



UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA MECÂNICA

ES828 - Laboratório de Controle de Sistemas

Pré Relatório - Experimento 3

Controle de plantas eletrônicas utilizando um
controlador PID digital

Nome:

Daniel Dello Russo Oliveira

Marcelli Tiemi Kian

RA

101918

117892

16 de março de 2015

1 Objetivos

O objetivo desse experimento é a familiarização com as ferramentas de design e simulação de sistemas de controle pelo Matlab e o estudo do projeto de controladores PID.

2 Projeto dos Controladores

Consideramos a planta cuja função de transferência representada pela equação 1 que foi obtida usando as medidas realizadas durante o experimento 2, consideramos também para todos os controladores uma envoltória $\epsilon = 2\%$. Cujos valores estão na tabela 1

$$G(s) = \frac{\kappa_1 * \kappa_2 * \kappa_3 * \kappa_4}{(s * \tau_2 + 1)(s * \tau_3 + 1)s} \quad (1)$$

Tabela 1: Parâmetros numéricos da função de transferência

Componente	Valor
κ_1	-0.1005
κ_2	-2.1508
κ_3	-4.6448
κ_4	-5.6307
τ_2	0.0210
τ_3	0.0244

2.1 Projeto do controlador proporcional

Através do método do lugar das raízes, projetamos um controlador proporcional $C(s) = \kappa$ de forma a obter um tempo de estabilização mínimo, para isto inicialmente plotamos com o auxílio do Matlab o diagrama de lugar das raízes para a planta, que pode ser observado na figura 1 e analisamos o tempo de estabilização do sistema para dois pontos que poderiam minimizar esse valor, um com amortecimento crítico e o outro com sobrelevação de 2%.

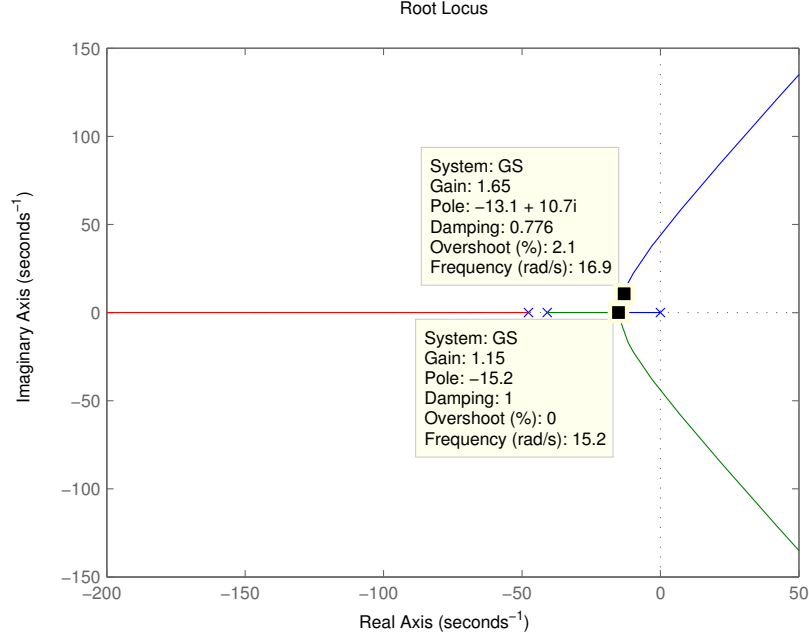


Figura 1: Diagrama de lugar das raízes para a $G(s)$

Utilizando as equações fornecidas do [1]:

$$t_e = -\frac{\ln(\epsilon)}{\xi\omega_n} \quad (2)$$

$$\psi(\xi) = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}} \quad (3)$$

$$\epsilon = \lim_{s \rightarrow 0} \frac{s}{1 + k * G(s)} * \frac{1}{s^k} \quad (4)$$

Logo, utilizando os ganhos representados no gráfico calculamos o coeficiente de amortecimento e a frequência natural dos sistemas em malha fechada com o auxílio do Matlab e obtivemos os seguintes valores:

- **Amortecimento crítico**

Verificamos o ganho desse item calculando o ponto onde $\frac{k*G(s)}{1+K*G(s)}$ tem multiplicidade dois, para isso encontramos as raízes s_1 e s_2 da derivada de $\frac{1}{G(s)}$, substituindo então na relação $k = -\frac{1}{(G(s))}$ encontramos dois valores

possíveis para k , comparando os valores com os obtidos no diagrama de lugar das raízes, confirmamos o ganho $k = 1.1515$. Para $\xi = 1$ e $\omega_n = 14.0$

$$t_e = 0.28s \quad (5)$$

$$\psi(\xi) = 0 \quad (6)$$

Erro estático da resposta ao degrau:

$$\epsilon = 0 \quad (7)$$

Erro estático da resposta a uma rampa:

$$\epsilon = 0.153633 \quad (8)$$

- **Sobrelevação de 2%**

Com o auxílio do sisotool encontramos o valor para o ponto de sobrelevação 2% com maior precisão, utilizamos então o ganho de 1.6649 para o controlador. Para $\xi = 0.776$ e $\omega_n = 17.2$

$$t_e = 0.2931s \quad (9)$$

$$\psi(\xi) = 2\% \quad (10)$$

Erro estático da resposta ao degrau:

$$\epsilon = 0 \quad (11)$$

Erro estático da resposta a uma rampa:

$$\epsilon = 0.106247 \quad (12)$$

Plotamos então as respostas ao degrau e rampa dos sistemas com cada um dos controladores que podem ser vistas nas figuras 2, 4, 3, 5.

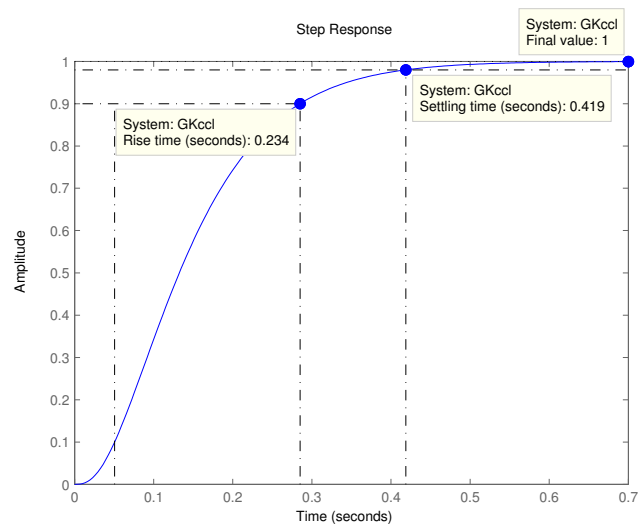


Figura 2: Resposta ao degrau do sistema com controlador projetado para amortecimento crítico

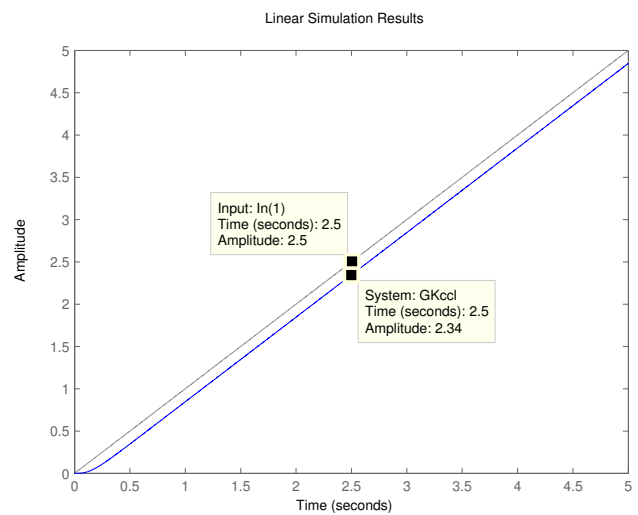


Figura 3: Resposta à rampa do sistema com controlador projetado para amortecimento crítico

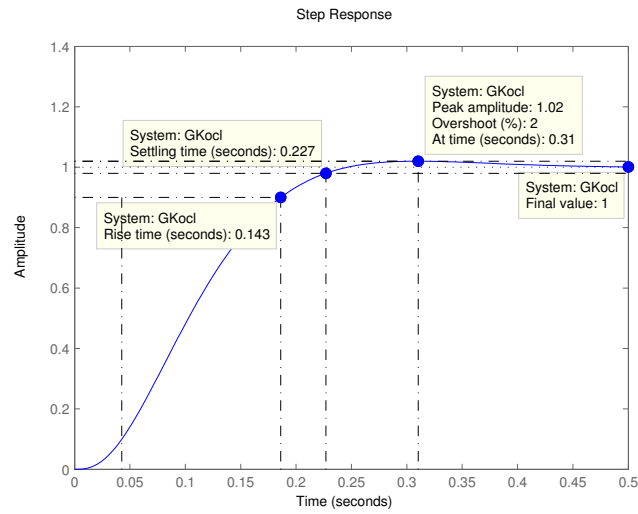


Figura 4: Resposta ao degrau do sistema com controlador projetado para sobrelevação de 2%

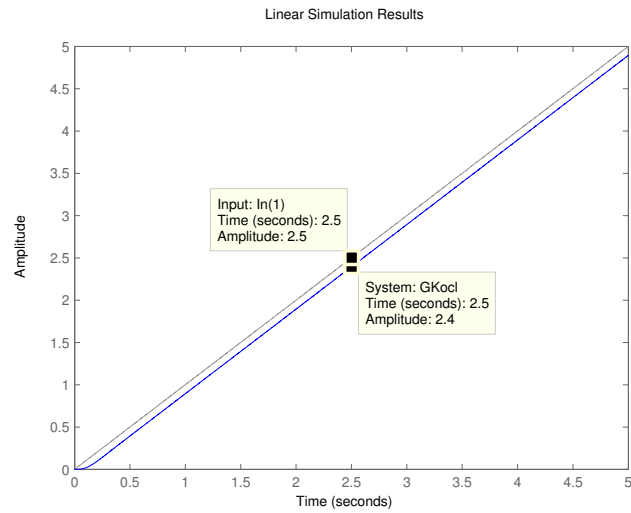


Figura 5: Resposta à rampa do sistema com controlador projetado para sobrelevação de 2%

Comparando os valores calculados, com os obtidos na simulação, podemos perceber que eles são brm próximos, porém ainda existe uma discrepância no

tempo de estabilização, isso ocorre devido às aproximações que fizemos durante ao cálculo (só levando em consideração os dois maiores polos) e devido às aproximações que são feitas pelo Matlab durante a simulação do sistema. O ponto com amortecimento crítico é o que minimiza a função representada na equação 2, porém pela simulação, podemos observar que o sistema com sobrelevação de 2% apresenta resultados melhores.

3 Projeto do controlador PID - Ziegler Nichols

Através do método de Ziegler Nichols, projetamos um controlador PID $C(s) = k_p * (1 + \frac{1}{T_i s} + T_d s)$. Para isto inicialmente plotamos com o auxílio do Matlab o diagrama de lugar das raízes para a planta, que pode ser observado na figura 6 encontramos o ponto limite de estabilidade. Também determinamos este ponto utilizando o critério de Routh, obtendo a equação $k_{osc} = \frac{\tau_3 + \tau_2}{k_1 * k_2 * k_3 * k_4 * \tau_3 * \tau_2}$ que nos fornece o mesmo valor para k_{osc} de 15.66.

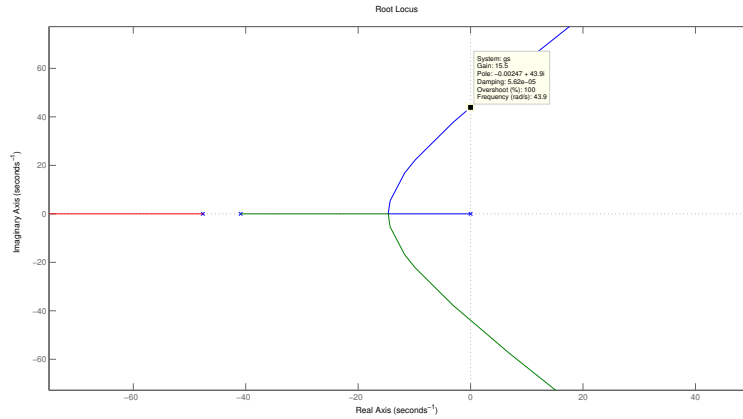


Figura 6: Diagrama de lugar das raízes para a $G(s)$ mostrando o limite de estabilidade

Com esse valor, utilizamos a tabela disponível no roteiro [1] para calcular os parâmetros $T_i = 0.0712$ e $T_d = 0.0178$. Pelo método do lugar das raízes, tentamos ajustar o valor de k_p , porém, como podemos ver na figura 7 existe uma assíntota que limita o aumento de $\xi\omega_n$ por mais que o ganho k aumente, levando o sistema a ter menor amortecimento (maior sobrelevação) e mesmo tempo de estabilização. Levando em conta essas limitações, escolhemos $k_p = 14$.

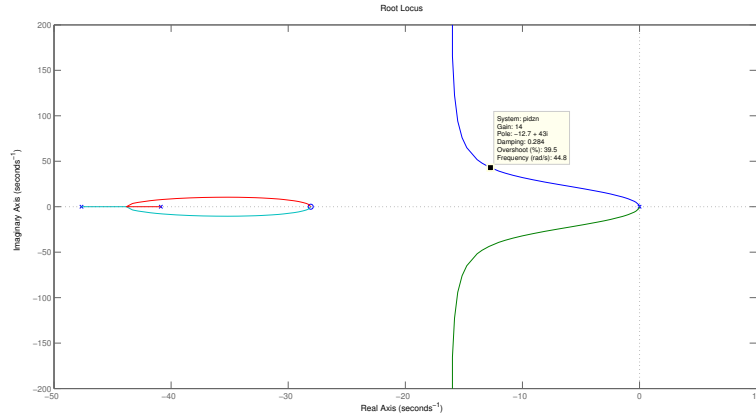


Figura 7: Diagrama de lugar das raízes para sistema com controlador calculado pelo método de Ziegler-Nichols

Esse sistema em malha fechada apresenta $\xi = 0.284$ e $\omega_n = 44.8$. Com esses valores e as equações 2, 3 e 4 calculamos: Tempo de estabilização:

$$t_e = 0.3075s \quad (13)$$

Sobrelevação:

$$\psi(\xi) = 39.43\% \quad (14)$$

Erro estático da resposta ao degrau:

$$\epsilon = 0 \quad (15)$$

Erro estático da resposta à rampa:

$$\epsilon = 0 \quad (16)$$

Discretizamos esse controlador por tusting utilizando a função "c2d" do Matlab, com frequência de amostragem de 0.001s. Obtendo a função de transferência discreta:

$$[H]C(z) = \frac{518.8z^2 - 997.1z + 484.4}{z^2 - 1} \quad (17)$$

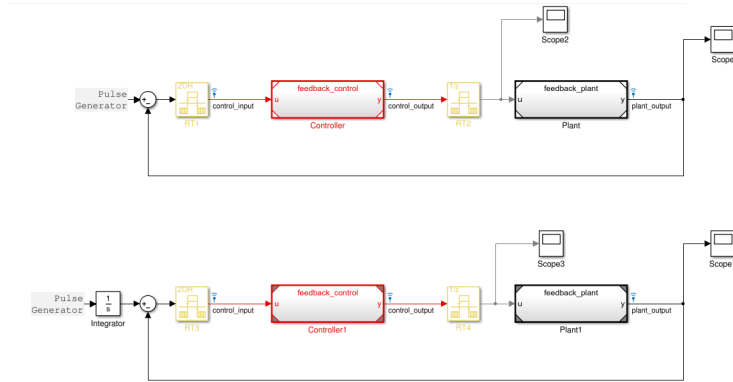


Figura 8: Esquema do sistema no simulink

Simulamos o sistema com essa função de transferência no Simulink, como pode ser visto na figura 8 e obtivemos como resposta à uma onda quadrada de amplitude $1V$ e frequência $4Hz$ o resultado que pode ser visto nas figuras 9 e 10 e a resposta à integral dessa onda quadrada (logo uma rampa) pode ser vista nas figuras 11 e 12.

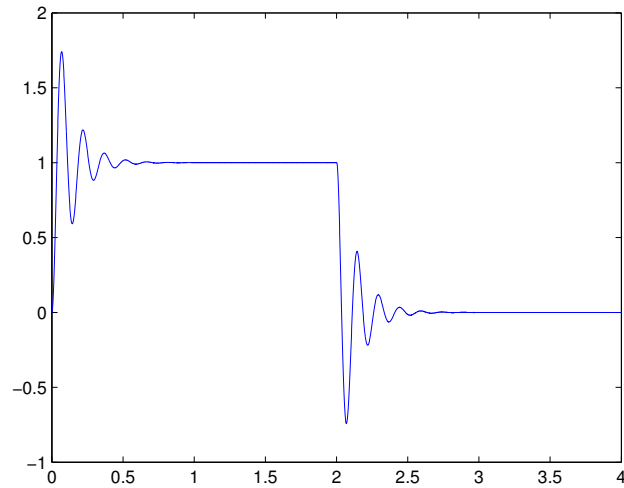


Figura 9: Resposta ($y(t)$) à onda quadrada do sistema com controlador calculado pelo método Ziegler-Nichols discretizado

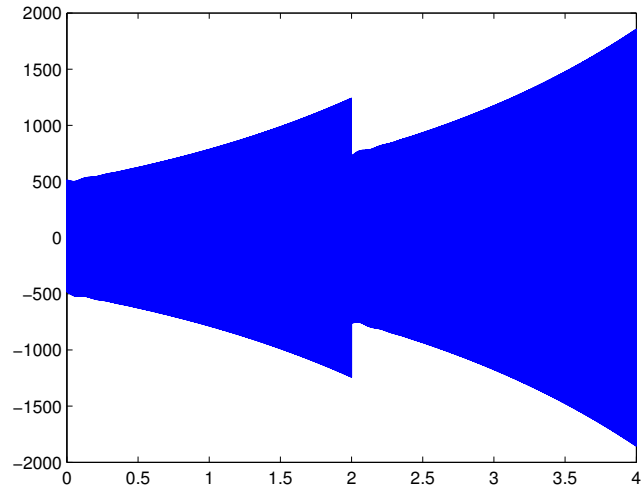


Figura 10: Esforço de controle ($y(t)$) em resposta a uma onda quadrada do sistema com controlador calculado pelo método Ziegler-Nichols discretizado

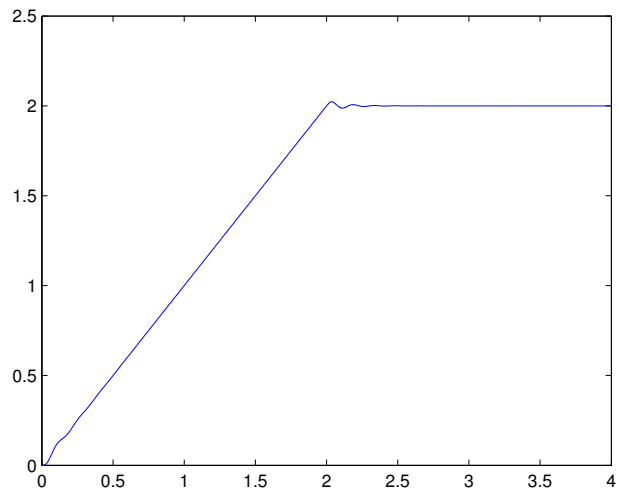


Figura 11: Resposta ($y(t)$) à uma rampa do sistema com controlador calculado pelo método Ziegler-Nichols discretizado

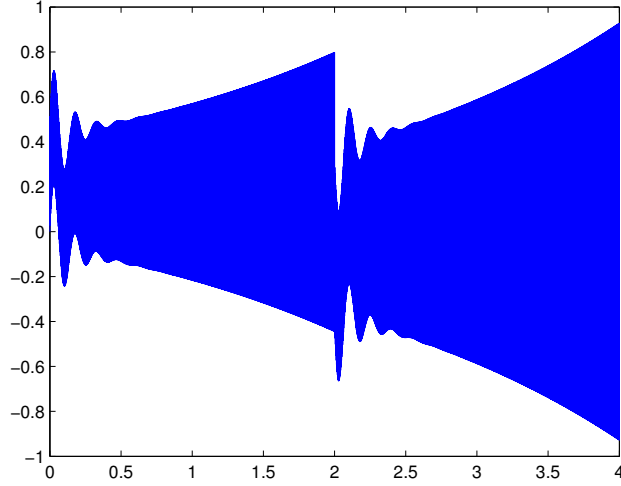


Figura 12: Esforço de controle ($y(t)$) em resposta à uma rampa do sistema com controlador calculado pelo método Ziegler-Nichols discretizado

Como podemos ver, o esforço de controle desse sistema quando discretizado oscila com um período extremamente elevado tornando a sua implementação praticamente impossível. Para essas resposta calculamos, com o auxílio da função "stepinfo" do Matlab, os mesmos valores acima e chegamos no resultado: Tempo de estabilização:

$$t_e = 0.24471s \quad (18)$$

Sobrelevação:

$$\psi(\xi) = 74.19\% \quad (19)$$

Erro estático da resposta ao degrau:

$$\epsilon = 0 \quad (20)$$

Erro estático da resposta à rampa:

$$\epsilon = 0.002 \quad (21)$$

Notamos que os valores calculados teoricamente são próximos dos medidos durante a simulação, exceto a sobrelevação, essas discrepâncias vêm muito provavelmente da discretização do controlador.

4 Projeto do controlador PID - Sisotool

Projetamos o controlador PID para a planta com o auxílio da ferramenta Sisotool do Matlab, e configuramos o método de design para "PID Tuning" e o tipo de controlador para PID, encontrando o controlador $11.716 * \frac{0.0324s^2 + 0.654s + 1}{0.00017s^2 + s}$. Notamos que um dos polos do controlador foi colocado muito longe da origem conforme apresentado no diagrama de lugar das raízes 13, e seu efeito no sistema é baixo.

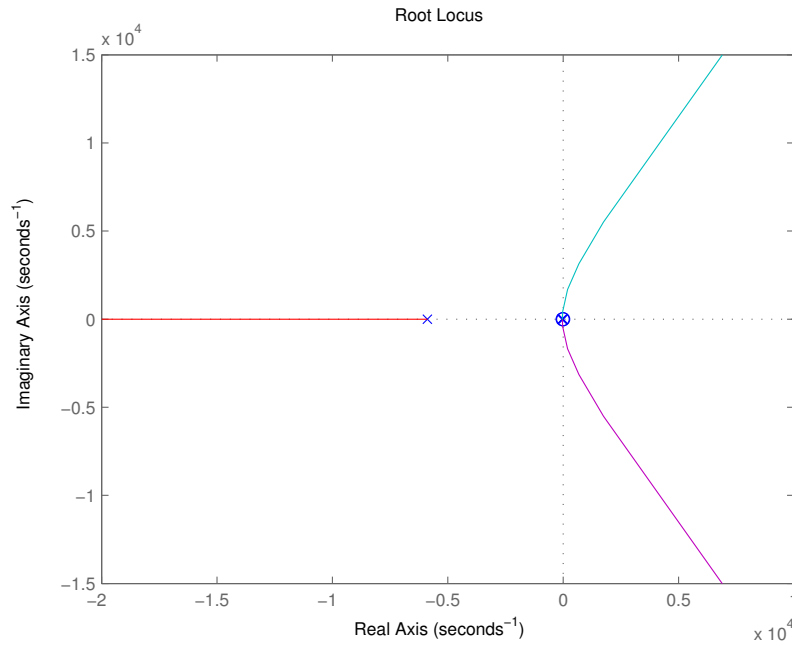


Figura 13: Diagrama de lugar das raízes para sistema com controlador PID projetado com o auxílio do SISOTool

Para que o tempo de estabilização deste sistema se aproximasse do tempo de estabilização do sistema com o controlador proporcional calculado anteriormente, seu ganho foi alterado para 15.53, levando seu tempo de estabilização para 0.294 segundos. Entretanto, outras características como a sobrelevação e tempo de subida não puderam ser reproduzidas, conforme mostrado em 14, e o sistema não apresenta erro à entrada rampa, como podemos ver na 15.

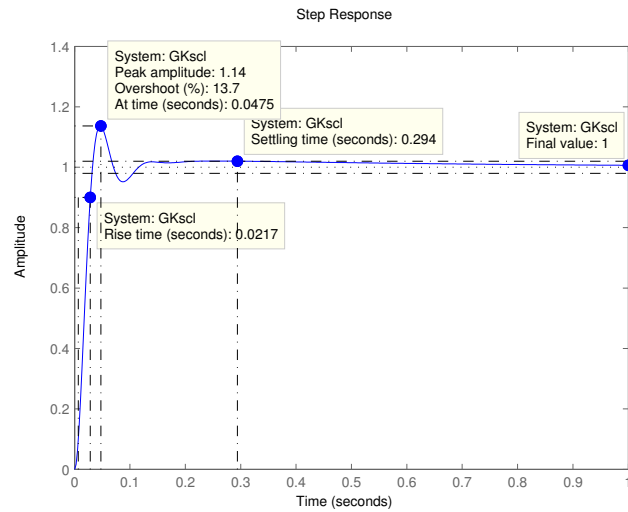


Figura 14: Resposta ao degrau para o sistema com controlador PID projetado com o auxílio do SISOTool

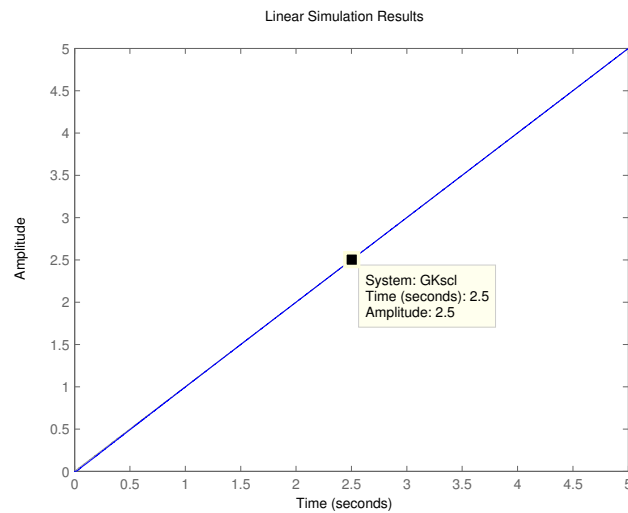


Figura 15: Resposta à rampa para o sistema com controlador PID projetado com o auxílio do SISOTool

Para que o sistema tenha tempo de estabilização menor que o sistema com controlador Ziegler Nichols, seu ganho é aumentado para 20, seu tempo de

estabilização passa a ser 0.124 segundos e sua sobrelevação 20%, como podemos ver na figura 16.

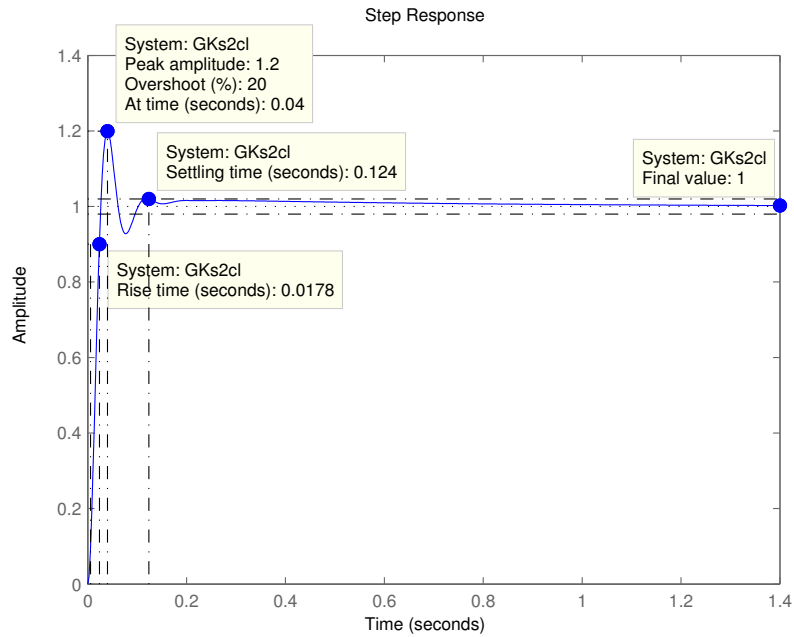


Figura 16: Resposta ao degrau para o sistema com controlador PID projetado com o auxílio do SISOTool

Utilizando o mesmo método de simulação descrito no projeto do PID Ziegler-Nichols obtvemos os resultados mostrados nas figuras 17 e 18 e a resposta à integral dessa onda quadrada (logo uma rampa) pode ser vista nas figuras 19 e 20.

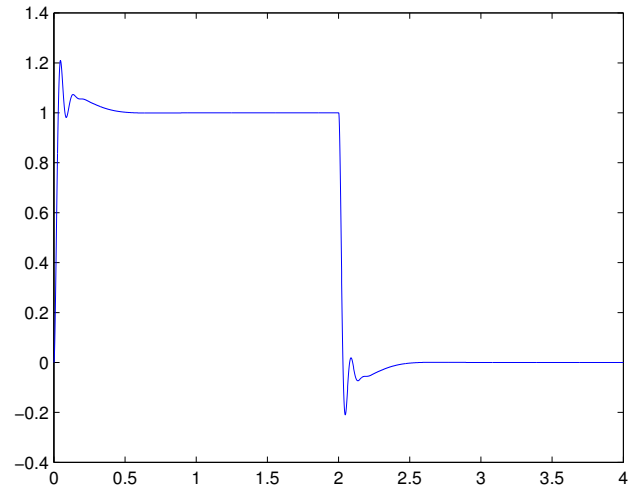


Figura 17: Resposta ($y(t)$) à onda quadrada do sistema com controlador projetado com o auxílio do SISOTool discretizado

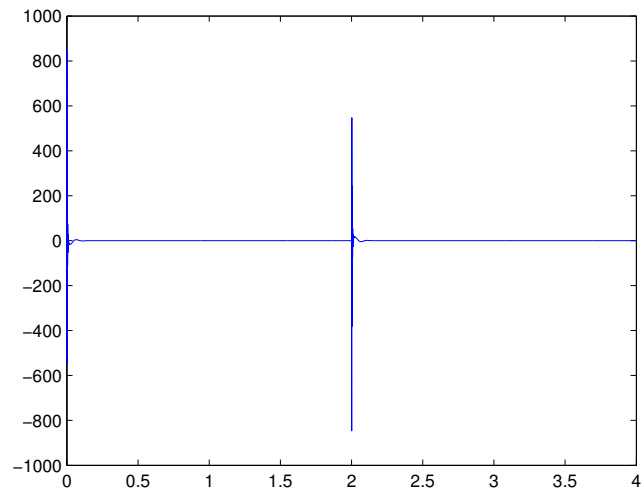


Figura 18: Esforço de controle ($y(t)$) em resposta a uma onda quadrada do sistema com controlador projetado com o auxílio do SISOTool discretizado

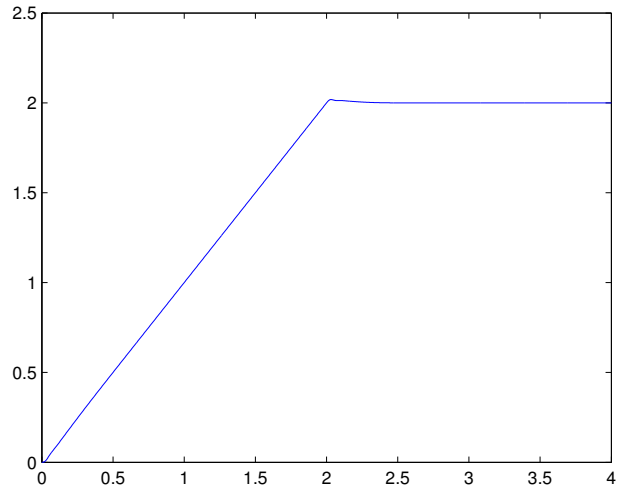


Figura 19: Resposta ($y(t)$) à uma rampa do sistema com controlador projetado com o auxílio do SISOTool discretizado

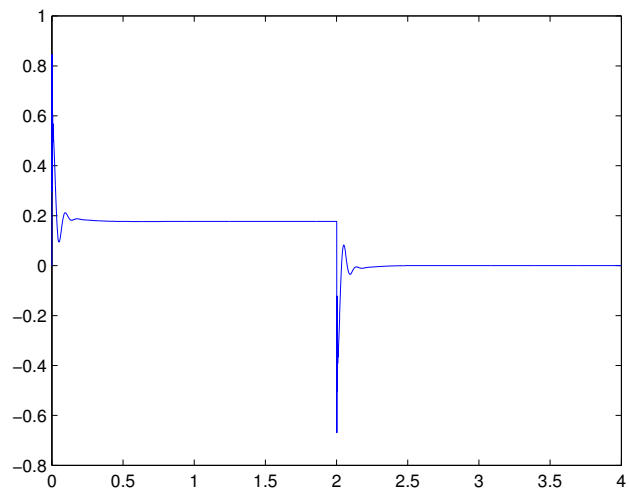


Figura 20: Esforço de controle ($y(t)$) em resposta à uma rampa do sistema com controlador projetado com o auxílio do SISOTool discretizado

Como podemos ver, o esforço de controle desse sistema na resposta à onda quadrada ultrapassa nosso limiar de 10V, logo acrescentamos um threshold de

saturação na simulação, obtendo os resultados das figuras 21 e 22

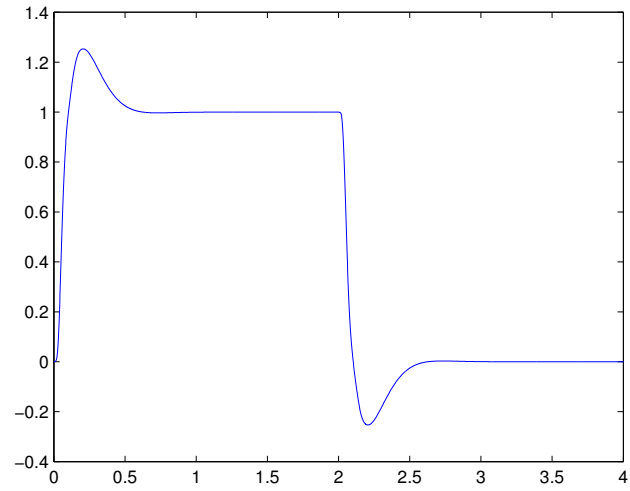


Figura 21: Resposta ($y(t)$) à onda quadrada do sistema com controlador projetado com o auxílio do SISOTool discretizado e com saturação

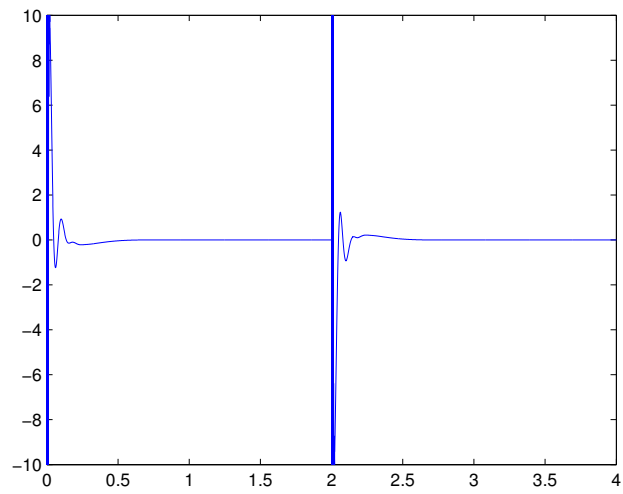


Figura 22: Esforço de controle ($y(t)$) em resposta a uma onda quadrada do sistema com controlador projetado com o auxílio do SISOTool discretizado e com saturação

Para essas resposta calculamos, com o auxílio da função "stepinfo" do Matlab, os mesmos valores obtidos nos gráficos acima e chegamos no resultado: Tempo de estabilização (sem saturação):

$$t_e = 0.23306s \quad (22)$$

Tempo de estabilização (com saturação):

$$t_e = 0.25010s \quad (23)$$

Sobrelevação (sem saturação):

$$\psi(\xi) = 20.97\% \quad (24)$$

Sobrelevação (com saturação):

$$\psi(\xi) = 25.33\% \quad (25)$$

Erro estático da resposta ao degrau (sem saturação):

$$\epsilon = 0 \quad (26)$$

Erro estático da resposta ao degrau (com saturação):

$$\epsilon = 0 \quad (27)$$

Erro estático da resposta à rampa:

$$\epsilon = 0.002 \quad (28)$$

Notamos que os valores calculados para o sistema contínuo são próximos dos medidos durante a simulação, com uma pequena diferença que provavelmente provém da discretização do controlador. Podemos ver que esse controlador tem uma curva de esforço de controle muito mais plausível que o controlador obtido pelo método de Ziegler-Nichols.

5 Referências

- [1] Roteiro do experimento disponibilizado para os alunos