



Instituto Superior Técnico

Universidade de Lisboa

Mestrado Integrado em Engenharia Aeroespacial

Controlo por Computador

*Identificação e controlo por computador
de um braço robótico flexível*
Parte 2: Desenho e teste de um controlador

Autores:

86635, Graça São Pedro

86661, Maria Carvalho

86668, Miguel Morgado

Docentes:

Prof. João Pedro Santos Gomes

Prof. João Miranda Lemos

Dezembro de 2019

Conteúdo

1	Introdução	1
2	<i>Design</i> do controlador e observador	2
2.1	<i>Design</i> do controlador LQ	3
2.1.1	Pólos do controlador LQ	4
2.2	<i>Design</i> do observador	5
2.2.1	Observador preditivo	5
2.2.2	Observador corrente	6
2.2.3	Filtro de Kalman	6
2.2.4	Pólos do observador	7
2.3	<i>Design</i> do controlador LQG	8
3	Implementação do controlador LQG com o modelo matemático	9
3.1	Implementação do controlador LQ	9
3.1.1	<i>Decay rate</i> do estado do sistema controlado	11
3.2	Implementação do observador	12
3.2.1	<i>Decay rate</i> da estimativa do estado do sistema	14
3.3	Implementação do controlador LQG - sistema completo	16
4	Validação do controlador LQG em simulação	20
4.1	Aproximação do sistema simulado ao sistema real	20
4.2	Implementação do integrador	21
4.3	Efeito da variação dos parâmetros do controlador LQG nas simulações	22
4.4	Validação do controlador	23
5	Implementação do controlador LQG com o sistema real	25
5.1	Efeito da variação do parâmetro R	26
5.2	Resposta à variação da frequência de entrada	27
5.3	Resposta a entradas com diferentes formas de onda	28

5.4	Resposta à variação da amplitude de entrada	31
5.5	Rejeição de perturbações	32
6	Conclusão	33
	Bibliografia	33
A	Código MATLAB utilizado	35

Lista de Figuras

1.1	Diagrama de blocos representativo do sistema que se pretende obter. Ref - sinal de referência; e - erro; u - entrada do sistema; y - saída do sistema; θ - posição angular do eixo do motor em relação a uma posição de referência; α - desvio angular da ponta da barra em relação à direção do eixo do motor.	1
2.1	Diagrama de blocos correspondente ao sistema controlado com recurso ao regulador quadrático linear que constitui o controlador ótimo.	2
2.2	Diagrama de blocos correspondente ao sistema com estimação de estado com recurso a um filtro de Kalman, que constitui o observador.	2
2.3	Diagrama de blocos correspondente ao acoplamento do controlador e do observador com o sistema em estudo.	3
2.4	Variação da potência do sinal de entrada $u(k)$ e de saída $y(k)$ em função de R	4
2.5	Diagrama de blocos do observador de estado corrente.	6
2.6	Diagrama de blocos correspondente ao controlador linear quadrático gaussiano com seguimento de referência.	8
3.1	Diagrama de blocos que implementa o controlador LQ.	9
3.2	Evolução do sinal de saída $y(k)$ do controlador impondo um degrau e amplitude 1° na entrada.	9
3.3	Mapa de pólos e zeros da malha fechada para $R = 100$	10
3.4	Mapa de pólos e zeros da malha fechada para $R = 200$	10
3.5	<i>Root-locus</i> simétrico da malha fechada para $R = 100$	11
3.6	<i>Root-locus</i> simétrico da malha fechada para $R = 200$	11
3.7	Diagrama de Bode para $R = 100$	11
3.8	Diagrama de Bode para $R = 200$	11
3.9	Evolução logarítmica de $ x(k) $ em função do tempo para avaliar o <i>decay rate</i>	12
3.10	Diagrama de blocos correspondente ao sistema com observador.	13
3.11	Diagrama de blocos que implementa o observador em <i>Simulink</i>	13
3.12	Comparação dos valores da estimativa do estado para diferentes valores de Q_E	13
3.13	Comparação dos valores da estimativa do estado para diferentes valores de R_E	14

3.14	Localização dos pólos do observador.	14
3.15	Evolução logarítmica de $ x(t) - x_{est}(t) $ em função do tempo para avaliar o <i>decay rate</i>	15
3.16	Diagrama de blocos do controlador LQG implementado em Simulink, ainda sem nenhum seguimento de referência.	16
3.17	Seguimento de uma referência nula utilizando o sistema com implementação de LQG da figura 3.16.	16
3.18	Visualização dos efeitos da imposição de ganho estático unitário.	17
3.19	Diagrama de Bode do sistema com LQG para $R = 100$	18
3.20	Diagrama de Bode do sistema com LQG para $R = 200$	18
3.21	Localização dos pólos do controlador e do observador.	18
3.22	Mapa de pólos e zeros do sistema total.	18
4.1	Diagrama de blocos do sistema final - ganho estático unitário, modelação da <i>dead zone</i> e inclusão de integrador.	20
4.2	Visualização do efeito da modelação da zona morta.	21
4.3	Visualização do efeito da inclusão do integrador.	21
4.4	Implementação condicional do integrador.	22
4.5	Efeito da variação de R na resposta do sistema.	22
4.6	Efeito da variação de Q_E na resposta do sistema.	23
4.7	Efeito da variação de R_E na resposta do sistema.	23
4.8	Efeito da variação da constante k do integrador na resposta do sistema.	23
4.9	Efeito da variação da amplitude na resposta do modelo do sistema.	24
4.10	Resposta do modelo do sistema a uma entrada sinusoidal.	24
4.11	Resposta do modelo do sistema a uma onda em dente de serra.	24
4.12	Resposta do modelo do sistema a um sinal do tipo PRBS.	24
5.1	Diagrama de blocos em <i>Simulink</i> que implementa o controlador LQG com o sistema físico.	25
5.2	Diagrama de blocos em <i>Simulink</i> que implementa o sistema físico.	26
5.3	Evolução da posição angular da ponta da barra em função do tempo para $R = \{150, 200, 250\}$ e uma onda de referência quadrada de amplitude 40° e frequência 0.4 Hz.	26
5.4	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 40° e frequência 0.4 Hz.	27
5.5	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 40° e frequência 0.5 Hz.	27
5.6	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 40° e frequência 0.25 Hz.	28

5.7	Drama de blocos que implementa o gerador de sinal multinível.	28
5.8	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência sinusoidal de amplitude 40° e frequência 0.4 Hz.	29
5.9	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência em dente de serra de amplitude 40° e frequência 0.4 Hz.	29
5.10	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência multinível com 5 níveis e amplitudes máximas 80° e 120°	30
5.11	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma sequência binária pseudo-aleatória de amplitude 40° e largura de banda 0.08 Hz.	30
5.12	Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 20° , 40° e 60° e frequência 0.4 Hz.	31
5.13	Evolução da posição angular da ponta da barra em função do tempo para uma referência nula e introdução de perturbações.	32

1. Introdução

O presente relatório foi produzido no âmbito da unidade curricular de Controlo por Computador, na sequência das sessões laboratoriais respeitantes ao projeto *Identificação e Controlo por Computador de um braço robótico flexível*.

Este projeto encontra-se dividido em duas partes. Na primeira, correspondente ao primeiro relatório, realizou-se a identificação do modelo matemático para o sistema físico em estudo e sua validação. Agora, pretende-se desenhar e testar um controlador adequado ao sistema identificado que permita cumprir os objetivos do projeto, descritos no primeiro relatório e que se enunciam novamente de seguida.

O principal objetivo deste trabalho é identificar e desenhar um sistema para controlar a posição angular do braço robótico flexível. Ao fornecer uma posição de referência para a extremidade do braço, pretende-se que esta seja seguida o mais fielmente possível. Assim, estuda-se a interação entre um sistema físico - o braço - e a componente computacional - o algoritmo. Na figura 1.1, representa-se o diagrama do sistema que se pretende obter no final de toda a atividade.

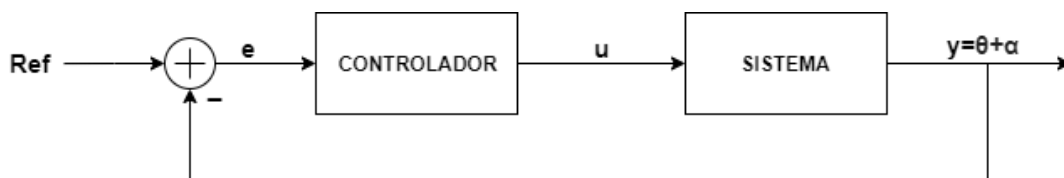


Figura 1.1: Diagrama de blocos representativo do sistema que se pretende obter. Ref - sinal de referência; e - erro; u - entrada do sistema; y - saída do sistema; θ - posição angular do eixo do motor em relação a uma posição de referência; α - desvio angular da ponta da barra em relação à direção do eixo do motor.

Deste modo, nesta fase do trabalho determina-se e implementa-se um controlador ótimo LQR (*linear-quadratic regulator*, ou regulador quadrático linear) que minimiza uma função de custo e permite realizar o controlo da posição angular da extremidade do braço robótico flexível. Uma vez que não é possível adquirir em cada instante o estado do sistema, implementa-se também um observador baseado em filtragem de Kalman, que permite estimar o estado da barra flexível. O acoplamento do controlador LQR e do observador ao sistema oferece, pois, os meios necessários para cumprir os objetivos de controlo do projeto.

Neste relatório, documentam-se todos os passos realizados na segunda parte do projeto, começando-se por uma explicação da teoria de controlo subjacente ao desenho do controlador LQG. De seguida, procede-se ao dimensionamento do regulador quadrático linear e do observador. Realizam-se depois simulações com o modelo matemático acoplado ao sistema de controlo e, de seguida, estuda-se a implementação deste no sistema físico.

Observa-se, deste modo, a influência dos vários parâmetros que caracterizam o controlador LQG na resposta simulada e real do sistema, bem como a forma como o dimensionamento da referência condiciona a capacidade de o sistema efetuar o seu seguimento.

2. Design do controlador e observador

Na primeira parte do projeto, tinha-se determinado um modelo matemático do sistema em espaço de estados da forma

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned} \quad (2.0.1)$$

em que $x(k)$ é o vetor estado do sistema, $y(k)$ a sua saída e $u(k)$ a sua entrada. As matrizes A , B , C e D determinadas são:

$$A = \begin{bmatrix} 3.2607 & -4.1348 & 2.4332 & -0.5591 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, C = \begin{bmatrix} -0.1554 \\ 0.2569 \\ 0 \\ 0 \end{bmatrix}, D = 0.$$

Este modelo em espaço de estados corresponde à função de transferência digital da equação 2.0.2.

$$G(z) = \frac{-0.1554z^3 + 0.2569z^2}{z^4 - 3.2607z^3 + 4.1348z^2 - 2.4332z + 0.5591} \quad (2.0.2)$$

Tal como referido na introdução deste relatório, de modo a cumprir os objetivos de controlo impostos, é necessário dimensionar e testar dois blocos: o controlador, representado por um bloco de ganho, e o observador, que estima o estado do sistema em cada instante. O diagrama de blocos que representa o sistema com o controlador na realimentação corresponde à figura 2.1, enquanto na figura 2.2 se ilustra o sistema realimentado com o observador que estima o seu estado atual. A figura 2.3 evidencia a forma como estes dois diagramas se ligam, permitindo a estimação do estado do sistema e o seu controlo.

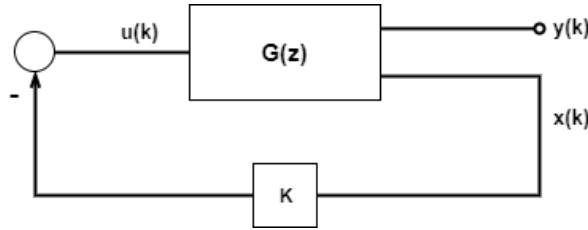


Figura 2.1: Diagrama de blocos correspondente ao sistema controlado com recurso ao regulador quadrático linear que constitui o controlador ótimo.

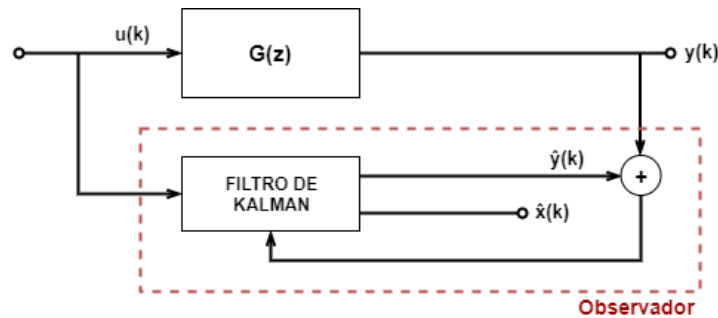


Figura 2.2: Diagrama de blocos correspondente ao sistema com estimação de estado com recurso a um filtro de Kalman, que constitui o observador.

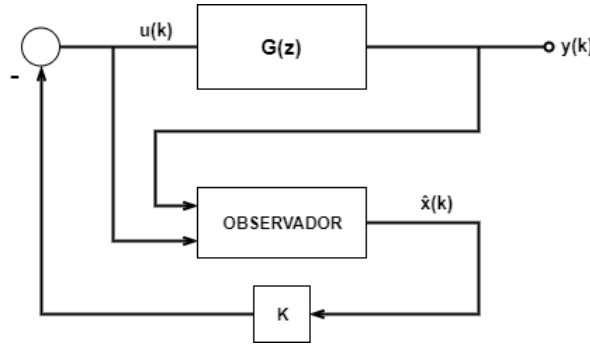


Figura 2.3: Diagrama de blocos correspondente ao acoplamento do controlador e do observador com o sistema em estudo.

2.1 Design do controlador LQ

A figura 2.1 mostra que o controlador se trata de um ganho - na verdade, é uma matriz de ganhos de dimensões adequadas - que usa o estado atual do sistema para realimentar a entrada do mesmo, não existindo qualquer *input* externo. Esta matriz de ganhos K é dimensionada utilizando técnicas de controle ótimo, nomeadamente o regulador quadrático linear (LQR - *Linear Quadratic Regulator*). O controle ótimo consiste em operar o sistema minimizando uma função de custo que, no caso em apreço, é definida pela equação 2.1.1.

$$J(K) = \sum_{k=0}^{\infty} [x^T(k)Qx(k) + u^T(k)Ru(k)] \quad (2.1.1)$$

Nesta equação, a matriz Q é semi-definida positiva ($Q \succcurlyeq 0$) e pondera o estado do sistema, $x(k)$, no cálculo do custo J , enquanto a matriz R é definida positiva ($R \succ 0$), determinando o peso das entradas $u(k)$ no cálculo do mesmo parâmetro.

O ganho K da realimentação do estado do sistema $u(k) = -Kx(k)$ é dado por

$$K = (B^T S B + R)^{-1} B^T S A \quad (2.1.2)$$

onde a matriz S é a única solução positiva definida da equação algébrica de Riccati - equação 2.1.3.

$$A^T S A - A^T S B (B^T S B + R)^{-1} B^T S A + Q = 0 \quad (2.1.3)$$

A função *dlqr* do *Matlab* permite resolver o problema do cálculo da matriz de ganhos K da realimentação de estado que constitui o controlador ótimo que minimiza a função de custo 2.1.1 em tempo discreto. A sintaxe desta função é

$$[K, S, E] = \text{dlqr}(A, B, Q, R),$$

onde K é matriz de ganhos da realimentação de estado, S é a matriz que resolve a equação algébrica de Riccati (equação 2.1.3) e E são os pólos do sistema em malha fechada, dados pelos valores próprios de $A - BK$. Os *inputs* da função *dlqr* são as matrizes A e B do modelo em espaço de estados discreto do sistema e as matrizes Q e R , que constituem os botões de ajuste (*tuning knobs*) no desenho do controlador LQR.

Para um sistema SISO (*single-input and single-output*) como o considerado neste projeto em que se pretende regular a saída do mesmo, atendendo ao modelo em espaço de estados (equação 2.0.1) e a que a matriz D é nula, a matriz Q é fixa e dada por

$$Q = C^T C. \quad (2.1.4)$$

Para além disso, a matriz R reduz-se a um escalar, bem como $y(k)$ e $u(k)$, pelo que a função de custo 2.1.1 pode ser simplificada, resultando na equação 2.1.5.

$$\begin{aligned}
 J(K) &= \sum_{k=0}^{\infty} [x^T(k)Qx(k) + u^T(k)Ru(k)] = \\
 &= \sum_{k=0}^{\infty} [x^T(k)C^T Cx(k) + u^T(k)Ru(k)] = \\
 &= \sum_{k=0}^{\infty} [y^2(k) + Ru^2(k)]
 \end{aligned} \tag{2.1.5}$$

Deste modo, a minimização da função de custo depende apenas do dimensionamento do valor de R , que determina o peso das parcelas $y^2(k)$ e $u^2(k)$ no cálculo de $J(K)$. A variação da potência do sinal de entrada $u(k)$ e do sinal de saída $y(k)$ em função de R está ilustrada no gráfico da figura 2.4.

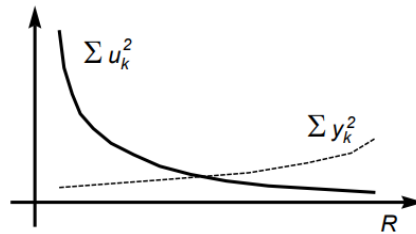


Figura 2.4: Variação da potência do sinal de entrada $u(k)$ e de saída $y(k)$ em função de R .

Idealmente, a função de custo minimizar-se-ia com valores simultaneamente pequenos de $\sum u^2(k)$ e de $\sum y^2(k)$. No entanto, o gráfico da figura 2.4 mostra que tal é impossível, uma vez que $\sum u^2(k)$ diminui com o aumento de R , ao passo que $\sum y^2(k)$ varia no mesmo sentido de R , sendo maior quanto maior for R .

Ora, a seleção de R acarreta algumas consequências que é necessário ter em conta no seu dimensionamento. Quanto menor for R , maior é a potência da entrada do sistema, $u(k)$, o que pode levar à saturação do controlador. Para além disso, quando R tende para zero a largura de banda do controlador torna-se mais ampla, o que resulta num controlo mais “nervoso”, já que os pólos tendem para posições assintóticas e mais próximas da instabilidade, e pode resultar na excitação de efeitos não considerados no modelo. Contudo, diminuir R resulta numa diminuição da sensibilidade da saída $y(k)$ a perturbações e o sistema apresenta uma resposta mais rápida.

A escolha de R passa, então, por um compromisso entre a rapidez de resposta do sistema e a sua estabilidade. Assim, realizam-se múltiplas simulações e testes do sistema físico de modo a determinar o valor de R que oferece a melhor resposta do sistema. Visto que R é sempre positivo, o sistema é sempre estável, sendo apenas necessário considerar o “nervosismo” e a rapidez de resposta do mesmo.

2.1.1 Pólos do controlador LQ

De modo a avaliar a estabilidade e o tipo de resposta do sistema controlado com recurso ao regulador linear quadrático, importa analisar a colocação dos seus pólos. Utiliza-se uma técnica denominada *root locus* simétrico.

Segundo o teorema de Chang-Letov, os valores próprios de $A - BK$ correspondem às raízes estáveis, ou seja, dentro do círculo unitário, da equação 2.1.6, onde $G(z) = \frac{b(z)}{a(z)}$ é a função de

transferência do sistema a controlar.

$$1 + \frac{1}{R}G(z)G(z^{-1}) = 0 \quad (2.1.6)$$

Esta equação pode ser reescrita, resultando em:

$$Ra(z)a(z^{-1}) + b(z)b(z^{-1}) = 0. \quad (2.1.7)$$

Quando R é muito pequeno, esta equação resume-se a

$$b(z)b(z^{-1}) = 0,$$

e, para R muito grande, reduz-se a

$$a(z)a(z^{-1}) = 0.$$

No primeiro caso, os pólos do sistema em malha fechada cancelam os zeros que se encontram dentro do círculo unitário ou são os seus simétricos ($p_i = \frac{1}{z_i}$), caso contrário. Na segunda situação, os pólos da malha fechada são os pólos da malha aberta quando estes são estáveis e, caso contrário, são os seus simétricos.

Em ambas as situações, a utilização dos ganhos do regulador linear quadrático garante que o sistema em malha fechada é sempre assintoticamente estável.

2.2 Design do observador

O controlador que se irá implementar no sistema faz uso do estado deste em cada instante para regular a sua entrada $u(k)$ de acordo com $u(k) = -Kx(k)$. Ora, não é possível adquirir em tempo real o estado do sistema, pelo que é necessário desenhar um observador cuja função é estimar $x(k)$ em cada instante de tempo tendo em conta as observações realizadas anteriormente. O diagrama de blocos que implementa o observador do sistema está representado na figura 2.2.

2.2.1 Observador preditivo

O observador preditivo consiste num estimador assintótico do estado que faz uso das observações do *input* e do *output* do sistema em cada instante de modo a calcular uma estimativa do estado no instante seguinte. Assim, a estimativa do estado no instante $k + 1$ é dada pela equação 2.2.1, onde L é um vetor de ganhos a determinar.

$$\hat{x}(k+1|k) = A\hat{x}(k|k-1) + Bu(k) + L[y(k) - C\hat{x}(k|k-1)] \quad (2.2.1)$$

Este observador apresenta uma estrutura semelhante à equação da dinâmica do sistema em espaço de estados, introduzindo apenas um termo de correção dado pela diferença entre a saída real no instante k ($y(k)$) e a saída estimada nesse instante ($C\hat{x}(k|k-1)$). Subtraindo a equação 2.2.1 à equação de estado 2.0.1, obtém-se o erro de estimação, \tilde{x} , dado pela equação 2.2.2.

$$\tilde{x} = x - \hat{x} : \quad \tilde{x}(k+1) = (A - LC)\tilde{x}(k) \quad \Rightarrow \quad \tilde{x}(k) = (A - LC)^k \tilde{x}(0) \quad (2.2.2)$$

O vetor de ganhos do observador, L , é determinado impondo os valores próprios de $A - LC$, que constituem os pólos do observador, tal que o polinómio característico A_o deste é dado por

$$A_o(z) = \det(zI - A + LC). \quad (2.2.3)$$

2.2.2 Observador corrente

O observador preditivo cujo funcionamento foi enunciado considera as estimativas do estado realizadas até ao instante $k - 1$ de modo a estimar $\hat{x}(k + 1)$, como evidenciado na equação 2.2.1. Este problema é abordado pelo observador corrente, que utiliza as estimativas mais recentes para estimar o estado do sistema no instante k , $\hat{x}(k)$. A expressão matemática que define o observador corrente corresponde à equação 2.2.4.

$$\hat{x}(k|k) = A\hat{x}(k-1|k-1) + Bu(k-1) + M[y(k) - C(A\hat{x}(k-1|k-1) + Bu(k-1))] \quad (2.2.4)$$

A equação 2.2.4 do observador corrente pode ser rescrita alternativamente sob a forma de uma predição e uma correção, dadas respetivamente pelas equações 2.2.5 e 2.2.6.

$$\text{Predição: } \bar{x}(k+1) = A\bar{x}(k) + Bu(k) \quad (2.2.5)$$

$$\text{Correção: } \hat{x}(k) = \bar{x}(k) + M[y(k) - C\bar{x}(k)] \quad (2.2.6)$$

O diagrama de blocos do estimador de estado que traduz as equações 2.2.1 e 2.2.6 está representado na figura 2.5, onde a parte a laranja corresponde ao observador preditivo e a parte a azul à correção efetuada para obter a estimativa do estado atual.

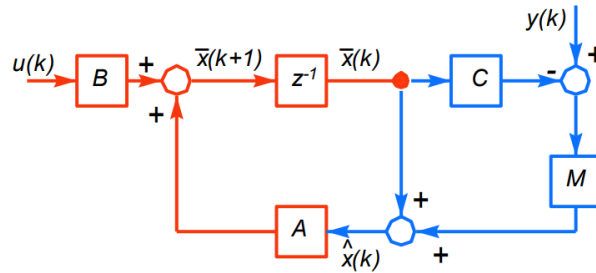


Figura 2.5: Diagrama de blocos do observador de estado corrente.

2.2.3 Filtro de Kalman

O observador implementado no projeto baseia-se no observador corrente descrito e é realizado com recurso a um filtro de Kalman que recebe como *input* a entrada do sistema $u(k)$ e a sua saída $y(k)$ e devolve uma estimativa do estado $\hat{x}(k)$. Para além disso, o filtro de Kalman assume que o sistema é afetado por perturbações, modeladas por ruído ($w(k)$ e $v(k)$), pelo que as equações da dinâmica do sistema em espaço de estados são dadas por

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Gw(k) \\ y(k) &= Cx(k) + Du(k) + v(k). \end{aligned} \quad (2.2.7)$$

O sinal $w(k)$ representa o ruído de processo que modela perturbações aleatórias e $v(k)$ corresponde ao ruído dos sensores, sendo ambos os sinais aleatórios, de média nula e independentes no tempo. Tratam-se, portanto, de ruídos brancos. As matrizes de covariância destes ruídos são dadas por:

$$Q_w = Q_E = E(w w^T) \quad R_v = R_E = E(v v^T).$$

As equações que descrevem o funcionamento do filtro de Kalman baseadas nas expressões 2.2.5 e 2.2.6 do observador corrente são dadas por 2.2.8 e 2.2.9.

$$\hat{x}(k|k-1) = A\hat{x}(k-1|k-1) + Bu(k-1) \quad (2.2.8)$$

$$\hat{x}(k|k) = \hat{x}(k|k-1) + M[y(k) - C\hat{x}(k|k-1)] \quad (2.2.9)$$

Por um lado, a equação 2.2.8 determina uma estimativa do estado no instante k conhecendo-se a estimativa ($\hat{x}(k-1|k-1)$) e a entrada ($u(k-1)$) anteriores. Por outro lado, a equação 2.2.9 corrige a predição efetuada com um termo proporcional à saída estimada por $C\hat{x}(k|k-1)$ e a saída real $y(k)$.

O ganho do filtro de Kalman, M , é selecionado de modo a otimizar a relação sinal-ruído da estimativa na presença do ruído que afeta o sistema. Ora, para um dado valor de M , o observador é caracterizado por uma certa largura de banda que permite a passagem não só das componentes da estimativa do estado dentro dessa banda, como de ruído. Manipulando progressivamente o valor de M tal que a largura de banda do observador aumenta, atinge-se um ponto em que deixam de passar os sinais de interesse da estimativa, admitindo que o estado x é um sinal passa-baixo. No entanto, o ruído de processo continua a passar, já que a sua densidade espectral de potência é independente da frequência. Assim, deverá existir um valor do ganho de Kalman que oferece a relação sinal-ruído ótima e que é calculado de forma a minimizar o custo dado pela equação 2.2.10.

$$J_o(M) = E \left[\sum_{k=0}^{\infty} \|x(k) - \hat{x}(k)\|^2 \right] \quad (2.2.10)$$

Atendendo à equação do observador corrente definida por 2.2.4 e ao modelo em espaço de estados com ruído das equações 2.2.7, o ganho de Kalman que minimiza o custo J_o é dado por

$$M(k) = APC^T \frac{1}{R_E + CPC^T}, \quad (2.2.11)$$

onde a matriz de covariância do erro de estimativa do estado, P , é solução da equação algébrica de Riccati:

$$P = APA^T + Q_E - \frac{APC^T CPA^T}{R_E + CPC^T}. \quad (2.2.12)$$

A função *dlqe* do *Matlab* permite calcular o valor M , sendo a sua sintaxe

$$[M, P, Z, EE] = dlqe(A, G, C, QE, RE).$$

Esta função recebe como *inputs* as matrizes A e C que descrevem o sistema em espaço de estados e a matriz G da equação 2.2.7, que pondera as perturbações $w(k)$, bem como as matrizes de covariância dos ruídos que afetam o modelo, Q_E e R_E . Tal como no caso do controlador, estas duas matrizes constituem os botões de ajuste do observador; pretende-se, pois, variar Q_w e R_w até se encontrar as matrizes que melhor regulam o observador tendo em conta os objetivos do projeto. A função *dlqe* devolve ainda como *output* a matriz de ganhos M que minimiza o custo dado pela equação 2.2.10, a matriz P que resolve a equação algébrica de Riccati, dada por 2.2.12, a matriz de covariância dos erros, Z , e os valores próprios de $A - MCA$ que constituem os pólos do observador.

2.2.4 Pólos do observador

Os pólos do observador que estima o estado do sistema podem ser calculados com recurso a um método semelhante ao descrito na secção 2.1.1 para o controlador LQ. Assim, os pólos do observador ótimo satisfazem

$$1 + \frac{Q_E}{R_E} G_o(z^{-1}) G_o(z) = 0, \quad (2.2.13)$$

onde $G_o(z)$ é dado por

$$G_o(z) = C(zI - A)^{-1}B. \quad (2.2.14)$$

2.3 Design do controlador LQG

Acoplando o controlador LQR e o estimador de estado dimensionados, obtém-se um controlador linear quadrático gaussiano (LQG). O diagrama de blocos que representa o acoplamento do observador e do controlador ao sistema está representado na figura 2.3. Este sistema é definido pela equação de estado 2.3.1 e a sua saída é dada por 2.3.2.

$$\hat{x}(k) = A\hat{x}(k-1) + Bu(k-1) + M[y(k) - C(A\hat{x}(k-1) + Bu(k-1))] \quad (2.3.1)$$

$$u(k) = -K\hat{x}(k) \quad (2.3.2)$$

Estas fórmulas foram definidas para o seguimento de uma referência nula. No entanto, pretende-se que o sistema real siga um sinal de referência na sua entrada, apresentando ganho unitário relativamente à mesma. Assim, a equação da entrada do sistema 2.3.2 é substituída por 2.3.3, onde r é o sinal de referência e N é um ganho a dimensionar de modo a que o ganho da saída relativamente à entrada seja unitário.

$$u(k) = Nr - K\hat{x}(k) \quad (2.3.3)$$

O diagrama de blocos que representa o sistema com controlador LQG e seguimento da referência está ilustrado na figura 2.6.

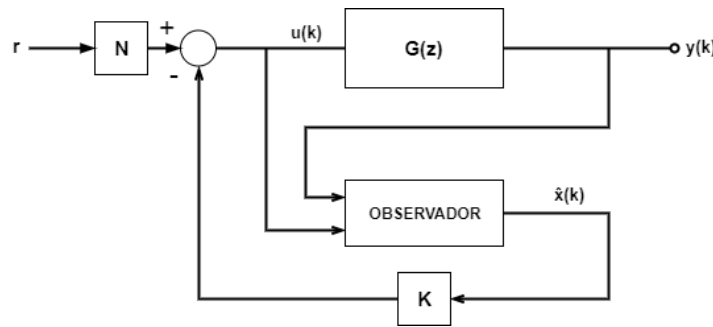


Figura 2.6: Diagrama de blocos correspondente ao controlador linear quadrático gaussiano com seguimento de referência.

De acordo com o livro *Digital Control of Dynamic Systems* de Franklin, Powell e Workman, 3ª edição, secção 8.4.1 [1], o ganho N pode ser determinado separando-se nas duas componentes N_x e N_u , que regulam respetivamente a equação de estado e a da saída. As equações que permitem determinar N são, pois, 2.3.4 e 2.3.5.

$$\begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} A - I & B \\ C & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (2.3.4)$$

$$N = N_u + KN_x \quad (2.3.5)$$

Outra forma equivalente de dimensionar N , é calcular o ganho em malha fechada da função de transferência do sistema com controlador LQG e invertê-lo, tal que

$$N = \frac{1}{\text{ganho da FT em malha fechada}} \quad (2.3.6)$$

O conjunto dos pólos em malha fechada do controlador LQG corresponde à união dos conjuntos de pólos do controlador LQ e do observador, ou seja:

$$\text{Pólos em malha fechada do LQG: } \text{eig}(A - BK) \cup \text{eig}(A - MCA), \quad (2.3.7)$$

onde $\text{eig}(A - BK)$ corresponde aos valores próprios do controlador LQ e $\text{eig}(A - MCA)$ aos valores próprios do observador.

3. Implementação do controlador LQG com o modelo matemático

3.1 Implementação do controlador LQ

Numa primeira fase, procedeu-se à construção do controlador e verificação do seu funcionamento. Para tal, implementou-se o esquema da figura 3.1 em *Simulink*. No bloco *Discrete State-Space* está caracterizado o modelo encontrado no processo de identificação levado a cabo na primeira parte deste projeto. O bloco de ganho na realimentação do sistema corresponde à matriz de ganhos do controlador de realimentação linear de variáveis de estado que se pretende incorporar no sistema. Como já vem da teoria previamente descrita, o botão de ajuste de que se faz uso de modo a dimensionar o controlador LQ é o fator R , pelo que se procede à análise da sua influência no sistema.

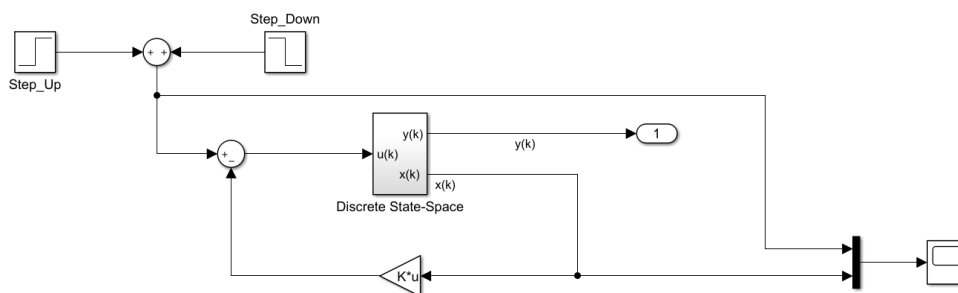


Figura 3.1: Diagrama de blocos que implementa o controlador LQ.

Para avaliar a influência da variação de R na resposta do sistema, obteve-se a figura 3.2 que apresenta a resposta da saída do sistema quando se impõe na entrada um degrau, no caso com 1° , para $R=\{50, 100, 200, 500\}$. Note-se que para nenhum destes valores a saída é aquela que se impôs à entrada, na medida em que não foi implementado ainda nenhum controlo de seguimento de referência que imporia um ganho estático unitário, pelo que não existe problema neste facto.

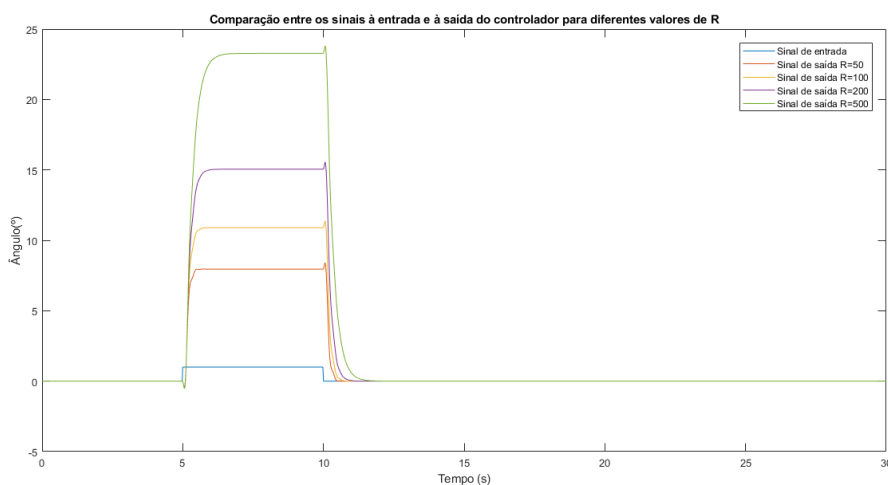


Figura 3.2: Evolução do sinal de saída $y(k)$ do controlador impondo um degrau e amplitude 1° na entrada.

Da representação da figura 3.2 pode-se inferir que, quando R aumenta, a estabilização do sistema é mais demorada e o valor para o qual este estabiliza é superior. Contudo, veja-se que para $R = 50$ a transição para a estabilidade não ocorre de forma tão suave, indicando que se está perante uma resposta mais “nervosa”. Como já foi abordado na teoria, há que existir um compromisso entre a rapidez da resposta e a suavidade das transições.

Como ainda não se está em condições de escolher o R adequado a utilizar para as simulações do sistema real, abordam-se os resultados de estabilidade para os valores de R intermédios $R = 100$ e $R = 200$. Obtiveram-se para estes valores de R as matrizes de ganho do controlador e respetivos pólos discriminadas de seguida. Verifica-se que o aumento do valor de R impõe um aumento no módulo das entradas da matriz de ganhos do controlador, K , e a aproximação dos pólos da circunferência unitária e consequente afastamento das posições assintóticas na origem.

$$K_{100} = [0,1692 \quad -0,3772 \quad 0,3058 \quad -0,0886]$$

$$K_{200} = [0,1229 \quad -0,2751 \quad 0,2245 \quad -0,0655]$$

$$\text{Pólos para } R = 100: \begin{bmatrix} 0,7374 + 0,4180i \\ 0,7374 - 0,4180i \\ 0,8084 + 0,0386i \\ 0,8084 - 0,0386i \end{bmatrix} \text{ e pólos para } R = 200: \begin{bmatrix} 0,7414 + 0,4158i \\ 0,7414 - 0,4158i \\ 0,8677 \\ 0,7873 \end{bmatrix}.$$

Com um menor R , estando os pólos mais próximos da origem do referencial, o sistema está menos protegido contra perturbações, ainda que o transitório seja mais rápido. Quando os pólos estão mais afastados da origem, ainda que mais lento, o sistema é mais robusto a perturbações. Note-se ainda que a existência de zeros de fase não-mínima para os vários R é facilmente verificada na figura 3.2 pela existência do efeito da chicotada, mostrando que o modelo escolhido foi adequado. A localização destes pólos e zeros pode ser observada nas figuras 3.3 e 3.4 e a evolução dos mesmos é estudada nos *root locus* simétricos das figuras 3.5 e 3.6.

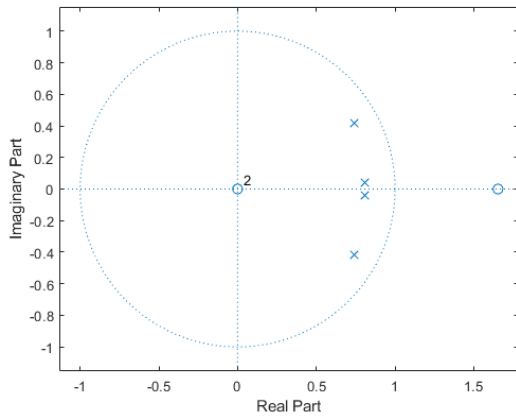


Figura 3.3: Mapa de pólos e zeros da malha fechada para $R = 100$.

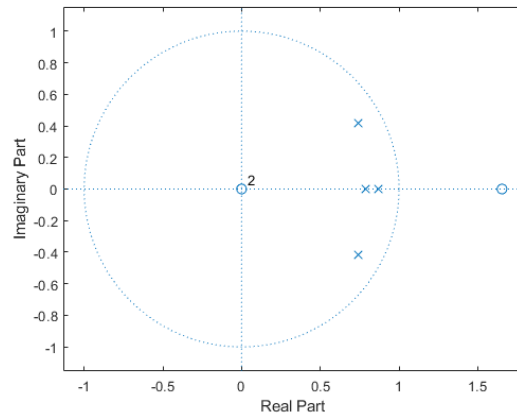


Figura 3.4: Mapa de pólos e zeros da malha fechada para $R = 200$.

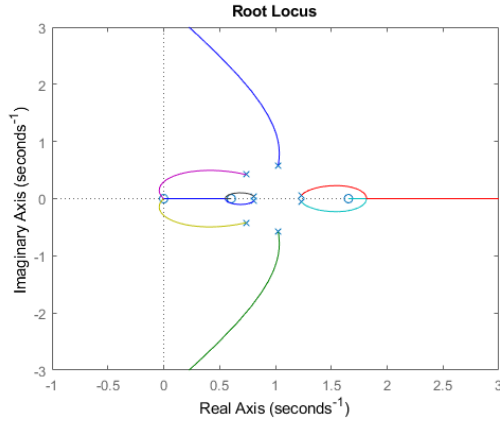
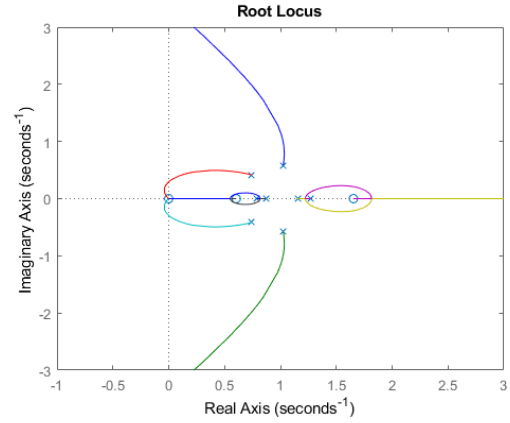
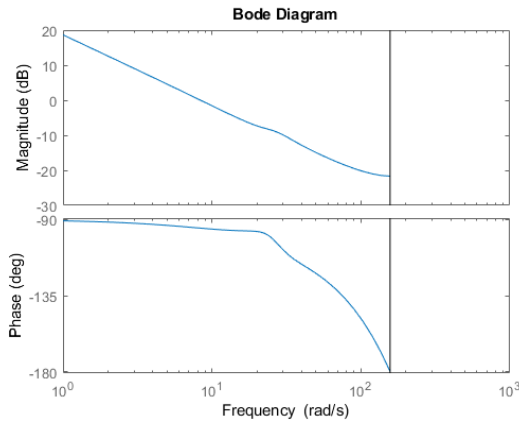
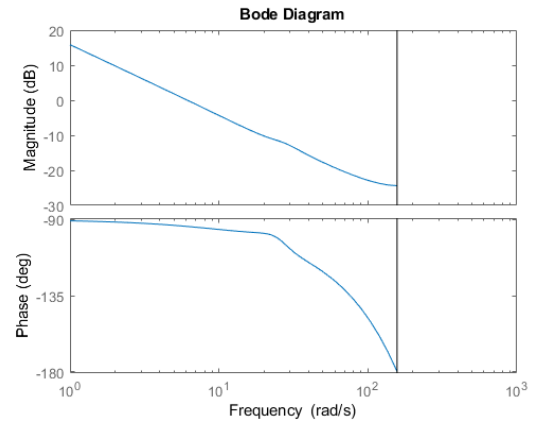

 Figura 3.5: *Root-locus* simétrico da malha fechada para $R = 100$.

 Figura 3.6: *Root-locus* simétrico da malha fechada para $R = 200$.

 Figura 3.7: Diagrama de Bode para $R = 100$.

 Figura 3.8: Diagrama de Bode para $R = 200$.

 Tabela 3.1: Margens de ganho e de fase para $R = 100$ e $R = 200$

	R=100	R=200
Margem de ganho (dB)	12.11	16.55
Margem de fase (°)	84.96	85.97

Daqui conclui-se mais uma vez o que tem vindo a ser abordado: o aumento do parâmetro R aumenta a estabilidade do sistema, uma vez que as margens de funcionamento também se afastam mais dos valores de estabilidade marginal.

3.1.1 *Decay rate* do estado do sistema controlado

Para aferir o correto funcionamento do controlador, obteve-se a figura 3.9, que representa a evolução logarítmica do estado do controlador na fase de decaimento após a entrada do sistema regressar a zero.

Seja λ_{max} o módulo do maior valor próprio da matriz $A - BK$. Sabe-se que o declive da reta que modela o decaimento do estado quando este regressa ao valor inicial é dado teoricamente por

$$m_{teo} = \log_{10}(\lambda_{max}) = -0.0718. \quad (3.1.1)$$

Para $R = 100$, e utilizando o *script* 3.1 desenvolvido em MATLAB verificou-se com os dados de simulação que o declive da reta que traduz o decaimento do estado é

$$m_{ctrl} = -0.0718.$$

Listing 3.1: *Script* desenvolvido em MATLAB para calcular o *decay rate* do estado do sistema controlado com o LQR.

```

1  load matrizes_ss.mat;
2  cria_ganhos;
3  out_cont = sim('conta', 'SaveState', 'on');
4
5  %controlador
6  x=out_cont.x;
7
8  figure('NumberTitle', 'off', 'Name', 'Estado x(t) em funcao do tempo t');
9  plot(x.time,x.signals.values);
10
11 figure('NumberTitle', 'off', 'Name', 'Log de |x(t)| em funcao do tempo t');
12 plot(x.time,log10(abs(x.signals.values)));
13
14 [pks1,locs1] = findpeaks(log10(abs(x.signals.values(:,4))));
15 sz_1=size(pks1);
16 m_ctrl = polyfit(locs1(14:sz_1(1)),pks1(14:sz_1(1)),1);
17
18 %logaritmo do modulo dos polos
19 eig_ctrl=log10(abs(E));
    
```

Tendo-se apontado um erro de cerca de 0%, comprova-se então que o controlador projetado tem um comportamento semelhante ao previsto pela teoria.

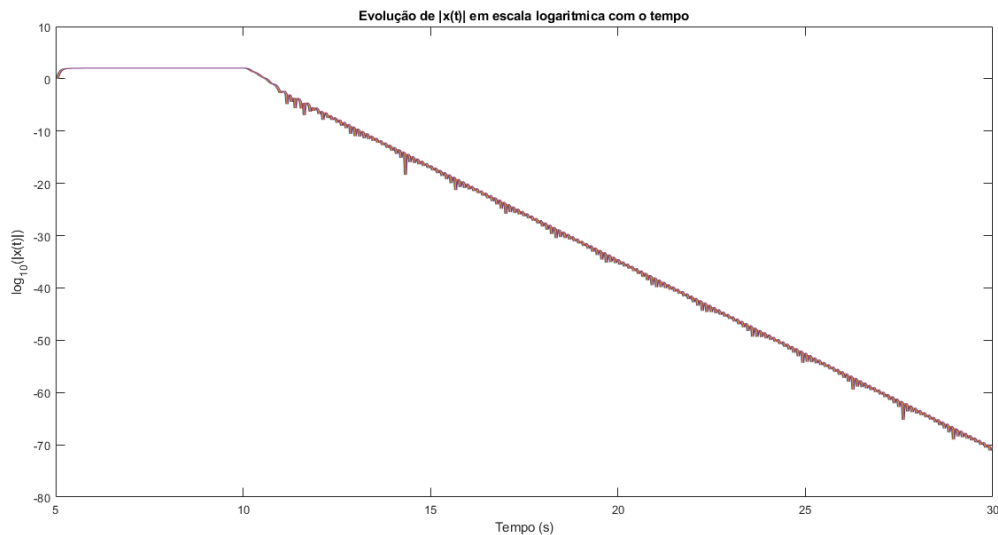


Figura 3.9: Evolução logarítmica de $|x(k)|$ em função do tempo para avaliar o *decay rate*.

3.2 Implementação do observador

De seguida, verificou-se o funcionamento do observador, o qual calcula uma estimativa do estado do sistema em cada instante. Note-se que o sinal na entrada do observador e do bloco *Discrete State-Space* são diferentes. Para tal, implementou-se o esquema da figura 3.10 em

Simulink. A figura 3.11 mostra o detalhe da implementação do observador, tal como foi descrito na introdução teórica do presente relatório.

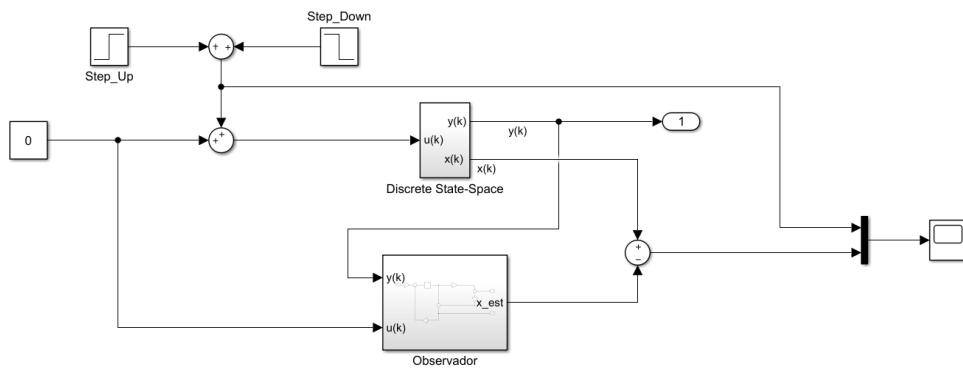


Figura 3.10: Diagrama de blocos correspondente ao sistema com observador.

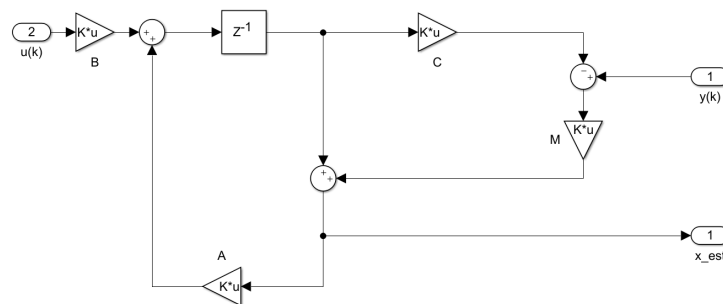


Figura 3.11: Diagrama de blocos que implementa o observador em *Simulink*.

Uma vez que existem dois graus de liberdade em vez de apenas um, como aconteceu para os botões de ajuste do controlador, será necessário analisar os seus efeitos separadamente. Começando por variar a matriz de covariância do ruído do processo, Q_E , fixando $R_E = 1$, demonstra-se na figura 3.12 que o impacto deste botão de ajuste é praticamente desprezável, uma vez que aumentar 50 vezes o seu valor representou uma diminuição de menos de 1° no erro da estimativa. Geralmente, um aumento de perturbações aleatórias, traduzido por um aumento de Q_E , induz uma diminuição nas estimativas de estado realizadas.

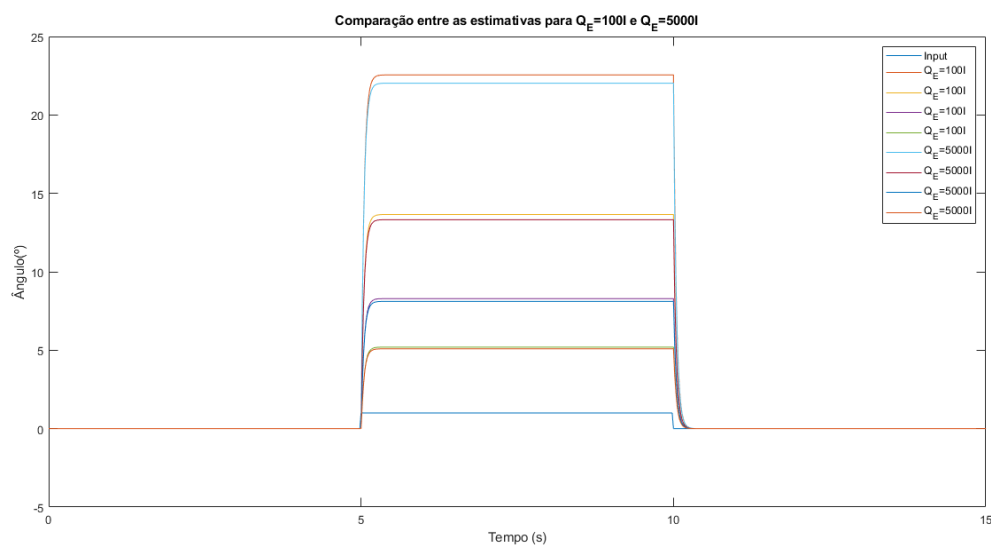


Figura 3.12: Comparação dos valores da estimativa do estado para diferentes valores de Q_E .

Contrariamente, um aumento dos coeficientes da matriz de covariância do ruído dos sensores, fazendo $Q_E = I$, induz um aumento do erro das estimativas bem como a suavização das transições. Veja-se na figura 3.13 que o transitório inicial para valores de R_E inferiores é menos suave do que aquele encontrado para $R_E = 400$.

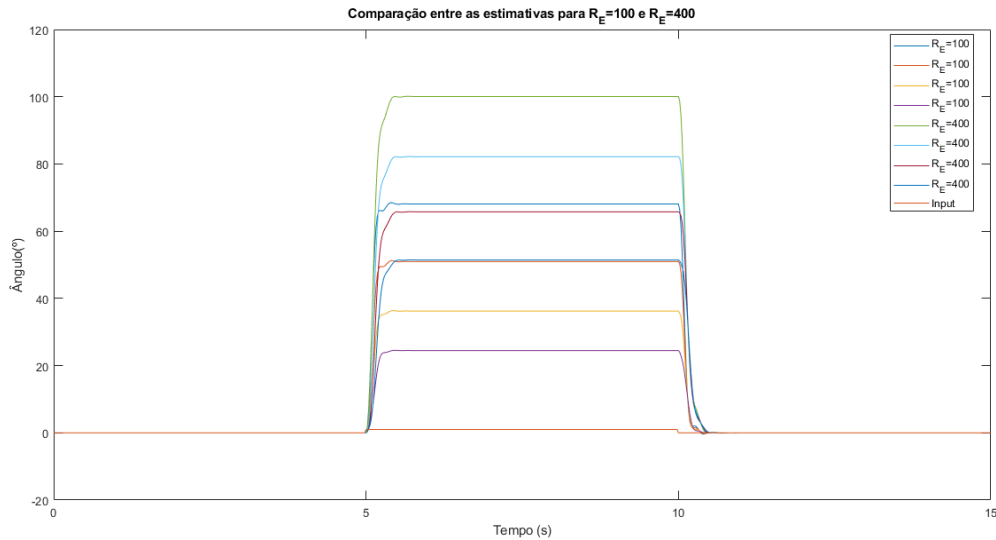


Figura 3.13: Comparação dos valores da estimativa do estado para diferentes valores de R_E .

Para $Q_E = 100I$ e $R_E = 1$, obteve-se a seguinte matriz de ganhos de Kalman e pólos do observador:

$$M = \begin{bmatrix} 42.4238 \\ 29.5072 \\ 17.7829 \\ 10.2467 \end{bmatrix}, \text{ Pólos: } \begin{bmatrix} 0,6244 \\ -0,0634 + i0,1959 \\ -0,0634 - i0,1959 \\ 0,2336 \end{bmatrix}.$$

Os pólos do observador estão representados na figura 3.14.

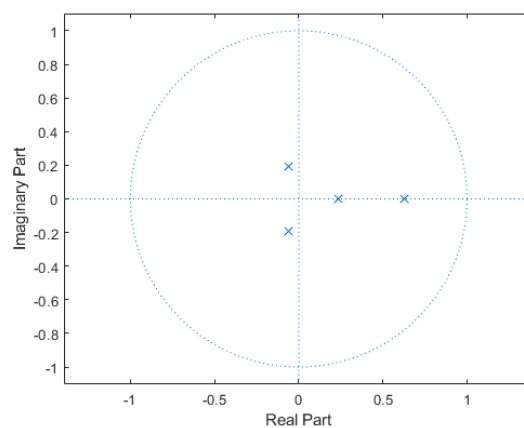


Figura 3.14: Localização dos pólos do observador.

3.2.1 Decay rate da estimativa do estado do sistema

Novamente, pelo mesmo processo que se usou para averiguar o funcionamento do controlador, pretende-se comprovar que o observador projetado é adequado. Para tal, obteve-se a figura 3.15, que representa a evolução logarítmica do estado do erro entre o valor real e a estimativa do estado.

Seja λ_{max} o módulo do maior valor próprio da matriz $A - MCA$. Sabe-se que o declive da reta que modela o decaimento do erro de estimação do estado quando este regressa ao valor inicial é dado teoricamente por

$$m_{teo} = \log_{10}(\lambda_{max}) = -0.2046. \quad (3.2.1)$$

Para $R = 100$ e utilizando o *script* 3.2, verificou-se com os dados de simulação, que se observam na figura 3.9 que

$$m_{obs} = -0.1979.$$

Listing 3.2: *Script* desenvolvido em MATLAB para calcular o *decay rate* do erro da estimativa do estado do sistema.

```

1 load matrizes_ss.mat;
2 cria_ganhos;
3 out_obs = sim('obs', 'SaveState', 'on');
4
5 %observador
6 x_til=out_obs.x_til;
7
8 figure('NumberTitle', 'off', 'Name', 'Estado x~(t) em funcao do tempo t');
9 plot(x_til.time,x_til.signals.values);
10
11 figure('NumberTitle', 'off', 'Name', 'Log de |x~(t)| em funcao do tempo t');
12 plot(x_til.time,log10(abs(x_til.signals.values)));
13
14 [pks2,locs2] = findpeaks(log10(abs(x_til.signals.values(:,4))));
15 sz_2=size(pks2);
16 m_obs = polyfit(locs2(57:58),pks2(57:58),1);
17
18 %logaritmo do modulo dos polos
19 eig_obs=log10(abs(EE));

```

Tem-se então um erro associado de 3.27%. É, portanto, um resultado muito coerente com a previsão teórica e comprova que o observador também é adequado para os objetivos do projeto.

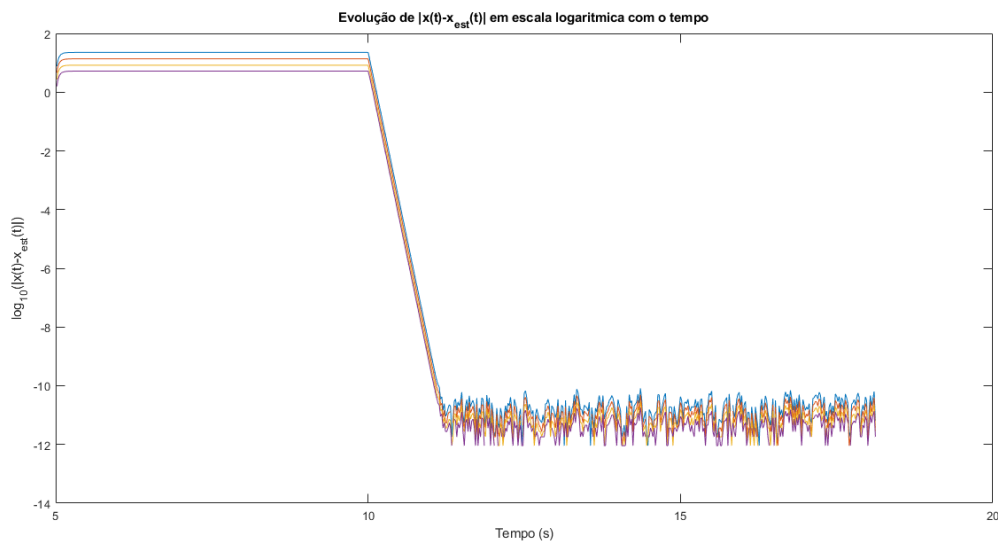


Figura 3.15: Evolução logarítmica de $|x(t) - x_{est}(t)|$ em função do tempo para avaliar o *decay rate*.

3.3 Implementação do controlador LQG - sistema completo

Finalmente, pode-se fundir o controlador desenhado anteriormente e representado na figura 3.1 com o observador, ilustrado na figura 3.11, de forma a que o estado do sistema estimado $\hat{x}(k)$ seja fornecido ao controlador. Com isto, obtém-se o sistema de controlo pretendido para o modelo previamente identificado, que se pode representar pela figura 3.16. É de notar que, nesta fase, o sistema ainda não tem implementado nenhum seguimento de referência.

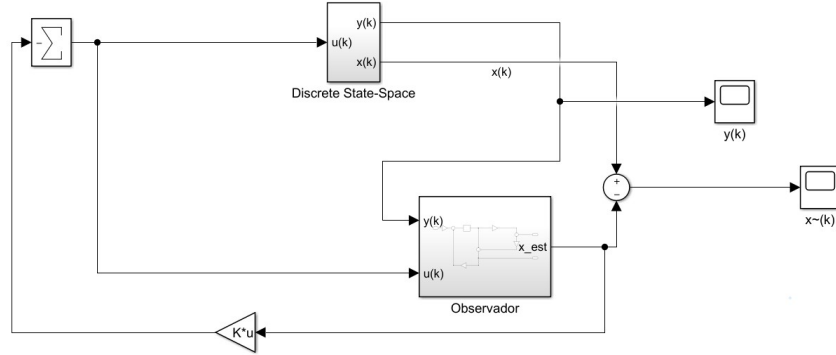


Figura 3.16: Diagrama de blocos do controlador LQG implementado em Simulink, ainda sem nenhum seguimento de referência.

Como seria de esperar, o sistema, sem nenhuma referência para seguir e com condições iniciais nulas, mantém a saída nula. Na figura 3.17 encontra-se o resultado de uma simulação de trinta segundos com $R = 200$, $R_E = 1$ e $Q_E = 100I$.

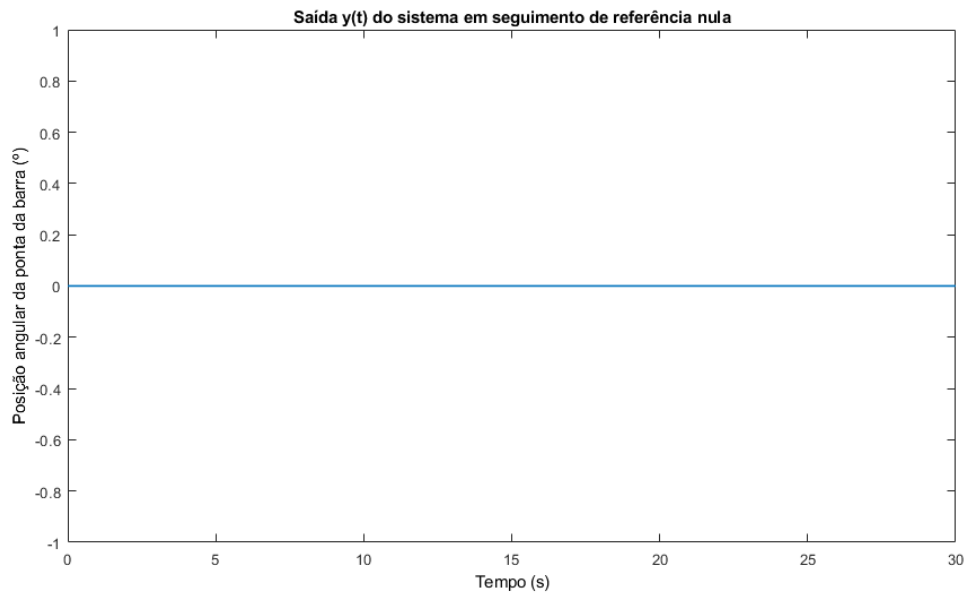


Figura 3.17: Seguimento de uma referência nula utilizando o sistema com implementação de LQG da figura 3.16.

Considere-se agora o caso em que a saída efetua o seguimento de uma referência não nula. Atendendo à equação 2.3.4, foi imposto à entrada do bloco *Discrete State Space* um ganho tal que o ganho estático em malha fechada fosse unitário. Para tal, usou-se o programa de MATLAB e importou-se do *workspace* as variáveis necessárias. O *script* seguinte permite calcular este ganho N de acordo com as equações enunciadas na teoria.

```

1 N = inv([A-eye(size(A)), B; C, 0])*[zeros(size(A,1),1);1];
2 Nx = N(1:end-1,:);
3 Nu = N(end,:);
4 Nbar = Nu+K*Nx;
    
```

Na figura 3.18, simulação feita para $R = 200$, $R_E = 1$ e $Q_E = 100I$, está demonstrado o efeito que a imposição do ganho unitário tem na resposta do sistema. Note-se que a resposta tem uma amplitude estável igual à amplitude da entrada, mas nem a rapidez da resposta nem a estabilidade da mesma foram alteradas, indicando que este ganho não perturba o sistema.

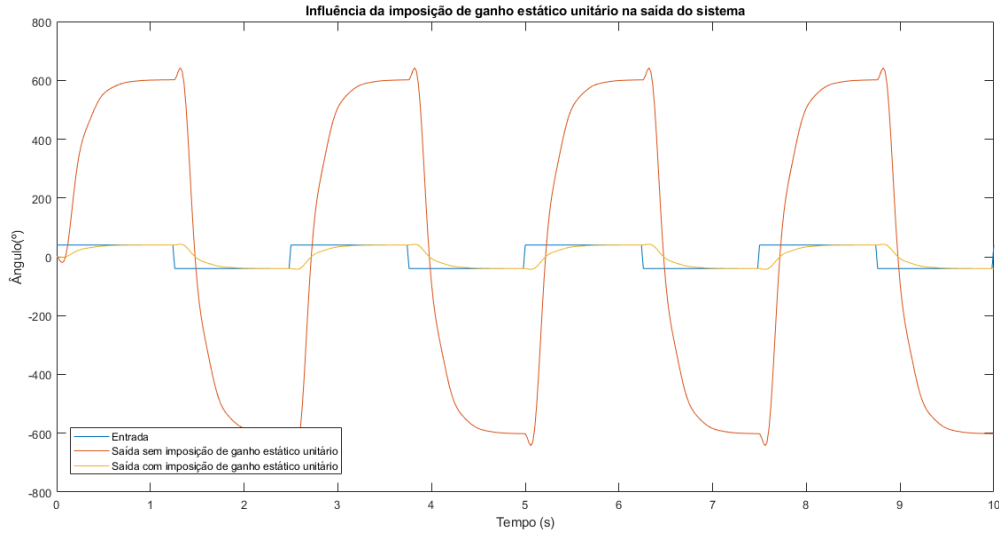


Figura 3.18: Visualização dos efeitos da imposição de ganho estático unitário.

O sistema completo com seguimento de referência rege-se pelas equações 2.3.1 e 2.3.3, não esquecendo que

$$y(k+1) = Cx(k+1) = C(Ax(k) + Bu(k)), \quad (3.3.1)$$

tal que, substituindo na equação 2.3.1, se obtêm as equações gerais para a solução de controle:

$$\begin{cases} x(k+1) = Ax(k) - BK\hat{x}(k) + BNr \\ \hat{x}(k+1) = MCAx(k) + (A - MCA - BK)\hat{x}(k) + BNr. \end{cases} \quad (3.3.2)$$

De forma a avaliar a influência do controlador LQG na estabilidade do sistema, apresentam-se os diagramas de Bode do mesmo para diferentes valores de R , nas figuras 3.19 e 3.20, assim como a tabela 3.2 onde se resumem os valores das margens de ganho e de fase, para $R = 100$ e $R = 200$, obtidos em *MATLAB*. É importante dimensionar o ganho de Kalman e o ganho de realimentação do controlador LQ de modo a que as margens de ganho e de fase sejam suficientemente amplas e a estabilidade do sistema não seja comprometida.

Tabela 3.2: Margens de ganho e de fase para $R = 100$ e $R = 200$ com controlador LQG.

	R=100	R=200
Margem de ganho (dB)	6.11	8.32
Margem de fase (°)	-180	-180

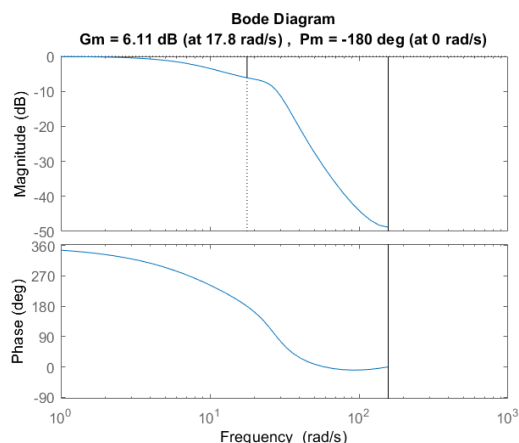


Figura 3.19: Diagrama de Bode do sistema com LQG para $R = 100$.

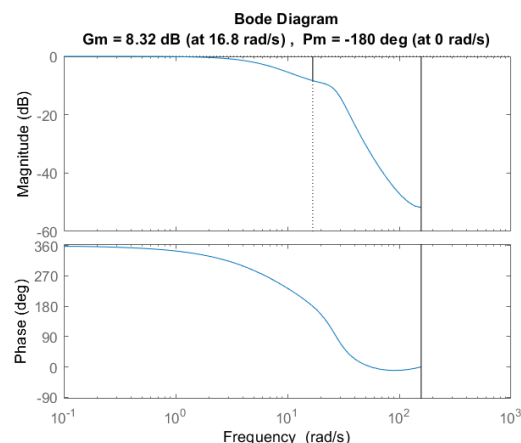


Figura 3.20: Diagrama de Bode do sistema com LQG para $R = 200$.

Obteve-se ainda a função de transferência do sistema com implementação do controlador LQG e seguimento de referência. Utilizando o comando *ss2tf* da linguagem *MATLAB* e introduzindo as respectivas matrizes do sistema em espaço de estados, calcularam-se os coeficientes da função de transferência.

$$G(z) = \frac{-0.01z^7 + 0.025z^6 - 0.013z^5 + 0.0015z^4 - 0.00037z^3 + 0.0001z^2 + 4 \times 10^{-19}z - 1.51 \times 10^{-19}}{z^8 - 3.87z^7 + 6.23z^6 - 5.30z^5 + 2.48z^4 - 0.62z^3 + 0.10z^2 - 0.02z + 0.00305}$$

Mais uma vez, é visível que o aumento do parâmetro R tem um efeito estabilizador no controle do sistema, uma vez que as margens de ganho e de fase crescem e afastam-se da instabilidade. É igualmente relevante a comparação com o controlador LQ , já que daí se podem retirar conclusões acerca da influência da estimação de $x(k)$ na estabilidade geral do sistema. Por comparação dos valores da tabela 3.1 com os da tabela 3.2, afirma-se que, de forma geral, a margem de ganho diminui para metade depois do acoplamento do observador, mas ainda assim o sistema é estável.

Finalmente, na figura 3.21 assinalam-se os pólos do sistema controlado com o LQG, que são efetivamente a reunião dos pólos do controlador LQ e do observador. Na figura 3.22, observa-se que os pólos do observador coincidem com parte dos zeros do sistema total. Além disso, comprova-se que no sistema final existe ainda um zero de fase não mínima responsável pelo efeito de “chicotada”. Desta figuras é possível concluir que o Princípio da Sobreposição se verifica e o trabalho realizado até aqui de projeção separada do controlador e do observador é viável. O controlador LQG está implementado corretamente.

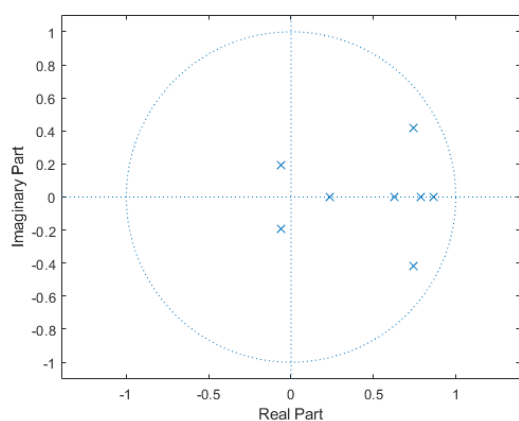


Figura 3.21: Localização dos pólos do controlador e do observador.

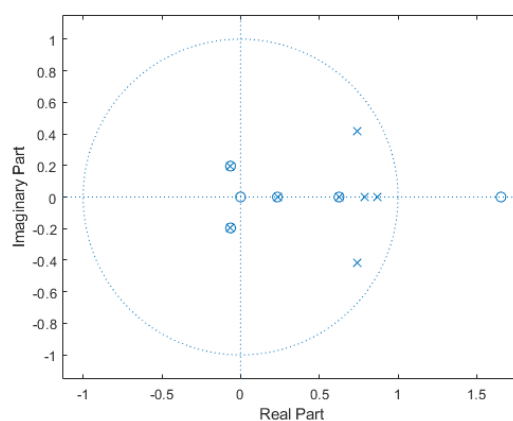


Figura 3.22: Mapa de pólos e zeros do sistema total.

Os pólos em cadeia fechada do sistema acoplado ao controlador LQG para $R = 200$, $Q_E = 100I$ e $R_E = 1$ são, assim:

$$\text{Pólos: } \begin{bmatrix} -0,7414 + i0,4158 \\ -0,7414 - i0,4158 \\ 0,8677 \\ 0,7873 \\ 0,6244 \\ -0,0634 + i0,1959 \\ -0,0634 - i0,1959 \\ 0,2336 \end{bmatrix}.$$

4. Validação do controlador LQG em simulação

4.1 Aproximação do sistema simulado ao sistema real

Nesta secção, abordam-se os complementos ao sistema modelado que o aproximam do sistema real, nomeadamente a inclusão da “zona morta”, e outros fatores que permitem um melhor seguimento da entrada imposta. Assim, implementou-se em *Simulink* o esquema da figura 4.1. Nesta fase da implementação, optou-se por gerar formas de onda em MATLAB tal como foi feito na primeira parte deste projeto e importá-las para o *Simulink*, acrescentando depois um bloco de ganho para impor amplitudes que fossem coerentes e viáveis no sistema físico.

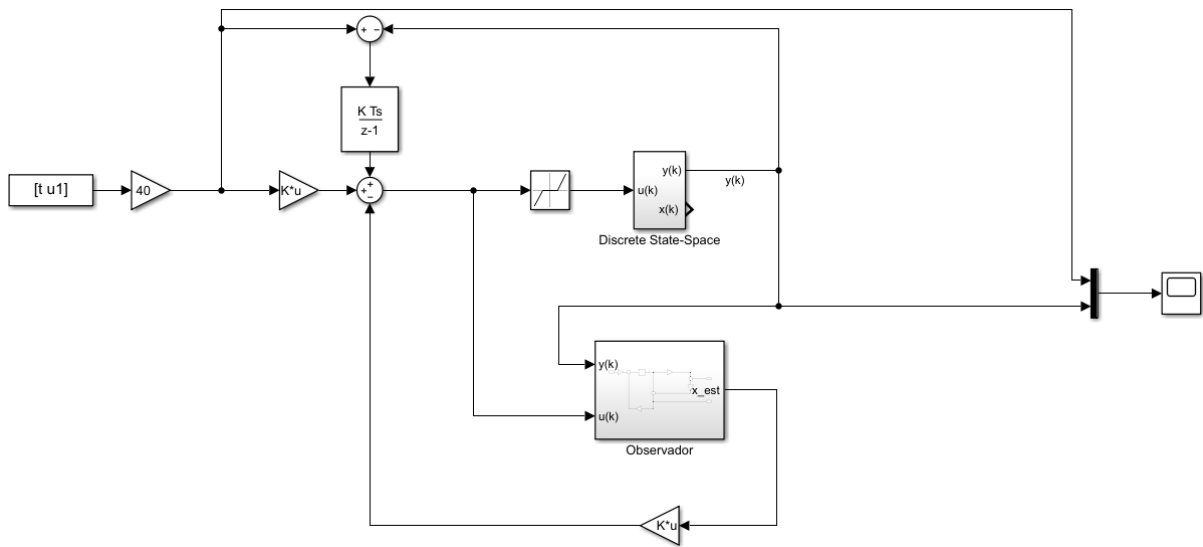


Figura 4.1: Diagrama de blocos do sistema final - ganho estático unitário, modelação da *dead zone* e inclusão de integrador.

Durante a primeira parte deste projeto, observou-se que o motor tem uma “zona morta” - uma gama de valores de tensão à qual o motor não consegue reagir -, isto é, existe uma tensão de limiar a partir da qual o motor irá responder mas, abaixo desse limiar, não existe resposta. Para validar o controlador construído, é necessário modelar este efeito para que se possam fazer testes com o mesmo. Como tal, adicionou-se à entrada do bloco que simula um motor ideal um bloco *dead zone* que simula as não idealidades do sistema, cujo efeito é a dificuldade do modelo projetado em atingir os valores de referência. A figura 4.2 mostra este fenómeno e a diferença entre o modelo ideal e o mais próximo do real. Denota-se também que a velocidade de reação não é afetada e os transitórios de variação de estado continuam a existir como ocorria anteriormente. Mais uma vez, as simulações foram realizadas para $R = 200$, $R_E = 1$ e $Q_E = 100I$, tendo-se considerado uma “zona morta” entre $[-0.15V, 0.15V]$.

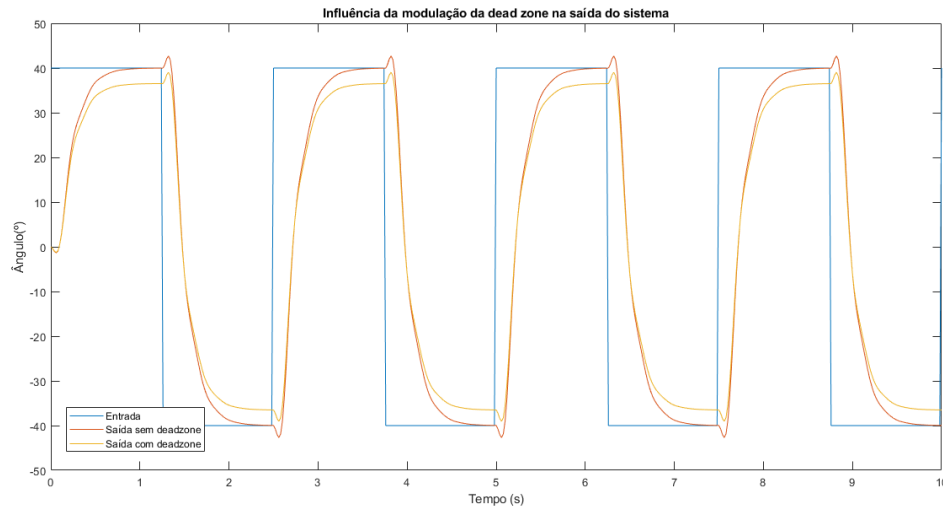


Figura 4.2: Visualização do efeito da modelação da zona morta.

4.2 Implementação do integrador

Denota-se agora que o problema de seguimento da referência que se tentou colmatar inicialmente, com a inserção de um ganho à entrada do sistema, ocorre novamente. Poder-se-ia optar por ajustar o ganho N previamente calculado; porém, isto teria de ser feito para cada teste que se realizasse e, ainda que funcionasse, não constitui uma solução robusta e eficaz. Como tal, opta-se pela adição de um integrador ao sistema que integra o erro de seguimento e, como tal, diminui o efeito da “zona morta”, aproximando a resposta do sistema à entrada imposta. Ao longo da simulação, o seu efeito tende a ser melhorado. Na figura 4.3, verifica-se que existe uma melhoria significativa na ação do controlador - estes gráficos foram elaborados para uma constante do integrador $k = 1.3 \times 10^{-2}$.

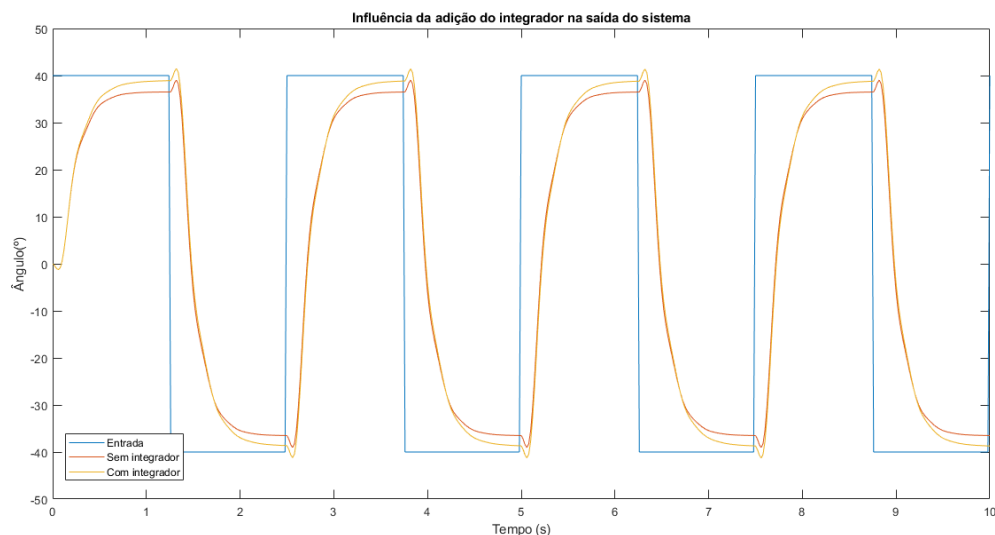


Figura 4.3: Visualização do efeito da inclusão do integrador.

Apesar disso, o integrador pode também ter efeitos nocivos na dinâmica do sistema, pelo que se optou por uma implementação condicional do mesmo associada a um ganho muito pequeno.

Assim, utilizou-se os blocos de lógica do *Simulink* para que o integrador só estivesse ativo quando a resposta está se estiver a aproximar da frequência. Na figura 4.4 observa-se esta implementação.

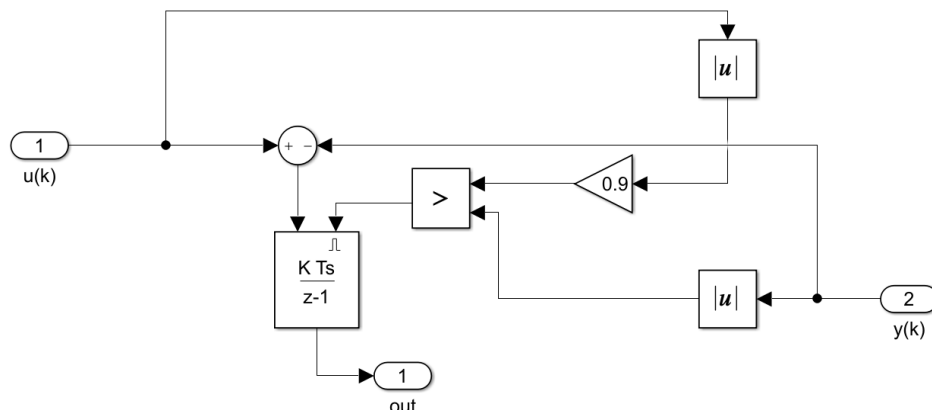


Figura 4.4: Implementação condicional do integrador.

4.3 Efeito da variação dos parâmetros do controlador LQG nas simulações

Está-se então em condições de introduzir o controlador projetado no modelo real. Contudo, antes disso, realizaram-se as simulações em ambiente computacional das condições que se pretendem impor. Começou-se por fazer variar o valor do parâmetro R , mantendo-se Q_E e R_E , e os resultados estão expostos na figura 4.5. Observa-se o efeito já explorado anteriormente e, além disso, comprova-se que a resposta se aproxima mais da referência com o aumento de R , ultrapassando por vezes quando ocorre o efeito da chicotada. Daqui infere-se que os melhores valores para aplicar no sistema real devem aproximar-se de $R = 200$, sendo este o ponto onde ocorre um sistema rápido mas não muito “nervoso”.

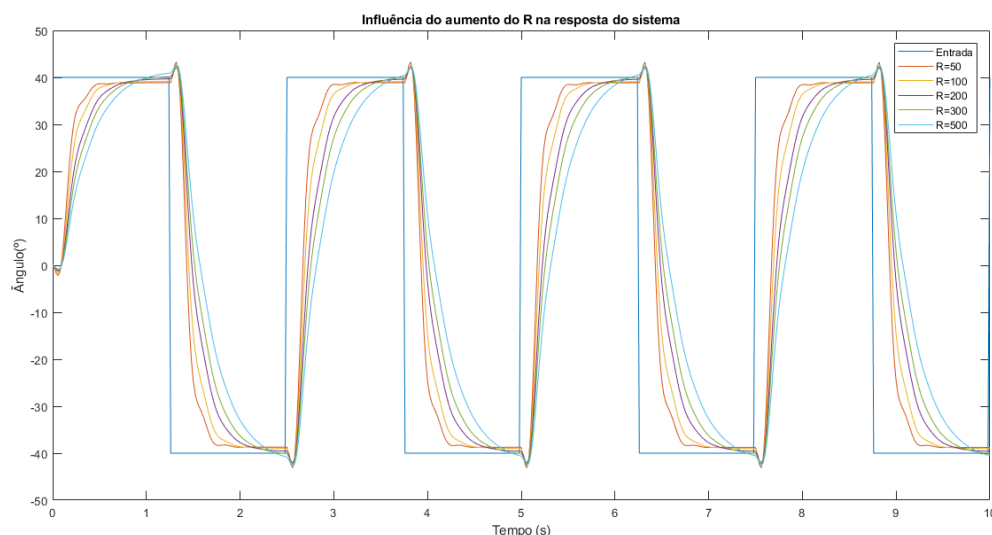


Figura 4.5: Efeito da variação de R na resposta do sistema.

Quanto à influência de Q_E e R_E na resposta do sistema, observa-se que esta é residual, quando comparada com a de R , excepto para valores baixos de Q_E , pelo que estes serão evitados. Opta-se por usar nos testes do sistema real $Q_E = 100I$ e $R_E = 1$, pois são os menores valores

para os quais se obtêm resultados aceitáveis e aumentá-los não tem efeitos significativos na resposta.

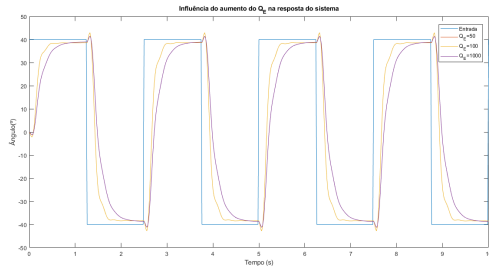


Figura 4.6: Efeito da variação de Q_E na resposta do sistema.

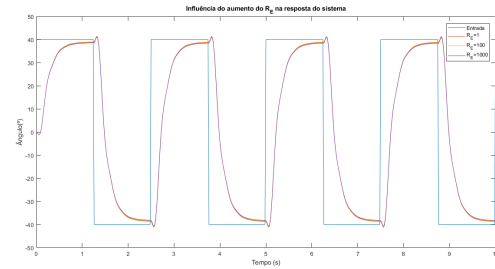


Figura 4.7: Efeito da variação de R_E na resposta do sistema.

Finalmente, verificou-se quais as respostas do sistema a diferentes constantes do integrador. Note-se que, quanto mais alto for o ganho do integrador, mais próxima da referência será a resposta. Porém, aumentar em demasia o efeito do integrador compromete a dinâmica do sistema e esse efeito é indesejado. Procurou-se então um *trade-off* e definiu-se $k = 1.3 \cdot 10^{-2}$ para as restantes simulações.

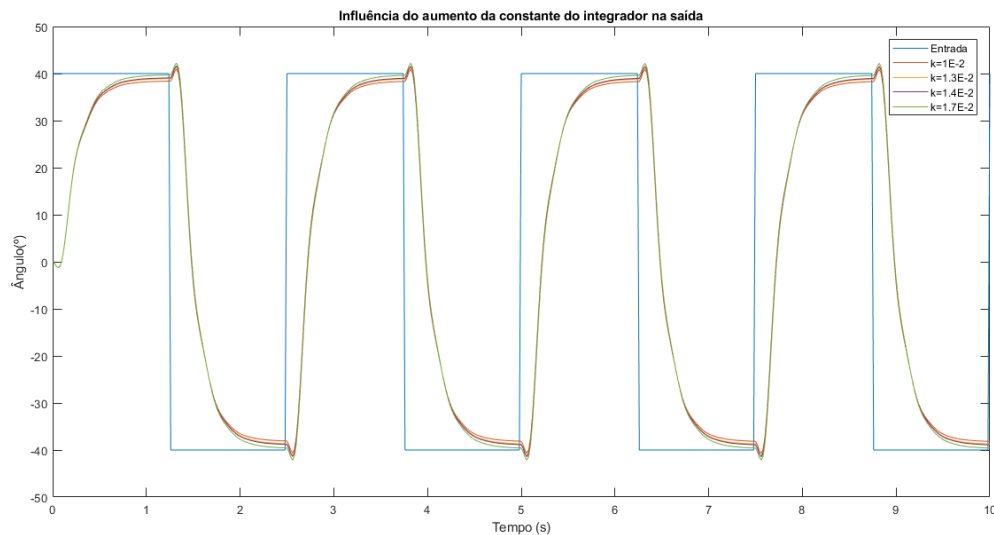


Figura 4.8: Efeito da variação da constante k do integrador na resposta do sistema.

4.4 Validação do controlador

Apresentam-se de seguida várias simulações que comprovam o funcionamento correto do controlador para o modelo encontrado durante a primeira parte deste trabalho laboratorial. Note-se que estas simulações foram realizadas para $R = 200$, $Q_E = 100I$, $R_E = 1$, $k = 1.3 \cdot 10^{-2}$ e sem implementar a ação condicional do integrador, pelo que o seu efeito é sentido ao longo de todo o tempo de simulação. A evolução destas respostas será discutida em detalhe no próximo capítulo.

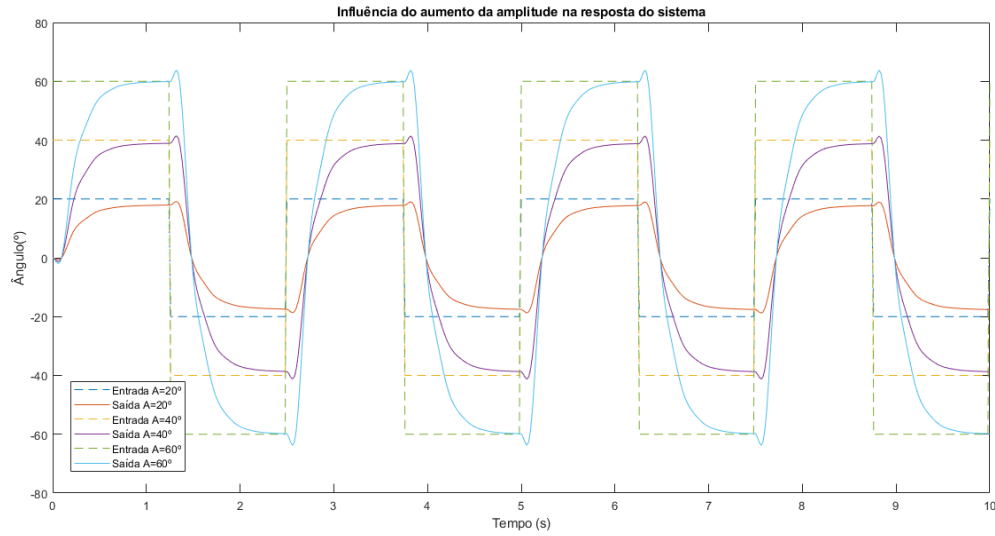


Figura 4.9: Efeito da variação da amplitude na resposta do modelo do sistema.

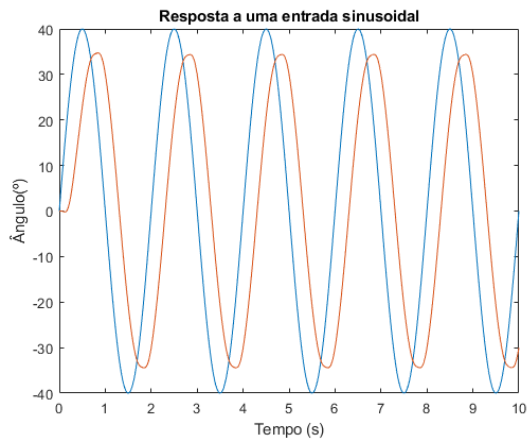


Figura 4.10: Resposta do modelo do sistema a uma entrada sinusoidal.

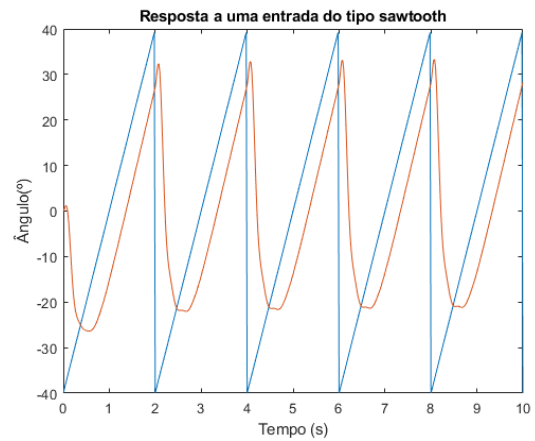


Figura 4.11: Resposta do modelo do sistema a uma onda em dente de serra.

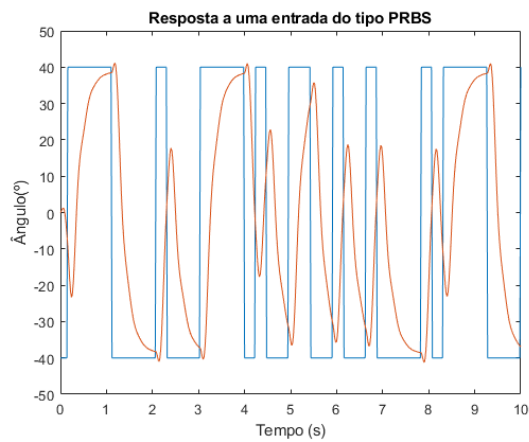


Figura 4.12: Resposta do modelo do sistema a um sinal do tipo PRBS.

5. Implementação do controlador LQG com o sistema real

Na seção anterior, implementou-se o controlador LQG com o modelo matemático do sistema em estudo. Assim, determinou-se a resposta deste para diferentes valores dos botões de ajuste R , Q_E e R_E , tendo-se observado que o primeiro apresentava a maior influência na saída $y(k)$ do modelo. Concluiu-se que os valores destes parâmetros que ofereciam a melhor resposta do modelo atendendo aos objetivos de projeto eram $R = 200$, $Q_E = 100I$ e $R_E = 1$, onde I é a matriz identidade de dimensões 4×4 . Consequentemente, o ganho do LQR, K , o ganho de Kalman do observador, M , e o ganho N que impõe ganho unitário na saída são, respetivamente:

$$K = \begin{bmatrix} 0.1229 & -0.2751 & 0.2245 & -0.0655 \end{bmatrix}, M = \begin{bmatrix} 42.4238 \\ 29.5072 \\ 17.7829 \\ 10.2467 \end{bmatrix} \text{ e } N = 0.0664.$$

Nesta secção, implementa-se o controlador LQG com o sistema real e verifica-se a resposta da barra flexível a diferentes entradas e características do controlador. Começa-se por fazer variar o botão de ajuste R em torno de $R = 200$; de seguida, estuda-se o movimento da barra para um sinal de entrada com diferentes formas de onda; depois, avalia-se a posição angular da ponta da barra para algumas frequências e amplitudes do sinal de entrada; finalmente, avalia-se a capacidade do sistema de rejeitar perturbações.

O diagrama de blocos em *Simulink* que permite realizar os testes com o sistema físico é semelhante ao utilizado na execução das simulações, mudando-se apenas o sentido do interruptor, e está ilustrado na figura 5.1. Note-se que também aqui se utilizou o integrador de modo a integrar o erro de seguimento, reduzindo o efeito da zona morta do motor, sendo o seu efeito apenas sentido em parte da resposta. O bloco que implementa o sistema real em *Simulink* e permite que este receba os sinais de entrada e devolva a sua resposta sob a forma de sinais digitais está representado na figura 5.2. Tal como para as simulações com o modelo matemático, as diferentes entradas do sistema são geradas a partir do *script cria_entradas.m* e os ganhos M e K são calculados com recurso ao *script cria_qanhos.m*.

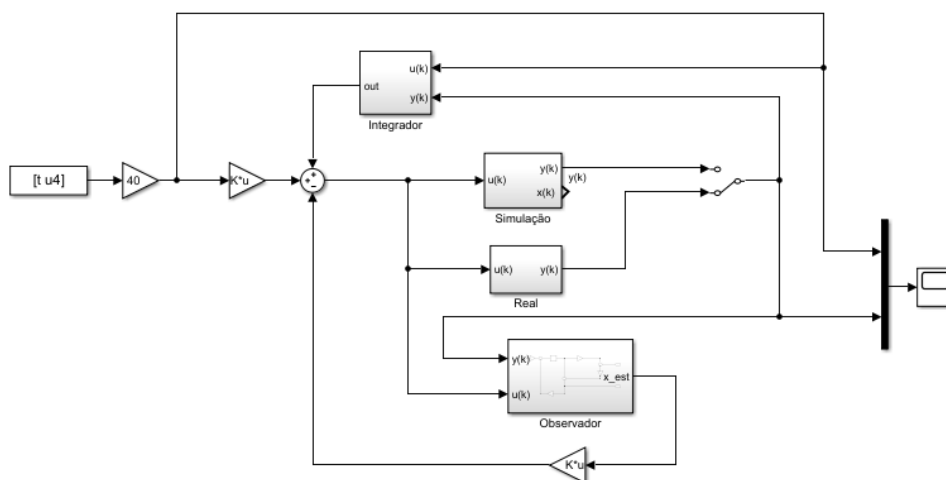
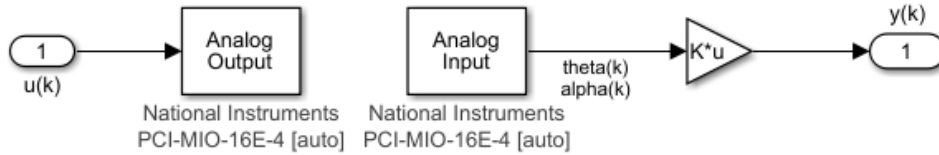
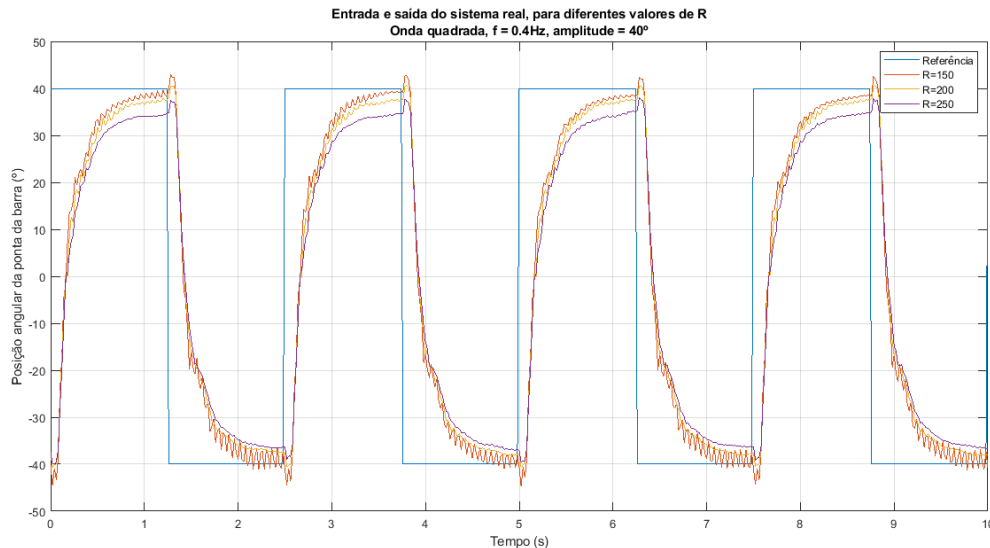


Figura 5.1: Diagrama de blocos em *Simulink* que implementa o controlador LQG com o sistema físico.


 Figura 5.2: Diagrama de blocos em *Simulink* que implementa o sistema físico.

5.1 Efeito da variação do parâmetro R

Recorrendo a simulações com o modelo matemático do sistema, determinara-se que a constante de ponderação R que oferecia um bom compromisso entre rapidez e estabilidade do modelo era $R = 200$. Pretende-se verificar a resposta do sistema real não só quando se utiliza este valor, mas também outros valores em torno do mesmo. Assim, fixou-se a frequência dos sinais de entrada em $f = 0.4$ Hz e introduziu-se uma onda quadrada de amplitude igual a 40° na referência, verificando-se o comportamento da saída para $R = 150$, $R = 200$ e $R = 250$. Os gráficos observados estão representados na figura 5.3.


 Figura 5.3: Evolução da posição angular da ponta da barra em função do tempo para $R = \{150, 200, 250\}$ e uma onda de referência quadrada de amplitude 40° e frequência 0.4 Hz.

Esta figura evidencia que, tal como enunciado pela teoria e verificado nas simulações, o valor mais baixo de $R = 150$, a laranja, oferece uma evolução da posição angular da ponta da barra mais “nervosa” e instável do que para $R = 200$ e $R = 250$. Este efeito deve-se à maior largura de banda do controlador LQ e é particularmente visível quando a barra tende para $y(k) = -40^\circ$, onde as oscilações em torno do valor de equilíbrio são da ordem de 2° . Em contrapartida, a utilização do valor do botão de ajuste $R = 150$ implica uma resposta mais rápida, havendo consequentemente um melhor seguimento da referência introduzida. Quando R toma o valor de 250, verifica-se que o sistema tem dificuldade em seguir a referência, apesar da introdução do integrador, apresentando um erro de seguimento próximo dos 5° , devido à existência da zona morta do motor. Nesta situação, apesar de o sistema ser mais estável, evidenciando menos e menores oscilações na sua saída, torna-se também mais lento.

Os resultados obtidos em laboratório demonstram, pois, que o valor $R = 200$ oferece um compromisso razoável entre rapidez de resposta e estabilidade. Por um lado, as oscilações da

saída são apreciavelmente menores do que as que se verificam para $R = 150$, o que mostra que o controlador é menos “nervoso”. Por outro lado, o sistema reage suficientemente rápido, apresentando um seguimento da referência com erros de cerca de 2° .

5.2 Resposta à variação da frequência de entrada

Pretende-se agora verificar a influência da frequência do sinal que se impõe como referência do sistema na sua resposta. Deste modo, definiu-se $R = 200$ e introduziu-se uma onda quadrada de amplitude igual a 40° na entrada do sistema. Observou-se a evolução da posição angular da barra quando a frequência da referência era 0.4 Hz, 0.5 Hz e 0.25 Hz. Os resultados estão discriminados nos gráficos das figuras 5.4, 5.5 e 5.6, respetivamente.

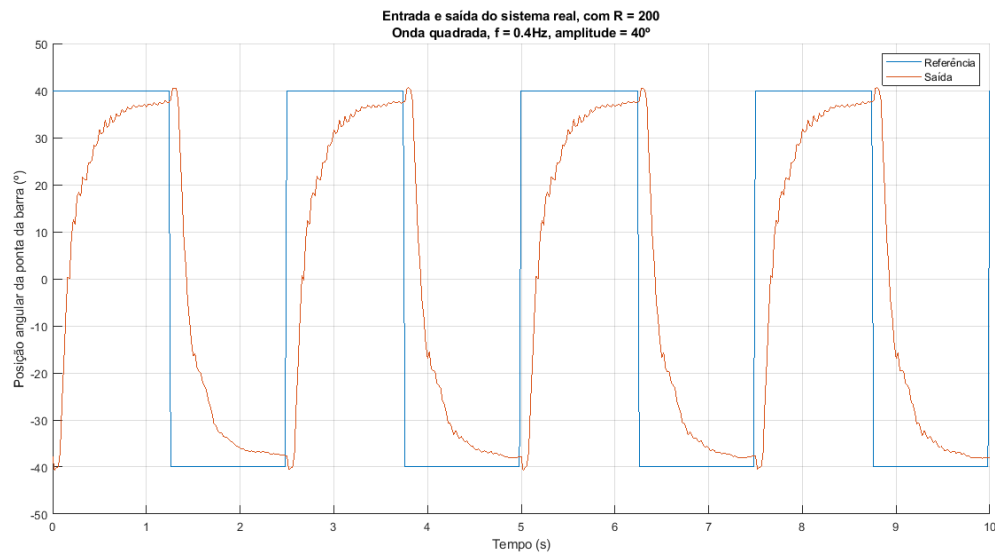


Figura 5.4: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 40° e frequência 0.4 Hz.

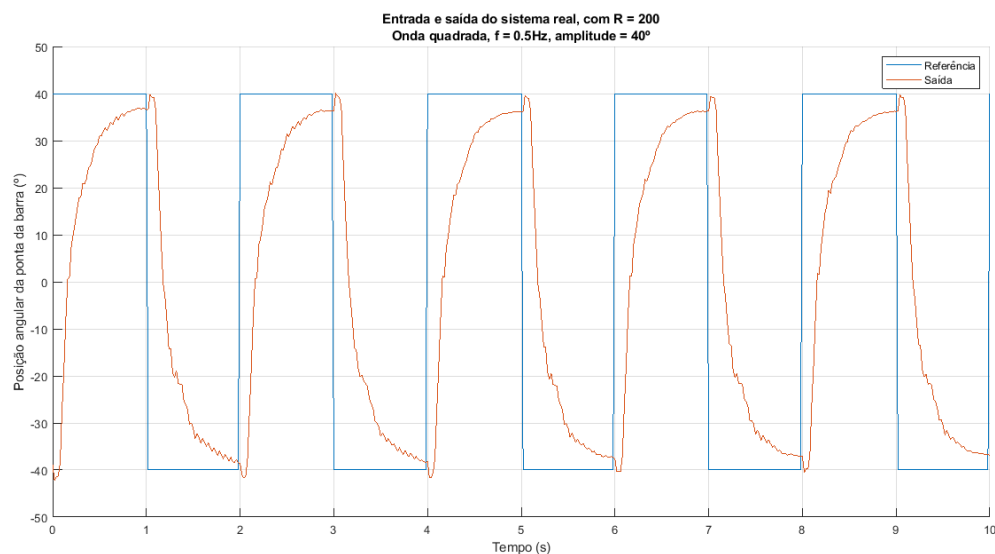


Figura 5.5: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 40° e frequência 0.5 Hz.

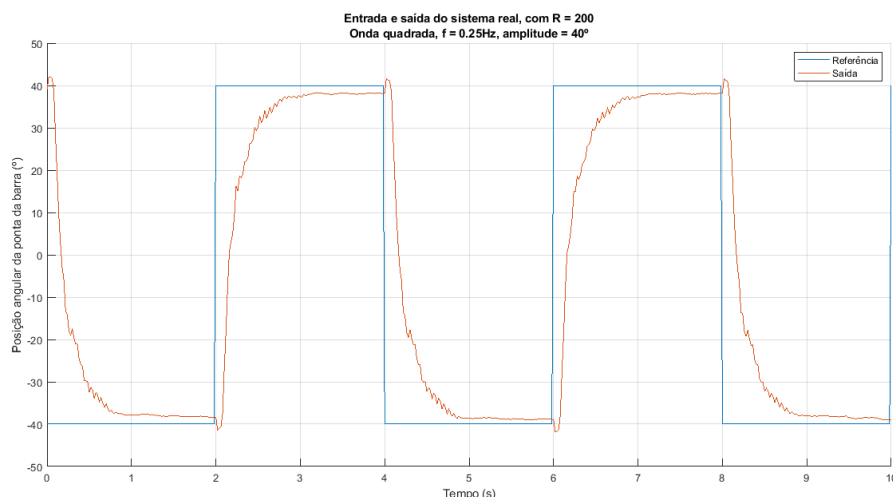


Figura 5.6: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 40° e frequência 0.25 Hz.

Estas figuras revelam que existe uma relação entre a frequência da referência e a capacidade de a saída realizar o seu seguimento: quanto maior for a frequência do sinal de entrada, menor é o intervalo de tempo que a saída tem para atingir o nível da referência. Para a entrada com $f = 0.5$ Hz, por exemplo, a saída nunca chega a estabilizar perto dos -40° . Diminuindo a frequência da onda quadrada para $f = 0.25$ Hz, a saída tem um maior período para se estabelecer, pelo que o seguimento da referência é melhor. Comprova-se, assim, que a frequência de funcionamento do sistema deve ser dimensionada adequadamente, de forma a que este siga corretamente a referência. A frequência a partir da qual o sistema deixa de se conseguir aproximar das amplitudes máxima e mínima da referência depende de múltiplos fatores, nomeadamente R , que influencia a rapidez de resposta, e a amplitude da referência, que se estudará na secção 5.4.

5.3 Resposta a entradas com diferentes formas de onda

Até agora, consideraram-se apenas ondas quadradas na entrada do sistema. Importa verificar a evolução da posição angular da barra quando a referência introduzida apresenta diferentes formas. Em particular, estuda-se o andamento da saída do sistema para uma referência sinusoidal (figura 5.8), em dente de serra (figura 5.9) e em escada multinível com 5 níveis e amplitude máxima 80° e 120° (figura 5.10). Inclui-se ainda o gráfico da figura 5.11, que ilustra a resposta do sistema quando a onda quadrada introduzida à sua entrada se trata de uma sequência binária pseudo-aleatória (PRBS - *pseudorandom binary sequence*), de largura de banda igual a 0.08 Hz e amplitude igual a 40° . No caso das ondas sinusoidal e dente de serra, a amplitude utilizada é também de 40° e a frequência é igual a 0.4 Hz. Em todas as experiências foi utilizado o valor de $R = 200$, uma vez que este provou oferecer um bom compromisso rapidez-estabilidade. O diagrama de blocos que implementa o gerador de sinal multinível com 5 níveis de amplitude e período de nível de 2 segundos está representado na figura 5.7.

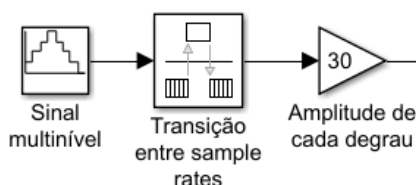


Figura 5.7: Diagrama de blocos que implementa o gerador de sinal multinível.

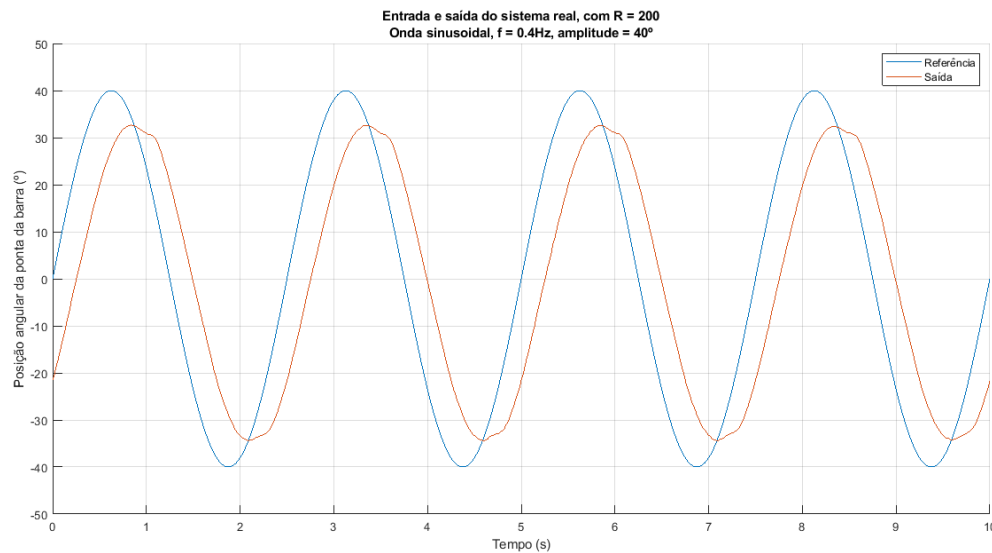


Figura 5.8: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência sinusoidal de amplitude 40° e frequência 0.4Hz .

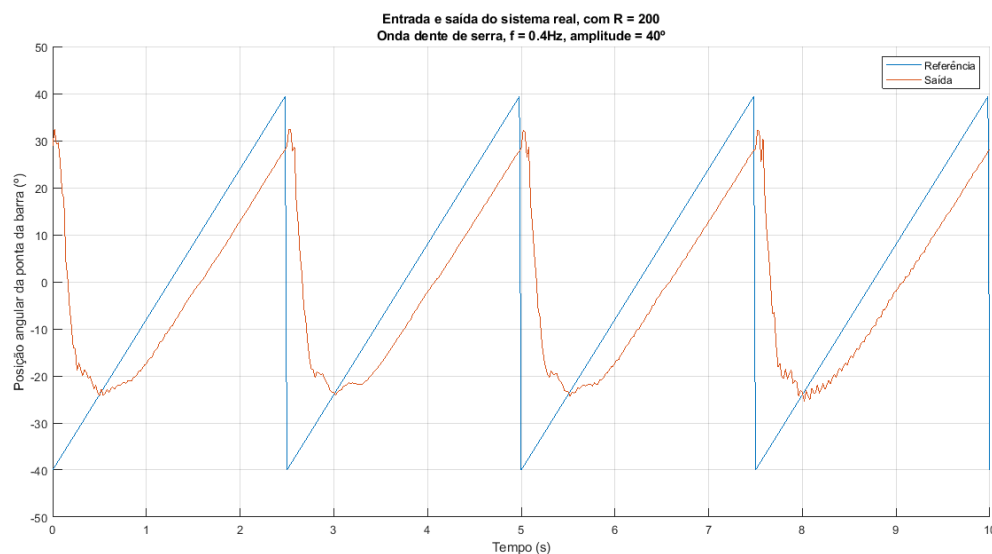


Figura 5.9: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência em dente de serra de amplitude 40° e frequência 0.4Hz .

As figuras 5.4, 5.8 e 5.9 apresentam a evolução da saída do sistema físico quando se introduz na sua entrada uma onda periódica quadrada, sinusoidal e em dente de serra, respetivamente. Essencialmente, verifica-se para os três casos aquilo que já tinha sido enunciado para a onda quadrada: o ganho estático é inferior a 1, o que se verifica pelo facto de a saída nunca atingir os valores correspondentes aos extremos da amplitude da referência, apesar da inclusão do integrador do erro de seguimento. Para além disso, a saída apresenta um pequeno desfasamento temporal relativamente à referência. Este efeito está bem patente na figura 5.8, onde a senoide da saída acompanha a da entrada, mas desfasada por algumas décimas de segundo. Isto é também visível para a onda em dente de serra, em que a velocidade angular é igual à da referência nas regiões de declive positivo. Devido ao atraso da saída relativamente à entrada, a primeira nunca chega a aproximar-se do valor mínimo da amplitude, 40° , já que a referência começa a crescer quando a saída ainda está a diminuir.

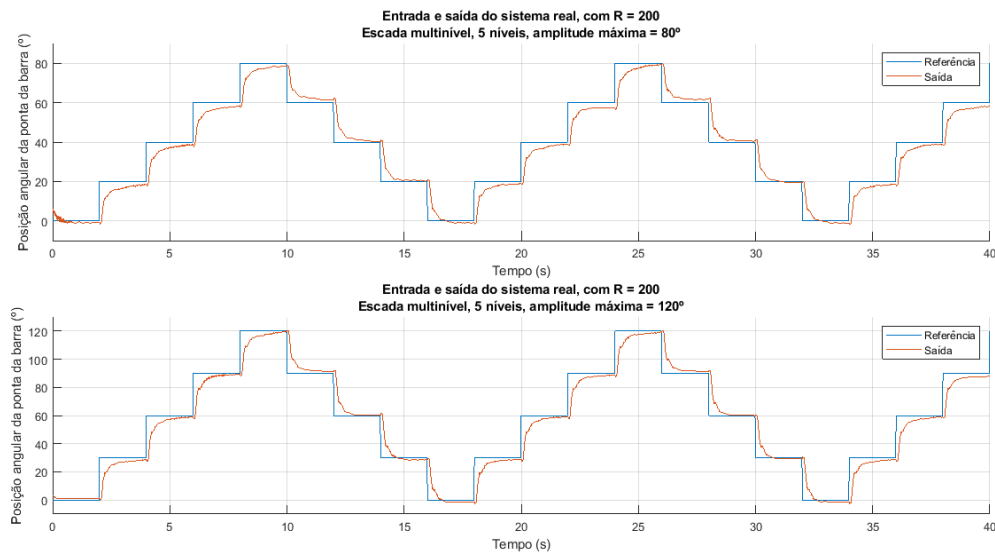


Figura 5.10: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência multinível com 5 níveis e amplitudes máximas 80° e 120° .

A figura 5.10 apresenta a resposta do sistema físico quando a referência é um sinal multinível com 5 níveis distintos. No gráfico de cima, o ângulo de referência avança 20° a cada 2 segundos, atingindo uma amplitude máxima de 80° , enquanto no de baixo os incrementos são de 30° com o mesmo período e uma amplitude máxima de 120° . Em geral, a saída apresenta um bom seguimento da referência, sendo evidente que a rapidez de resposta é superior quando a amplitude da referência é decrementada. Caso o intervalo de tempo que a saída demora a atingir a amplitude máxima não seja condicionante, este tipo de sinal pode ser utilizado em detrimento de uma onda quadrada simples quando se pretende atingir valores elevados da posição angular da barra. Isto evita picos de tensão abruptos no sistema e permite uma transição mais gradual entre as posições angulares que constituem os extremos.

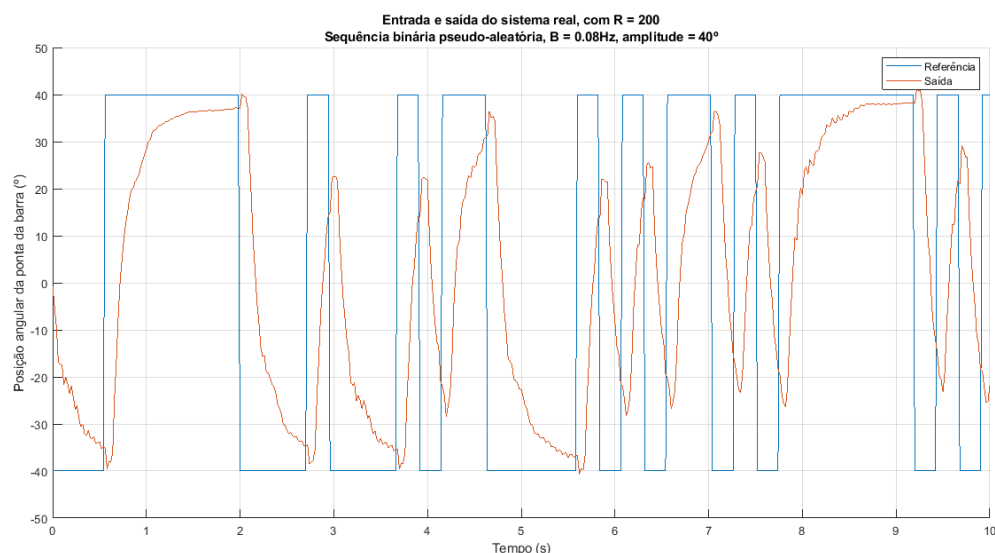


Figura 5.11: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma sequência binária pseudo-aleatória de amplitude 40° e largura de banda 0.08 Hz .

Na primeira fase do trabalho, a identificação e validação do modelo matemático do sistema foi realizada com recurso a ondas periódicas quadradas e sinais PRBS. Será, pois, interessante

verificar a resposta do sistema real controlado a uma entrada deste último tipo. Definiu-se, então, um sinal PRBS com largura de banda igual a 0.08 Hz e amplitude 40° . A referência e a saída do sistema estão representadas na figura 5.11.

Havia-se concluído que, para frequências da referência demasiado elevadas, a saída podia não ter tempo para estabilizar perto no valor de amplitude pretendido. Esse efeito verifica-se nesta imagem, por exemplo entre os instantes $t = 5.5$ s e $t = 8$ s. Neste intervalo de tempo, a frequência com que a referência muda de nível não permite que a saída se aproxime de qualquer um dos valores extremos da amplitude da referência, implicando que a barra apenas oscile em torno da posição central ($y(k) = 0^\circ$), sem nunca estabilizar. Assim, os resultados observados para uma sequência binária pseudo-aleatória vêm corroborar aquilo que se tinha determinado na secção 5.2: a frequência da referência deve ser sempre tida em conta no funcionamento do sistema físico.

5.4 Resposta à variação da amplitude de entrada

Do mesmo modo que a frequência do sinal de entrada pode condicionar a capacidade de a barra seguir a referência, é plausível questionar também a influência da amplitude do sinal no seguimento. De modo a averiguar as consequências da variação da amplitude da referência no seguimento desta por parte da barra, produziu-se o gráfico da figura 5.12. Utilizando-se uma onda quadrada de frequência 0.4 Hz e definindo-se $R = 200$, estudou-se a resposta do sistema físico quando a amplitude da referência era de 20° , 40° e 60° .

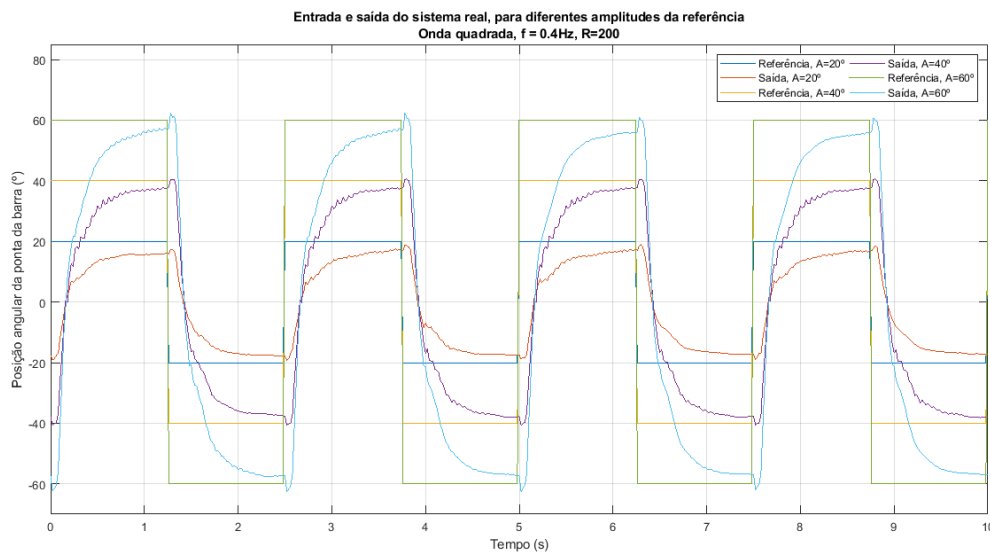


Figura 5.12: Evolução da posição angular da ponta da barra em função do tempo para $R = 200$ e uma onda de referência quadrada de amplitude 20° , 40° e 60° e frequência 0.4 Hz.

Para as amplitudes de 20° e de 40° , o erro de seguimento é semelhante, mas a saída estabiliza durante um maior período de tempo para a referência de menor amplitude - veja-se, por exemplo, o intervalo de tempo entre $t = 0$ s e $t = 2.5$ s. Quando a amplitude é de 60° , o erro de seguimento é ligeiramente superior e a saída apresenta mais dificuldades em estabilizar antes de o nível da referência ser invertido. Este facto sugere que, para que o sistema apresente um melhor seguimento da referência, estando condicionado à sua rapidez de resposta, a frequência da entrada deve variar no sentido contrário ao da amplitude: quanto maior for a amplitude do sinal que se pretende seguir, menor deve ser a frequência com que este inverte o seu sentido.

Verifica-se ainda que o efeito de chicotada é mais proeminente quanto maior for a amplitude da referência.

5.5 Rejeição de perturbações

De modo a terminar os testes ao sistema físico, avaliou-se a capacidade deste de rejeitar com sucesso as perturbações externas a que esteja sujeito. Assim, introduziu-se uma referência nula na sua entrada e fez-se rodar manualmente o eixo do motor, de forma a simular perturbações. O objetivo deste teste é averiguar se a barra retorna à posição inicial definida pela referência nula. As observações estão registadas nos gráficos das figuras 5.13.

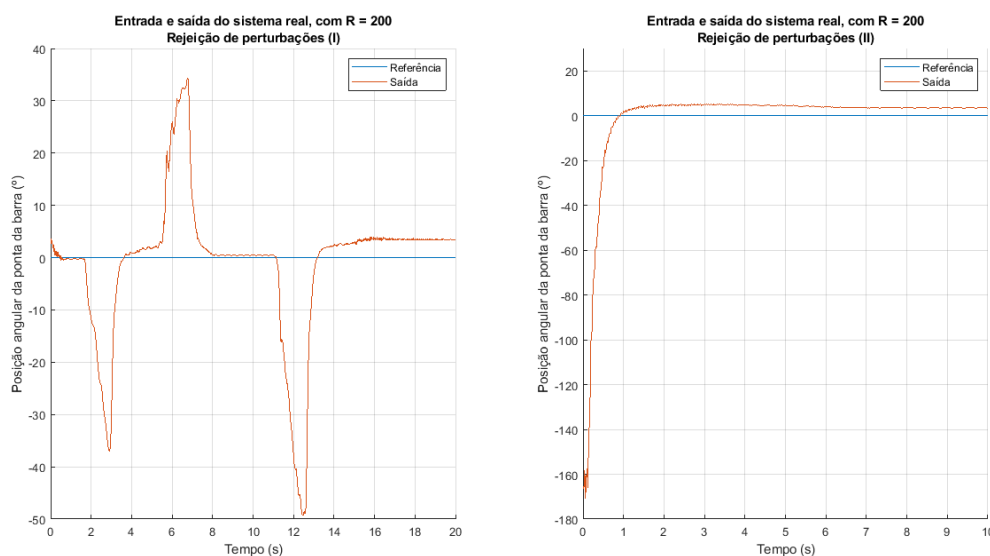


Figura 5.13: Evolução da posição angular da ponta da barra em função do tempo para uma referência nula e introdução de perturbações.

Na figura da esquerda, começou-se com a barra na posição de referência, rodando-se o eixo em ambos os sentidos e esperando que a barra estabilizasse na posição de referência entre perturbações. Na segunda figura, forçou-se a barra a iniciar o teste numa posição de cerca de 170°.

Ambas as experiências mostram que a resposta da barra é praticamente instantânea quando se deixa de aplicar a perturbação, regressando à posição de referência num intervalo de tempo inferior a 1 segundo. Mesmo quando a perturbação tinha amplitude perto do máximo possível (180°), o retorno à posição de referência deu-se dentro desse período de tempo. Todavia, é evidente que, quando a barra avança de amplitudes negativas para zero, existe um *offset* na posição angular em que estabiliza. Este efeito pode dever-se à zona morta do motor - o valor em que a barra estabiliza é tão próximo da referência, que a tensão a aplicar ao motor para que a barra se mova exatamente para $y(k) = 0$ é tão pequena que se localiza dentro da zona morta do motor.

Conclui-se, deste modo, que o sistema é robusto a perturbações, apesar das limitações que o caracterizam, nomeadamente a zona morta e a velocidade de resposta.

6. Conclusão

Ao longo deste projeto, estudou-se o funcionamento de um braço robótico flexível. Na primeira parte, cujas conclusões se apresentaram no primeiro relatório, procedeu-se à identificação e validação de um modelo numérico da dinâmica do sistema físico. Este modelo provou ser crucial para a segunda parte do projeto, referente ao controlo por computador do sistema, baseado no desenho de um controlador linear quadrático gaussiano (LQG). Este relatório discrimina todo o processo de desenho do controlador e respetivos testes, quer com o modelo matricial do sistema, quer com a barra flexível propriamente dita.

Iniciou-se o relatório com uma breve introdução que explica todos os aspetos teóricos inerentes ao dimensionamento do controlador LQG, nomeadamente o desenho do controlador linear quadrático (LQ) com o qual se realiza a realimentação linear das variáveis de estado e do observador corrente com filtro de Kalman que calcula a estimativa do estado do sistema, já que este não é observável em tempo real pelo algoritmo.

Na secção seguinte, implementou-se o controlador LQG com o modelo matemático. Avaliou-se o comportamento do controlador LQ e do observador separadamente, tendo-se determinado valores preliminares adequados dos botões de ajuste R , Q_E e R_E que regulam o controlador LQG. Concluiu-se que o parâmetro R tinha a maior influência na estabilidade e rapidez de resposta do modelo, enquanto as matrizes de covariância dos ruídos que afetam o sistema, Q_E e R_E , praticamente não influenciavam a sua resposta numa ampla gama de valores.

De seguida, procedeu-se à validação do controlador LQG em simulação, em que se aproximaram as características do modelo em *Simulink* dos aspetos que caracterizam o sistema real. Este processo requereu a introdução de uma zona morta, que modela o intervalo de tensões para as quais o motor não oferece qualquer resposta, implicando a existência de um erro de seguimento da saída relativamente à referência introduzida. De modo a contrariar este efeito, incluiu-se um integrador cuja função é integrar o erro de seguimento, aproximando a saída da referência. Realizou-se, então, um conjunto de simulações em ambiente computacional, de modo a averiguar a resposta do sistema controlado aproximado antes de se proceder aos testes no sistema físico.

O último capítulo deste relatório encerra as conclusões que se retiraram da resposta do sistema físico a diferentes características do mesmo e da referência introduzida à sua entrada. Estudou-se, assim, o efeito da variação do botão de ajuste R na resposta do sistema, tendo-se observado aquilo que já se tinha concluído anteriormente: quanto menor for R , mais rápida é a resposta, apesar de ser também mais instável e oscilatória. Verificou-se ainda a forma como a frequência, forma de onda e amplitude da referência podem ser fatores condicionantes no seguimento da mesma; se a frequência e/ou a amplitude da onda de entrada não forem dimensionadas adequadamente, o sistema pode não reagir suficientemente rápido, resultando num seguimento insatisfatório da referência. Para concluir, avaliou-se a capacidade de rejeição de perturbações, tendo-se observado que o sistema é robusto, apesar das suas limitações.

Importa ainda referir que o sistema apresenta alguns limites de operação. Em particular, está condicionado pela zona morta que caracteriza o motor, pela frequência e amplitude da referência e pelas tensões de operação do motor, que se encontram no intervalo $[-10, 10]$ V - quando algum destes limites é ultrapassado, verifica-se uma *death line*. Este efeito foi particularmente crítico na identificação e validação do modelo matemático do sistema.

Terminado o desenho e implementação do controlador LQG, conclui-se que se cumpriram os objetivos de controlo definidos no início do projeto.

Bibliografia

- [1] G. F. Franklin, J. D. Powell e M. Workman. *Digital Control of Dynamic Systems*. 3^a ed. Half Moon Bay, CA: Ellis-Kagle Press, 1998.
- [2] J. M. Lemos. *Computer Control. Parts 1, 2*. Departamento de Engenharia Eletrotécnica e de Computadores, Área Científica de Sistemas, Decisão e Controlo, Instituto Superior Técnico. 2018.
- [3] J. P. Gomes, A. Bernardino e J. M. Lemos. *Identification and Computer Control of a Flexible Robot Arm Joint*. Departamento de Engenharia Eletrotécnica e de Computadores, Área Científica de Sistemas, Decisão e Controlo, Instituto Superior Técnico. 2019.
- [4] J. M. Lemos e J. P. Gomes. *Controlo por Realimentação de Variáveis de Estado em Tempo Discreto*. Instituto Superior Técnico. Out. de 2016.
- [5] J. R. Azinheira. *Controlo de Voo. MEAer, 3^o ano, 2^o sem.* Departamento de Engenharia Mecânica, Instituto Superior Técnico. Fev. de 2019.

A. Código MATLAB utilizado

cria_entradas.m

```
1 %criacao dos estímulos a aplicar ao sistema
2 t = (0:1/50:60)'; %fs=50Hz; T=120s
3 u1 = square(2*pi*0.5*t); %f=0.4 Hz
4 u2 = idinput(length(t)+100, 'prbs', [0 0.08]); %B=0.08
5 u2(1:100) = []; %descartar os primeiros 2 segundos
6 u3=sawtooth(2*pi*t*0.4); %f=0.4 Hz
7 u4=sin(2*pi*t*0.4); f=0.4 Hz
```

cria_ganhos.m

```
1 load 'total.mat'
2 % Regulator design
3 Q = C'*C;
4 R = 200;
5 [K,S,E] = dlqr(A,B,Q,R);
6
7 % Localizacao dos polos da cadeia fechada no plano Z:
8 [num, den] = ss2tf(A - B*K, B, C, D);
9 figure(1);
10 zplane(num, den);
11
12 % Root-Locus Simétrico
13 sim_rlocus = tf(conv(num, fliplr(num)), conv(den, fliplr(den)));
14 figure(2)
15 rlocus(sim_rlocus);
16 hold on;
17 zplane([], []);
18 axis([-3 3 -3 3]);
19
20 Ts=0.02;
21 cont=ss(A,B,K,0,Ts);
22 figure(3)
23 bode(cont)
24 [Gm,Pm,~,~] = margin(cont);
25
26 % Current estimator design
27 w2 = 100;
28 QE = eye(size(A))*w2;
29 RE = 1;
30 G = eye(size(A));
31 [M,P,Z,EE] = dlqe(A,G,C,QE,RE);
32
33 % Estimator implementation
34 PHIE = A-M*C*A;
35 GAMMAE = B-M*C*B;
36 CE = eye(size(PHIE));
37 DE = zeros(size(CE,1),size(GAMMAE,2));
38
39 lgg_sr= ss([A zeros(size(A)); M*C*A PHIE-GAMMAE*K], [B; M*C*B], [zeros(size(K)) ...
    K],0,Ts);
40 [Gm1,Pm1,~,~] = margin(lgg_sr);
41
42 % External reference input
43 N = inv([A-eye(size(A)), B; C,0])*[zeros(size(A,1),1);1];
44 Nx = N(1:end-1,:);
```

```

45 Nu = N(end,:);
46 Nbar = Nu+K*Nx;
47
48 lgg_cr= ss([A -B*K; M*C*A PHIE-GAMMAE*K-M*C*B*K], [B; M*C*B+GAMMAE]*Nbar, [C ...
    zeros(size(C)),0,Ts);
49 figure(4)
50 bode(lgg_cr);
51 [Gm2,Pm2,-, -] = margin(lgg_cr);
52
53 figure(6)
54 zplane([], [E; EE]);
55
56 %LQG com referencia
57 % Localizacao dos polos da cadeia fechada no plano Z:
58 [num_cr, den_cr] = ss2tf([A -B*K; M*C*A PHIE-B*K], [B; B]*Nbar, [C ...
    zeros(size(C)),0]);
59 figure(7);
60 zplane(num_cr, den_cr);
61 ganho_cr=sum(num_cr)/sum(den_cr);
62
63 % Root-Locus Simetrico
64 locus_cr = tf(conv(num_cr, fliplr(num_cr)), conv(den_cr, fliplr(den_cr)));
65 figure(8)
66 rlocus(locus_cr);
67 hold on;
68 zplane([], []);
69 axis([-3 3 -3 3]);

```