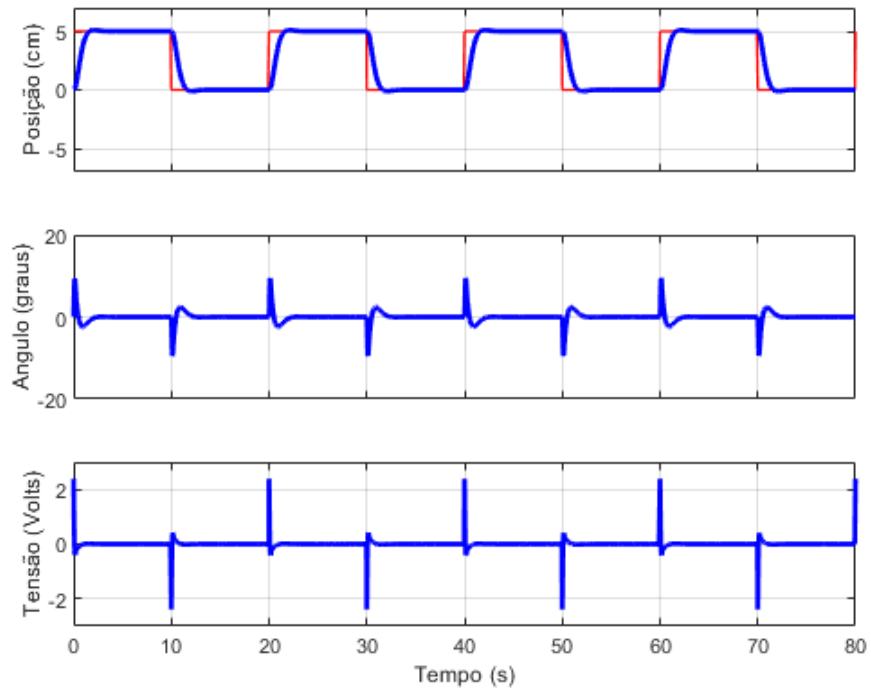
Figura 15 - Posição  $X(s)$  para simulação PD inicial com  $bsd=0$ 

Fonte: Elaborado pelo autor

Neste caso tem-se um tempo de pico de 1,9s, tempo de estabelecimento de 2,5s e uma porcentagem de *overshoot* de 7,4%, que estão dentro do valor esperado, mas muito próximos do limite.

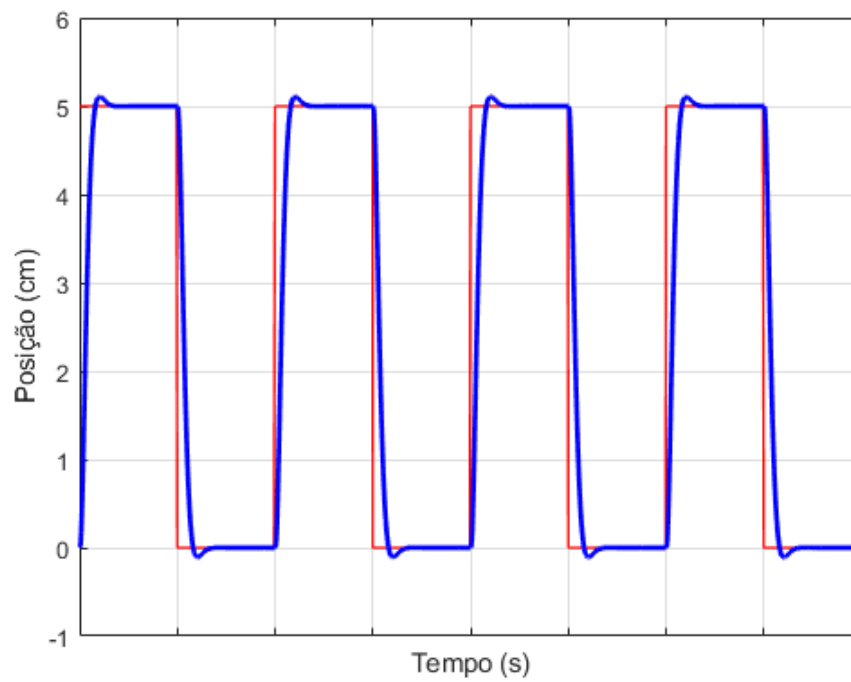
Como este projeto visa uma melhora de controle, para diminuir este efeito, o valor do ganho  $K_{dbb}$  foi aumentado de 2,0496 para 2,4 e nas Figura 16 e Figura 17 é mostrado o resultado desta mudança. Este valor foi escolhido pois, conforme se aumenta essa constante, o *overshoot* e o tempo de estabelecimento diminuem, mas após 2,4 o tempo de pico começa a aumentar.

Figura 16 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PD melhorada



Fonte: Elaborado pelo autor

Figura 17 - Posição  $X(s)$  para simulação PD melhorada



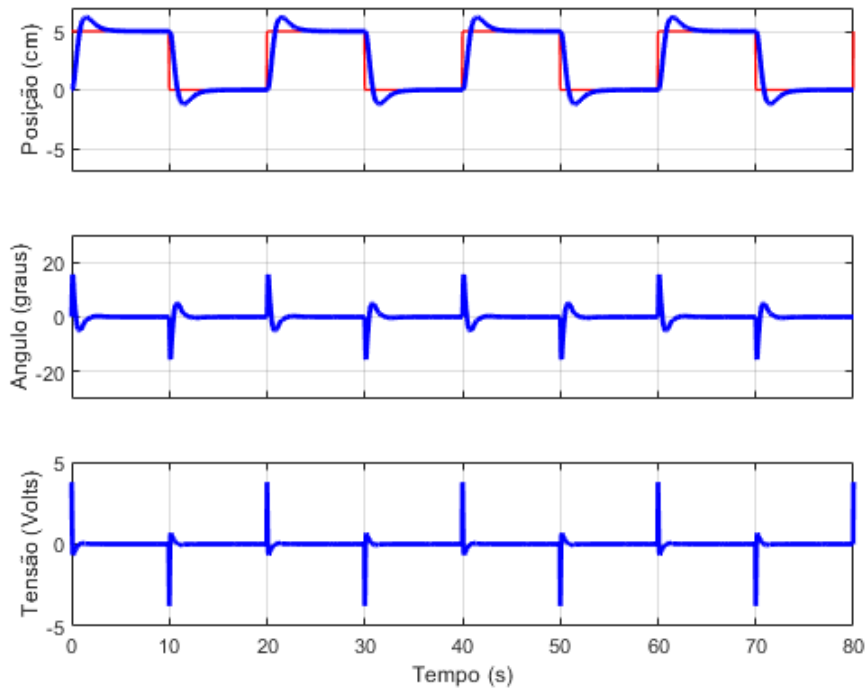
Fonte: Elaborado pelo autor

A porcentagem de *overshoot* agora foi diminuída para 2,2%, o tempo de pico se mantém em 1,9s e o tempo de estabilização passa a 2,1s. Esta simulação apresenta um resultado satisfatório e com certa melhoria em relação aos dados do fabricante, principalmente se considerado que ainda não foi introduzida a constante de integração.

### 5.2.2 Controlador PD para o motor e PID para a mesa

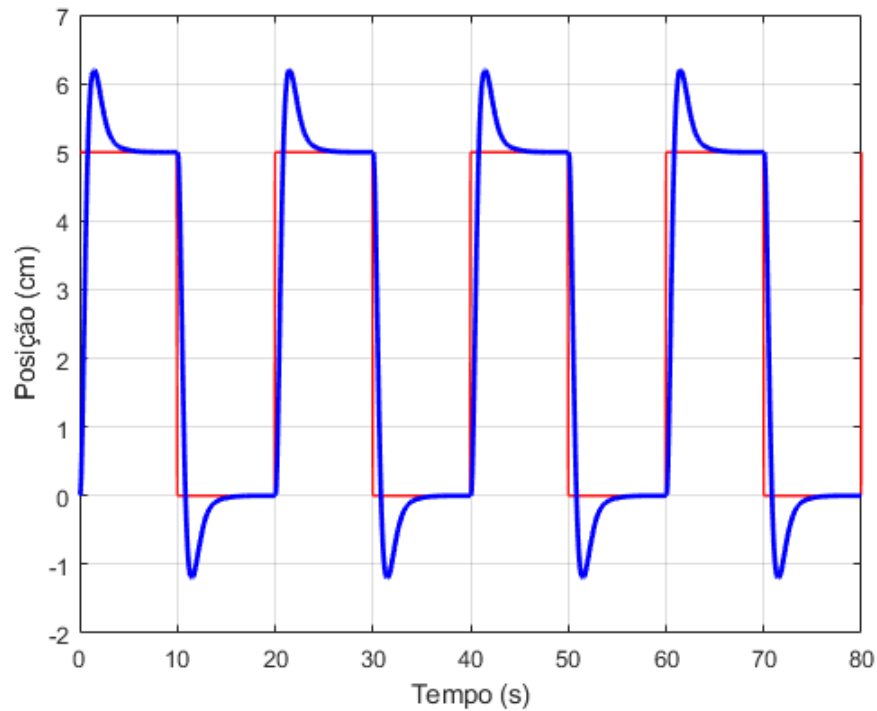
A seguir foram feitas as simulações para um controlador PD para o motor e controlador PID para a mesa, utilizando o esquema da Figura 11, montado no Simulink. Nas Figura 18 e Figura 19 são ilustrados os resultados obtidos com os valores calculados anteriormente.

Figura 18 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PID inicial



Fonte: Elaborado pelo autor

Figura 19 - Posição X(s) para simulação PID inicial

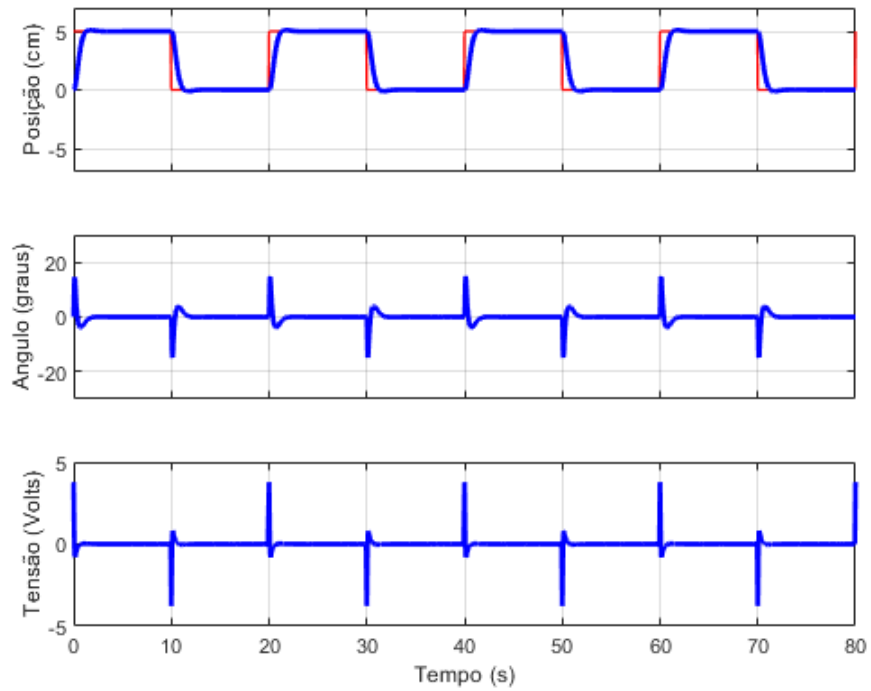


Fonte: Elaborado pelo autor

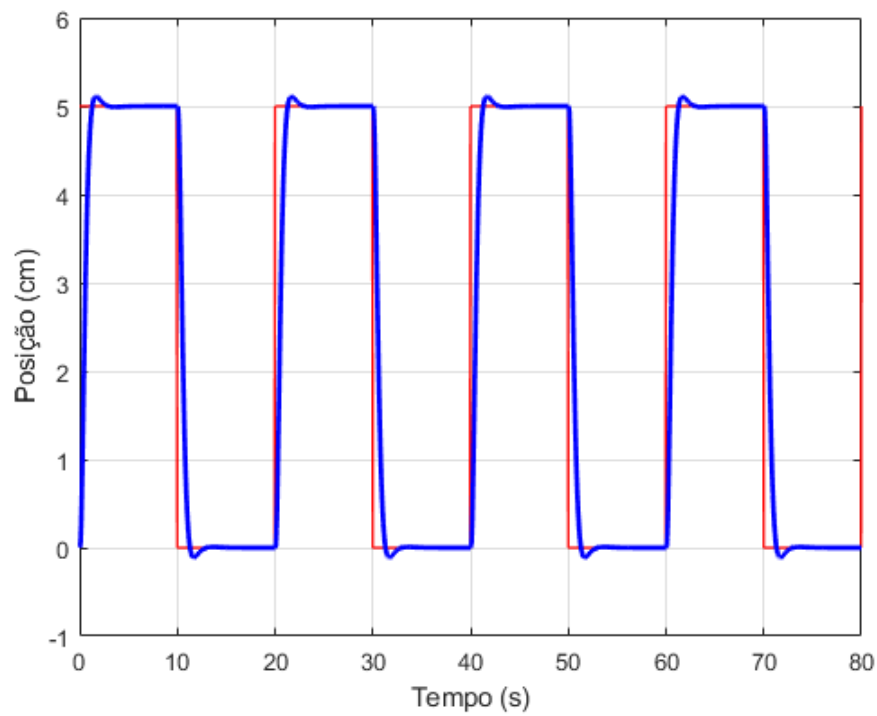
Comparando as Figura 14 e Figura 18, observa-se que o controlador PID exige uma tensão ligeiramente maior do que o controlador PD.

A simulação da Figura 18 possui tempo de pico de 1,6s, tempo de estabilização de 3,8s e porcentagem de *overshoot* de 23,4%. Estes valores estão acima dos limites estabelecidos portanto, para cumprir os objetivos deste trabalho, o ângulo de saturação INT\_MAX foi alterado de 3° para 0,15°, o que contribui para diminuição do *overshoot*. Ao diminuir Tr de 1 para 0,1, é diminuído o tempo de estabelecimento. A constante de derivação foi aumentada para 3,1, o que também causa redução de *overshoot*. Assim como no caso anterior, estes valores foram escolhidos de forma a otimizar os parâmetros, sem ocasionar piora em outros.

Nas Figura 20 e Figura 21 são mostrados o resultado da simulação melhorada.

Figura 20 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PID melhorada

Fonte: Elaborado pelo autor

Figura 21 - Posição  $X(s)$  para simulação PID melhorada

Fonte: Elaborado pelo autor

O tempo de pico desta simulação é de 1,7s, o tempo de estabilização de aproximadamente 1,8s e a porcentagem de *overshoot* de 2,2%, uma melhora considerável em relação aos valores iniciais. Por isso o segundo resultado deste controlador será considerado como o melhor.

Comparando os resultados dos controladores PD e PID, vemos que a porcentagem de *overshoot* obtida chegou ao mesmo valor, no entanto, o controlador PID possui um tempo de estabelecimento mais baixo.

Conforme foi visto na Tabela 2, o controlador PID tende a aumentar o *overshoot*, como aconteceu na primeira simulação, mas isso foi compensado pelo loop de *anti-windup*.

Durante as simulações não foi analisado o erro em regime estacionário (que representa a maior vantagem do uso do controlador PID), já que os valores foram baixos o suficiente para serem desconsiderados. No entanto, isto será analisado mais a frente durante os resultados com o próprio *Ball Balancer*.

### 5.2.3 Simulação de falhas

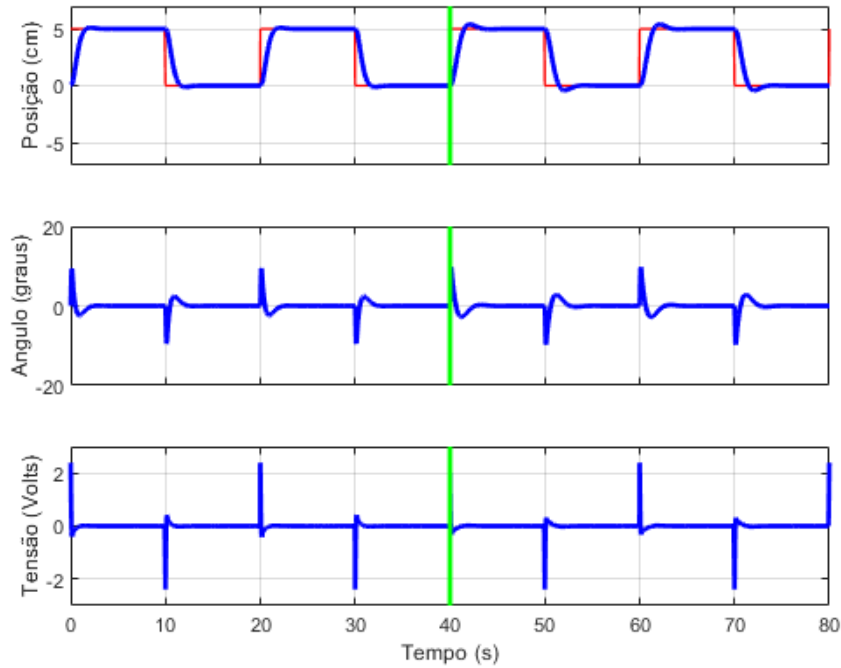
Considerando que o dispositivo em algum momento apresente falhas, foram feitas as simulações a seguir, para prever o comportamento dos controladores para falhas de 30%, 50% e 70% de perda de potência do motor.

Para isso, foi aplicado na planta do sistema um sinal com 70% do sinal de entrada para falha de 30%, 50% da entrada para falha de 50% e 30% do sinal de entrada para simular a falha de 70% de potência.

Na Figura 22 é ilustrado o esquema utilizado para o caso de 30% de perda no controlador. Foram considerados os controladores com melhorias.



Figura 23 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PD com 30% de falha

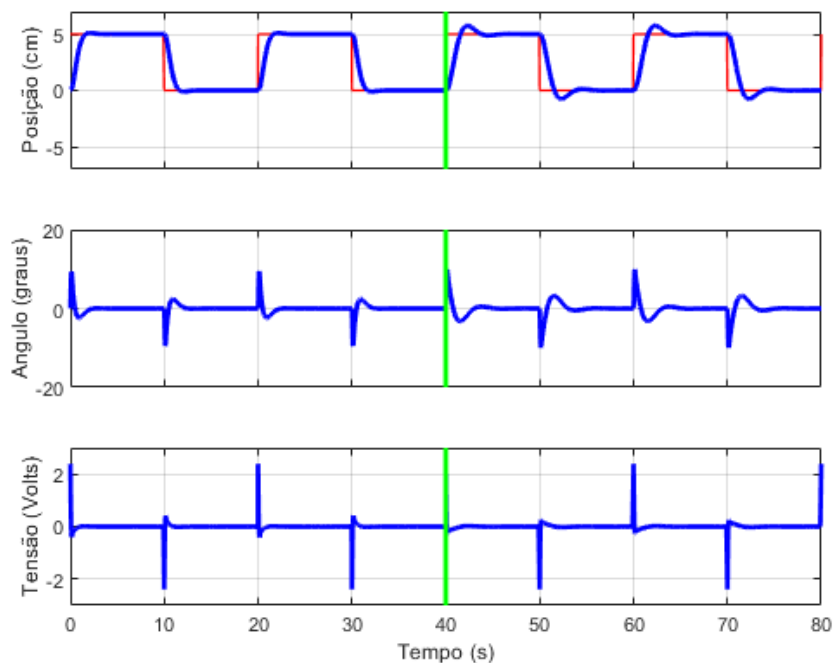


Fonte: Elaborado pelo autor

Após os 40s já é observada uma pequena falha de aumento de *overshoot*, mas no geral o sistema ainda funciona bem.

A seguir foram feitas as simulações para 50% de perda de potência para este mesmo controlador, os resultados estão representados na Figura 24.

Figura 24 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PD com 50% de falha

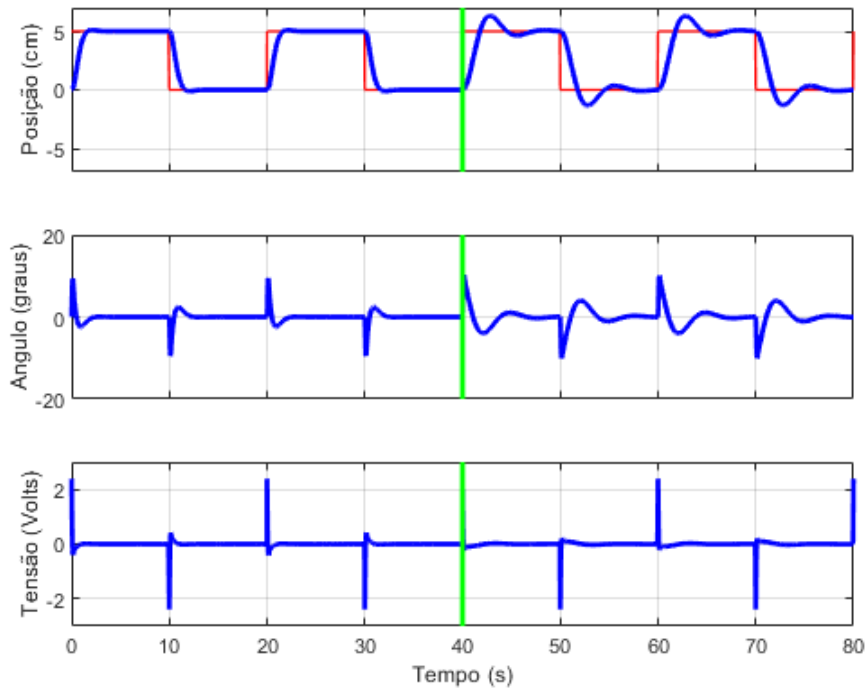


Fonte: Elaborado pelo autor

Nesta fase as falhas já começam a aumentar e ficam perceptíveis também no ângulo e na tensão. Mesmo assim ainda consegue realizar o trajeto.

Ainda com o controlador PD, são apresentados os resultados da simulação de 70% de perda, na Figura 25.

Figura 25 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PD com 70% de falha

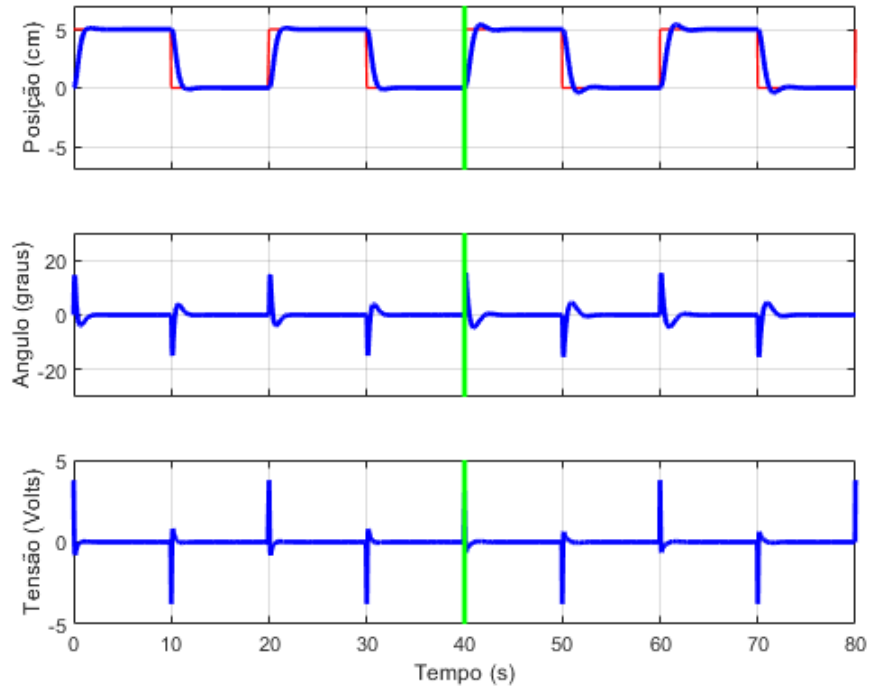


Fonte: Elaborado pelo autor

Com 70% de perda, o motor já não consegue se manter bem dentro da trajetória, isso fica ainda mais visível pelas curvas grotescas no gráfico do ângulo de rotação.

Seguindo o mesmo procedimento para o controlador PID, é utilizado o esquema da Figura 22 para simular uma falha de 30% no dispositivo. Os resultados da simulação com este sistema são apresentados na Figura 26.

Figura 26 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PID com 30% de falha

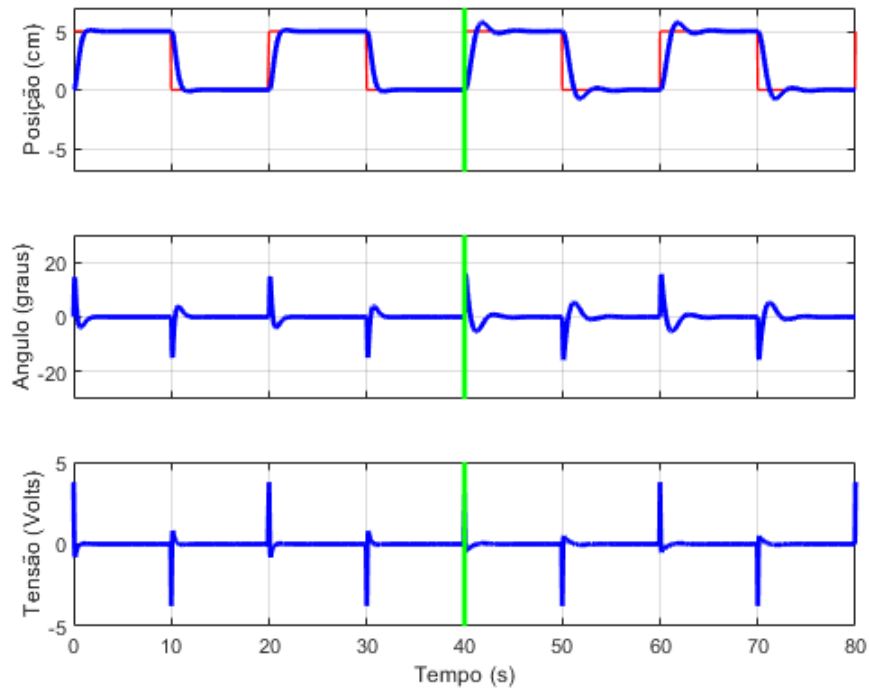


Fonte: Elaborado pelo autor

Assim como nas simulações sem falha, o PID tem um overshoot pouco maior que o PD, mas um tempo de estabelecimento menor e ainda realiza sua trajetória de forma satisfatória com esta porcentagem.

A seguir foi feita a mesma simulação para 50% de falha do motor. O resultado foi anexado na Figura 27.

Figura 27 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PID com 50% de falha

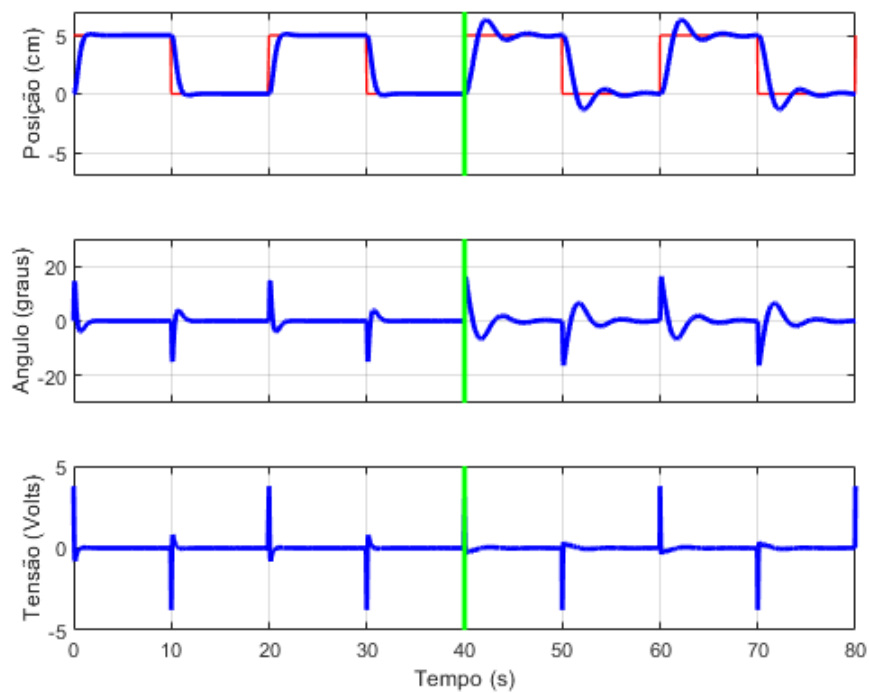


Fonte: Elaborado pelo autor

Neste caso tem-se uma diferença menor entre os 30% e os 50% do que em relação ao controlador anterior.

Por fim, a simulação do controlador PID com 70% de falha é apresentada na Figura 28.

Figura 28 - Posição  $X(s)$ , tensão  $V(s)$  e ângulo  $\theta(s)$  para simulação PID com 70% de falha



Fonte: Elaborado pelo autor

Com este experimento já é possível observar a diferença proporcionada pela parte integrativa no erro em regime permanente, já que o controlador PID mostra, apesar de pequenas variações, que consegue se manter melhor dentro da trajetória após o primeiro impulso, sendo que no controlador PD ele demora muito para retornar ao ponto desejado após o pico da subida.

## 6 RESULTADOS EXPERIMENTAIS COM O BALL BALANCER

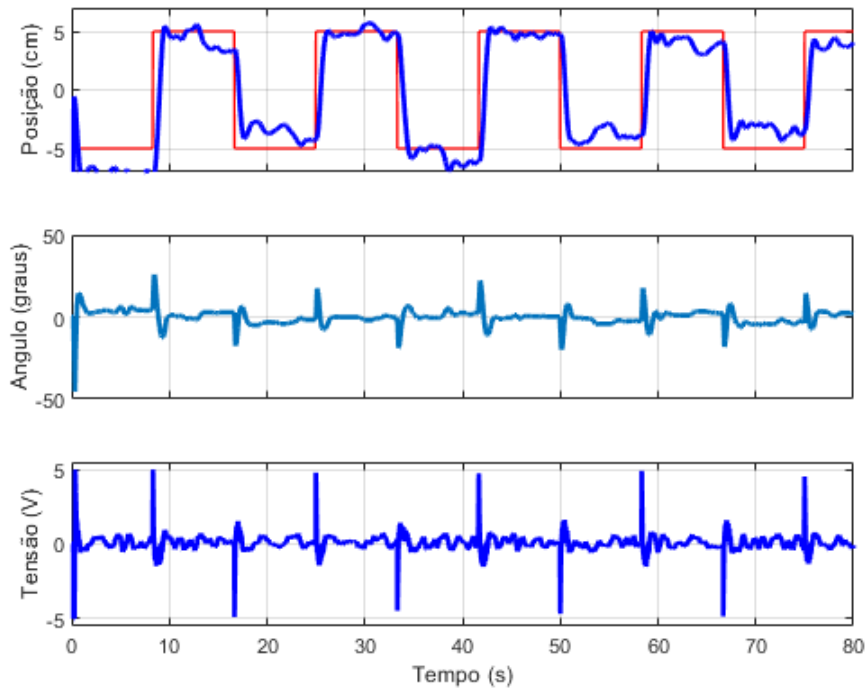
Após testados nas simulações, os controladores desenvolvidos foram aplicados no *Ball Balancer*, com os valores encontrados anteriormente.

O equipamento é conectado a um computador de bancada e possui arquivos padrão que podem ser executados pelo MATLAB. Para rodar um controlador basta substituir o circuito de controle original pelo novo.

### 6.1 CONTROLADOR PD PARA ENTRADA QUADRADA

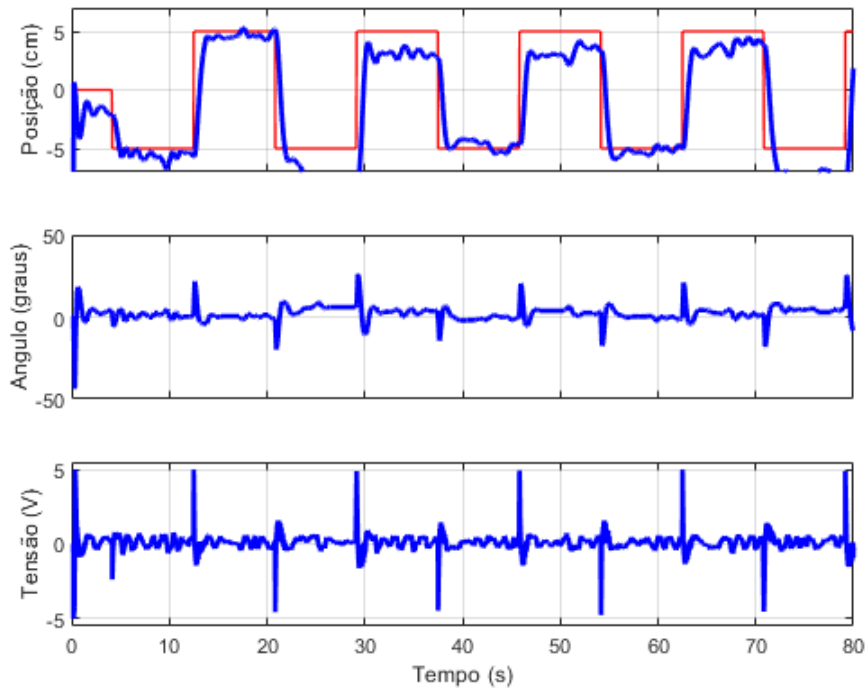
Ao rodar o circuito do controlador PD com uma forma de onda quadrada de amplitude 5cm em ambos os eixos, foi obtido como resultado, o gráfico da Figura 29, de tensão do motor, ângulo de rotação e posição da bola, para o eixo X e da Figura 30 para o eixo Y.

Figura 29 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PD com entrada quadrada



Fonte: Elaborado pelo autor

Figura 30 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PD com entrada quadrada



Fonte: Elaborado pelo autor

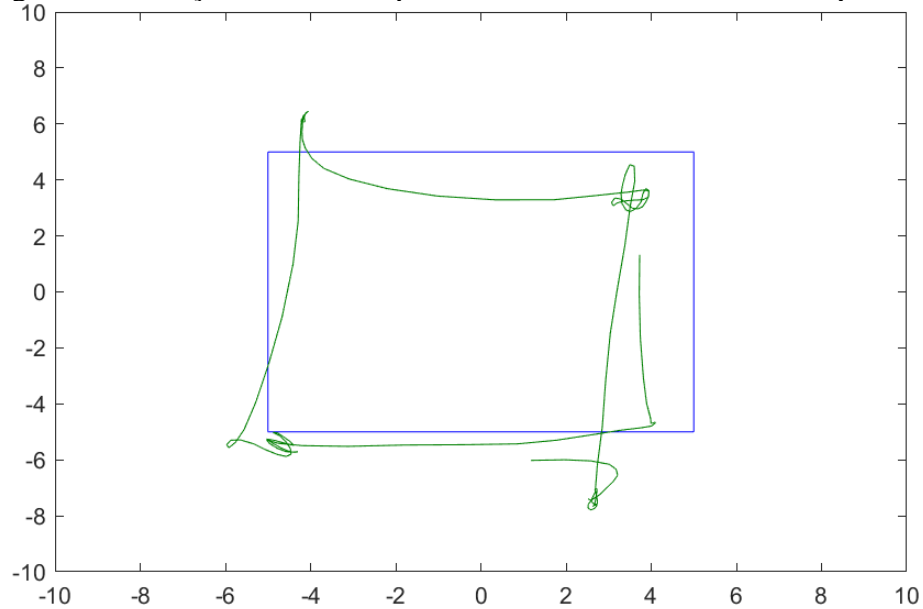
Assim como foi assumido antes das simulações, os eixos X e Y são simétricos e apresentam resultados muito similares.

O sistema real apresenta muito mais ruídos do que a simulação, tendo certa dificuldade em manter a bola fixa em um ponto. No entanto, desconsiderando os ruídos, o resultado é positivo, não tem um alto *overshoot* inicial e consegue entrar em regime estacionário rapidamente.

Os picos de tensão são um pouco maiores do que os da simulação, mas ainda dentro de uma faixa de tensão baixa e que não apresentaria nenhum risco para o equipamento.

Para mais uma análise, foi plotado também o gráfico da posição no eixo X versus a posição no eixo Y, deste mesmo experimento. Este gráfico está representado na Figura 31. Ele mostra de forma mais visual, o trajeto feito pela bola ao redor da mesa, sendo que o resultado deveria formar um quadrado com lado de 5cm. No entanto, só é possível analisar um pequeno intervalo de tempo dentro destes 40 segundos, já que ao voltar a um ponto por onde já passou, o gráfico começa a se sobreescrever, por isso é vista uma trajetória de aproximadamente 360°.

Figura 31 - Posição X versus Y para controlador PD com entrada quadrada



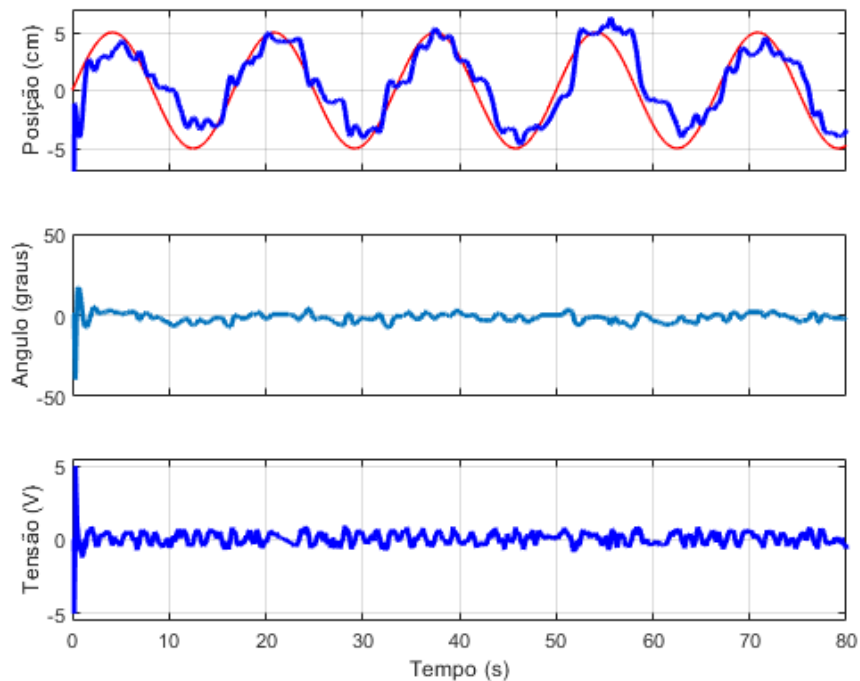
Fonte: Elaborado pelo autor

O objeto realizou a trajetória esperada, apenas com uma pequena defasagem.

## 6.2 CONTROLADOR PD PARA ENTRADA SENOIDAL

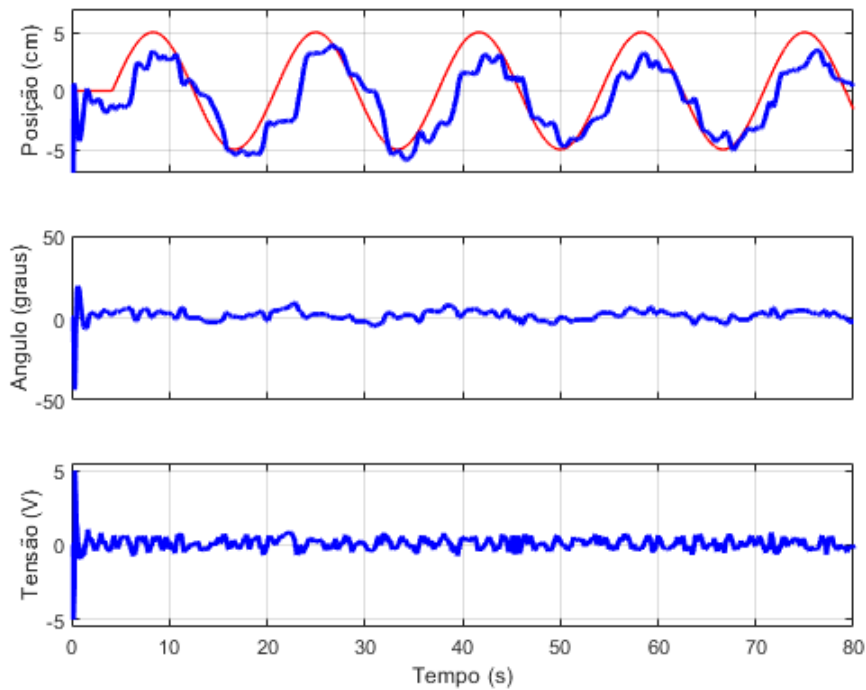
Executando o mesmo circuito de controle PD, com uma entrada senoidal de frequência 0,06Hz para ambos os eixos, foram obtidos os resultados mostrados nas Figura 32 e Figura 33.

Figura 32 - Posição X(s), tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PD com entrada senoidal



Fonte: Elaborado pelo autor

Figura 33 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PD com entrada senoidal

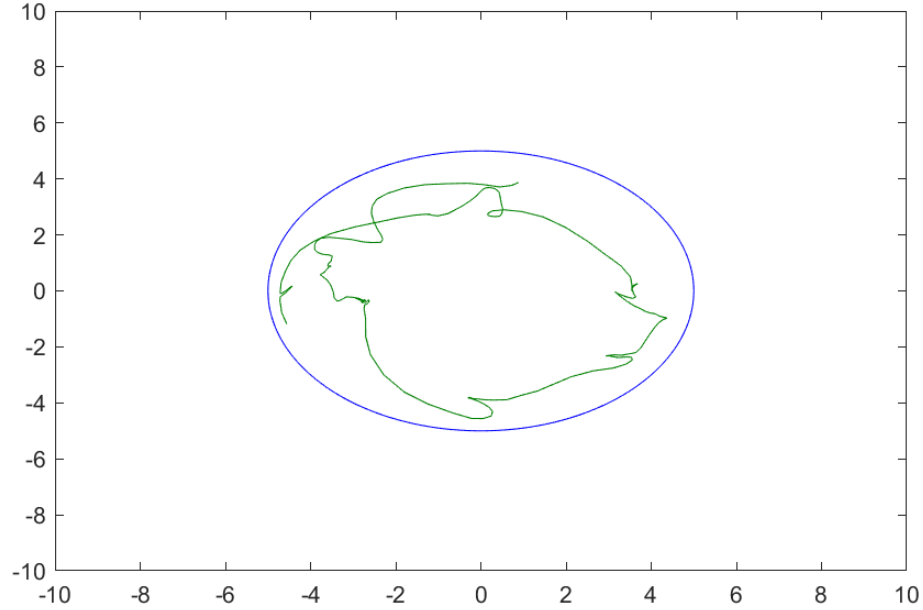


Fonte: Elaborado pelo autor

A trajetória senoidal, ao contrário da quadrada, não exige picos de tensão, mas apresenta maiores ruídos, já que o objeto fica em movimento em ambos os eixos durante todo o tempo, enquanto na quadrada apenas se move em um eixo por vez.

O trajeto feito pela bola em um pequeno intervalo de tempo desta execução é ilustrado na Figura 34. A junção de uma onda senoidal nos dois eixos forma uma imagem circular de 5cm de diâmetro.

Figura 34 - Posição X versus Y para controlador PD com entrada senoidal



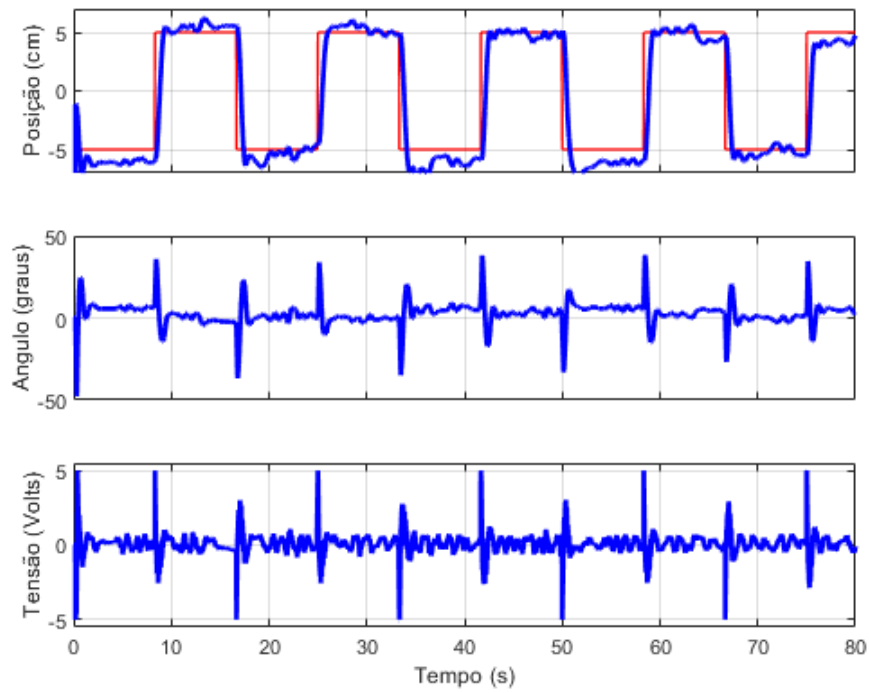
Fonte: Elaborado pelo autor

A defasagem se apresenta um pouco maior para este caminho, mas ainda sendo possível identificar bem a forma desejada.

### 6.3 CONTROLADOR PID PARA ENTRADA QUADRADA

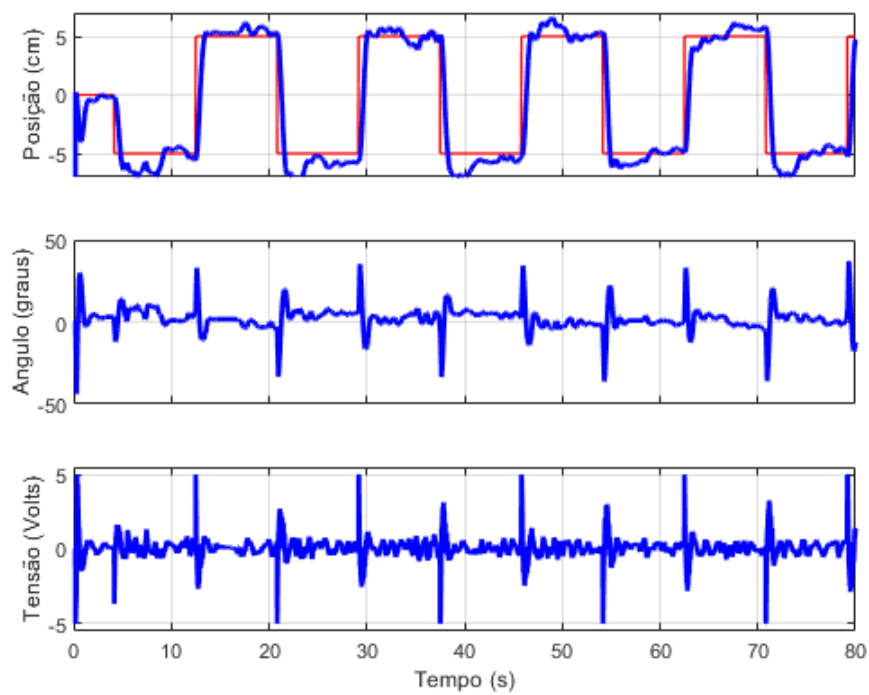
A seguir, foi trocado o controlador para o modelo que utiliza PD para o motor e PID para a mesa. O resultado desta etapa, considerando uma entrada quadrada, é mostrado nas Figura 35 e Figura 36.

Figura 35 - Posição  $X(s)$ , tensão  $Vx(s)$  e ângulo  $\theta x(s)$  para controlador PID com entrada quadrada



Fonte: Elaborado pelo autor

Figura 36 - Posição  $X(s)$ , tensão  $Vx(s)$  e ângulo  $\theta x(s)$  para controlador PID com entrada quadrada

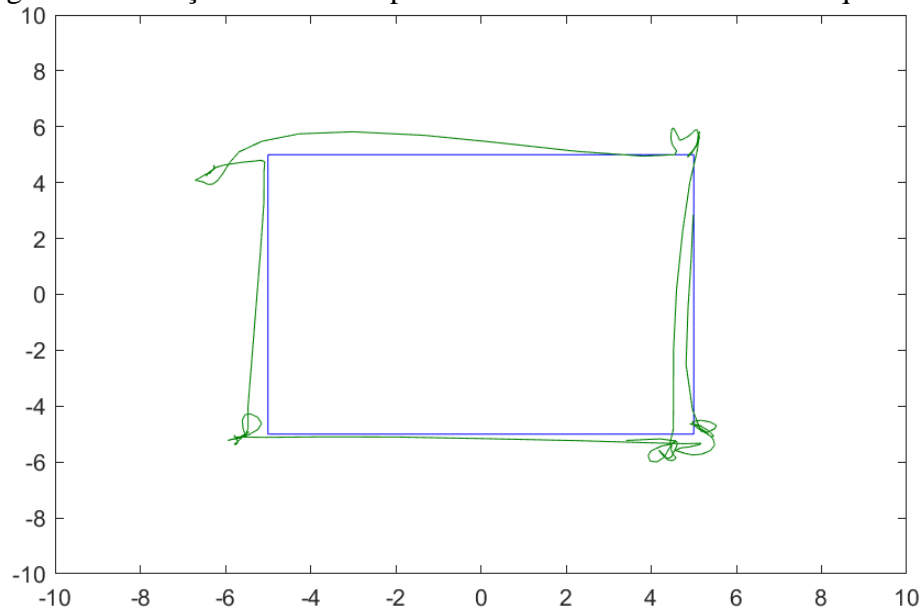


Fonte: Elaborado pelo autor

Assim como esperado pela teoria, o controlador PID apresenta menores ruídos, constituindo um menor erro em regime permanente visivelmente menor do que o anterior. Também se tem um tempo de estabelecimento ligeiramente menor e o *overshoot* continua insignificante.

Na Figura 37 é representado o curso realizado pela bola durante a execução com este controlador, que resulta em uma trajetória quadrada.

Figura 37 - Posição X versus Y para controlador PID com entrada quadrada



Fonte: Elaborado pelo autor

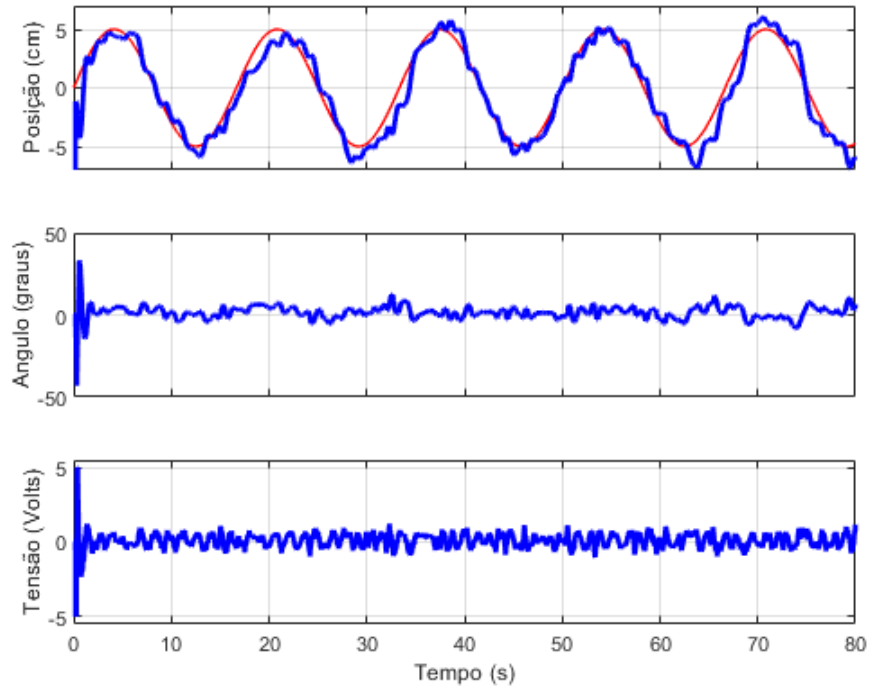
A melhora em relação ao controlador PD também é visível neste gráfico, com uma defasagem menor, o objeto se mantém mais próximo do trajeto esperado.

#### 6.4 CONTROLADOR PID PARA ENTRADA SENOIDAL

Por fim, tendo uma forma de onda senoidal como entrada do controlador PID, são obtidos os gráficos das Figura 38 e

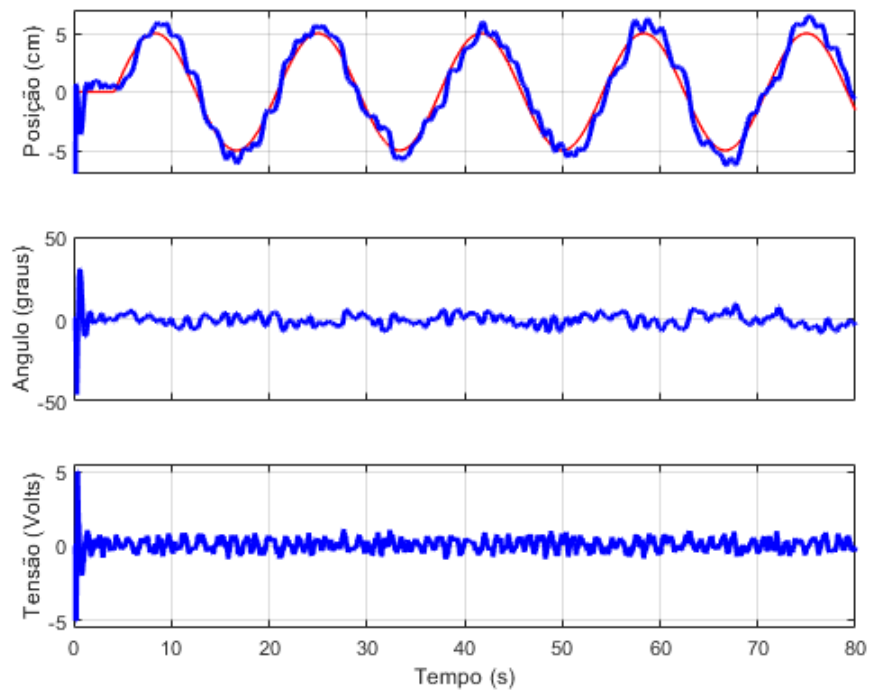
Figura 39, para cada eixo.

Figura 38 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PID com entrada senoidal



Fonte: Elaborado pelo autor

Figura 39 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PID com entrada senoidal



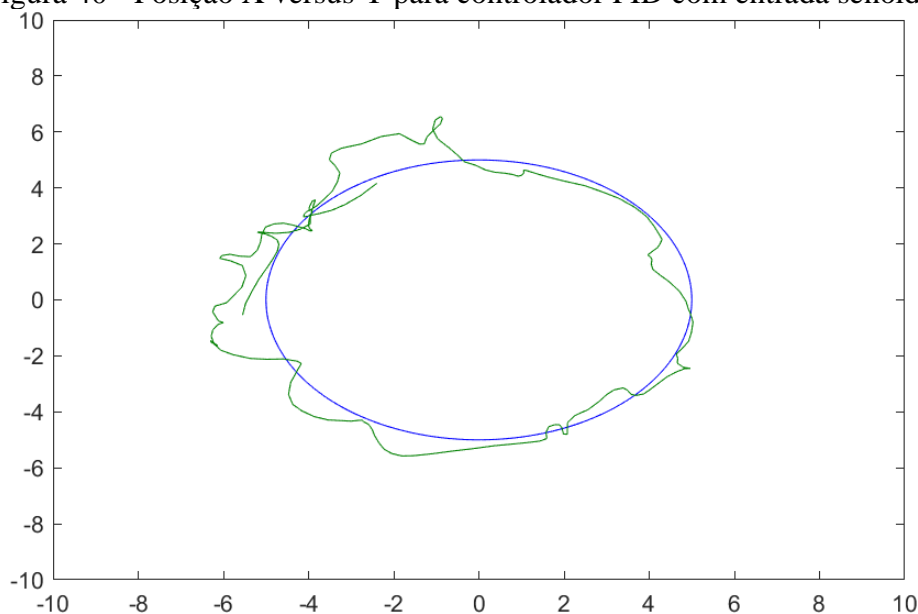
Fonte: Elaborado pelo autor

O mesmo que foi notado para a trajetória quadrada, vale para o caso com entrada senoidal. O movimento constante causa maiores ruídos do que a entrada quadrada, mas ele se mantém mais próximo da trajetória do que o controlador sem o integrador.

Em todos os resultados vistos até agora, durante os primeiros segundos há um pico de tensão do motor. Isso se deve ao reset inicial do equipamento, para depois retornar para a posição inicial e começar o movimento.

Na Figura 40 é mostrado o percurso circular resultante deste experimento.

Figura 40 - Posição X versus Y para controlador PID com entrada senoidal



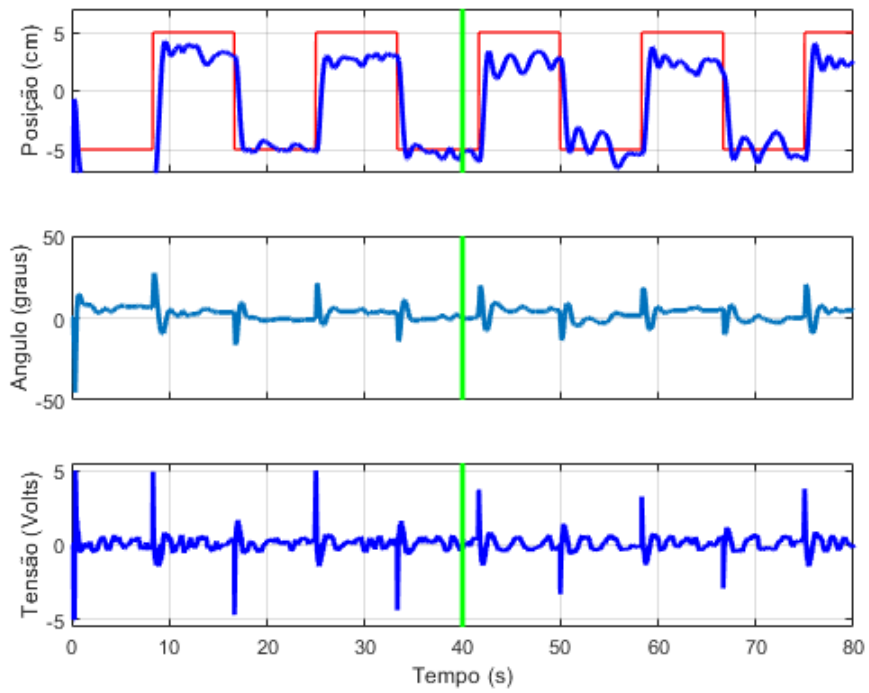
Fonte: Elaborado pelo autor

## 6.5 RESULTADOS COM FALHAS

Considerando agora que o motor apresentasse falhas de funcionamento de 30%, 50% e 70% da sua potência, foi implementado o circuito com chave mostrado anteriormente, que considera perdas a partir de 40s de execução, todos com entrada quadrada.

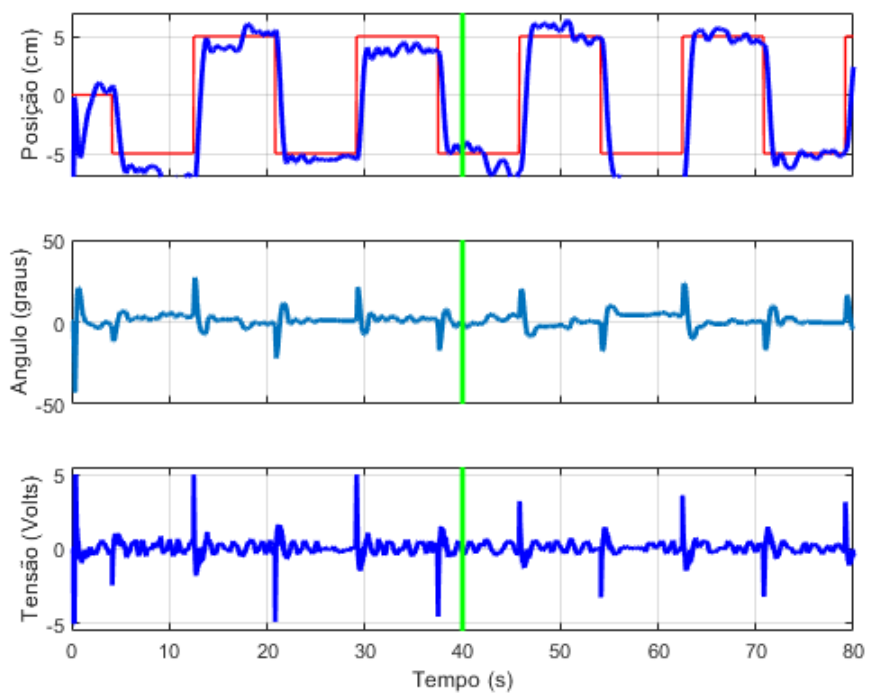
No caso do controlador PD com 30% de falha, nas Figura 41 a Figura 43 são mostrados os resultados. As figuras de X versus Y consideram falha em todo o percurso, ao contrário dos gráficos por tempo.

Figura 41 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PD com falha de 30%



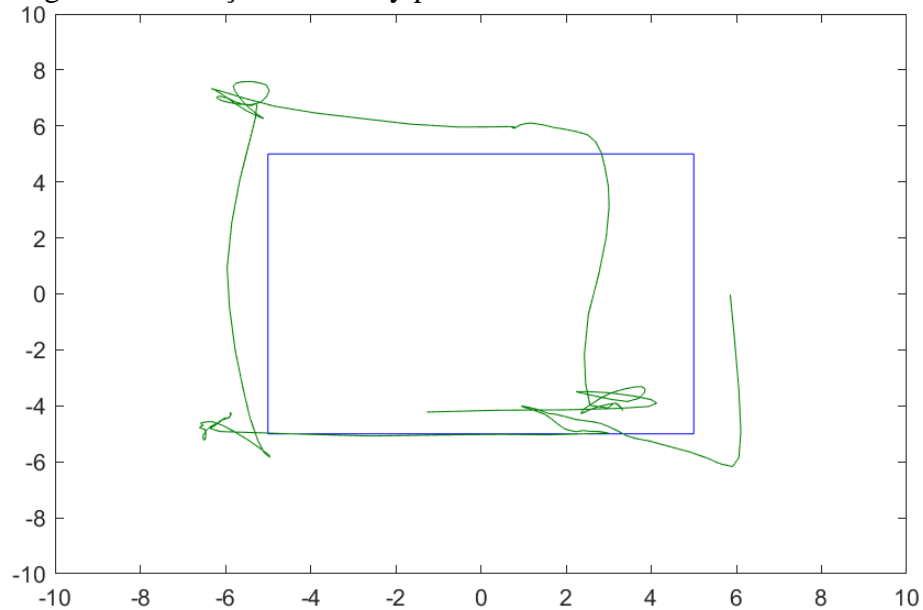
Fonte: Elaborado pelo autor

Figura 42 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PD com falha de 30%



Fonte: Elaborado pelo autor

Figura 43 - Posição x versus y para controlador PD com falha de 30%

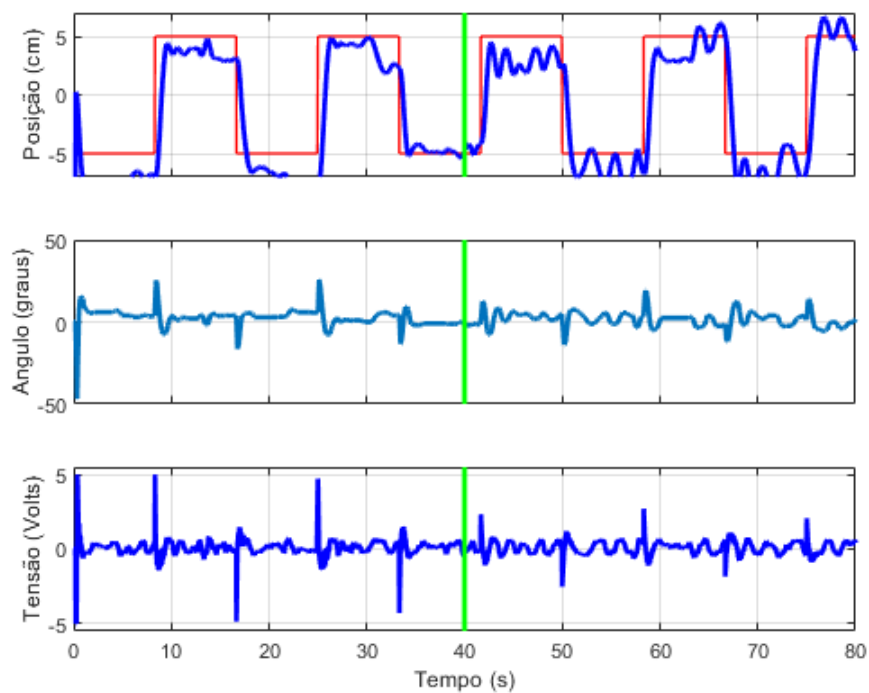


Fonte: Elaborado pelo autor

As Figuras anteriores mostram que após o início da falha, o controlador opera com mais ruídos, mas ainda capaz de realizar o movimento.

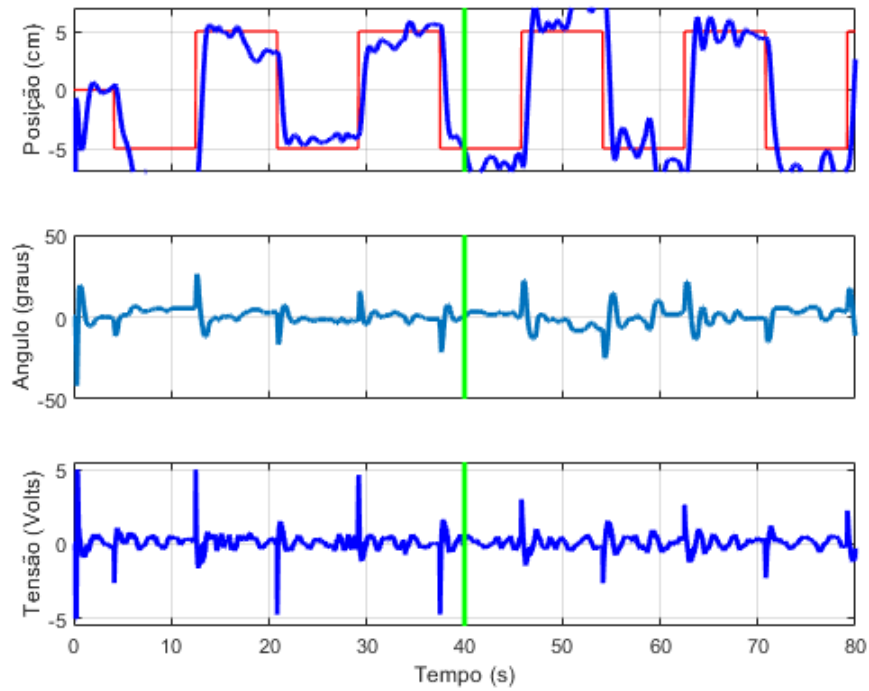
Para falha de 50% com o controlador PD, obtém-se os gráficos das Figura 44, Figura 45 e Figura 46.

Figura 44 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PD com falha de 50%



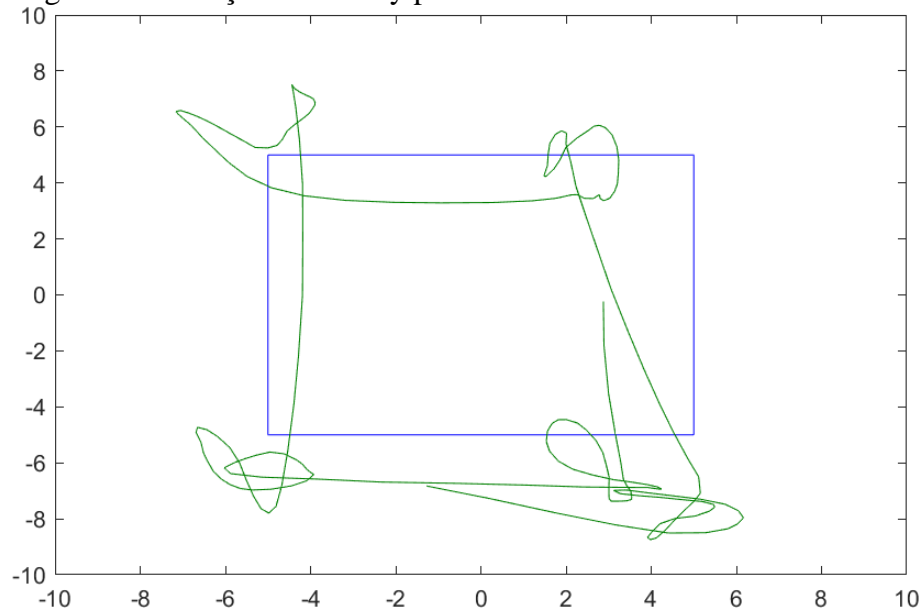
Fonte: Elaborado pelo autor

Figura 45 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PD com falha de 50%



Fonte: Elaborado pelo autor

Figura 46 - Posição  $x$  versus  $y$  para controlador PD com falha de 50%



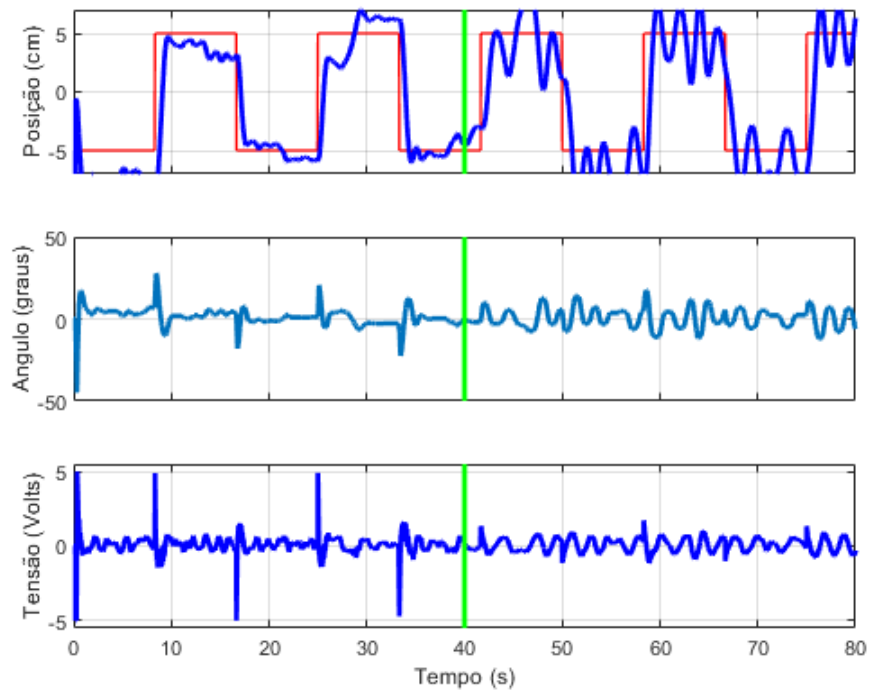
Fonte: Elaborado pelo autor

Os ruídos agora se tornam mais significativos e fica clara a dificuldade em manter a bola parada em um ponto. No gráfico de  $X$  versus  $Y$ , são feitas muitas voltas em torno do vértice do quadrado até ela conseguir atingir o ponto para o próximo movimento.

É interessante observar como o motor já não consegue atingir os picos de tensão, que foram restringidos.

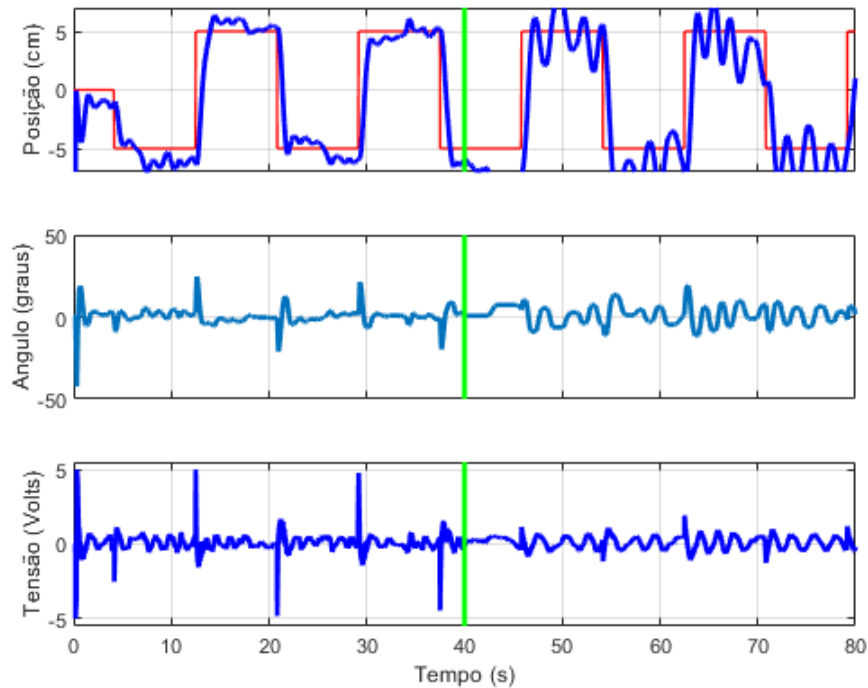
Com 70% de falha no motor, o controlador PD atua como mostrado nas Figura 47 a Figura 49.

Figura 47 - Posição  $X(s)$ , tensão  $Vx(s)$  e ângulo  $\theta x(s)$  para controlador PD com falha de 70%



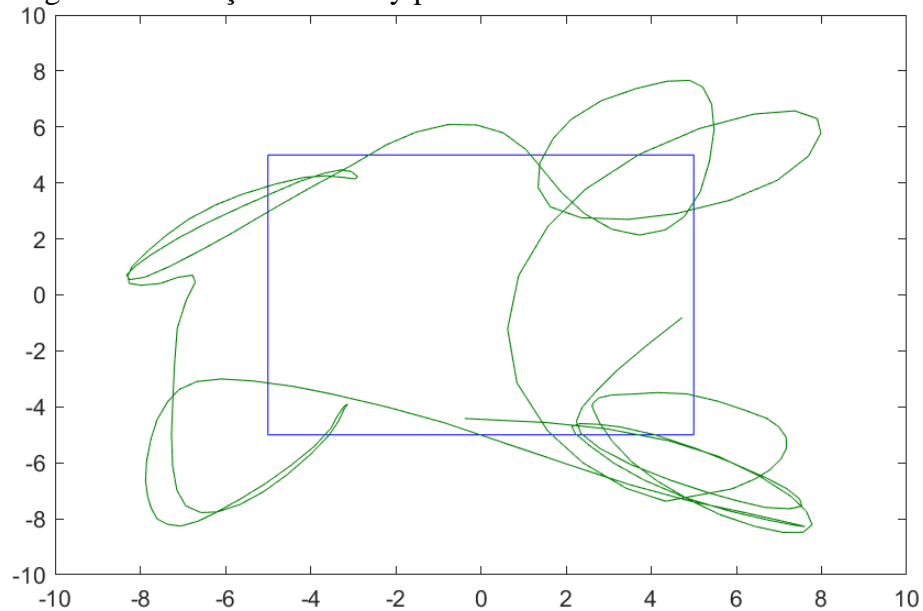
Fonte: Elaborado pelo autor

Figura 48 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PD com falha de 70%



Fonte: Elaborado pelo autor

Figura 49 - Posição  $x$  versus  $y$  para controlador PD com falha de 70%

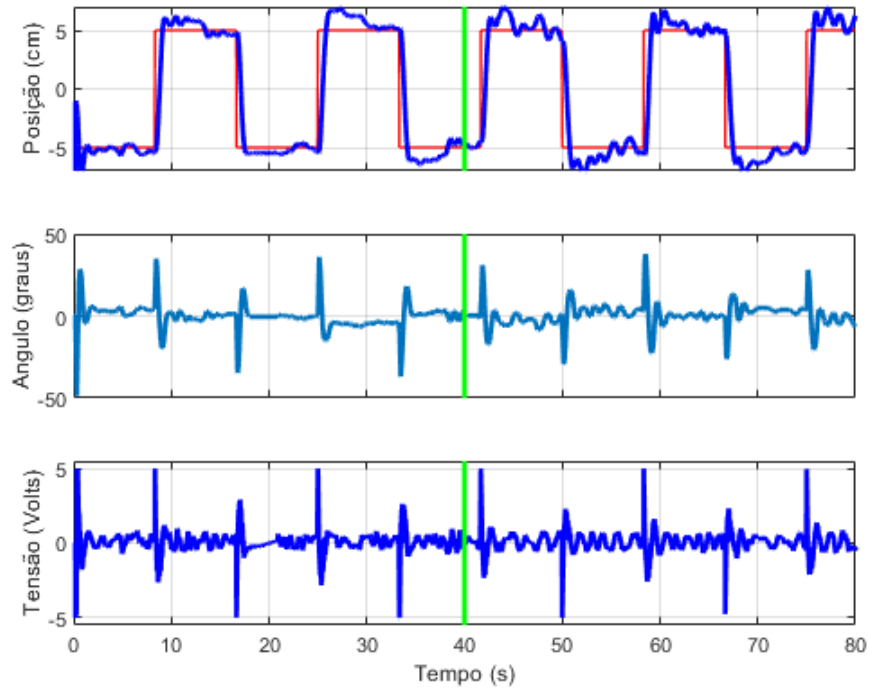


Fonte: Elaborado pelo autor

O objeto agora se afasta completamente da trajetória, fazendo voltas muito largas para se manter em uma posição e não conseguindo realizar o movimento reto da aresta do quadrado, já que não consegue atingir a tensão para a rotação necessária do motor. Mesmo assim, em nenhum momento o objeto foi derrubado da mesa.

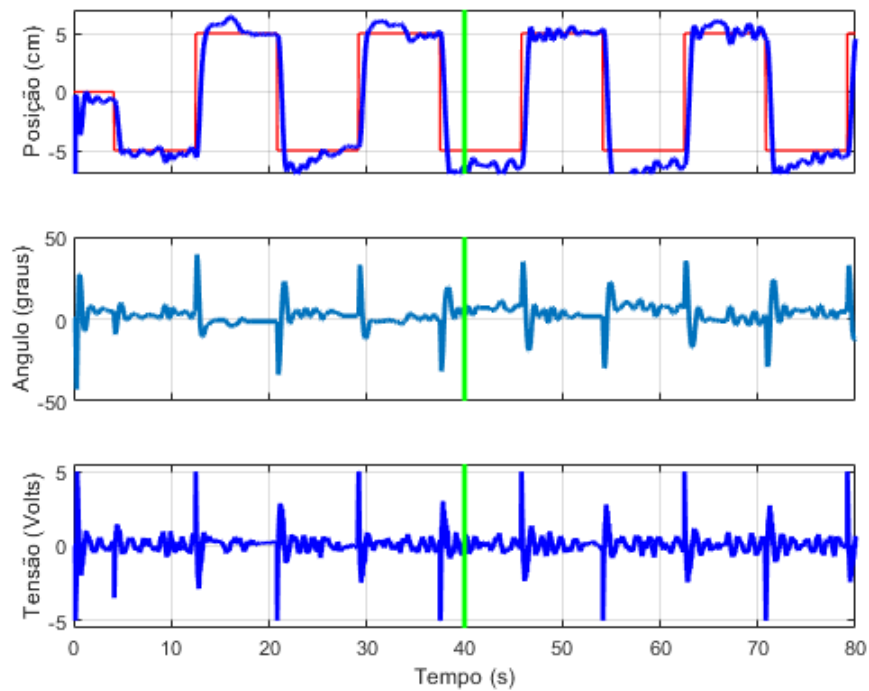
Também foi rodado o circuito do controlador PID para caso de perdas no motor. As Figura 50 a Figura 52 mostram as respostas para 30% de perda.

Figura 50 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PID com falha de 30%



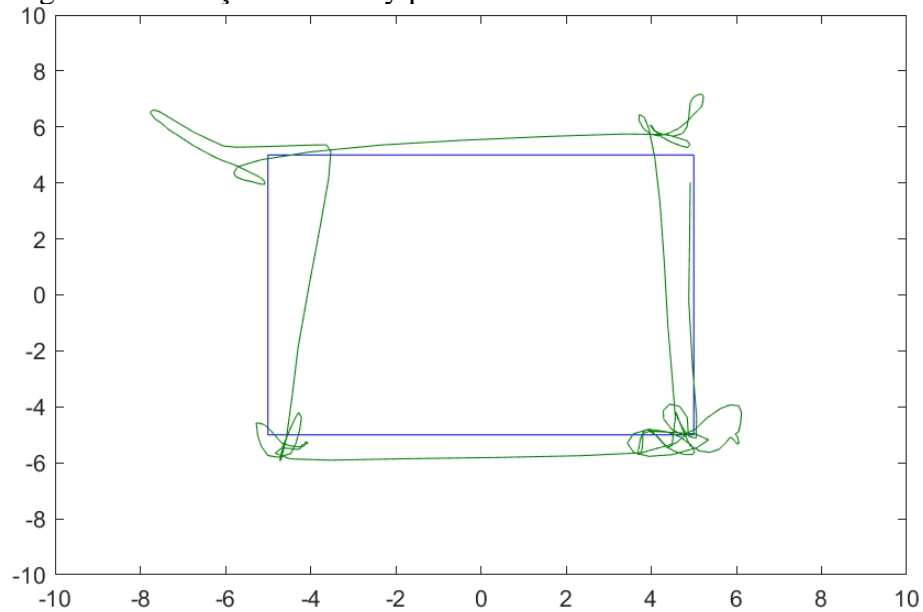
Fonte: Elaborado pelo autor

Figura 51 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PID com falha de 30%



Fonte: Elaborado pelo autor

Figura 52 - Posição x versus y para controlador PID com falha de 30%

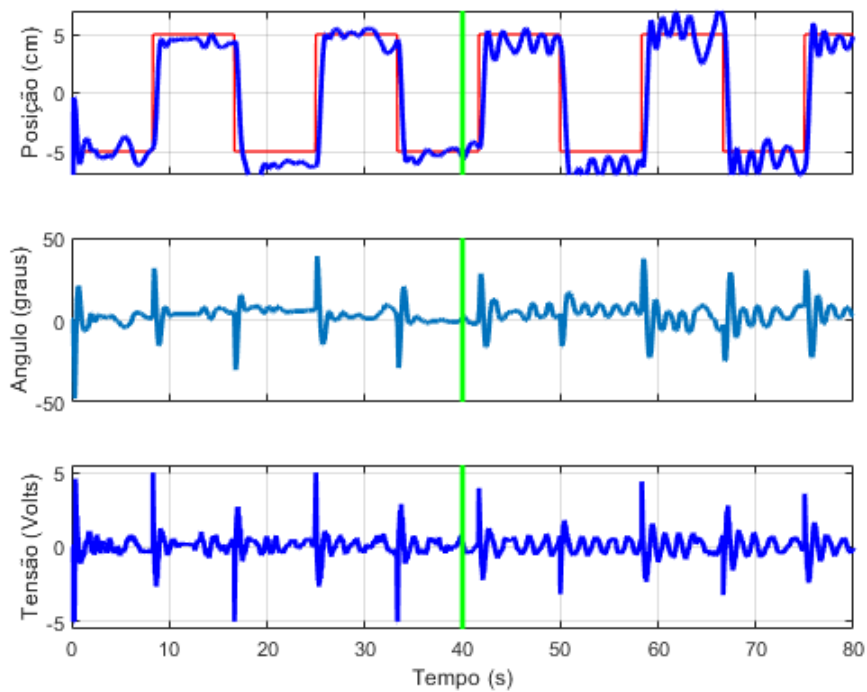


Fonte: Elaborado pelo autor

Nas Figura 50 a Figura 52, há um aumento de ruído após a perda, mas o controlador ainda consegue seguir o percurso com algumas defasagens.

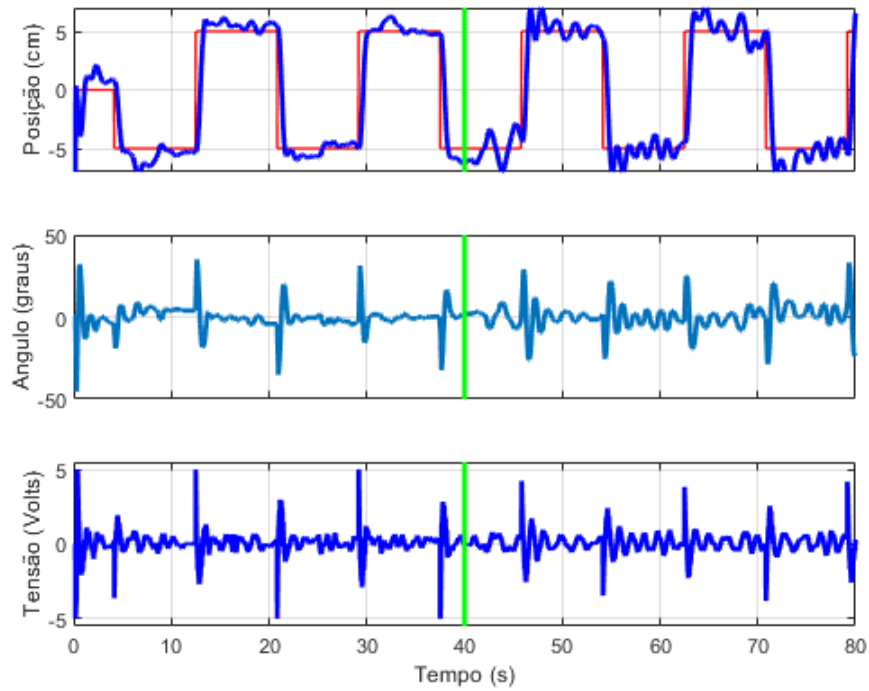
Nas Figura 53 a Figura 55 é mostrado o efeito de 50% de perda da potência do motor.

Figura 53 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PID com falha de 50%



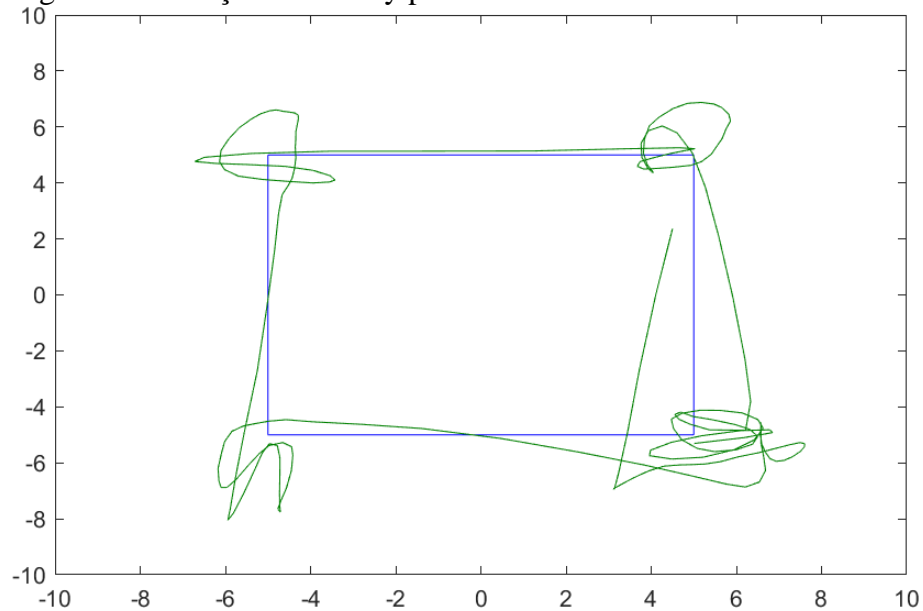
Fonte: Elaborado pelo autor

Figura 54 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PID com falha de 50%



Fonte: Elaborado pelo autor

Figura 55 - Posição  $x$  versus  $y$  para controlador PID com falha de 50%

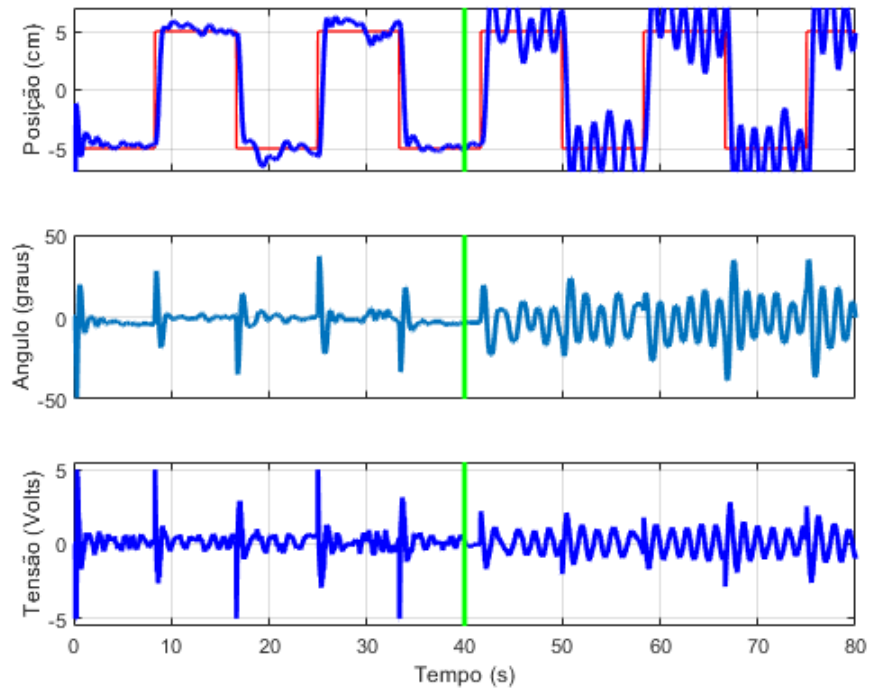


Fonte: Elaborado pelo autor

Apesar do problema em se manter enquanto parado, durante o movimento ele ainda se aproxima do esperado e a resposta continua com tempo de subida baixo.

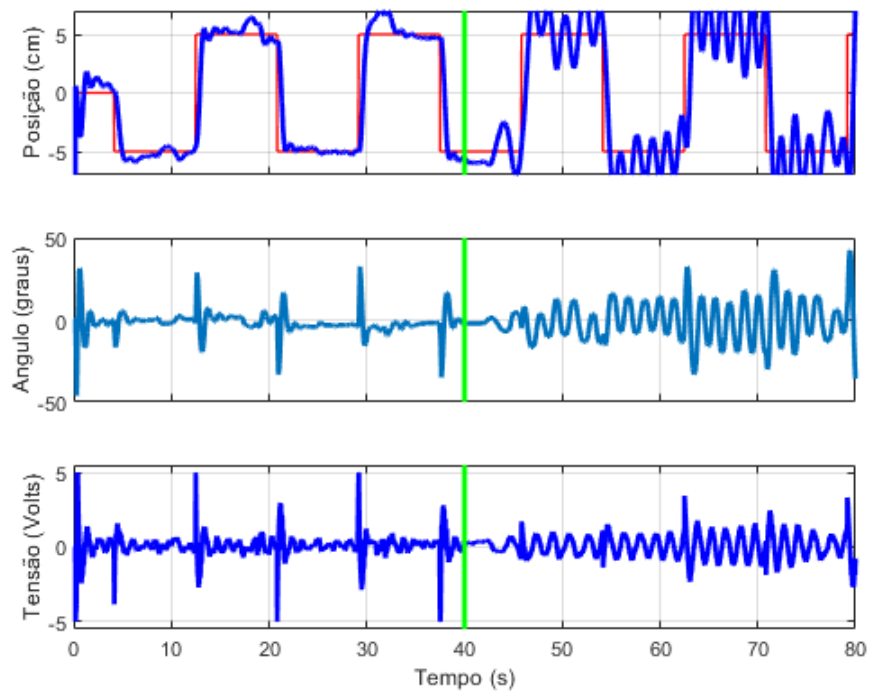
Já com 70% de falha no motor, o controlador PID demonstra o resultado das Figura 56 a Figura 58.

Figura 56 - Posição  $X(s)$ , tensão  $V_x(s)$  e ângulo  $\theta_x(s)$  para controlador PID com falha de 70%



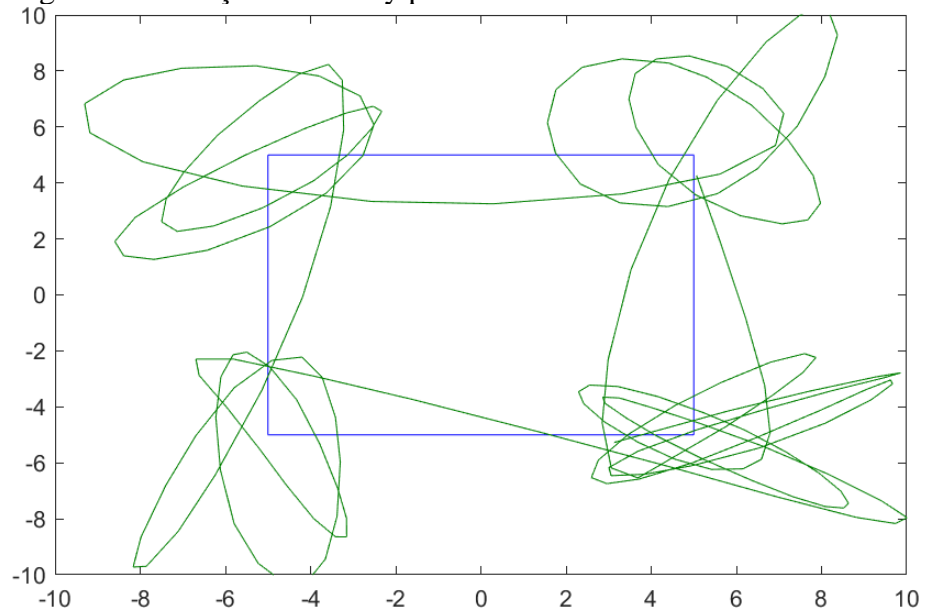
Fonte: Elaborado pelo autor

Figura 57 - Posição  $Y(s)$ , tensão  $V_y(s)$  e ângulo  $\theta_y(s)$  para controlador PID com falha de 70%



Fonte: Elaborado pelo autor

Figura 58 - Posição x versus y para controlador PID com falha de 70%



Fonte: Elaborado pelo autor

Com 70% de falha, o objeto se afasta muito do percurso, mas ainda capaz de retornar a ele sem cair da mesa.

Analisando os gráficos pelo tempo, a resposta do controlador PID é sempre mais rápida do que a do controlador PD quando há falhas.

## 7 CONCLUSÃO

As teorias de controle clássico PID, existentes há mais de 80 anos, continuam sendo uma boa base para estudo e pesquisa. Este trabalho teve por objetivo fazer um estudo aprofundado destes conceitos e aplicar os resultados em um sistema instável e não linear, de forma a complementar os assuntos estudados no curso de graduação de Engenharia Elétrica e dar continuidade aos trabalhos anteriores realizados no Laboratório de Pesquisa em Controle da Faculdade de Engenharia de Ilha Solteira.

O módulo *2DOF Ball Balancer* da Quanser se mostrou uma ótima ferramenta de aprendizado e demonstração, tanto das leis físicas utilizadas, mas principalmente das teorias de controle clássico.

As ferramentas do MATLAB se mostraram muito úteis, de forma a prever possíveis problemas, antes da implementação no equipamento real, com simulações que se aproximaram muito da realidade.

Foi comprovada a validade dos cálculos matemáticos, em atingir as expectativas para o controlador, estando os resultados iniciais dentro dos limites impostos, mesmo havendo margem para melhoria.

As melhorias aplicadas mostram a eficácia do loop *anti-windup* em impedir a saturação do integrador, diminuindo o *overshoot* e o tempo de estabelecimento. Também foi comprovada a eficácia da constante derivativa na diminuição do *overshoot* em ambos os controladores, como referia a Tabela 2.

Segundo a teoria apresentada, o maior benefício proporcionado pela parte integrativa do controlador seria em relação ao erro em estado estacionário. Apesar de não ser possível visualizar isto pelas simulações iniciais, já que o erro já é baixo o suficiente para ser desconsiderado, as simulações com falha demonstram a veracidade desta afirmação, pois o controlador PID, mesmo perdas altas na potência do motor, consegue se manter melhor dentro da trajetória.

Os resultados com o *Ball Balancer* também mostram uma vantagem do controlador PID, que ocasionou menores ruídos e menor tempo de estabelecimento. Isto fica evidente nas figuras de X versus Y, em que o controlador PD realiza a trajetória com uma defasagem muito maior do que o controlador PID.

Levando em consideração tudo que foi analisado, conclui-se que a parte integrativa do controlador contribui para um resultado geral melhor.

Observa-se também que o módulo tem maior dificuldade em realizar uma trajetória circular, do que uma trajetória quadrada e que seu maior obstáculo é manter o objeto parado em um ponto fixo. Mesmo assim, não houve queda em nenhum momento, nem mesmo no caso de o usuário causar uma interferência.

Considerando tudo que foi exposto acima, este trabalho cumpre o propósito de complementar a formação da graduação em Engenharia Elétrica, aprofundando a matéria estudada nas disciplinas de Controle Linear 1 e 2, além dos conceitos de física, de programação e simulação em MATLAB, mostrando uma aplicação experimental das teorias.

## 8 REFERÊNCIAS

- ASSUNÇÃO, E.; TEIXEIRA, M. C. M. “Controle Linear I. Parte A – Sistemas contínuos no tempo”. Apostila de Controle Linear I. Univ. Estadual Paulista (UNESP). Departamento de Engenharia Elétrica da Faculdade de Engenharia de Ilha Solteira (FE-IS). 240 p., 2013.
- C. Knospe, "PID control," in *IEEE Control Systems Magazine*, vol. 26, no. 1, pp. 30-31, Feb. 2006, doi: 10.1109/MCS.2006.1580151.
- DORF, R. C.; BISHOP, R. H. “Sistemas de controle modernos”. Livros Técnicos e Científicos, 8ª ed. Rio de Janeiro, 2011.
- H. Bang and Y. S. Lee, "Embedded Model Predictive Control for Enhancing Tracking Performance of a Ball-and-Plate System," in *IEEE Access*, vol. 7, pp. 39652-39659, 2019, doi: 10.1109/ACCESS.2019.2907111.
- H. Bang and Y. S. Lee, "Implementation of a Ball and Plate Control System Using Sliding Mode Control," in *IEEE Access*, vol. 6, pp. 32401-32408, 2018, doi: 10.1109/ACCESS.2018.2838544.
- HALLIDAY, D.; RESNICK, R. “Fundamentos de Física”. Livros Técnicos e Científicos Editora, 9ª ed. Rio de Janeiro, 2012.
- MUNIZ, S. R.; "Resumo sobre controladores PID". USP, 2017.
- OGATA, K. “Engenharia de Controle Moderno”. Person, 5ª ed. São Paulo, 2014.
- QUANSER. “2D Ball Balancer Control using QUARC – Instructor Manual”. Canada, 2008a.
- QUANSER. “2D Ball Balancer Control using QUARC – Student Manual”. Canada, 2008c.
- QUANSER. “2D Ball Balancer – User Manual”. Canada, 2008b.
- QUANSER, “The Rotatory Control Lab - A Modular Single Source Solution You Can Control”, 2020.
- S. U. Din, Q. Khan, F. -U. Rehman and R. Akmeliawanti, "A Comparative Experimental Study of Robust Sliding Mode Control Strategies for Underactuated Systems," in *IEEE Access*, vol. 5, pp. 10068-10080, 2017, doi: 10.1109/ACCESS.2017.2712261.

## APÊNDICE A: CÓDIGO PARA CÁLCULO E SIMULAÇÃO

```

clc;
clear;
close all;
fprintf('CONTROLE PD/PID - BALL BALANCER \n');

% Variáveis predefinidas
g = 9.8;
L = 0.275;
r = 0.0254;
K = 1.76;
tau = 0.0285;
Kbb = (3*g*r)/(2*L);

% Especificações de projeto do motor
tp = input('Qual o máximo tempo de pico do motor? \n');
POs = input('Qual a máxima porcentagem de overshoot do motor? \n');

% Cálculo de ganhos do controlador do motor
fprintf('Calculando ganhos do controlador do motor... \n');
Es = sqrt((log(POs/100))^2/((pi^2)+(log(POs/100))^2));
wns = pi/(tp*sqrt(1-(Es^2)));
Kps = (wns^2)*tau/K;
Kds = tau*(2*Es*wns-(1/tau))/K;

% Especificações de projeto da mesa
ts = input('Qual o máximo tempo de estabelecimento da mesa? \n');
POb = input('Qual a máxima porcentagem de overshoot da mesa? \n');
cts = input('Qual a máxima porcentagem de tempo de assentamento da mesa? \n');

% Cálculo de ganhos do controlador da mesa
Eb = sqrt((log(POb/100))^2/((pi^2)+(log(POb/100))^2));
wnb = -log(cts*sqrt(1-(Eb^2)))/(ts*Eb);

% Escolha do tipo de controlador da mesa
type = input('Qual o tipo de controlador? Digite 1 para controlador PD
ou 2 para controlador PID \n');
fprintf('Calculando ganhos do controlador da mesa... \n');
if (type==1)
    po = 0;
end
if (type==2)
    Tp = 1;
    po = 1/Tp;
end
Kib = (wnb^2)*po/Kbb;
Kpb = ((wnb^2)+2*Eb*wnb*po)/Kbb;
Kdb = (2*Eb*wnb+po)/Kbb;

% Mostrar resultados para o usuário
fprintf(' \n RESULTADOS \n');
fprintf('Ganho proporcional do motor Kps = %f \n',Kps);
fprintf('Ganho derivativo do motor Kds = %f \n',Kds);
fprintf('Ganho proporcional da mesa Kpbb = %f \n',Kpb);
fprintf('Ganho derivativo da mesa Kdbb = %f \n',Kdb);

```

```

fprintf ('Ganho integral da mesa Kibb = %f \n',Kib);

% Simulação
exe = input(' \n Gostaria de executar a simulação? Se sim digite 1 \n');
if (exe==1)

    % Escolha do usuário em alterar os ganhos
    melhoria = input('Deseja prosseguir com os ganhos calculados? Se sim
digite 1, se não digite 2 \n');
    if (melhoria==2)
        Kps = input('Qual o novo valor de Kps? \n');
        Kds = input('Qual o novo valor de Kds? \n');
        Kpb = input('Qual o novo valor de Kpbb? \n');
        Kdb = input('Qual o novo valor de Kdbb? \n');
        Kib = input('Qual o novo valor de Kibb? \n');
    else
        fprintf ('Proseguindo com os valores calculados... \n');
    end

    % Escolha do usuário em considerar perdas no motor
    falha = input('Gostaria de considerar falhas no motor? Se sim digite 1,
se não digite 2 \n');
    fprintf ('Executando simulação... \n');
    bsd = input('Qual o valor de bsd? \n');

    % Simulação com falha
    if (falha==1)
        perda = 1-0.01*(input('Qual a porcentagem de perda do motor? \n'));
        if (type==1)
            INTMAX = 0.1;
            Tr = 1;
        end
        if (type==2)
            INTMAX = deg2rad(input('Qual o limite de saturação do
integrador em graus? \n'));
            Tr = input('Qual a constante de tempo do anti-windup? \n');
        end

        diagramapdpidfalha;
        sim diagramapdpidfalha;

        % Plotar gráfico com falha
        load x;
        load theta;
        load v;
        load xd;
        figure (1);
        subplot(3,1,1);
        plot(xd(1,:),xd(2,:), 'r', 'LineWidth',1);
        hold on;
        plot(x(1,:),x(2,:), 'b', [40 40], [-10 10], 'g', 'LineWidth',2);
        hold off;
        set(gca, 'XTickLabel', []);
        ylabel('Posição (cm)');
        axis([0 80 -7 7]);
        grid on;
        subplot(3,1,2);
        plot(theta(1,:),theta(2,:), 'b', [40 40], [-10 10], 'g', 'LineWidth',2);
        set(gca, 'XTickLabel', []);
        ylabel('Angulo (graus)');

```

```

axis([0 80 -20 20]);
grid on;
subplot(3,1,3);
plot(v(1,:),v(2:,:), 'b', [40 40], [-10 10], 'g', 'LineWidth',2);
xlabel('Tempo (s)');
ylabel('Tensão (Volts)');
axis([0 80 -3 3]);
grid on;
end

% Simulação sem falha
if (falha==2)
    if (type==1)
        INTMAX = 0.1;
        Tr = 1;
    end
    if (type==2)
        INTMAX = deg2rad(input('Qual o limite de saturação do
integrador em graus? \n'));
        Tr = input('Qual a constante de tempo do anti-windup? \n');
    end
end

diagramapdpid;
sim diagramapdpid;

% Plotar gráfico sem falha
load x;
load theta;
load v;
load xd;
figure (1);
subplot(3,1,1);
plot(xd(1,:),xd(2,:), 'r', 'LineWidth',1);
hold on;
plot(x(1,:),x(2,:), 'b', 'LineWidth',2);
hold off;
set(gca, 'XTickLabel', []);
ylabel('Posição (cm)');
axis([0 80 -7 7]);
grid on;
subplot(3,1,2);
plot(theta(1,:),theta(2,:), 'b', 'LineWidth',2);
set(gca, 'XTickLabel', []);
ylabel('Angulo (graus)');
axis([0 80 -20 20]);
grid on;
subplot(3,1,3);
plot(v(1,:),v(2:,:), 'b', 'LineWidth',2);
xlabel('Tempo (s)');
ylabel('Tensão (Volts)');
axis([0 80 -3 3]);
grid on;
end

end
fprintf ('PROGRAMA FINALIZADO \n');

```