



Universidade de Lisboa
Instituto Superior Técnico
Mestrado Integrado em Engenharia Aeroespacial
4º Ano 1º Semestre - Ano Lectivo 2014/2015

Controlo por Computador

Segundo Relatório do Projecto

Identificação e Controlo por Computador de uma Barra Flexível

Miguel Freitas da Silva Mendes, nº 72740
João Lucas dos Santos Munhão, nº 72842
João Paulo da Rocha Silva, nº 73411

Professor João Pedro Gomes

Lisboa, 21 de Dezembro de 2014

Conteúdo

1	Introdução	2
2	Especificações do Sistema de Controlo	2
3	Fundamentação Teórica	2
3.1	Controlador LQ	2
3.2	Filtro de Kalman	5
3.3	Controlador LQG	7
3.3.1	Recuperação do Ganho de Malha	8
3.3.2	Seguimento de Referência	8
3.3.3	Estimador Corrente	9
3.3.4	Modelo em Espaço de Estados Total	9
4	Projecto do Controlador LQG	11
4.1	Controlador LQ	11
4.2	Root Square Locus	14
4.3	Observador	17
4.4	Inverso do Ganho Estático	20
4.5	Sistema Completo	20
5	Implementação do Controlador LQG	21
5.1	Simulação	21
5.1.1	Representação em Simulink	21
5.1.2	Resultados na Simulação	22
5.2	Realidade	25
5.2.1	Representação em Simulink	25
5.2.2	Resultados no Sistema Real	26
6	Características do Sistema	30
6.1	Sistema em Malha Fechada	30
6.2	Função de Transferência do Sistema em Malha Fechada	31
6.3	Resposta no Tempo	33
6.4	Resposta em Frequência	35
7	Considerações Finais	38
7.1	Inclusão de um pré-filtro	38
7.2	Avaliação do Desempenho do Controlador e Limites do Desempenho do Sistema	38
7.3	Vários Sinais de Referência	38
8	Conclusão	41
9	Bibliografia	41

1 Introdução

O objectivo desta 2ª parte laboratorial consiste em desenvolver um sistema de controlo para o modelo identificado na 1ª parte do projecto que posicione a ponta da barra flexível no ângulo desejado.

Utilizar-se-á um controlador LQG que se baseia na junção de um controlador LQ com um Observador construído através de um Filtro de Kalman, cuja base de funcionamento será explicitada ao longo deste relatório.

Em ambiente de simulação, determinar-se-ão os parâmetros associados ao controlador LQG e testar-se-á o comportamento do sistema com a variação dos mesmos de forma a avaliar o melhor desempenho que este poderá apresentar.

Finalmente, testaremos o controlador obtido no sistema real, verificando a sua validade e funcionalidade.

2 Especificações do Sistema de Controlo

Antes de iniciarmos o projecto do sistema de controlo, é necessário definir quais as especificações do controlador, para sabermos que tipo de controlador implementar e quais as suas características. Assim, queremos um controlador que:

- Estabilize o sistema;
- Permita o seguimento da referência fornecida tão rapidamente quanto possível;
- Provoque um movimento suave e sem vibrações na barra;
- Permita o seguimento da referência com pouco atraso em relação a esta;
- Não provoque a saturação do actuador do motor, nem que seja muito exigente com este.

3 Fundamentação Teórica

Como fundamento teórico do trabalho desenvolvido nesta fase do projecto, foram utilizados conceitos, estudados nas aulas teóricas da disciplina de Controlo por Computador, que se detalham nesta secção.

3.1 Controlador LQ

Para sistemas controláveis, podemos calcular os ganhos do controlador por realimentação de variáveis de estado por forma a colocar os pólos da cadeia fechada em qualquer posição. Põe-se no entanto o problema de determinar as posições adequadas a escolher.

Obviamente, os pólos da cadeia fechada deverão estar dentro do círculo unitário para que o sistema em cadeia fechada seja assintoticamente estável.

Uma possibilidade consiste em projectar o controlador por forma a minimizar o custo quadrático:

$$J = \frac{1}{2} \sum_{k=0}^{\infty} [\mathbf{x}^T(k) \mathbf{Q} \mathbf{x}(k) + R u^2(k)] \quad (1)$$

Em que \mathbf{Q} é uma matriz quadrada $n \times n$, semidefinida positiva, e R é um escalar, $R > 0$.

Repare-se que se escolhermos:

$$Q = C^T C \quad (2)$$

E tendo em conta que:

$$x^T Q x = x^T C^T C x = (Cx)^T Cx = y^T y = y^2$$

Pelo que o custo fica:

$$J = \frac{1}{2} \sum_{k=0}^{\infty} y^2(k) + R u^2(k) \quad (3)$$

Este custo é formado por duas parcelas:

- Uma parcela (y^2) tem por objectivo que a potência da saída seja pequena;
- A outra parcela (u^2) tem por objectivo que a variável manipulada (u) seja pequena.

De um modo geral, estes dois objectivos são antagónicos, e o parâmetro R (o “peso” na acção de controlo) pondera a importância relativa de cada um deles. R pequeno significa que o custo da parcela do controlo é menor, pelo que o controlador vai reduzir y , deixando u ser maior.

Uma variação genérica das potências do sinal de entrada e do sinal de saída com o aumento de R encontra-se na figura (1).

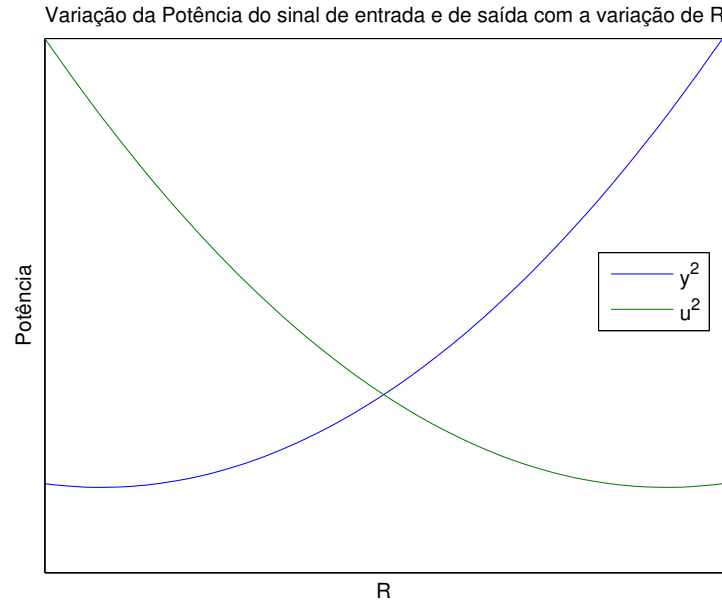


Figura 1: Variação típica das potências de entrada e da saída com o parâmetro R .

De um ponto de vista do domínio da frequência, quando R aumenta o sistema em cadeia fechada fica mais lento, ou seja, a sua largura de banda diminui.

O parâmetro R proporciona uma variável de ajuste muito importante para ajustar o controlador. Quando reduzimos R , a largura de banda da cadeia fechada aumenta e a resposta no tempo fica mais rápida, o que é bom, mas dois fenómenos indesejáveis podem ocorrer:

- A amplitude de u aumenta, o que pode levar à saturação do actuador;
- O aumento da largura de banda leva a que haja efeitos não considerados no modelo (por exemplo, picos de ressonância de alta frequência, tal como na barra flexível considerada no laboratório), que são amplificados e que podem conduzir à instabilidade do sistema em cadeia fechada.

A escolha de R é assim um compromisso entre conseguir uma resposta rápida e a estabilidade.

A lei de controlo obtida pela minimização de J denomina-se linear (pois o modelo é linear) quadrática (do custo ser quadrático), ou controlo LQ.

Assim, a solução é dada por uma retroacção constante do estado:

$$u(k) = -K_c x(k) \quad (4)$$

Em que o vector de ganhos óptimo é dado por:

$$K_c = \frac{1}{R + \Gamma^T S \Gamma} \Gamma^T S \Phi \quad (5)$$

E a matriz S ($n \times n$) é a solução da equação algébrica de Ricatti:

$$S = \Gamma^T \left[S - S \Gamma^T \Gamma S \frac{1}{R} \right] \Gamma + H^T H \quad (6)$$

No MATLAB, os ganhos do controlador LQ (a matriz K_c) podem ser obtidos com a função `dlqr`.

Esta função também retorna os pólos da cadeia fechada quando se utiliza o controlador LQ, isto é, os valores próprios de $\Phi - \Gamma K_c$.

Segundo o teorema de Chang-Letov, os pólos da cadeia fechada do controlador LQ são dados pelas n raízes estáveis (dentro do círculo unitário) de:

$$Ra(z)a(z^{-1}) + b(z)b(z^{-1}) = 0 \quad (7)$$

Em que $b(z)/a(z)$ é a função de transferência em cadeia aberta.

Repare-se que para R muito pequeno, a equação (7) reduz-se a:

$$b(z)b(z^{-1}) = 0 \quad (8)$$

Pelo que, para esta escolha de R , os pólos da cadeia fechada cancelam os zeros se estes estiverem dentro do círculo unitário, ou são os seus “inversos” (a um zero em z_i corresponde um pólo em $1/z_i$), se os zeros estiverem fora do círculo unitário.

Analogamente, se R for muito grande, a equação (7) fica:

$$a(z)a(z^{-1}) = 0 \quad (9)$$

Nesta situação, os pólos da cadeia fechada são os pólos da cadeia aberta, se estes forem estáveis, ou os seus inversos se os pólos da cadeia aberta forem instáveis (e por tanto tornam-se pólos estáveis).

Em qualquer caso, o sistema em cadeia fechada é sempre estável quando se usam os ganhos do LQ.

Para valores intermédios do peso R , é necessário resolver a equação (7) e tomar os pólos estáveis. Deste modo ficamos a saber qual a posição dos pólos da cadeia fechada que corresponde o ganho K_c .

3.2 Filtro de Kalman

Na maior parte dos sistemas de controlo não é possível medir directamente a variável x , mas pela relação que esta variável apresenta com a saída y é possível estimar o seu valor:

$$x(k+1) = \Phi x(k) + \Gamma u(k) \quad (10)$$

$$y(k) = Hx(k) \quad (11)$$

Com $x \in \mathbb{R}^n$, $y \in \mathbb{R}$ e $u \in \mathbb{R}$.

Para estimar o estado de um sistema uma montagem exemplificativa seria a seguinte:

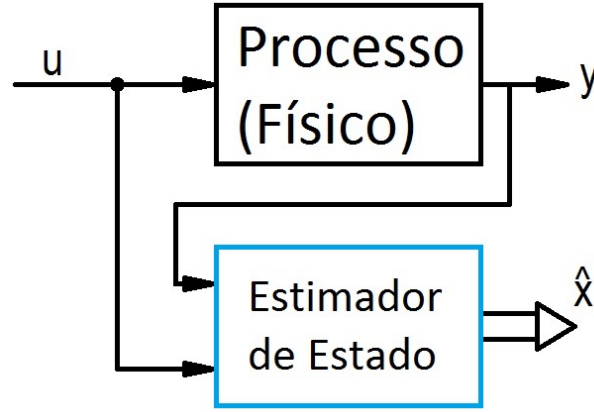


Figura 2: Montagem de um possível estimador. O bloco em azul é realizado computacionalmente.

$$\begin{aligned} \hat{x} &\rightarrow \text{estimativa do estado } x \\ \tilde{x} = x - \hat{x} &\rightarrow \text{erro de estimação} \end{aligned}$$

Para além do mais é desejável que a estimativa do estado, que inicialmente poderá ser diferente do valor verdadeiro de estado, com a passagem do tempo discreto k tenda rapidamente para o valor verdadeiro.

Assim, e com base na figura (2), podemos escrever a equação do estimador preditivo:

$$\hat{x}(k+1) = \Phi \hat{x}(k) + \Gamma u(k) + K_0[y(k) - H\hat{x}(k)] \quad (12)$$

Em que o último termo é proporcional a $y(k) - H\hat{x}(k)$, ou seja, a diferença entre o que observamos na saída $y(k)$ e o que estamos a espera de observar dada a estimativa e o nosso conhecimento do modelo das observações definido por (11).

O erro $\tilde{x} = x - \hat{x}$ satisfaz uma equação que se obtém quando subtrai-se (12) de (10).

$$\begin{aligned} \tilde{x}(k+1) &= \Phi[x(k) - \hat{x}(k)] - K_0[y(k) - H\hat{x}(k)] \\ \tilde{x}(k+1) &= \Phi \tilde{x}(k) - K_0[y(k) - H\hat{x}(k)] \end{aligned} \quad (13)$$

E tendo em conta (11) vemos que:

$$\tilde{x}(k+1) = [\Phi - K_0 H] \tilde{x}(k) \quad (14)$$

O projecto do ganho K_0 é feito de forma a impor os valores próprios de $\Phi - K_0H$. Sendo $\alpha_0(z)$ o polinómio característico desejado para o estimador de estado, K_0 satisfaz a seguinte equação:

$$\det[zI - \Phi + K_0H] = \alpha_0(z) \quad (15)$$

A equação anterior pode ser resolvida se e só a matriz:

$$O(H, \Phi) \triangleq \begin{bmatrix} H \\ - - - \\ H\Phi \\ - - - \\ H\Phi^2 \\ - - - \\ \vdots \\ - - - \\ H\Phi^{n-1} \end{bmatrix} \quad (16)$$

Chamada matriz de observabilidade, tiver característica n:

$$\text{car}[O(H, \Phi)] = n \quad (17)$$

Esta equação pode ser resolvida no MatLab pela fórmula de Ackerman através da função **acker**.

No domínio da frequência a escolha dos valores próprios desejados para a equação de erro é equivalente à escolha do valor da largura de banda do estimador do modelo. Quanto mais perto da origem estiverem os valores próprios especificados, maior será a largura de banda do estimador.

O aumento da largura de banda indiscriminadamente não é bom por dois motivos:

- Pode haver erros de modelação na alta frequência. O aumento da largura de banda do estimador pode levar à amplificação dos erros e consequentemente à divergência da estimativa.
- Há perturbações aleatórias e ruído do sensor.

Assim podemos melhorar o modelo anteriormente construído em (10) e (11) e adicionar a componente relacionada ao ruído obtendo assim obtemos:

$$x(k+1) = \Phi x(k) + \Gamma u(k) + w(k) \quad (18)$$

$$y(k) = Hx(k) + v(k) \quad (19)$$

O ruído do sensor, $v(k)$, e o ruído do processo, $w(k)$, são sequências gaussianas, incorrelacionadas de ruído branco, com média nula e variância:

$$\varepsilon[w(k)w^T(k)] = Q_w \quad (20)$$

$$\varepsilon[v(k)v^T(k)] = R_v \quad (21)$$

Admitindo válido este modelo, concluímos que o valor do ganho K_0 do estimador, que corresponde a uma dada largura de banda, permitirá a passagem para a estimativa \hat{x} as componentes na frequência do estado x que estão dentro da sua banda, e também ruído cuja frequência esteja dentro da banda.

Com o aumento do ganho K_0 mais e mais frequências do estado \mathbf{x} entrarão na estimativa $\hat{\mathbf{x}}$, mas também mais ruído passará. Haverá um ponto em que não há passagem de mais sinal de interesse \mathbf{x} (admitindo este passa-baixo), mas o ruído, por ter densidade espectral de potência constante na frequência, continuará a passar causando divergência da estimativa.

Para a determinação do K_0 ideal é necessário descobrir qual é o minimizador centrado, ou seja, $\varepsilon[\mathbf{x}(k) - \hat{\mathbf{x}}(k)] = 0$, que minimiza a seguinte função custo:

$$J_0 = \varepsilon \left[\sum_{k=0}^{\infty} \|\mathbf{x}(k) - \hat{\mathbf{x}}(k)\|^2 \right] \quad (22)$$

Os estados e observações geradas pelo modelo (18) e (19) levam a que o requisito de que o estimador seja centrado conduza à estrutura (13).

A minimização de J_0 conduz a um ganho calculado por:

$$K_0 = \Phi P H^T \frac{1}{R_v + H P H^T} \quad (23)$$

Em que a matriz de covariância do erro de estimação, P , satisfaz a equação de Riccati algébrica:

$$P = \Phi P \Phi^T + Q_w - \frac{\Phi P H^T H P \Phi^T}{R_v + H P H^T} \quad (24)$$

O vector de ganho óptimo correspondente ao filtro de Kalman pode ser calculado no MATLAB através da função `dlqe`.

3.3 Controlador LQG

Um controlador LQ utiliza a realimentação de todos os estados. Ora, se os estados não estão acessíveis para medição directa, então o um observador com base no filtro de Kalman é a solução, fazendo-se realimentação da estimativa dos estados ao invés dos estados em si.

Repare-se que, considerando o modelo em espaço de estados discreto total do controlador LQG, verifica-se que o processo contribui com n variáveis de estado para o modelo (o estado \mathbf{x}), e o controlador contribui com outras n variáveis de estado (as estimativas $\hat{\mathbf{x}}$).

Se cada um destes componentes tiver sido correctamente elaborado, então ao implementá-los em conjunto tudo deverá funcionar correctamente, já que os valores próprios do sistema total correspondem à união dos valores próprios do sistema em malha fechada apenas com o controlador (isto é, os valores próprios de $\Phi - \Gamma K_c$) com os valores próprios do observador (isto é, os valores próprios de $\Phi - K_0 H \Phi$).

Este resultado é assegurado pelo teorema da separação, que diz que para sistemas lineares em que o estado é estimado por um observador como o explorado na secção anterior, e o controlo é calculado através de uma retroacção linear da estimativa do estado, então os pólos do sistema em cadeia fechada são as raízes do polinómio característico que se obteria realimentando o estado:

$$|zI - \Phi + \Gamma K_c|$$

E as raízes do polinómio característico do erro do observador, quando não há retroacção do estado nem da sua estimativa:

$$|zI - \Phi + K_0 H \Phi|$$

Este resultado afirma a possibilidade de projectar separadamente o controlador e o observador, garantindo que, quando estes são acoplados através da realimentação do estado dada pelo

observador, os pólos da cadeia fechada são os que tinham sido projectados para o controlador e o observador.

3.3.1 Recuperação do Ganho de Malha

O ganho de malha (produto de todas as funções de transferência da cadeia de retroacção) desempenha um papel fundamental no projecto de controladores por realimentação. O projecto do controlador deve “moldar” o ganho de malha por forma a garantir os seguintes aspectos:

- Ganho de malha elevado nas baixas frequências, o que rejeita perturbações e permite o seguimento de referências;
- Ganho de malha reduzido nas altas frequências para garantir rejeição do ruído nos sensores e a condição de estabilidade face aos erros de modelação;
- Nas vizinhanças dos 0 dB a taxa de variação do ganho de malha com a frequência não pode ser muito elevada (em módulo) para não causar instabilidades, garantindo que o sistema tem margens de estabilidade (de ganho e de fase) adequadas.

Quando se faz directamente uma realimentação dos estados, o controlador LQ proporciona margens de estabilidade relativamente boas (por exemplo, margem de ganho infinita e margem de fase de 60°).

Contudo, quando se utiliza a estimativa do estado, dada por um observador, para efectuar a realimentação, as garantias de estabilidade desaparecem por completo. Dependendo das matrizes de coavariância dos ruídos no modelo, Q_E e R_E , as margens de estabilidade relativa do sistema podem ser quase nulas.

Para contornar este problema, usa-se uma técnica denominada Loop Transfer Recovery (LTR), ou recuperação do ganho de malha, na qual as matrizes Q_E e R_E são escolhidas não com base nas covariâncias dos ruídos (que são difíceis de determinar), mas sim como botões de ajuste, de tal forma que o ganho de malha do controlador LQG aproxima o ganho de malha do controlador LQ.

3.3.2 Seguimento de Referência

Uma referência $\text{ref}(k)$ (o valor que se pretende colocar na saída do sistema $y(k)$), é introduzida no sistema através da lei de controlo:

$$u(k) = N\text{ref}(k) - K_c \hat{x}(k) \quad (25)$$

N é um escalar tal que o ganho estático da saída, i.e. o ganho da função de transferência da referência $y_{ref}(k)$ para a saída $y(k)$, seja a própria referência:

$$y_\infty = y_{ref}(k)$$

Como:

$$y_\infty = G_\infty \cdot \text{ref}(k)$$

Para que se obtenha $y_\infty = y_{ref}(k)$ então é necessário que:

$$G_\infty \cdot \text{ref}(k) = y_{ref}(k)$$

Logo:

$$\text{ref}(k) = \frac{1}{G_\infty} y_{ref}(k) = N \cdot y_{ref}(k) \quad (26)$$

O cálculo de N pode ser feito de acordo com as fórmulas:

$$\begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} \Phi - I & \Gamma \\ H & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ I \end{bmatrix} \quad (27)$$

$$N = N_u + K_0 N_x \quad (28)$$

3.3.3 Estimador Corrente

É de se notar que o estimador da equação (12) baseia o seu cálculo no instante de amostragem anterior: é um estimador preditivo. Uma melhor alternativa é definida pela equação:

$$\hat{x}(k) = \Phi \hat{x}(k-1) + \Gamma u(k-1) + K_0 \{y(k) - H[\Phi \hat{x}(k-1) + \Gamma u(k-1)]\} \quad (29)$$

É o chamado estimador corrente, que ao contrário do preditivo, utiliza a informação mais recente para o controlo.

Simplificando a equação tem-se:

$$\hat{x}(k) = (\Phi - K_0 H \Phi) \hat{x}(k-1) + (I - K_0 H) \Gamma u(k-1) + K_0 y(k) \quad (30)$$

Definindo novas variáveis,

$$\Phi_E \triangleq \Phi - K_0 H \Phi \quad (31)$$

$$\Gamma_E \triangleq (I - K_0 H) \Gamma \quad (32)$$

Obtém-se finalmente a equação do estimador utilizado no trabalho laboratorial:

$$\hat{x}(k) = \Phi_E \hat{x}(k-1) + \Gamma_E u(k-1) + K_0 y(k) \quad (33)$$

3.3.4 Modelo em Espaço de Estados Total

Já vimos que o estado total do sistema em malha fechada com o controlador LQG terá $2n$ variáveis de estado, n correspondentes às variáveis de estado do sistema e n correspondentes às suas estimativas.

Encontremos então este modelo.

A equação da entrada de actuação é a equação (25):

$$u(k) = N \text{ref}(k) - K_c \hat{x}(k) \quad (34)$$

As equações de estado do processo são:

$$x(k+1) = \Phi x(k) + \Gamma u(k) \quad (35)$$

$$y(k) = H x(k) \quad (36)$$

Note-se que a matriz D é a matriz nula pelo que não se inclui neste modelo.

Colocando a equação (34) em (35), obtém-se a equação para os primeiros n estados:

$$\mathbf{x}(k+1) = \Phi \mathbf{x}(k) - \Gamma \mathbf{K}_c \hat{\mathbf{x}}(k) + \Gamma N_{\text{ref}}(k) \quad (37)$$

A equação de estado do observador é a equação (33):

$$\hat{\mathbf{x}}(k) = \Phi_E \hat{\mathbf{x}}(k-1) + \Gamma_E u(k-1) + \mathbf{K}_0 y(k) \quad (38)$$

Explicitemos a dependência de $y(k)$, utilizando as equações (36) e (37), sendo que escrevemos esta última equação para o instante $k-1$:

$$y(k) = \mathbf{H}\mathbf{x}(k) = \mathbf{H}[\Phi \mathbf{x}(k-1) - \Gamma \mathbf{K}_c \hat{\mathbf{x}}(k-1) + \Gamma N_{\text{ref}}(k)]$$

Assim, a equação (38) fica:

$$\hat{\mathbf{x}}(k) = \Phi_E \hat{\mathbf{x}}(k-1) + \Gamma_E u(k-1) + \mathbf{K}_0 \mathbf{H} \Phi \mathbf{x}(k-1) - \mathbf{K}_0 \mathbf{H} \Gamma \mathbf{K}_c \hat{\mathbf{x}}(k-1) + \mathbf{K}_0 \mathbf{H} \Gamma N_{\text{ref}}(k)$$

Pegando agora na equação (34) expressa para $k-1$, substitui-se nesta última equação e tem-se:

$$\hat{\mathbf{x}}(k) = \Phi_E \hat{\mathbf{x}}(k-1) + \Gamma_E N r - \Gamma_E \mathbf{K}_c \hat{\mathbf{x}}(k-1) + \mathbf{K}_0 \mathbf{H} \Phi \mathbf{x}(k-1) - \mathbf{K}_0 \mathbf{H} \Gamma \mathbf{K}_c \hat{\mathbf{x}}(k-1) + \mathbf{K}_0 \mathbf{H} \Gamma N_{\text{ref}}(k) \quad (39)$$

Que permite então obter a equação de estado para o observador incorporado no sistema:

$$\hat{\mathbf{x}}(k+1) = \mathbf{K}_0 \mathbf{H} \Phi \mathbf{x}(k) + [\Phi_E - \Gamma_E \mathbf{K}_c - \mathbf{K}_0 \mathbf{H} \Gamma \mathbf{K}_c] \hat{\mathbf{x}}(k) + [\Gamma_E + \mathbf{K}_0 \mathbf{H} \Gamma] N_{\text{ref}}(k) \quad (40)$$

A equação de saída do sistema total é a equação do modelo dinâmico do processo, mas agora acrescentamos zeros à matriz nas posições correspondentes às estimativas dos estados.

Assim, o modelo em espaço de estados do sistema completo ($r \rightarrow y$, i.e., referência para posição angular) é:

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \hat{\mathbf{x}}(k+1) \end{bmatrix} = \begin{bmatrix} \Phi & -\Gamma \mathbf{K}_c \\ \mathbf{K}_0 \mathbf{H} \Phi & \Phi_E - \Gamma_E \mathbf{K}_c - \mathbf{K}_0 \mathbf{H} \Gamma \mathbf{K}_c \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \hat{\mathbf{x}}(k) \end{bmatrix} + \begin{bmatrix} \Gamma \\ \Gamma_E + \mathbf{K}_0 \mathbf{H} \Gamma \end{bmatrix} N_{\text{ref}}(k) \quad (41)$$

$$y(k) = \begin{bmatrix} \mathbf{H} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \hat{\mathbf{x}}(k) \end{bmatrix} \quad (42)$$

4 Projecto do Controlador LQG

Como referido na secção anterior, o projecto do Sistema de Controlo baseia-se essencialmente num Controlador LQG, que é constituído por um Controlador LQ suportado por um estimador.

A obtenção do Controlador LQG consiste na determinação dos parâmetros R e Q do Controlador LQ e dos parâmetros R_v e Q_w do Observador. Em adição é ainda necessário inserir um ganho que compense o Ganho Estático do sistema em Malha Fechada. Esses passos vão ser explicados em detalhe nesta secção, onde se vai fazer também a ponte para o script gerado em Matlab, *Script de Controlo*, que pode ser consultado no *Apêndice A*, no final deste relatório. No final será ainda apresentado o Sistema Total, i.e., o obtido do fecho da Malha Aberta com o Controlador LQG.

4.1 Controlador LQ

O Controlador LQ resume-se a achar a matriz de ganhos K_c que permite uma resposta tão rápida e estável quanto possível. Para tal, é utilizada a função `dlqr` do MatLab. Para os argumentos desta função, temos apenas de fazer uma simples escolha das variáveis Q e R . Conforme visto na secção teórica, a matriz Q define o peso do sinal de saída. Esta matriz é na verdade uma pseudo-variável pois é constante e está definida como $Q=C^T C$. O peso dado ao sinal de entrada, R , funciona como variável de afinação, cujo valor óptimo é determinado em duas fases: primeiro, vemos em simulação qual o valor que permite a melhor resposta. Depois, em testes com o sistema real, verificamos a influência de R e fazemos possíveis alterações do seu valor. Para obter a matriz K_c é necessário usar a função `dlqr` do Matlab, que permite obter o ganho da seguinte forma:

$$[K, \text{Sol_Ric}, V_P] = \text{dlqr}(A, B, Q, C)$$

Em que :

- K é o ganho de retroacção do Controlador LQ;
- Sol_Ric é a solução da equação de Riccati;
- V_P são os pólos da malha fechada.

Para acharmos o melhor valor da matriz R , implementamos o diagrama de blocos na figura (3).

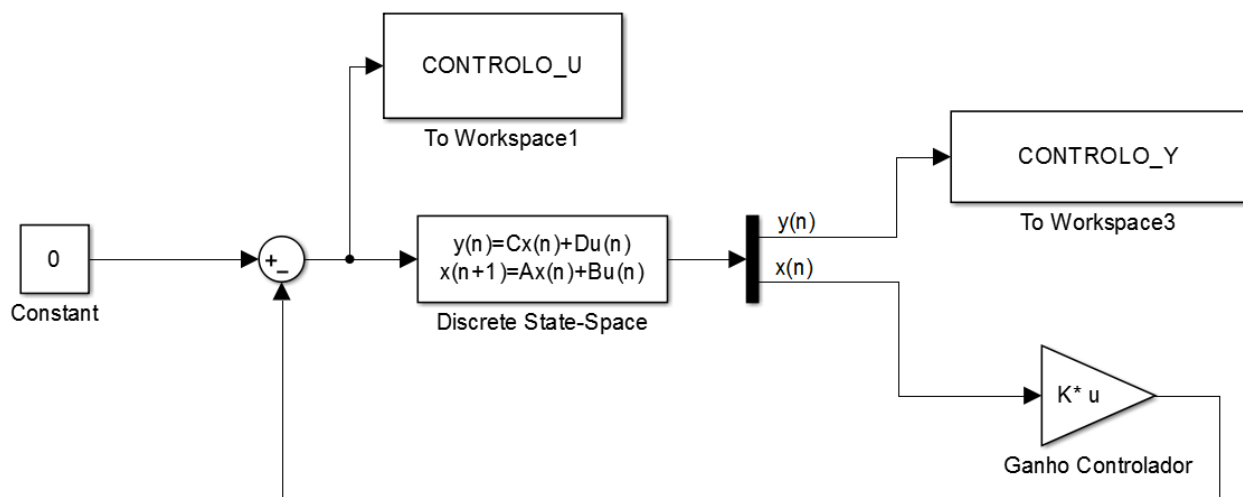


Figura 3: Diagrama de Blocos para determinação do Controlador LQ

Neste diagrama faz-se uma realimentação directa dos estados. Trata-se de um artifício para poder projectar o controlador sem usar um observador, dado que na realidade não dispomos destes estados. Para termos os estados disponíveis à saída do sistema, a matriz de saída (H_C) passa a ser:

$$H_C = \begin{bmatrix} H \\ I \end{bmatrix} \quad (43)$$

Assim, a saída do sistema contém a saída normal e os n estados do sistema.

Fazendo a realimentação destes estados, multiplicando-os pela matriz de ganhos K dado pela função `dlqr`, podemos estudar os efeitos que a variação de R provoca na resposta dinâmica do sistema, submetendo-o a uma entrada nula (com condições iniciais diferentes de zero). Estamos então a testar a rapidez com que o controlador leva a saída para zero.

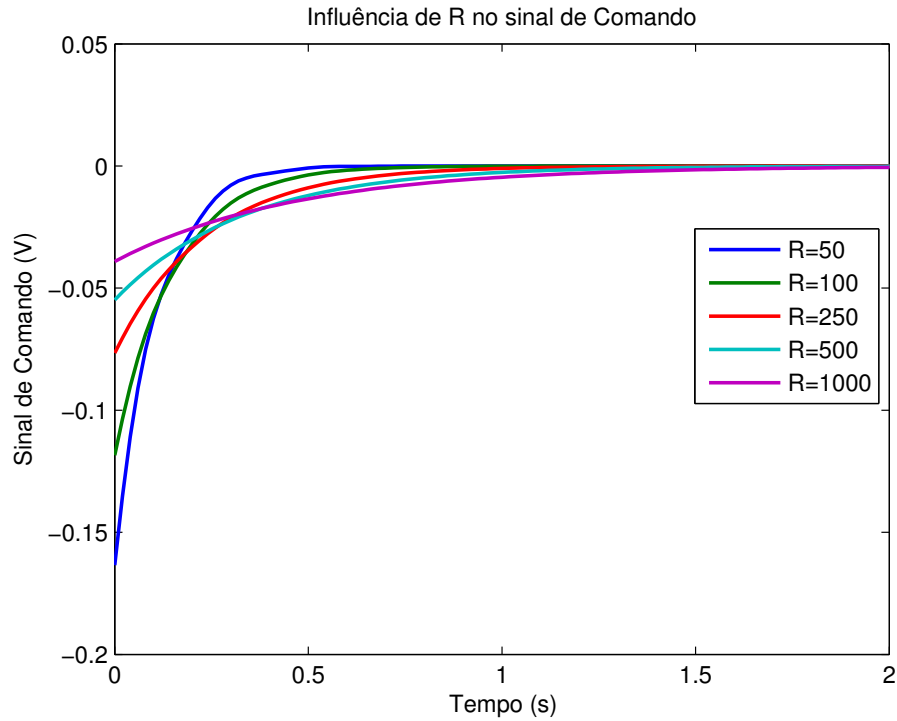


Figura 4: Influência da variação de R no sinal de actuação

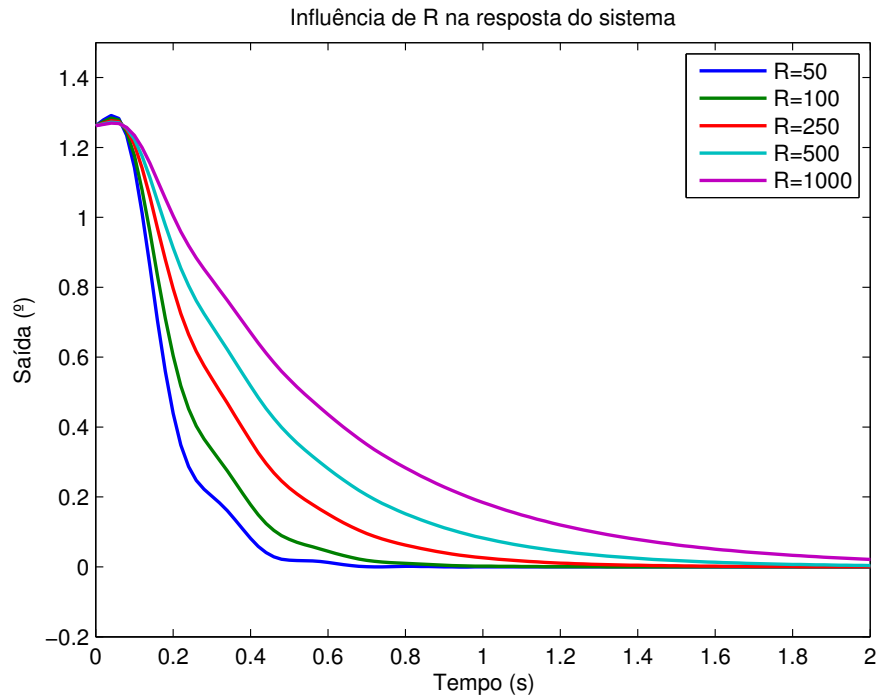


Figura 5: Influência da variação de R na saída

Nestes gráficos verifica-se que:

- Quanto maior o valor de R:
 - Mais lento fica o sistema;
 - Menor o pedido inicial ao actuador;

E pelo contrário:

- Quanto menor o valor de R:
 - Mais rápido fica o sistema;
 - Maior o pedido inicial ao actuador;

Pela análise destes gráficos chegamos à conclusão que o parâmetro R deverá ser superior a 100 e inferior a 500, para que a resposta seja suficientemente rápida para um bom desempenho do sistema, mas garantindo que não há saturação do motor. A decisão final do valor de R fica então adiada para a parte do teste com o sistema real.

4.2 Root Square Locus

Uma ferramenta bastante útil e poderosa para projectar o ganho do controlador que cumpra as especificações para a malha fechada é o Root Locus, ou Lugar Geométrico das Raízes, que permite visualizar graficamente onde ficarão os pólos do sistema em malha fechada quando a saída é realimentada. Permite ainda determinar o ganho que coloca os pólos do sistema em malha fechada nos valores que permitem cumprir as especificações para o sistema.

A aplicação desta ferramenta a sistemas discretos é ligeiramente diferente dos sistemas contínuos, pelo que a seguir se explora como se realizou o Root Locus neste sistema.

Root Locus para Sistemas Discretos

Para um sistema discreto da forma:

$$G(z) = \frac{B(z)}{A(z)}$$

Com A um polinómio de grau n , e cujo sistema de controlo obtém-se através de realimentação de variáveis de estado onde o vector de ganhos do controlador LQ é obtido através da minimização do potencial de custo definido na equação (3), que aqui se repete:

$$J = \frac{1}{2} \sum_{k=0}^{\infty} y^2(k) + Ru^2(k) \quad (44)$$

Esta função de custo é uma particularização do controlador LQ para casos SISO. Consultando a referência [1], página 373, verificamos que a equação (9.39) fica, nestes casos:

$$A(z)A(z^{-1}) + \frac{1}{R}B(z)B(z^{-1}) = 0$$

Note-se que esta equação terá $2n$ raízes, e se existir uma raiz em z_n então existirá certamente uma raiz em z_n^{-1} , isto é, teremos n raízes dentro do círculo unitário, e n raízes fora do círculo unitário.

A equação acima é equivalente à equação:

$$\frac{1}{R} \frac{B(z)B(z^{-1})}{A(z)A(z^{-1})} = -1 \quad (45)$$

Note-se que esta equação apresenta a habitual estrutura do root locus, onde se tem que a multiplicação de um escalar por uma função de transferência resulta em -1. Desta forma, o root square locus resume-se a realizar o habitual root locus ao sistema definido pela função de transferência:

$$G(z^{-1}) = \frac{B(z)B(z^{-1})}{A(z)A(z^{-1})} \quad (46)$$

E onde o ganho K devolvido pelo root locus corresponderá ao inverso do parâmetro R do controlador LQ. Se R for elevado, então $1/R$ será pequeno, logo as raízes óptimas estão perto dos pólos do sistema (ou as reflexões estáveis dos pólos, se $G(z)$ for instável). Se R for reduzido, então $1/R$ é grande, logo as raízes tendem para os zeros de $G(z)G(z^{-1})$.

Do root locus aplicado a esta função de transferência resultarão, obviamente, $2n$ pólos, em que n estarão dentro do círculo unitário e n fora deste. Para o projecto do controlador, apenas interessam os ramos do root locus que estão no interior do círculo unitário, pelo que se podem desprezar os ramos fora deste.

Implementação em MatLab

Iremos então fazer o root locus da função de transferência (46).

Primeiro, começamos por determinar o modelo na forma de função de transferência, fazendo:

```
[B, A] = ss2tf(A, B, C,D) ; B_1 = B ; A_1 = A
```

Precisamos agora de obter os polinómios $B(z^{-1})$ e $A(z^{-1})$. Para tal, invertemos a ordem dos elementos dos vectores que representam os polinómios B e A , pois isto corresponde a multiplicar ambos os polinómios por z^{-n} , sendo n o grau do polinómio A . De seguida, fazemos a multiplicação dos polinómios, multiplicando $B(z)$ por $B(z^{-1})$ e $A(z)$ por $A(z^{-1})$.

```
B_2=fliplr(B) ; A_2=fliplr(A) ; B=conv(B_1,B_2) ; A=conv(A_1,A_2)
```

Voltamos a fazer a transformação do sistema definido pela função de transferência B/A para espaço de estados, em que definimos o tempo de amostragem como 0.02 segundos conforme utilizado no laboratório, e estamos então em condições de chamar finalmente a instrução de root locus:

```
[AA, BB, CC, DD]=tf2ss(B,A) ; sys=ss(AA, BB, CC, DD, 0.02)
```

Com esta sequência de instruções aplicadas ao modelo dinâmico em cadeia aberta do sistema, obtivemos o seguinte root locus, com a instrução `rlocus(sys)`:

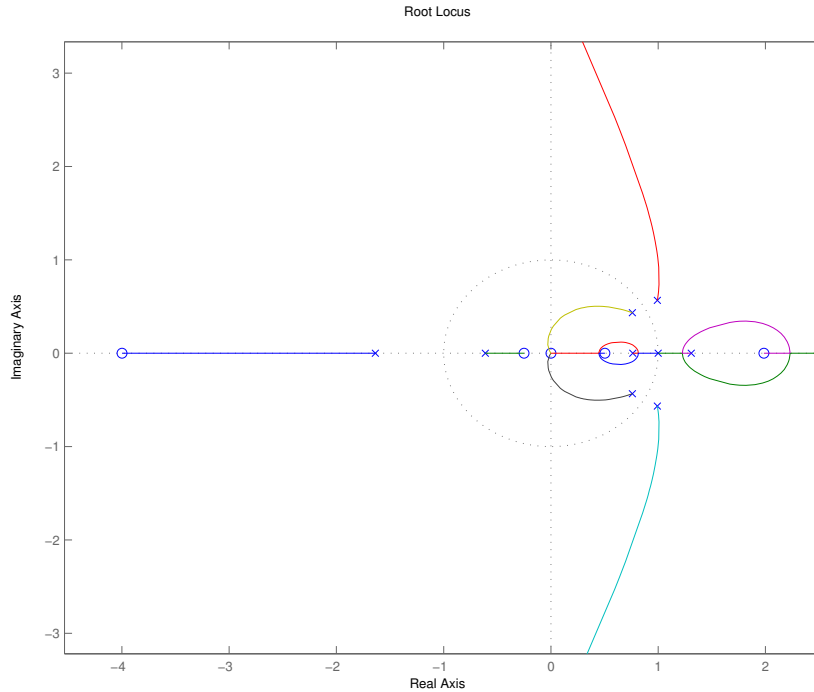


Figura 6: Root Locus do Sistema em Cadeia Aberta

Pela figura (6), verifica-se que o root locus do sistema tem duas partes distintas, dentro do círculo unitário e fora do círculo unitário. Apenas consideramos os ramos dentro do círculo unitário, visto que apenas estes pólos dizem respeito aos pólos do sistema a controlar.

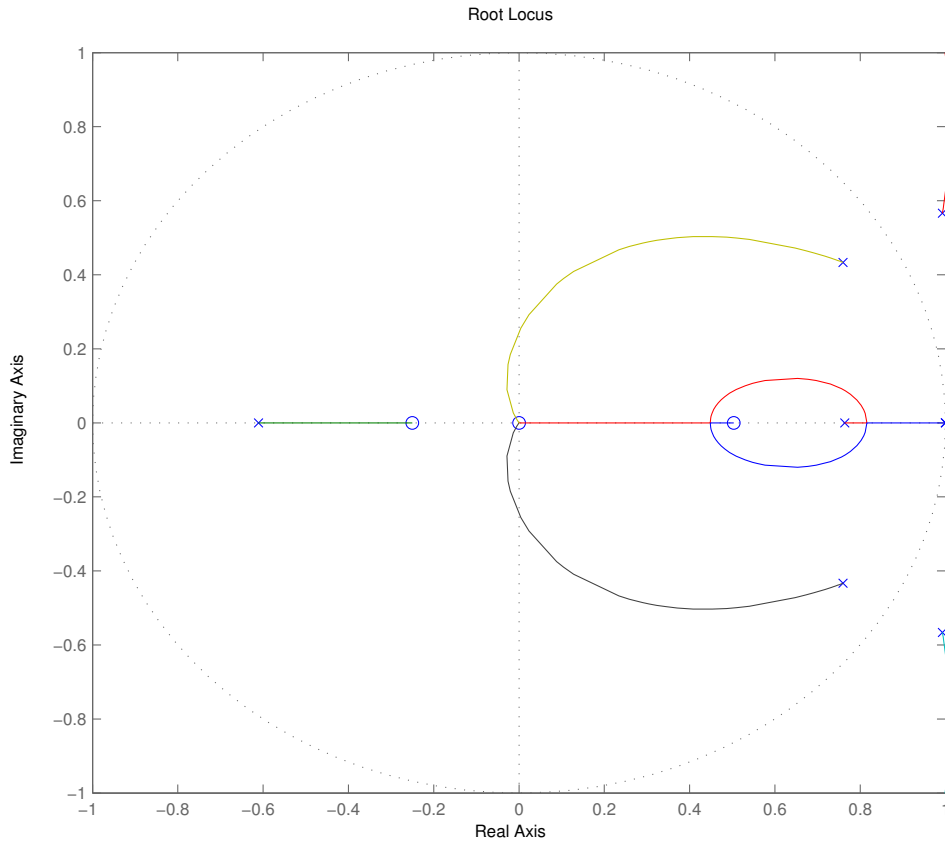


Figura 7: Detalhe do Root Locus do Sistema, apenas com a parte de interesse (o círculo unitário)

Considerando os pólos dentro do círculo unitário, na figura (7), vemos que a sua tendência é deslocarem-se para os zeros do sistema, aproximando-se então da origem, o que corresponde a dinâmicas do sistema mais rápidas e por tanto a melhores desempenhos do sistema.

Seria tentador escolher os pólos perto da origem, contudo tal exigira ganhos demasiado grandes, e pela secção anterior já verificámos que tal é impraticável pois exige demasiado do actuador do motor.

Recorrendo à função do MatLab `rlocfind(sys)`, podemos escolher directamente no gráfico o ponto onde queremos que os pólos da cadeia fechada estejam. Vimos, na secção anterior, que o parâmetro R deverá estar entre 100 e 500, o que corresponde a um parâmetro de ganho no root locus entre 0.01 e 0.002. Estes ganhos são muito reduzidos pelo que os pólos em malha fechada estão quase em cima dos pólos em malha aberta.

4.3 Observador

A design do observador consiste na simples escolha das variáveis Q_w e R_v , conforme visto na secção de fundamentação teórica onde se explorou a técnica de LTR. Para obter o ganho de Kalman é necessário usar a função `dlqe` do Matlab da seguinte forma:

```
[M, Sol_Ric_e, Error_cov, V_P_e] = dlqe(A, eye(5), C, Q_e, R_e)
```

Em que:

- M é o ganho de Kalman;
- Sol_Ric_e é a solução de Riccati;
- $Error_cov$ é a covariância do erro de estimação;
- V_P_e são os pólos da Matriz da Dinâmica da estimação.

Sendo assim, as matrizes de estimação, que vão ser inseridas no bloco de sistema em espaço de estados do observador, podem ser obtidas com:

$$\begin{aligned} A_e &= A - M * C * A \\ B_e &= [M \quad \Gamma_{\gamma_e}] \\ C_e &= eye(5) \\ D_e &= zeros(5, 2) \end{aligned}$$

Para se encontrar o valor das variáveis Q_w e R_v , construiu-se um diagrama de blocos Simulink em que se colocou o estimador de estados a funcionar em paralelo ao processo, e comparou-se a resposta de ambos quando submetidos à mesma entrada. Se o observador estiver bem dimensionado, então a diferença entre os estados reais (saída do processo) e os estados estimados (saída do observador) tem de tender rapidamente para zero. Assim, para testar a dinâmica do processo de observação, construiu-se o diagrama de blocos da figura (8), em que as condições iniciais do sistema e do observador são diferentes de zero (a condição inicial do observador foi 20 e a do processo -30).

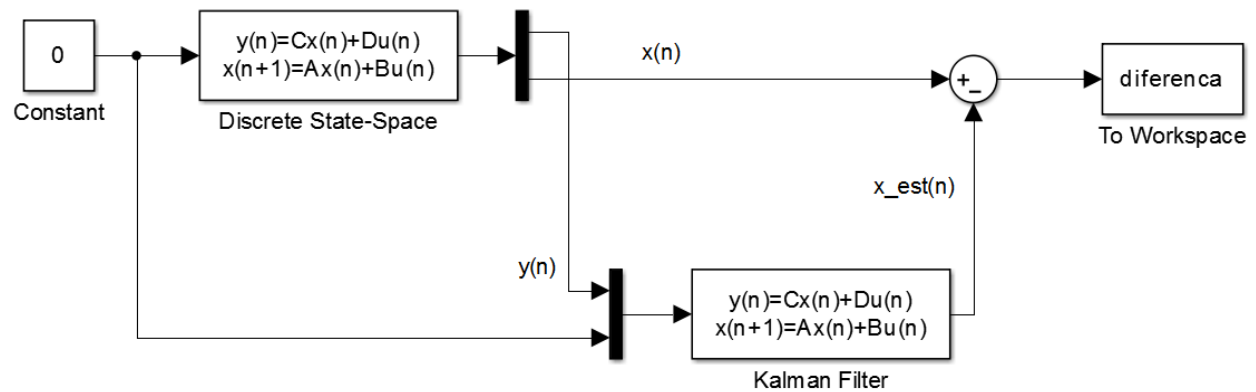


Figura 8: Diagrama de Blocos para determinação dos parâmetros Observador

Começou-se por adoptar os valores sugeridos no guia, $Q_w = 100I$, e $R_v = 1$. De seguida, mantendo o valor de Q_w constante, variou-se o valor de R_v , tendo obtido o gráfico da figura (9) para 5 valores diferentes de R_v .

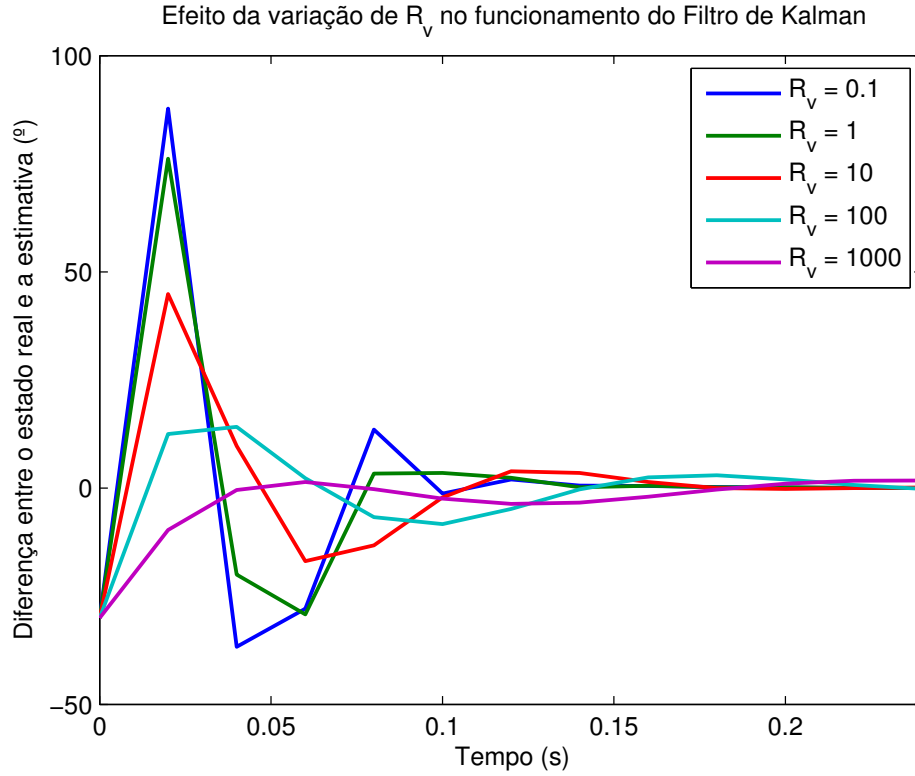


Figura 9: Efeito da variação de R_v na dinâmica do observador, mantendo $Q_w = 100I$; encontra-se representada a evolução da diferença entre o primeiro estado do processo e a estimativa deste

Verifica-se que, quanto maior for o valor da matriz R_v , mais tempo o estado estimado demora a atingir o estado real. Além de demorar mais tempo, verifica-se que a diferença segue uma evolução mais “suave”. Para o nosso objectivo, não nos é relevante a transição suave, apenas estamos interessados em fazer com que o observador chegue ao estado do processo o mais rápido possível, pelo que escolhemos $R_v = 1$, a linha verde na figura (repare-se que $R_v = 0.1$ é pior que $R_v = 1$).

De seguida, com o valor de $R_v = 1$ mantido constante, varia-se o valor da matriz Q , mantendo-a sempre diagonal e com as entradas da diagonal iguais entre si. Obteve-se o gráfico da figura (10) para 5 valores diferentes de Q_w .

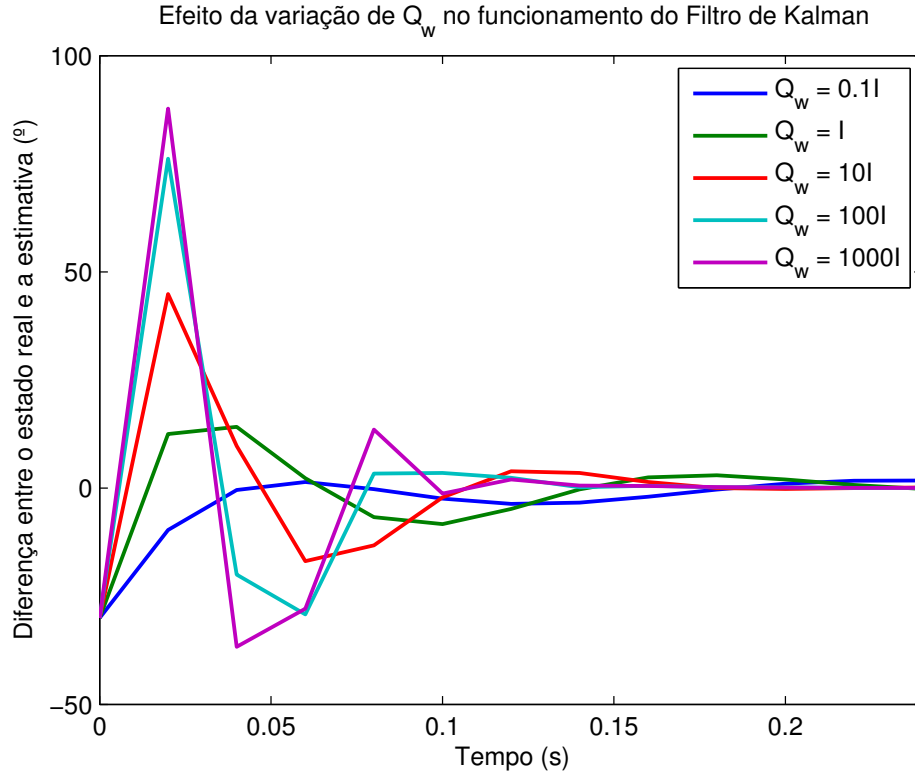


Figura 10: Efeito da variação de Q_w na dinâmica do observador, mantendo $R_v = 1$; encontra-se representada a evolução da diferença entre o primeiro estado do processo e a estimativa deste

Verifica-se que, quanto menor for o valor da matriz Q_w , mais tempo o estado estimado demora a atingir o estado real. Além de demorar mais tempo, verifica-se que a diferença segue uma evolução mais “suave”. Para o nosso objectivo, não nos é relevante a transição suave, apenas estamos interessados em fazer com que o observador chegue ao estado do processo o mais rápido possível, pelo que escolhemos $Q_w = 100I$, a linha azul clara na figura (repare-se que $Q_w = 1000I$ é pior que $Q_w = 100I$).

Assim, o observador a utilizar é o obtido da função `dlqe`, quando invocada com $R_v = 1$ e $Q_w = 100I$.

4.4 Inverso do Ganho Estático

O Ganho Estático é definido como o ganho de malha à frequência zero, i.e. o limite quando o tempo tende para infinito da saída quando a entrada é uma referência constante. É assim necessário multiplicar o sinal de referência pelo inverso deste ganho para que a saída o siga correctamente, como foi visto na secção 3, i.e.:

$$\text{ref}(k) = N \cdot y_{ref}(k)$$

A determinação do ganho N, aqui Nbar, é implementada pelo código seguinte:

```
N = inv([A-eye(5), B; C, 0])*[zeros(5,1);1]
N_x = N(1:5,:)
N_u = N(6,:)
Nbar = N_u+K*N_x
```

Este ganho é introduzido no bloco do sistema inserindo um simples bloco de ganho.

4.5 Sistema Completo

Tendo finalmente obtido:

- Controlador LQ;
- Observador;
- Inverso do Ganho Estático.

Estamos em condições de chegar ao Sistema Completo em Malha Fechada. Para isso é então necessário fazer uso das equações (41) e (42) deduzidas na secção 3.

Sendo assim, o Sistema em Malha Fechada é:

- Espaços de Estados

```
A_SISTEMA = [ A          -B*K;
              M*C*A      Phi_e-M*C*B*K-Gamma_e*K]
B_SISTEMA = [ B ; Gamma_e+M*C*B]*Nbar
C_SISTEMA = [ C 0 0 0 0 0 ]
D_SISTEMA = zeros(1, 1)
SYS=ss(A_SISTEMA, B_SISTEMA, C_SISTEMA, D_SISTEMA, Ts)
```

- Função de Transferência

```
[NUM_SISTEMA, DEN_SISTEMA] = ss2tf(A_SISTEMA, B_SISTEMA, C_SISTEMA, D_SISTEMA)
```

5 Implementação do Controlador LQG

A primeira parte da implementação do Controlador vai ser feita em Simulação, no ambiente Simulink. A segunda é, finalmente, testar o Controlador no Sistema Real.

Com a implementação em ambiente Simulink, pretendemos testar a validade dos parâmetros determinados na secção anterior, bem como melhorar o desempenho do Sistema Total através de afinações dos parâmetros. Tendo a simulação cumprido os objectivos definidos, passamos então para a implementação e teste no Sistema Real.

Na verdade, a passagem do sistema em Simulink para o Sistema Real requereu apenas uma readaptação dos blocos de modo a contornar as não idealidades do Modelo, para cumprir os mesmos objectivos. Finalmente o Controlador foi obtido e a sua influência nas características do Sistema estão discutidas na secção (6), *Características do Sistema*.

Em Simulink implementamos os dois Sistemas, i.e. o Modelo de Simulação e o Sistema Real. Fizemos uso do mesmo Controlador LQR e fizemos a separação em Blocos, aos quais pedimos o seguimento da mesma referência. O referido está exemplificado no seguinte Diagrama de Blocos:

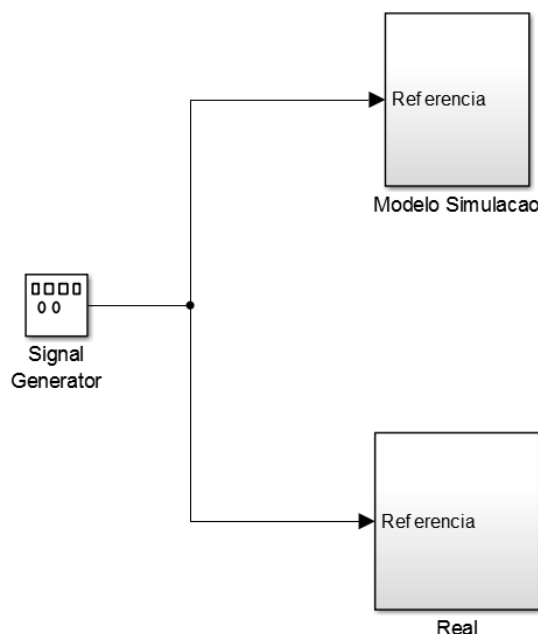


Figura 11: Projecto do Sistema de Controlo - Divisão entre Simulação e Realidade

5.1 Simulação

Nesta subsecção detalha-se a construção do modelo Simulink para teste, em ambiente computacional, do sistema de controlo.

5.1.1 Representação em Simulink

O Controlador foi implementado no Sistema Total em Simulink fazendo uso dos blocos do Controlador LQ e do Estimador dispostos e ligados da forma descrita no Diagrama de Blocos da figura (12).

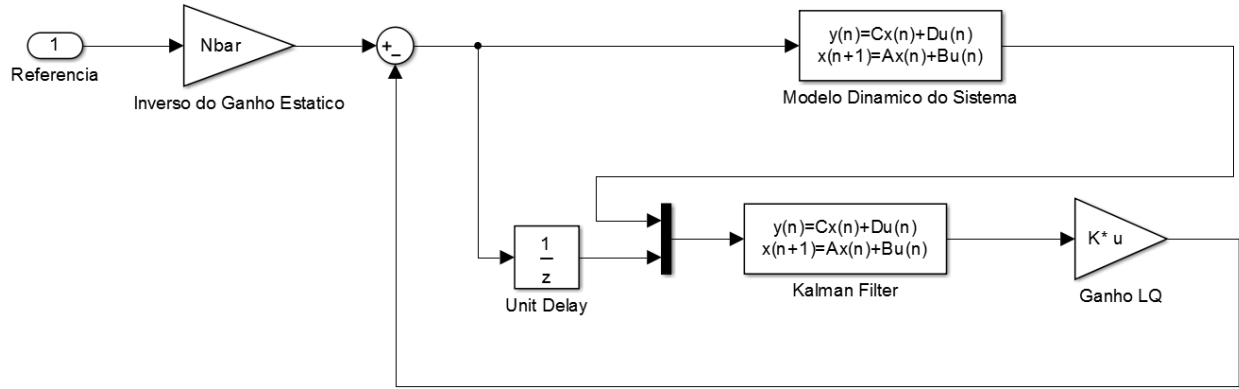


Figura 12: Diagrama de Blocos do Sistema Total para Simulação

Temos como usual o bloco de Modelação, i.e. o bloco *Modelo Dinâmico do Sistema* que contém as matrizes A, B, C e D.

O Controlador está dividido nos blocos:

- *Kalman Filter* que representa o Estimador e é um sistema em espaço de estados composto pelas matrizes de estimação A_e , B_e , C_e e D_e , que recebe a actuação no instante anterior e a saída no instante actual e a fornece o valor estimado dos estados no instante actual.
- *Ganho LQ* que representa o Controlador LQ, que é simplesmente um ganho de retroacção proporcional aos estados.

São inseridos ainda alguns blocos suplementares:

- *Unit Delay*, bloco de atraso unitário que é inserido para fornecer ao Estimador a actuação no instante anterior, em acréscimo à saída no instante actual.
- *Inverso de Ganho Estático*, bloco de ganho que é inserido por forma a que o seguimento estático da referência tenha erro nulo, isto porque o Ganho Estático tem um valor não unitário.

5.1.2 Resultados na Simulação

Conforme determinado anteriormente, os parâmetros do estimador serão $R_v = 1$ e $Q_w = 100I$. Para o Controlador LQ tínhamos visto que $Q = C^T C$ e que R deveria variar entre 100 a 500.

Parâmetro R

Fizemos então testes de Simulação para vários valores da variável de afinação R, desde 10^1 , até 10^3 , com incrementos da ordem de 10^2 . O comportamento do Sistema Total para o R mínimo, intermédio e máximo está representado nas seguintes figuras:

- R = 10

Para este valor de R a resposta é oscilatória com grande sobrelevação mas é rápida o suficiente e seria a desejada. Verifica-se ainda a existência do zero de fase não mínima pois existe uma grande chicotada no fim de cada variação do sinal de referência. No entanto o pedido de actuação é excessivamente exigente. Como tal na prática o motor vai saturar, e a resposta não vai ser como o aqui representado, pelo que não poderá depois ser implementado. O aqui observado vai de encontro ao previsto na secção 4.1. Vimos que para este valor de R o pedido de actuação ia ser demasiado exigente e chegaríamos facilmente à saturação, pelo que o pressuposto de $R > 100$ foi confirmado.

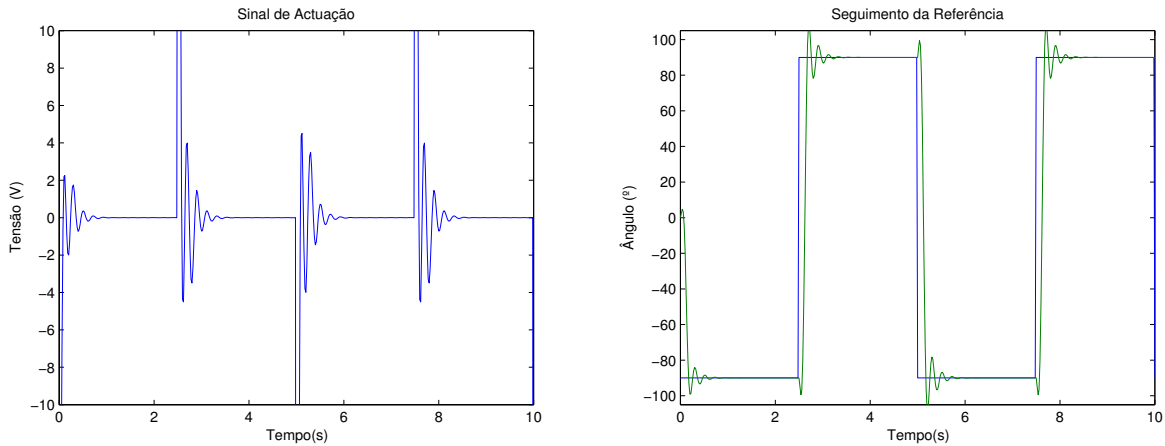


Figura 13: Resposta do sinal de Actuação e Saída para R = 10

- R = 250, R = 400

Para estes valores de R tanto a actuação como a saída estão de acordo com exigido. Isto é, temos uma resposta relativamente rápida, e uma actuação não muito exigente sem sobrecarga do motor. Todos os gráficos representam variações suaves e suficiente rápidas. Temos para ambos os casos tempos de estabelecimento de cerca de um segundo, para um pedido de variação de ângulo de 180° , o que é exagerado mas necessário para efeitos de teste. Verifica-se que a chicotada devido ao zero de fase não mínima é praticamente disfarçada e é praticamente desprezável face ao obtido nos gráficos anteriores para $R=10$. Temos ainda pedidos ao motor nunca superiores a 10V, tensão de saturação do motor, mas que só em curtos períodos de tempo ultrapassam os 6V, tensão que fomos aconselhados a não ultrapassar. Verifica-se então que a conclusão de manter $100 < R < 500$ vai ser respeitada quando partirmos para o modelo real, sendo que os pontos de partida vão ser $R=250$ e $R=400$, forçando R para o interior deste intervalo.

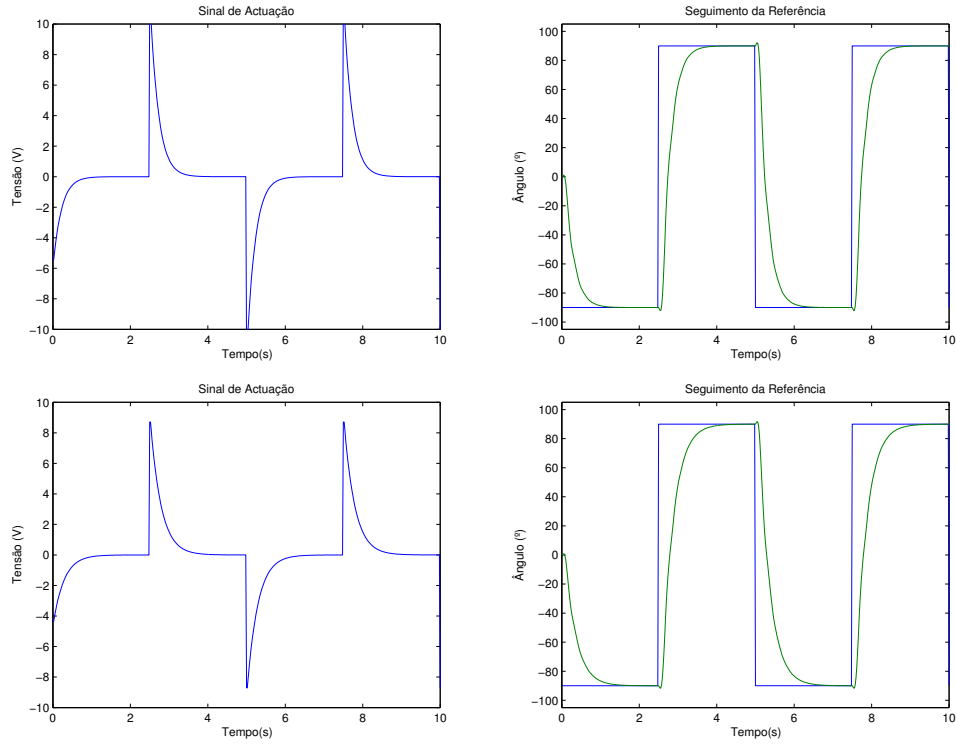


Figura 14: Resposta do sinal de Actuação e Saída para $R = 250$ e $R = 400$

- $R = 1000$

Para este valor de R o pedido de actuação não é tão exigente, é um pedido suave. A resposta é também ela suave e atinge a referência pedida, mas é demasiado lenta. Temos agora que o tempo de estabelecimento é de cerca de dois segundos, o dobro do observado anteriormente. Este tempo de estabelecimento é demasiado longo, pelo que apesar de o pedido ao motor ser bastante bom e até a chicotada praticamente ter desaparecido, este valor de R não poderá depois ser implementado, pois este sistema não vai ter uma dinâmica como a desejada. Confirma-se então o previsto anteriormente, que R necessariamente seria inferior a 500.

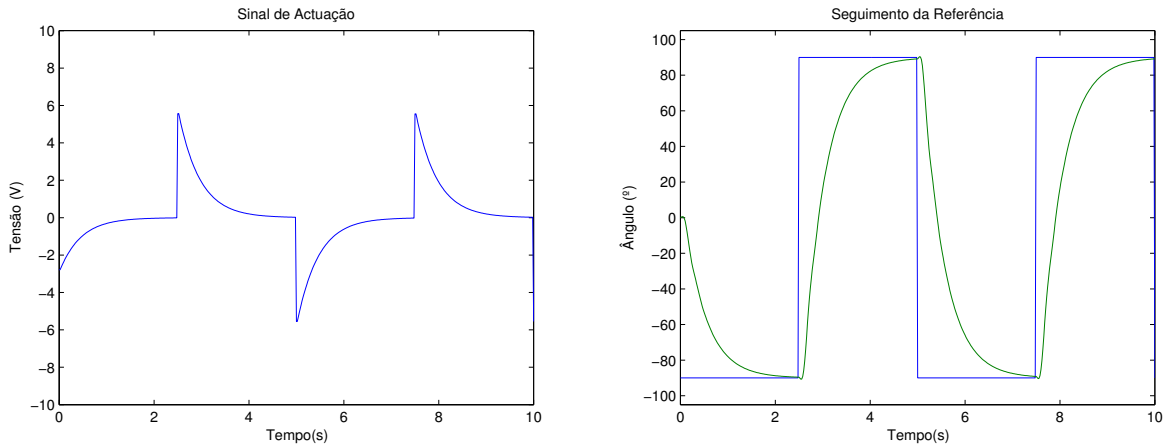


Figura 15: Resposta do sinal de Actuação e Saída para $R = 1000$

5.2 Realidade

Após validação em simulação do controlador LQG, estamos agora em condições de passar para o teste no Sistema Real.

5.2.1 Representação em Simulink

A forma de implementar o controlador no Diagrama de Blocos do sistema Total (Figura 16) foi em tudo semelhante ao da simulação, tendo para isso sido necessária a criação de um subsistema que foi chamado *Sistema Real* (Figura 17), em analogia ao bloco *Modelo Sistema* no Diagrama de Blocos da Simulação (12). Dentro deste bloco estão o conjuntos de Actuadores e Sensores permitem controlar o conjunto Motor + Barra Flexível.

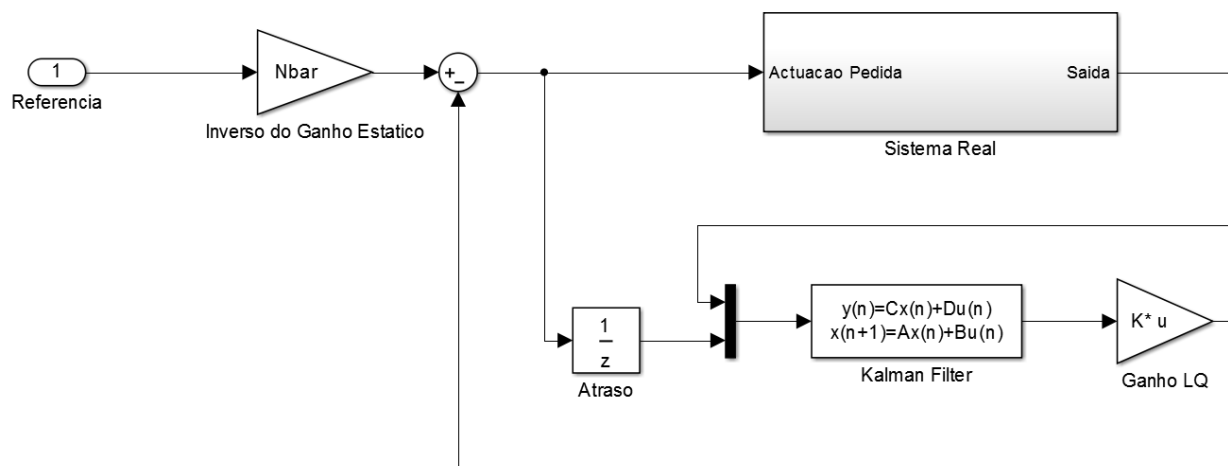


Figura 16: Diagrama de Blocos do Sistema Total para a Realidade

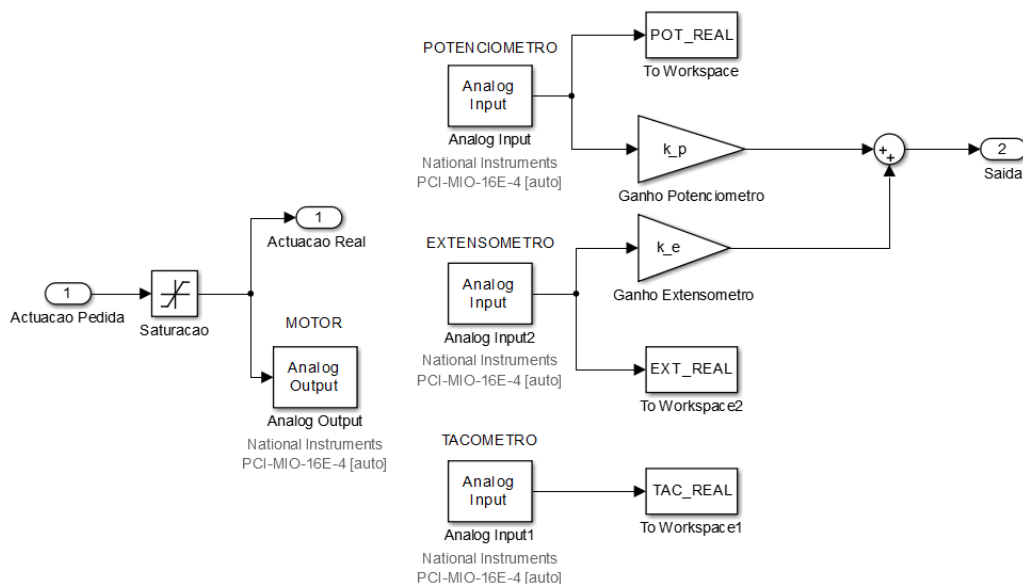


Figura 17: Bloco *Sistema Real* com sensores e actuador

5.2.2 Resultados no Sistema Real

No sistema real há factores determinantes que vão poder afectar o nosso controlador, nomeadamente:

- Saturação no Motor:

Como já implicitamente referido algumas vezes ao longo deste relatório, a saturação no motor é algo crítico. Existe uma tensão máxima de 10V acima dos quais não se podem fazer pedidos ao motor sem risco deste ser danificado. No sistema real optámos por inserir um bloco de saturação (6V) antes de enviarmos o sinal para o Amplificador.

- Efeito de Carga:

O controlo da posição angular da ponta da barra flexível apresenta por norma uma desvio da posição angular desejada como se vê na figura (18). Este efeito é notório com a inclinação da base de montagem relativamente à horizontal, nesta situação a força da gravidade poderá facilitar o movimento da barra para determinados ângulos e dificultar para outros.

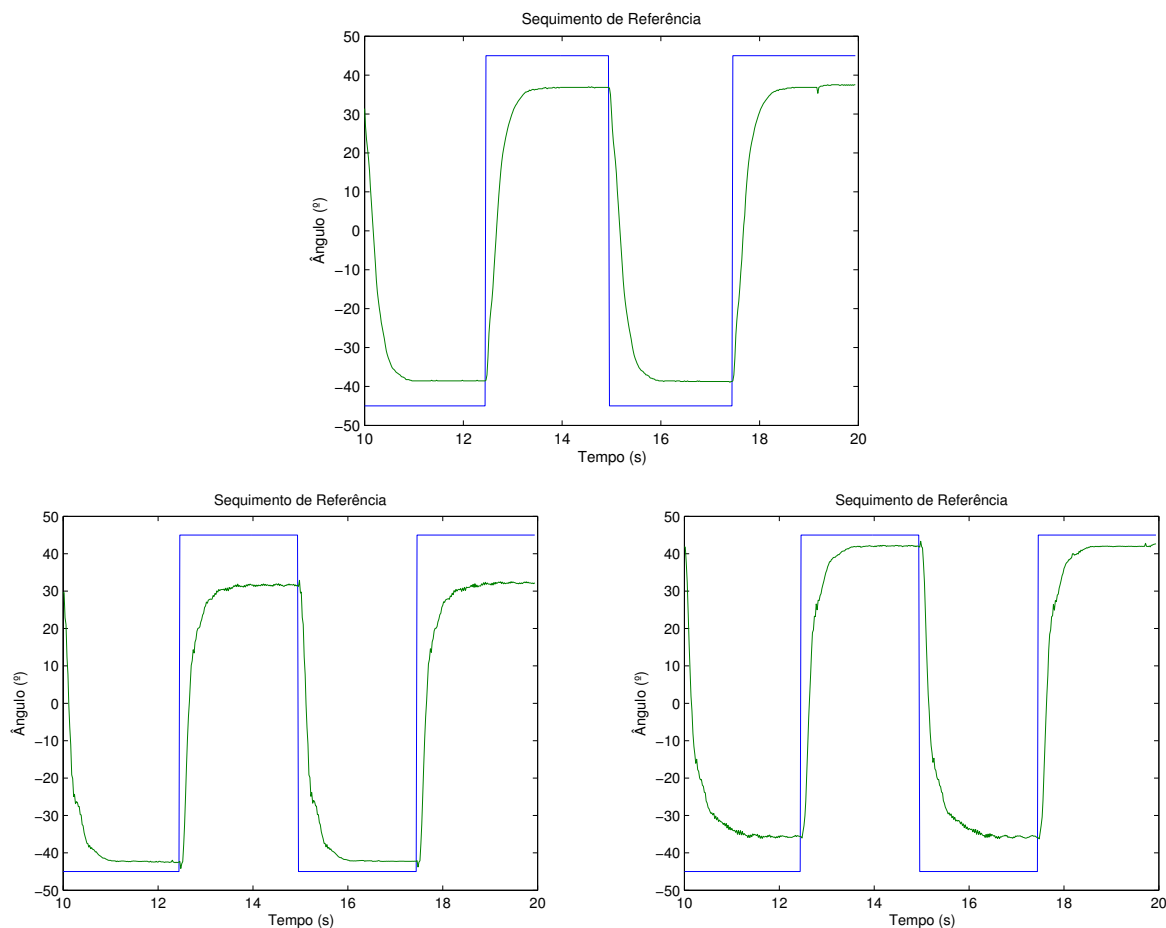


Figura 18: Diferença no offset entre a referência e a saída para o caso horizontal (figura de cima), e quando a montagem está inclinada para um lado e para o outro

- Ruído:

Os sensores apresentam ruído, no entanto o design do sistema de controlo baseia-se na realimentação de estimativas do estado, pois os estados do sistema não estão disponíveis para medição directa. O Estimador recebe os dados da saída e entrada do sistema e produz as estimativas dos estados. Contudo, estes estados são imunes ao ruído que afecta a saída, pois o Observador funciona como um filtro passa-baixo que exclui as componentes de alta frequência do ruído.

Dos dois valores de R escolhidos anteriormente obtivemos dois Controladores LQG diferentes que vão ter respostas diferentes. O primeiro **LQG1**, para $R=250$ pressupõe um maior pedido de actuação mas também uma maior rapidez de resposta a um pedido de seguimento de referência. Do segundo, **LQG2** com $R=400$ espera-se o contrário. Para os dois valores de R foram analisados essencialmente 4 tópicos:

- Resposta do Sinal de Actuação.
- Resposta do Sinal de Saída em Relação à Referência.
- Potência do Sinal de Entrada.
- Potência do Sinal de Saída subtraído da Referência (Erro).

• Resposta do sinal de Actuação e da Saída em Relação à Referência

Para decidirmos qual o melhor valor de R , analisamos o que acontece nos dois controladores, LQG1 e LQG2 e comparamos as duas respostas.

Facilmente se observa que esta resposta é em tudo semelhante ao previsto em simulação, para os dois controladores, à excepção do offset já discutido. Temos tempos de estabelecimento de cerca de 1 segundo, o que era o desejado, mesmo para um pedido de variação de ângulo tão exigente como são os 180° . Dificilmente se consegue distinguir com esta resposta qual o melhor controlador, as respostas são parecidas apesar de o LQG1 ser ligeiramente mais rápido.

São apresentadas os sinais de actuação (à esquerda), e a saída comparada com a referência (à direita).

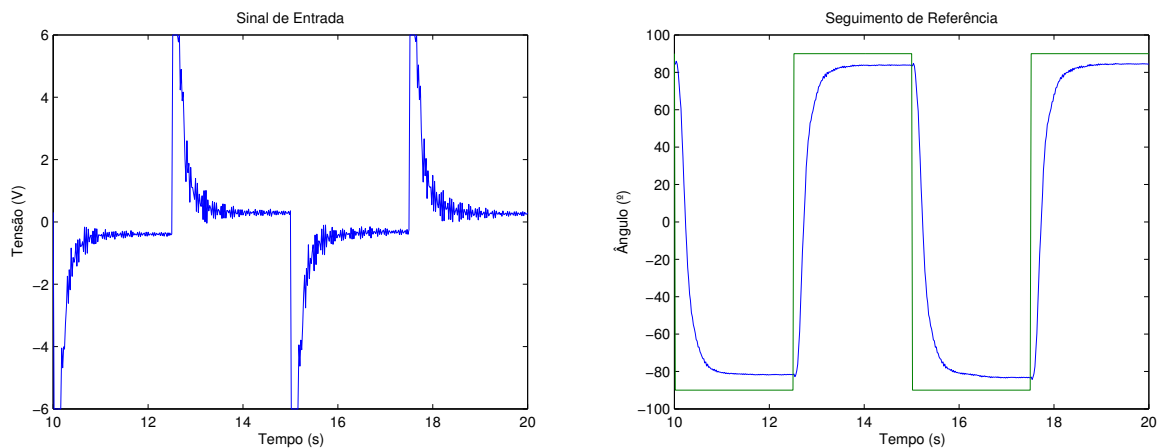


Figura 19: Resposta do Sinal de Actuação e Saída para $R = 250$

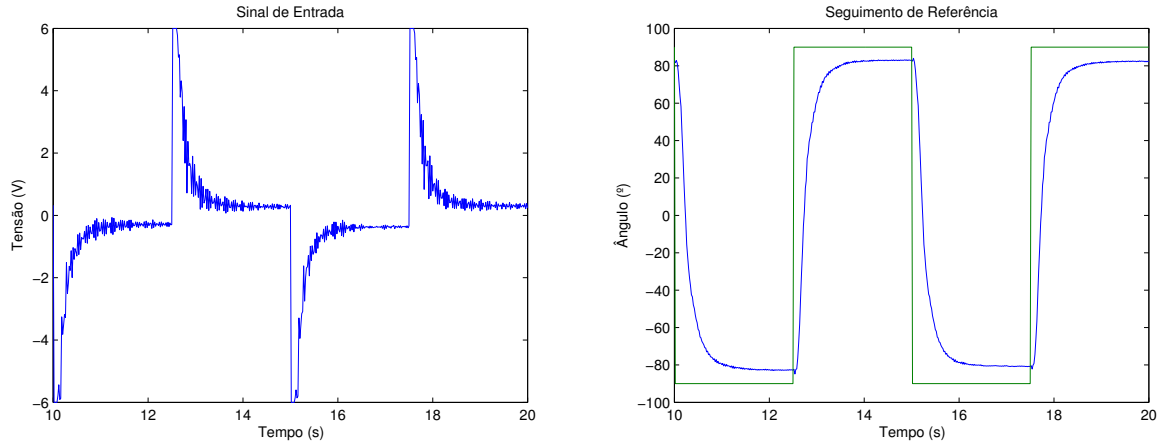


Figura 20: Resposta do Sinal de Actuação e Saída para $R = 400$

Findada a análise da resposta no tempo quer do sinal de actuação quer do sinal de saída, prevê-se que o LQG2 ($R = 400$) seja ligeiramente melhor, dado que exige ligeiramente menos dos actuadores e apresenta uma resposta de saída em tudo semelhante à do LQG1. Como as respostas são muito semelhantes, realizamos uma análise ainda mais detalhada.

• Potência do Sinal de Entrada

Para análise da Potência do sinal de entrada estão aqui apresentados os resultados não só obtidos no sistema real mas também em simulação.

Em simulação vê-se que a potência pedida ao actuador LQG2 é claramente menor. O pico para LQG1 é $120V^2$ (i.e. um pedido máximo de 11V) e para o LQG2 é $80V^2$ (i.e. um pedido máximo de 9V). Para o LQG1 poderá haver dano no motor, por se encontrar acima dos 10V. Isto era expectável pois quanto menor o valor de R mais exigente é o pedido à entrada.

Mas na realidade tal não acontece. Fruto de inserirmos o bloco de saturação no motor a potência máxima pedida em ambos é igual e de $36V^2$ (i.e. um pedido máximo de 6V). Verifica-se que as curvas estão praticamente sobrepostas. Em todos os instantes a curva do LQG1 está por cima mas é tão pouco que é desprezável em termos de potência de entrada.

Sendo assim que o factor de análise determinante vai ser o seguinte, a potência do Erro.

• Potência do Erro

A potência do Erro é a potência da diferença entre a saída e a referência. Esta é uma característica bastante interessante pois permite saber com que rapidez a saída tende para o desejado, ou seja, a potência do Erro tende para zero.

Em simulação vê-se claramente que o controlador LQG1 é melhor pois o erro cai para zero mais rapidamente. Há um pico em cerca de $8000^{\circ 2}$ (i.e. um erro de cerca de 90°) nos dois controladores, sendo o pico no LQG1 mais acentuado pelo facto de ser mais rápido. Isso não é tão importante pois a rapidez aqui é a chave.

O pico de erro máximo não é logo aos 15 ou 17.5 segundos pois estamos a considerar ângulos sempre em módulo. Imagine-se no primeiro caso que a barra está em -90° : Como a nova referência pedida passa a ser 90° , o erro entre o módulo do ângulo e a referência é nulo. Quando a barra passa pelos 0° atinge-se o erro máximo, de 90° a que corresponde uma

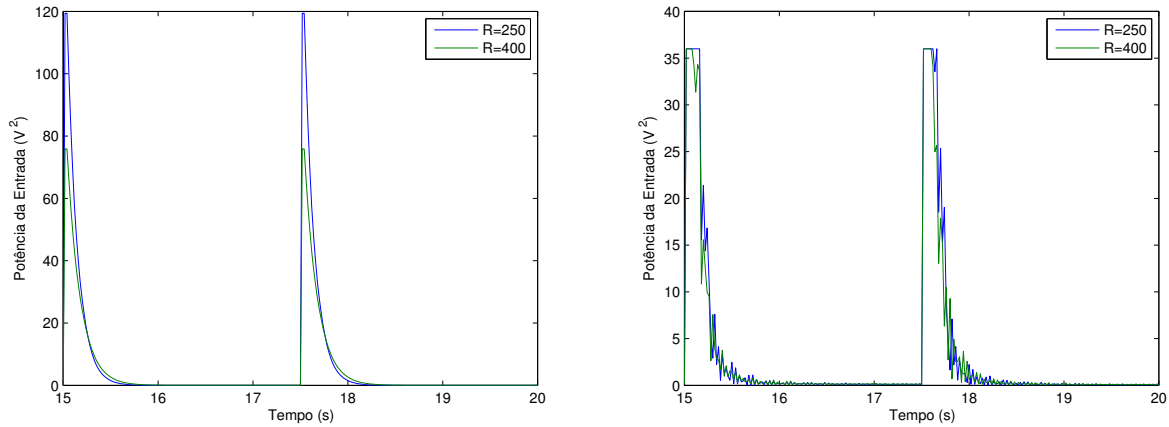


Figura 21: Potência da Entrada, à esquerda em simulação e à direita na realidade

potência de $8100^{\circ 2}$. Depois a barra continua o seu movimento e o erro decresce de novo até serem atingidos os 90° pedidos. Nesse caso o erro é de novo zero. Isso comprova que o pico acentuado não é muito importante, o mais importante é a segunda vez que a potência desce para zero e a rapidez com que isso acontece. Repare-se que na imagem estão representados dois pedidos, pelo que há dois picos diferentes.

Quando passamos para a realidade continua a confirmar-se que o LQG1 é melhor. O pico dos dois é agora semelhante, e continua a ver-se que o LQG1 é mais rápido. Note-se que agora a potência do Erro não tende para zero, dado o offset que vimos existir em relação à referência.

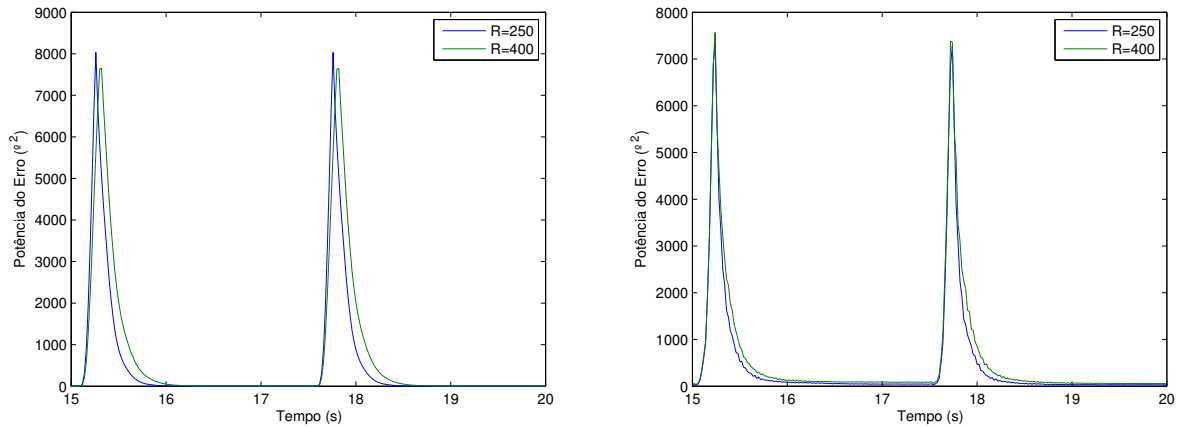


Figura 22: Potência do Erro, à esquerda em simulação e à direita na realidade

Finalmente podemos escolher o melhor controlador. Em termos de Potência da Entrada as respostas são semelhantes sendo que o LQG2 era ligeiramente melhor. Em relação à Potência do Erro observa-se que o LQG1 é consideravelmente melhor.

Como tal a escolha final de Controlador vai ser o LQG1, ou seja, um controlador composto por um Controlador LQ com $R = 250$ e $Q = 1$, assistido por um Estimador com $R_v = 1$ e $Q_w = 100I$.

Com este Controlador LQG podemos retirar agora algumas características finais acerca do Sistema Total de Controlo.

6 Características do Sistema

Obtido o sistema de controlo, efectua-se uma análise detalhada das características dinâmicas do sistema controlado.

6.1 Sistema em Malha Fechada

Como vimos na secção de fundamentação teórica o novo sistema, composto pelos estados mais os estados estimados é dado por:

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \hat{\mathbf{x}}(k+1) \end{bmatrix} = \begin{bmatrix} \Phi & -\Gamma\mathbf{K}_c \\ \mathbf{K}_0\mathbf{H}\Phi & \Phi_E - \Gamma_E\mathbf{K}_c - \mathbf{K}_0\mathbf{H}\Gamma\mathbf{K}_c \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \hat{\mathbf{x}}(k) \end{bmatrix} + \begin{bmatrix} \Gamma \\ \Gamma_E + \mathbf{K}_0\mathbf{H}\Gamma \end{bmatrix} N \cdot \text{ref}(k)$$

$$\mathbf{y}(k) = \begin{bmatrix} \mathbf{H} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \hat{\mathbf{x}}(k) \end{bmatrix}$$

E que podemos simplificar obtendo o novo Sistema Discreto em Espaço de Estados, com as matrizes da dinâmica, de entrada e de saída:

$$\mathbf{x}(k+1) = \Phi^* \mathbf{x}^*(k) + \Gamma^* \text{ref}(k) \quad (47)$$

$$\mathbf{y}(k) = \mathbf{H}^* \mathbf{x}^*(k) \quad (48)$$

Em que:¹

$$\Phi^* = \begin{bmatrix} 2.67 & -2.20 & -0.06 & 0.95 & -0.36 & -0.08 & 0.14 & -0.04 & -0.05 & 0.03 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2.98 & 8.60 & 0.20 & -3.02 & 1.14 & 5.57 & -10.67 & -0.30 & 3.92 & -1.47 \\ -1.97 & 5.69 & 0.13 & -2 & 0.75 & 2.97 & -5.69 & -0.13 & 2 & -0.75 \\ -0.98 & 2.85 & 0.07 & -1 & 0.37 & 0.98 & -1.84 & -0.07 & 1.00 & -0.38 \\ -0.47 & 1.36 & 0.03 & -0.48 & 0.18 & 0.47 & -1.36 & 0.97 & 0.48 & -0.18 \\ -0.22 & 0.62 & 0.01 & -0.29 & 0.08 & 0.22 & -0.62 & -0.01 & 1.22 & -0.08 \end{bmatrix} \quad (49)$$

$$\Gamma^* = \begin{bmatrix} 0,0607 \\ 0 \\ 0 \\ 0 \\ 0,0607 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (50)$$

$$\mathbf{H}^* = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (51)$$

$$\mathbf{x}^*(k) = \begin{bmatrix} x(k) \\ \hat{x}(k) \end{bmatrix} \quad (52)$$

¹Foram reduzidas casas decimais na matriz Φ^* para apresentação no relatório

6.2 Função de Transferência do Sistema em Malha Fechada

Tendo o sistema em espaços de estado facilmente se chega à função de transferência do mesmo:²

$$G_d(z) = \frac{-0.0062z^9 + 0.0126z^8 + 0.0009z^7 - 0.0028z^6 - 0.0006z^5 + 0.0001z^4}{z^{10} - 2.89z^9 + 2.67z^8 - 0.07z^7 - 1.28z^6 + 0.56z^5 + 0.07z^4 - 0.05z^3 - 0.01z^2 + 0.01z + 0.01} \quad (53)$$

Verifica-se que o sistema é de ordem 10 e tem um atraso puro, o qual era o expectável pois o sistema em Malha Aberta era de ordem 5 e o Observador introduz mais 5 estados, estas estimativas dos iniciais. O atraso do sistema mantém-se e é o atraso do sistema em Malha Aberta.

Representando os Pólos e os Zeros do Sistema obtém-se:

Pólos [rad/s]	Zeros [rad/s]
$-0.1168 + 0.1527i$	$-0.1168 + 0.1527i$
$-0.1168 - 0.1527i$	$-0.1168 - 0.1527i$
-0.2767	-0.2500
-0.6110	-0.2767
0.2488	0
0.5588	0
$0.7580 + 0.4342i$	0.2488
$0.7580 - 0.4342i$	0.5588
0.7702	1.9863
0.9163	

Figura 23: Pólos e Zeros do Sistema em Malha Fechada

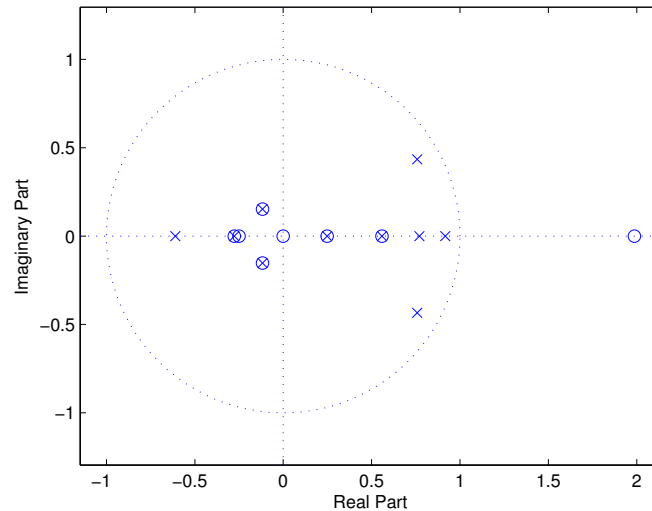


Figura 24: ZPlane dos Pólos e Zeros do Sistema em Malha Fechada

²Foram reduzidas casas decimais do denominador para apresentação no relatório

Facilmente se verifica que existem 5 pólos que cancelam com 5 zeros. Estes pólos/zeros são os que resultam do facto de introduzirmos o Estimador, que acrescenta mais 5 estados ao sistema. Naturalmente os pólos/zeros acrescentados teriam de se cancelar pois a ordem do sistema inicial não se poderia alterar. Os estados estimados não são “verdadeiros” estados, são apenas uma estimativa dos estados iniciais, i.e. não acrescentam nada à dinâmica do mesmo.

Sendo assim, resulta que o Sistema tem os seguintes Pólos e Zeros não cancelados:

Pólos [rad/s]	Zeros [rad/s]
-0.6110	-0.2500
$0.7580 + 0.4342i$	0
$0.7580 - 0.4342i$	0
0.7702	1.9863
0.9163	

Figura 25: Pólos e Zeros não cancelados do Sistema em Malha Fechada

Estes pólos e zeros podem ser comparados com os obtidos em Malha Aberta, que eram:

Pólos [rad/s]	Zeros [rad/s]
-0.6110	-0.2500
$0.7599 + 0.4334i$	0
$0.7599 - 0.4334i$	0
0.7638	1.9863
1	

Figura 26: Pólos e Zeros do Sistema em Malha Aberta

Verifica-se que quase todos os pólos não sofrem alterações significativas e os zeros são os mesmos.

A excepção é o pólo instável devido à existência do integrador (ou criticamente estável, o que para efeitos de controlo é o mesmo). Conseguimos contornar esse problema e puxar o pólo para dentro do círculo unitário, mantendo assim o Sistema estável em Malha Fechada. Quanto mais para dentro do círculo unitário o pólo estivesse mais rápido seria o sistema, no entanto isso iria exigir demasiado dos actuadores. Note-se que assim este pólo está sensivelmente à mesma distância da origem que os pólos complexos conjugados pelo que a constante de tempo do sistema total é dominada por ambos. Como tal se puxássemos o pólo ainda mais para dentro não teria grande influência em termos de rapidez pois os complexos passariam assim a dominar mas estaríamos a saturar os actuadores dada a diminuição de R.

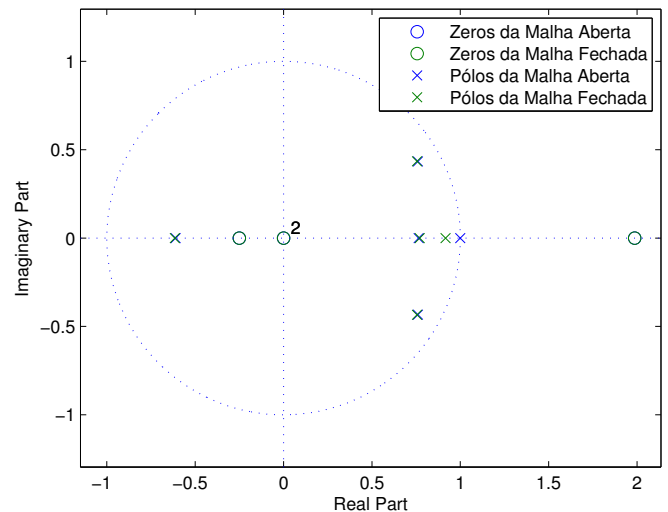


Figura 27: Pólos e Zeros não cancelados do Sistema em Malha Aberta e Fechada

Assim a função de Transferência obtida anteriormente (53) pode ser simplificada para:

$$G_{d_{MF}} = \frac{z^4 - 1.7363z^3 - 0.4966z^2}{z^4 - 2.5915z + 2.0690z^3 + 0.1026z^2 - 0.9014z + 0.3290} \quad (54)$$

A qual pode ser comparada com a Função de Transferência obtida em Malha Aberta:

$$G_{d_{MA}} = \frac{z^4 - 1.7363z^3 - 0.4966z^2}{z^4 - 2.6727 - 2.2036z^3 + 0.0613z^2 - 0.9494z + 0.3571} \quad (55)$$

O numerador da função de transferência é igual, enquanto que o denominador muda como consequência da alteração dos pólos pela realimentação das variáveis de estado.

6.3 Resposta no Tempo

Fazendo agora a análise e comparação entre a Resposta no Tempo entre o Sistema em Malha Aberta e em Malha Fechada verifica-se fisicamente o pretendido, i.e. o sistema é estável. Antes, quando existe um pedido unitário de 1° (step unitário) o sistema não responde e fica instável, aumentando o ângulo para infinito aproximadamente à velocidade de \hat{K}_t (Vide Parte 1). Agora o sistema evolui como queremos dada a introdução do Controlador LQG, nomeadamente segue rapidamente a referência com um erro estático nulo.

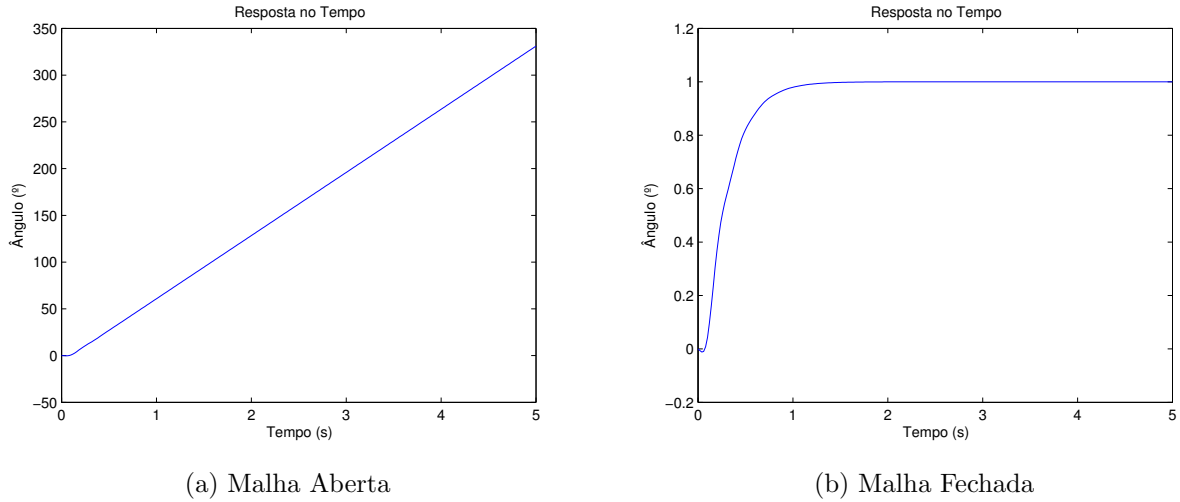


Figura 28: Resposta no tempo a um Step unitário em Malha Aberta e Fechada

Analisando agora apenas a Malha Fechada e em particular alguns aspectos importantes temos:

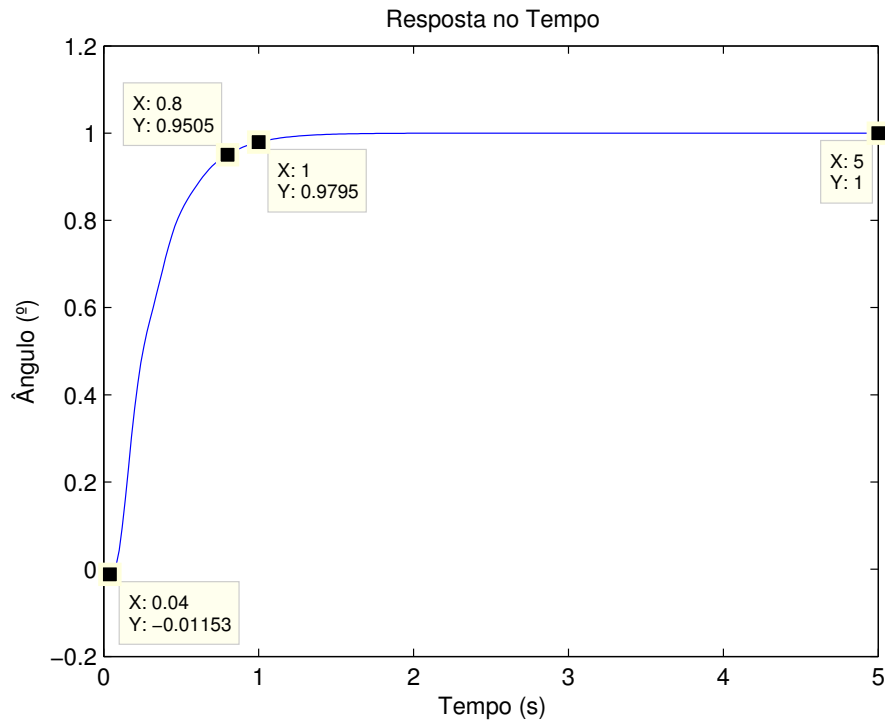


Figura 29: Resposta no tempo a um Step unitário em Malha Fechada

Tempo de Estabelecimento:

Temos que o tempo de estabelecimento é, 0.8 e 1.0 segundos a 95% e 98% respectivamente. Assim o tempo de estabelecimento é bastante reduzido e cumpre os requisitos pressupostos.

Ganho Estático:

Verifica-se que o ganho estático é unitário, i.e. o sistema segue a referência com erro estático nulo, como desejado, e também sem sobrelevação.

Chicotada Inicial:

Dado que o Sistema é de fase não mínima então existe uma chicotada inicial, mas que é praticamente abafada, acontece o seu máximo aos 0.04 segundos (apenas dois instantes de amostragem) e é de apenas 1%.

6.4 Resposta em Frequência

Para a análise da resposta em Frequência retiramos do sistema o ganho inverso ao ganho estático, pois trata-se de um simples ganho que poderia induzir em erro a análise (faria uma translação do gráfico da magnitude para baixo). Assim podemos obter o seguinte Diagrama de Bode:

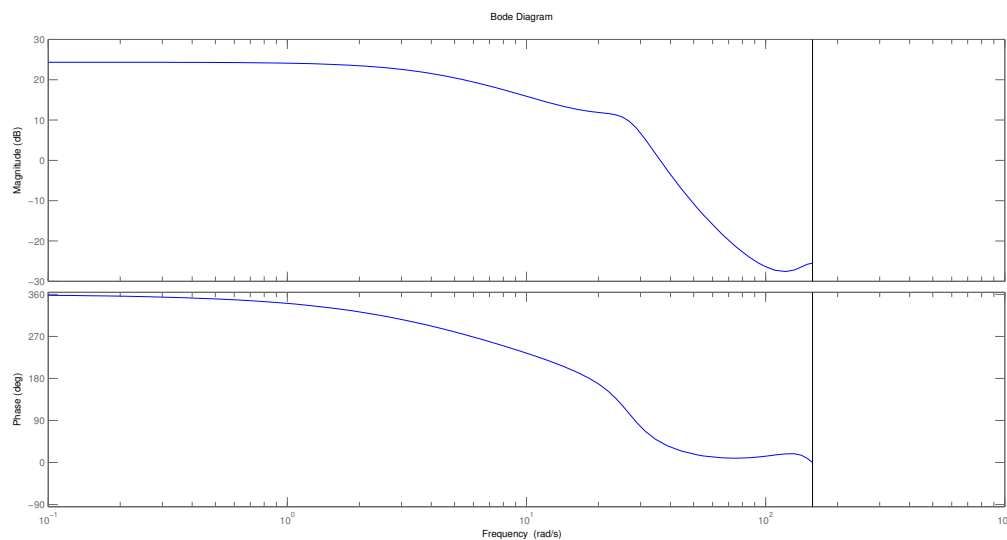


Figura 30: Diagrama de Bode do Sistema em Malha Fechada

Verifica-se que o sistema tende para um ganho alto de 24 dB a frequências baixas (à frequência “0”, sinais constantes, é o ganho estático em dB), filtrando assim perturbações. Às altas frequências o Sistema tende para um ganho bastante baixo, isto para filtrar o ruído.

Podemos agora fazer a comparação entre o obtido em Malha aberta e em Malha Fechada.

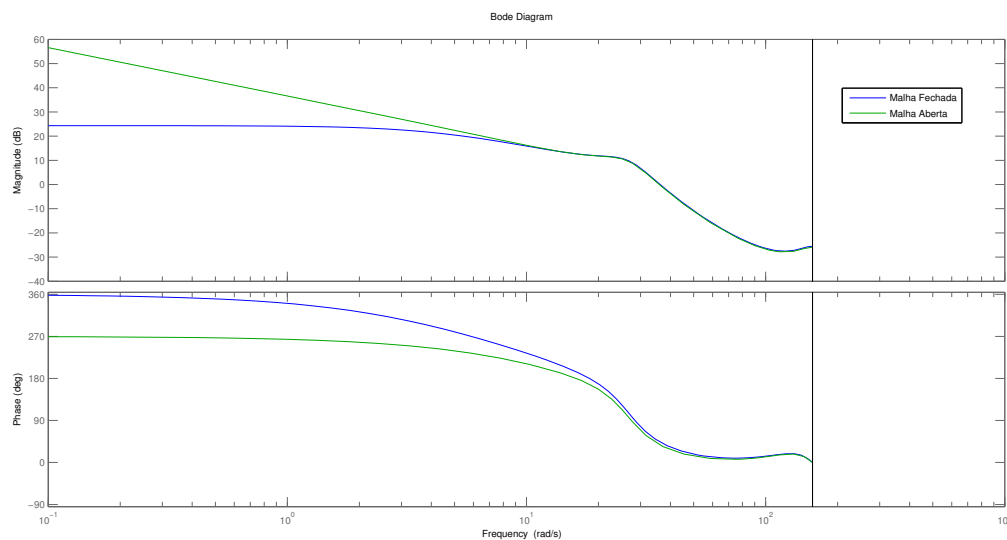


Figura 31: Diagrama de Bode do Sistema em Malha Aberta e Fechada

Retirando assim alguns pontos de interesse temos:

- Baixas Frequências:

Em Malha Aberta temos que em baixas frequências o ganho é muito alto. A 0.1 rad/s o ganho é cerca de 60 dB, o que numa análise ingênua poderia ser bom pois rejeitaria perturbações. Mas repare-se que este ganho tende para infinito quando a frequência tende para zero pois o ganho estático em malha aberta é infinito fruto do sistema ser instável.

Em Malha Fechada obtivemos um ganho a baixas frequências que tende para 24 dB, alto, rejeitando assim perturbações. No diagrama o ganho não é unitário (0 dB) como suposto para seguir a referência mas na realidade é fruto de inserirmos o bloco do inverso do ganho estático.

- Margem de Ganho:

Em Malha Aberta, quando a fase é 180° temos que o ganho é 12.7 dB. Sendo assim temos uma margem de ganho $G_M = 12.7$ dB.

Em Malha Fechada, quando a fase é 180° temos que o ganho é 12.1 dB. Sendo assim temos uma margem de ganho $G_M = 12.1$ dB. Assim sendo podemos fazer um aumento de ganho até 12.1 dB melhorando assim a rejeição das perturbações (mas piorando a rejeição do ruído nos sensores!)

- Margem de Fase:

Em Malha Aberta, quando o ganho é 0 dB temos que a fase é aproximadamente 39.2° . Sendo assim temos uma margem de fase $\phi_M = 180 - 39.2 = 140.8^\circ$.

Em Malha Fechada, quando o ganho é 0 dB temos que a fase é aproximadamente 44.0° . Sendo assim temos uma margem de fase $\phi_M = 180 - 44 = 136^\circ$.

- Altas Frequências:

Em Malha Aberta temos que na frequência mais elevada o ganho é -25.9 dB.

Quando passamos para a Malha Fechada o ganho mantém-se praticamente constante pelo que o ruído nos sensores é suficientemente filtrado.

- Pólo na Origem:

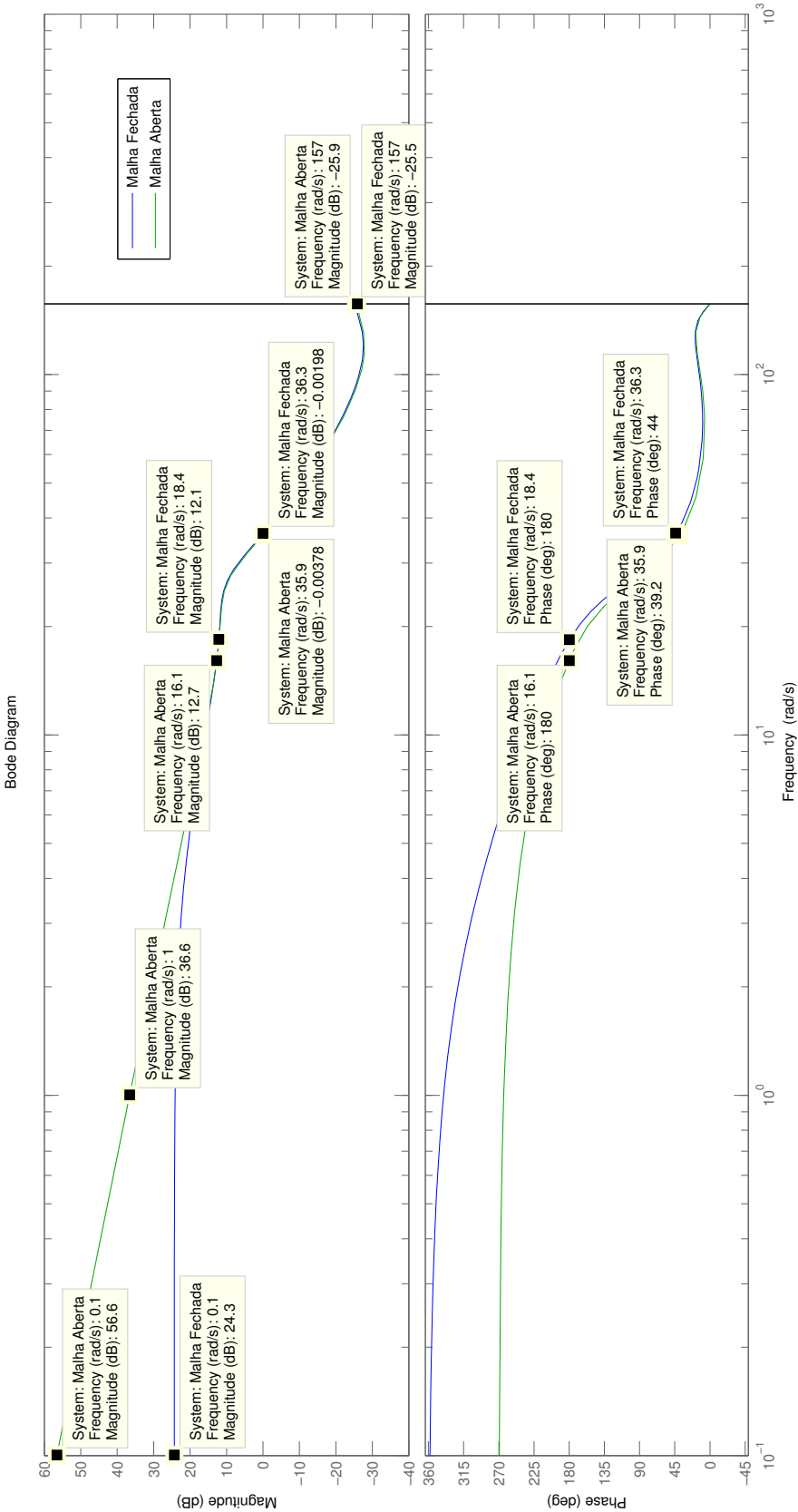
Como se comprova facilmente existe de um pólo na origem na Malha Aberta. Pode-se comprovar pois a fase começa a 270° ou pois o decaimento inicial é de -20 dB/dec ($56.6-36.6=20$).

- Frequência de Nyquist:

Um último pormenor é o pormenor do diagrama ser interrompido a partir dos 157.1 rad/s. Isso acontece pois o sistema é amostrado a 50 Hz. Como tal, e segundo o Teorema de Nyquist, a Frequência de Nyquist do sistema terá de ser inferior a metade da Frequência de Amostragem. Assim:

$$f_N < \frac{50}{2} \text{ Hz} = 157.1 \text{ rad/s}$$

Figura 32: Pontos de interesse do Diagrama de Bode do Sistema em Malha Aberta e Fechada



7 Considerações Finais

7.1 Inclusão de um pré-filtro

A entrada de actuação pode sofrer variações muito bruscas, principalmente nos inícios dos pedidos (quando a diferença entre a referência e a saída é máxima).

Para garantir que a entrada não satura nem varia demasiado bruscamente, a inclusão de um pré-filtro (filtrar, com um filtro passa-baixo, os valores do sinal de actuação) pode ser vantajoso, como uma medida de precaução caso o sistema de controlo se esteja a comportar deficientemente.

Contudo, da forma como o controlador foi projectado, sendo conservador no valor do parâmetro R e garantindo que não estamos a exigir demasiado do actuador, então o pré-filtro torna-se desnecessário, na medida em que o sinal de actuação nunca terá valores que provoquem problemas no desempenho do sistema.

7.2 Avaliação do Desempenho do Controlador e Limites do Desempenho do Sistema

Na secção 2 definimos as especificações para o controlador do sistema. Consideramos ter atingido todas as especificações, pelo que o desempenho do controlador é o desejado.

Em termos de aspectos negativos, destaca-se o facto de o controlador não ser capaz de colocar a saída exactamente na referência (conforme visto anteriormente, o controlador não é imune ao efeito da carga), o que se deverá a erros de modelação do sistema: repare-se que o sistema em simulação não apresenta erro entre a referência e a saída. Introduzindo um efeito integrativo no controlador, este defeito seria corrigido, permitindo a eliminação do *offset* constante entre a referência e a saída.

O sistema tem, por si só, limitações de desempenho associadas às limitações físicas do motor e da dinâmica da barra. O controlador não altera esses limites, e desta forma o desempenho do sistema controlado está limitado pelas características físicas do sistema em malha aberta.

7.3 Vários Sinais de Referência

Na 1ª parte do Projecto, a identificação, submetemos o nosso a Sinais de Onda Quadrada e PRBS.

Na 2ª parte do Projecto focamos-nos apenas em Sinais de Onda Quadrada, para verificar se o sinal atingia o ângulo desejado.

Para complementar a nossa análise do Desempenho do Sistema e a verificação dos Limites do mesmo decidimos testar o nosso sistema com outros tipos de Sinais de Referência que não a Onda Quadrada. O sinal PRBS não faz sentido, pois é uma Onda Quadrada que varia a frequência no tempo, pelo que o comportamento será semelhante ao já analisado. Os outros sinais escolhidos foram a Onda Triangular e a Onda Sinusoidal cujos resultados obtidos serão agora apresentados.

Seguimento de uma Onda Triangular de Referência

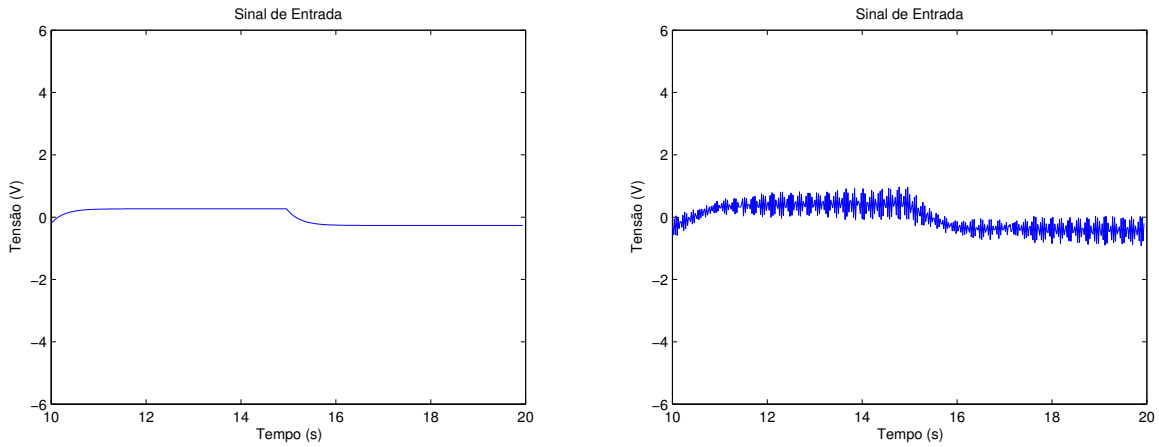


Figura 33: Referência Onda Triangular - Resposta do Sinal de Actuação

Em relação ao sinal de actuação, uma recta no ângulo implica velocidade angular constante, que é o que se verifica em simulação e aproximadamente na realidade.

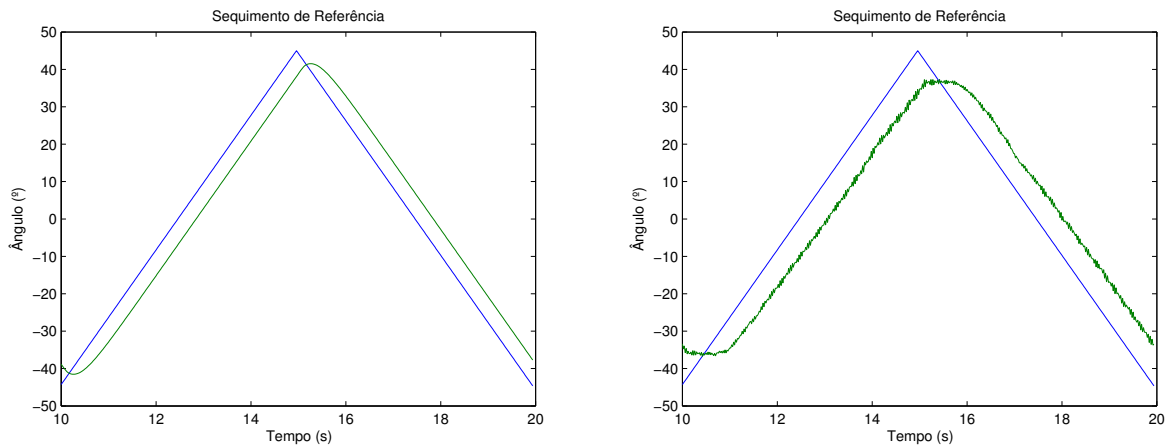


Figura 34: Referência Onda Triangular - Resposta do Sinal de Saída

Verifica-se, tanto em simulação como na realidade, um atraso entre a referência e a saída, o que faz com que o valor máximo da onda triangular nunca seja atingido.

A resposta do sistema segue a referência com o declive desejado, apenas com um *offset* constante.

Seguimento de uma Onda Sinusoidal ($T = 2.5$ s) de referência

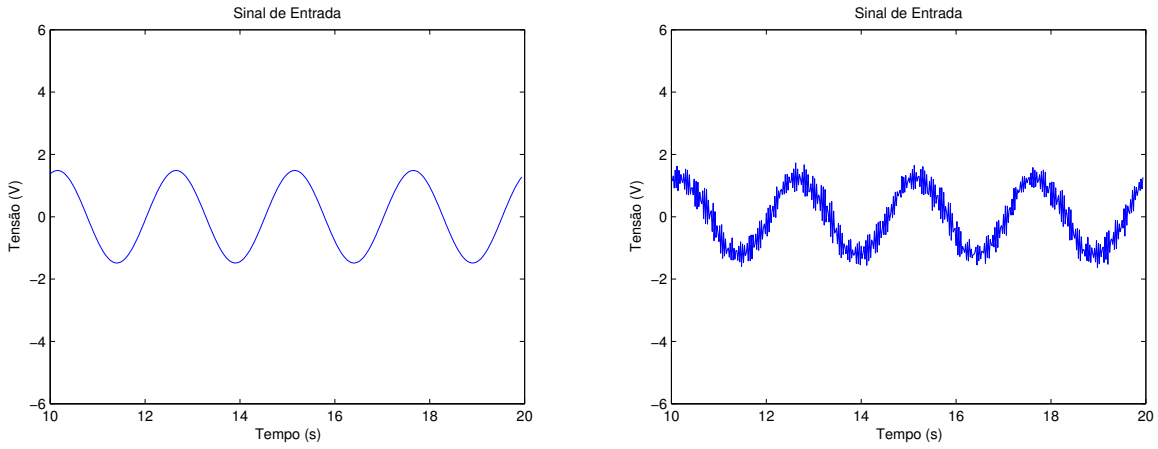


Figura 35: Referência Seno ($T = 2.5$ s) - Resposta do Sinal de Actuação

Para uma referência em seno, o sinal de entrada é um cosseno, que é a derivada do seno. Em simulação é uma onda sinusoidal perfeita; no sistema real verificam-se pequenas oscilações.

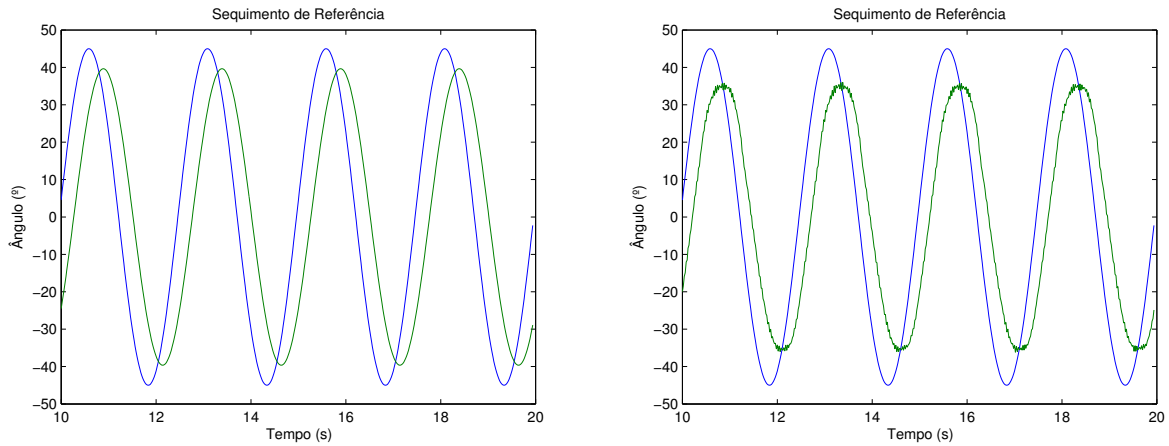


Figura 36: Referência Seno ($T = 2.5$ s) - Resposta do Sinal de Saída

A resposta do sistema é satisfatória, verificando-se novamente um atraso da resposta e um *offset*. De notar que o atraso em simulação é semelhante ao atraso na realidade.

8 Conclusão

O modelo obtido na 1ª parte laboratorial permitiu o desenvolvimento de um controlador LQG que satisfaz os objectivos definidos no início deste relatório.

Desta forma, concluímos que todos os objectivos do projecto foram alcançados com sucesso.

9 Bibliografia

- [1] *Digital Control of Dynamic Systems*, G. F. Franklin, J. D. Powell and M. Workman. Addison Wesley. 3ª Edição;
- [2] *Guia Laboratorial*, por J. Miranda Lemos e A. Bernardino;
- [3] *Ajudas para o Laboratório*, por J. Miranda Lemos, João P. Gomes e A. Bernardino;
- [4] *Slides das Aulas Teóricas de Controlo por Computador*, por J. Miranda Lemos;

Os elementos [2], [3] e [4] foram disponibilizados na página Fénix da unidade curricular de Controlo por Computador.

Apêndice A - Script de Controle

```
% LABORATÓRIO DE CONTROLO POR COMPUTADOR

% 72740 - Miguel Mendes
% 72842 - João Lucas Munhão
% 73411 - João Paulo Silva

% SCRIPT PARA CONTROLO DO SISTEMA

%-----%

% Assume-se modelo em espaço de estados da Malha Aberta nas matrizes:
% A - Matriz de Estado
% B - Matriz de Actuação
% C - Matriz de Saída
% D - Matriz de Perturbações

% Referencia

Y_ref = 90;

%-----%

% Controlador LQ

Q = C'*C;          % Peso da Saida
R = 250;           % Peso da Entrada

% Obtencao do Ganho do Controlador
[K, Sol_Ric, V_P] = dlqr(A, B, Q, R);

%-----%

% Filtro de Kalman

Q_e = 1000*eye(5); % Peso do ruído do processo
R_e = 1;           % Peso do ruído dos sensores

% Obtencao do Ganho de Kalman
[M, Sol_Ric_e, Error_cov, V_P_e] = dlqe(A, eye(5), C, Q_e, R_e);

% Construção do modelo em espaço de estados para o observador
Phi_e = A-M*C*A;
Gamma_e = (eye(5)-M*C)*B;

A_e = Phi_e;          % Matriz de Estado do Observador
B_e = [ M Gamma_e ]; % Matriz de ''Actuação'' do Observador
```

```

C_e = eye(5);           % Matriz de Saída do Observador
D_e = zeros(5,2);       % Matriz de Perturbações do Observador

%-----%

% Inverso do Ganho Estático

N = inv([A-eye(5), B; C, 0])*[zeros(5,1);1];
N_x = N(1:5,:);
N_u = N(6,:);
Nbar = N_u+K*N_x;

%-----%

% Sistema Completo (da referência para a saída)

% Sistema em Espaço de Estados
A_SISTEMA = [ A          -B*K;
              M*C*A      Phi_e-M*C*B*K-Gamma_e*K];

B_SISTEMA = [ B ; Gamma_e+M*C*B]*Nbar;

C_SISTEMA = [ C 0 0 0 0 0 ];

D_SISTEMA = zeros(1, 1);

% Sistema em Função de Transferência
[NUM_SISTEMA, DEN_SISTEMA] = ss2tf(A_SISTEMA, B_SISTEMA, C_SISTEMA, D_SISTEMA);
SYS=ss(A_SISTEMA, B_SISTEMA, C_SISTEMA, D_SISTEMA, Ts);

%-----%

% Visualizações para análise

% Sistema em Malha Fechada
[RESPOSTA, TEMPO]=step(SYS, 5);
figure;
plot(TEMPO, RESPOSTA);
figure;
zplane(NUM_SISTEMA, DEN_SISTEMA);
figure;
dbode(A_SISTEMA,B_SISTEMA,C_SISTEMA,D_SISTEMA,Ts);

ZEROS_SISTEMA = roots(NUM_SISTEMA);
POLES_SISTEMA = roots(DEN_SISTEMA);

% Sistema em Malha Aberta (da 1a parte do relatório)

```

```

[NUM_before, DEN_before]= ss2tf(A,B,C,D);
SYS_before=ss(A,B,C,D,Ts);

figure;
[RESPOSTA_before, TEMPO_before]=step(SYS_before, 5);
plot(TEMPO_before, RESPOSTA_before);
figure;
zplane(NUM_before, DEN_before);
figure;
dbode(A,B,C,D,Ts);

ZEROS_before = roots(NUM_before);
POLES_before = roots(DEN_before);

```