



TÉCNICO LISBOA

Aeronaves Robotizadas

Estimation of Motion Variables of the Parrot AR.Drone
Theoretical and Experimental part

Engenharia Aeroespacial

Abril 2018

Hugo Marques, **80929**

Pedro Trindade, **81612**

Pedro Afonso, **66277**

Índice

1	Introdução	1
2	Modelação e caracterização dos sensores	1
2.1	Experiência A	1
2.2	Experiência B	2
2.3	Experiência C	2
2.3.1	Remoção de <i>outliers</i>	3
2.4	Cálculo <i>raw</i> do <i>pitch</i> , <i>roll</i> com base em medições do acelerômetro	4
3	Filtro de Kalman com modelo dinâmico	5
3.1	Design, implementação e avaliação do filtro de Kalman	6
3.1.1	Ângulo de picada	6
3.1.2	Ângulo de rolamento	10
3.1.3	Altura	11
3.2	Funções transferência	13
3.2.1	Ângulo de picada	13
3.2.2	Ângulo de rolamento	14
3.2.3	Altura	14
3.3	<i>Tuning Parameters Q e R</i>	14
3.4	Com <i>bias</i>	15
3.4.1	Ângulo de picada	15
3.4.2	Ângulo de rolamento	16
3.4.3	Altura	17
3.5	Filtros complementares	18
4	Filtro de Kalman com modelo cinemático	18
4.1	Design, implementação e avaliação do filtro de Kalman	18
4.1.1	Ângulo de picada	18
4.1.2	Ângulo de rolamento	19
4.1.3	Altura	19
4.2	Funções de Transferência	20
4.2.1	Ângulo de Picada	20
4.2.2	Ângulo de Rolamento	21
4.2.3	Altura	21
4.3	Impacto das matrizes <i>Q</i> e <i>R</i>	22
4.4	Com bias	22
4.4.1	Ângulo de Picada	22
4.4.2	Ângulo de Rolamento	24
4.4.3	Altura	25

4.5	Filtros complementares	26
5	Resultados Experimentais	26
5.1	Experiência D	27
5.1.1	Ângulo de Picada	27
5.1.2	Ângulo de Rolamento	28
5.1.3	Altura	29
5.1.4	Conclusões	30
5.2	Experiência E	31
5.2.1	Resultados	31
5.2.2	Conclusões	32
6	Estimação integrada de <i>roll</i> e <i>pitch</i>	33
6.1	Design e implementação do filtro	33
6.1.1	Observabilidade do sistema em espaço de estados	33
6.1.2	Matriz do Ganho de Kalman	35
6.1.3	Estimação dos estados	36
6.2	Simulações	37

1 Introdução

2 Modelação e caracterização dos sensores

De entre a miríade de sensores e variáveis medidas disponibilizadas pelo *Parrot AR Drone*, alguns têm particular interesse. Neste caso iremos focar-nos nos acelerómetros, giroscópios e no altímetro. Para um estudo adequado de toda a atividade experimental, é fulcral a modelação destes sensores bem como a caracterização do ruído que os caracterizam e outras perturbações.

2.1 Experiência A

O objetivo da experiência A é a recolha de valores experimentais com o drone em repouso. Não existem perturbações dos motores elétricos, vibrações significativas ou deslocações de ar que afetem os valores medidos.

Sabemos à partida que o sensor de altura disponível apenas regista medições com interesse de estudo a partir de uma determinada altura (20 centímetros por indicação do professor). Como tal, quando este está em repouso, a altura tem um valor constante de $3,3351 \times 10^{-43}$, que doravante consideramos como sendo zero. Sendo um valor constante a sua variância é naturalmente zero (tabela 1).

Passando ao estudo dos acelerómetros, e tendo a experiência A as características acima mencionadas, esperaríamos que ambas as acelerações em x e y fossem zero, e que a aceleração vertical tivesse o valor da aceleração da gravidade. Foi aproximadamente o que obtivemos (tabela 1). Por comparação com o valor conhecido da aceleração gravítica ($\approx 9.81 m/s^2$), podemos concluir que os valores *raw* que nos são fornecidos pelos acelerómetros vêm em cm/s^2 . Os valores obtidos em x e y, são claramente muito pequenos ($\approx 1\%$ da aceleração gravítica), e por comparação com os valores obtidos em outras iterações da experiência A (com diferente orientação do drone), acreditamos que se devem a uma possível inclinação do acelerómetro, ou a uma possível inclinação do chão do laboratório em relação à verdadeira orientação horizontal. Terão pouco impactos nas experiências futuras. Quanto à covariância, nesta experiência, deve-se quase unicamente a ruído inerente aos sensores, dando uma ideia da sua dimensão, para futura modelação.

Passando à análise dos giroscópios, esperaríamos que todas as 3 velocidades angulares fossem 0. Foi o que obtivemos, aproximadamente, em valor médio para as 3. Novamente, referimos que, a covariância caracteriza quase unicamente o ruído inerente a estes sensores (tabela 1).

Tabela 1: Análise estatística experiência A

Experiência A	Altura	Aceleração x	Aceleração y	Aceleração vertical	Roll Rate	Pitch Rate	Yaw rate
Valor médio	3.3351×10^{-43}	13.5774	4.6363	-977.2680	0.0402	0.0345	0.0023
Covariância	0	13.7353	8.5353	10.2827	0.0022	0.0016	0.0017

2.2 Experiência B

Na experiência B, o drone permanece no chão, mas desta vez com as pás dos motores em funcionamento. Os motores elétricos introduzem vibrações significativas no nossos sistema, que no entanto tentam ser contrariadas pela força de um dos elementos do nosso grupo, que aplica pressão sobre o drone, introduzindo novos elementos de incerteza nas medições.

Aplica-se a mesma situação que na experiência A em relação à validade dos valores obtidos pelo altímetro. Continuamos a ter o mesmo valor constante de $3.3351 \times 10^{-43} m$, com algumas particularidades: considerando esta experiência B como já sendo uma *flight experiment*, existem muitas entradas inválidas (zonas descendentes até zero), que explicaremos adiante a sua origem, e como tal chegamos a um diferente valor médio e covariância para a altura. Estes encontram-se na tabela 2 a título ilustrativo mas devem ser ignorados, por não terem significado correto sem a remoção destes *outliers*.

Analizando os dados relativos às acelerações, continuamos a obter (aproximadamente), os mesmos valores para as 3 componentes das acelerações, mas desta vez, valores bastante maiores para as covariâncias (tabela 2). Algo expectável considerando todas as novas possíveis fontes de vibrações e ruído.

Quanto às velocidades angulares, são novamente próximas de zero, seguindo as covariâncias a mesma lógica que as acelerações (tabela 2).

Segue a lógica que os valores de covariância obtidos para os sensores nesta experiência, englobam as variadas fontes de ruído, vibrações ou interferências possíveis.

Tabela 2: Análise estatística experiência B

Experiência B	Altura	Aceleração x	Aceleração y	Aceleração vertical	Roll Rate	Pitch Rate	Yaw rate
Valor médio	2.8278×10^{-43}	-5.3464	16.4132	-964.4989	0.0561	0.1133	-0.0677
Covariância	1.5176×10^{-86}	1.3244×10^3	1.6847×10^3	140.2430	44.1893	69.0449	131.2013

2.3 Experiência C

Durante a experiência C, recolhemos dados com o drone em *hover*. Nesta *flight experiment*, praticamente todas as possíveis perturbações estão presentes, desde vibrações e ruído provenientes dos motores elétricos, a influências aerodinâmicas do meio envolvente. Com base nos dados obtidos nesta experiência, e na sua análise, poderemos efetivamente retirar os valores de *bias* e *noise* que melhor caracterizam e modelam os sensores no caso de voo.

O *hover* é feito com 1 metro de altura de referência e claramente podemos observar que o drone estabiliza em torno do valor médio dos nossos dados. Como tal, por simples correspondência, concluímos que para converter os valores obtidos no altímetro para metros (unidade útil no sistema em espaço de estados), devemos dividir esse valor por 1.4024×10^{-42} .

Tanto as 3 componentes de aceleração como as 3 velocidades angulares enquadram-se com o esperado para o modo de voo em questão, descritas na tabela 3.

Tabela 3: Análise estatística experiência C

Experiência C	Altura	Aceleração x	Aceleração y	Aceleração vertical	Roll Rate	Pitch Rate	Yaw rate
Valor médio	1.4024×10^{-42}	15.2959	1.7955	-975.2683	0.0329	0.0214	0.3205
Covariância	4.7262×10^{-89}	18.5751	20.6842	39.3808	2.3018	2.7550	1.5478

Algumas conclusões

Estão apenas descritos nas tabelas 1, 2 e 3 os valores obtidos para uma tentativa de cada uma das experiências. Várias foram feitas para cada uma delas, tendo sido escolhidas aquelas tentativas que nos permitem apresentar melhores resultados, aquelas em que houve menos entradas inválidas ou aquelas em que os movimentos pedidos ao drone foram melhor executados. No entanto, é de notar que os valores médios e covariâncias obtidas para as mesmas variáveis, nas mesmas experiências e sob as mesmas condições, por vezes variavam um pouco, sendo que nuns casos essas variações podiam ser consideradas importantes, mas noutras desprezáveis. Como tal, não confiamos totalmente nos valores que teremos como base para introduzir o *bias* e ruído na modelação dos nossos sensores, mas procurámos obter o melhor dentro do possível.

Por integração dos valores provenientes dos giroscópios e sua comparação com os ângulos que também nos são disponibilizados pelo drone, conseguimos concluir que as velocidades angulares se encontram em graus por segundo ($^{\circ}/s$), futuramente útil para utilização no nossos sistema.

Como melhor solução, mas que não implementámos por acharmos que se encontra fora do âmbito deste trabalho experimental e do seu objetivo, sugerimos que, em repouso, sejam recolhidos durante alguns segundos, dados, que nos permitam, detetar o *bias* e ruído dos sensores nas condições em que a experiência vai ter lugar, e após isto, fazer uma calibração do drone antes de iniciar o voo. No caso deste trabalho, optámos por generalizar e utilizar os valores médios e covariâncias para a experiência C.

2.3.1 Remoção de *outliers*

É de notar, que durante algumas *flight experiments*, alguns dos valores de altura que obtemos têm um comportamento extremamente estranho. Imagine-se o caso do drone em torno do 1 metro, e no momento a seguir, o altímetro assinala um valor bastante inferior durante alguns instantes, e logo de seguida novamente um valor em torno da unidade. Na prática, o que se verifica é que o formato da curva que descreve a altura, de um momento para o outro, sofre um *shift*, mas mantém o mesmo formato de curva. Várias ideias nos surgiram para explicar o fenômeno, mas no entanto nenhuma delas nos pareceu suficientemente adequada. Falou-se, por exemplo, no caso de perda de pacotes, ou seja, falhas na comunicação entre o drone e o nosso sistema. Este tipo de perdas são muito comuns em redes onde muitos dispositivos estejam a ser utilizados ao mesmo tempo, como é o caso do nosso laboratório experimental, havendo interferências. No entanto, se tal fosse o caso, perderíamos a comunicação

e não se manteria a curva em questão. Como tal, não conseguimos arranjar uma explicação para o que acontece, mas implementámos um algoritmo para lidar com estes casos.

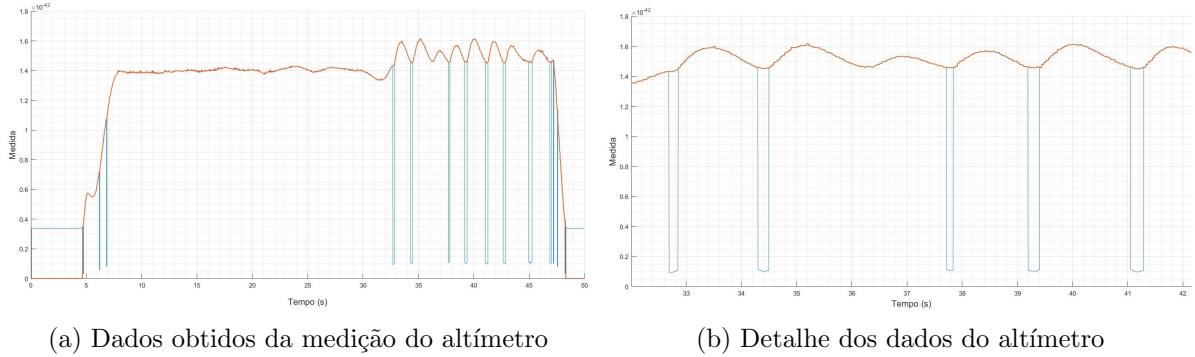


Figura 1: Observação dos dados obtidos pelo altímetro. A azul encontra-se os dados como obtidos do altímetro, e a vermelho os dados após filtragem com o filtro descrito

Aquando da deteção de um destes momentos de *shift*, é calculado o desvio e colocada a curva no sítio correto outra vez, obtendo aquilo que consideramos que deveriam ser as medições sem a ocorrência deste fenómeno. Note-se ainda que este filtro foi desenhado com o objetivo de obter um filtro causal, sem introduzir uma fase na resposta, podendo assim ser implementado em tempo real. Na figura 1b, é visível um detalhe das medições, que facilita a observação do fenómeno descrito, e da respetiva correção.

2.4 Cálculo *raw* do *pitch*, *roll* com base em medições do acelerómetro

É possível, apenas com base nos valores das 3 componentes da aceleração, obter medidas para o *pitch* e *roll*, utilizando o pensamento típico dos inclinômetros, para obter os *tilt angles*. Esta abordagem apenas é válida num voo não acelerado e considerando que $\omega \times v = 0$, isto é considerando que apenas se mede as componentes da aceleração gravítica no referencial do corpo, e que o bias é nulo. Com isto, pode-se escrever:

$$y_{accel} = R(\lambda)^T \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}$$

Em que $R(\lambda)^T$ representa a matriz de rotação do referencial para o referencial do corpo. Posto isto, podemos escrever:

$$\begin{aligned} y_{accel,x} &= g \sin(\theta) + \eta_{accel,x} \\ y_{accel,y} &= -g \cos(\theta) \sin(\phi) + \eta_{accel,y} \\ y_{accel,z} &= -g \cos(\theta) \cos(\phi) + \eta_{accel,z} \end{aligned}$$

Desprezando os ruídos, podemos, através de alguma manipulação algébrica, obter expressões para as medidas de *pitch* e *roll*:

$$\theta_{raw} = \arctan \left(\frac{y_{accel,x}}{\sqrt{y_{accel,y}^2 + y_{accel,z}^2}} \right)$$

$$\phi_{raw} = \arctan \left(\frac{y_{accel,y}}{y_{accel,z}} \right)$$

Note-se que as expressões obtidas não dependem do valor da aceleração da gravidade, sendo por isso imunes a variações deste valor e podendo assim obter-se valores válidos caso haja uma aceleração segundo a direção da gravidade, desde que a soma não mude o sentido da mesma. Note-se ainda que, devido às expressões conterem uma razão entre medidas, as unidades das mesmas não têm impacto no cálculo. No entanto, antes de efetuar este cálculo, estas medidas são convertidas para m/s^2 (Devido ao valor medido pelo acelerômetro segundo a direção z, na experiência A, conclui-se que as medidas dos acelerômetros vêm expressas em cm/s^2). É ainda importante notar que estas expressões nos permitem apenas obter medidas dos ângulos entre -90° e 90° , sendo isto suficiente para o tipo de movimento estudado. Poderia trabalhar-se com os sinais das componentes caso fosse desejado obter ângulos maiores que 90° .

Para garantir que os valores de atitude que obtemos com este método são coerentes, isto é, a representação dos ângulos de Euler utilizada é a considerada pelo fabricante do drone, comparámos estes valores com os valores das mesmas variáveis fornecidos pelo *Parrot AR Drone* acessíveis através do modelo de *Simulink* fornecido, tendo estes coincidido.

3 Filtro de Kalman com modelo dinâmico

Após uma devida modelação dos sensores, com base nos dados obtidos em laboratório iremos centrar-nos no estudo e aplicação de filtros de Kalman e filtros complementares.

Tendo como ideia alguma continuidade entre atividades laboratoriais, tomamos como ponto de partida, que deveríamos utilizar o modelo dinâmico do primeiro laboratório para aplicação do filtro de Kalman. Neste caso, este permitir-nos-ia a estimação do estado de um sistema dinâmico linear a partir de uma série de medições ruidosas, tendo como base os valores medidos para essas grandezas e os valores que são previstos para estas, pelo nosso modelo (em parte previamente desenvolvido). Este calcula a incerteza para o valor previsto, e de acordo com esta incerteza, faz uma média ponderada entre os valores predito e medido. As estimativas geradas pela aplicação deste método estão normalmente mais próximas dos valores reais que os valores medidos.

Fizemos todo o estudo teórico, produzimos o filtro e implementámo-lo em *Simulink*, testando-o em simulação e no laboratório, tudo com base neste modelo dinâmico. Detetámos inúmeros problemas. Como todo este processo é baseado num modelo dinâmico, que não

é exato, este está sujeito a muitos erros; desde erros na entrada, problemas com o atraso introduzido pela comunicação, mudanças não previstas pelo nosso modelo (como por exemplo as perturbações que provêm de uma má medição da altitude), à introdução de uma influência por parte da entrada (a nossa referência) no erro de estimativa. Isto ocorre devido ao facto de o modelo não ser exato, não sendo por isso o modelo introduzido no filtro de Kalman, uma réplica do real, não sendo possível portanto remover o efeito da entrada.

No entanto, mais tarde, e de acordo com indicações dos professores, surgiu-nos a hipótese de uma alternativa de modelo baseado na cinemática. Este modelo, baseado apenas na informação proveniente dos sensores, torna todo o processo muito mais simples, e está blindado contra qualquer um dos erros mencionados anteriormente.

Iremos guiar o leitor pelos vários passos de produção e implementação de 2 filtros de Kalman (um para o modelo dinâmico e outro para o cinemático), cada um deles para o ângulo de picada, ângulo de rolamento e altura, procedendo a uma análise dos resultados obtidos em cada um, e finalmente proporemos apenas um deles como melhor solução (não ótima, no entanto).

3.1 Design, implementação e avaliação do filtro de Kalman

3.1.1 Ângulo de picada

Na parte experimental do último trabalho laboratorial determinámos a função de transferência de segunda ordem de θ_{ref} para θ , tendo obtido:

$$\frac{\Theta(s)}{\Theta_{ref}(s)} = \frac{2.8065s + 7.0915}{s^2 + 5.21s + 16.9695}$$

Com recurso ao Matlab, foi possível representar a dinâmica do nosso sistema em espaço de estados, utilizando a função `tf2ss()`. No entanto, estes resultados foram confirmados, obtendo o sistema utilizando métodos matemáticos. O sistema é da forma:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t)\end{aligned}$$

O estado não tem significado físico para a representação em espaço de estados obtida. A entrada do sistema é θ_{ref} e a saída do sistema é θ . Assim, as matrizes obtidas são:

$$A = \begin{bmatrix} -5.21 & -16.9695 \\ 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 2.8065 & 7.0915 \end{bmatrix} \quad D = 0$$

Como saída $y(t)$, temos o ângulo de picada θ , como já referido. No entanto, pretende-se observar também $\dot{\theta}$, para posteriormente ser feito o design de um filtro de Kalman que utilize tanto os dados obtidos pelo acelerómetro, θ_{raw} , como pelo giroscópio, onde se considera $\dot{\theta} = \omega_y$,

aproximação válida para pequenos ângulos. Uma vez que $D = 0$, facilmente se deduz que:

$$\dot{\theta} = \dot{y}(t) = C\dot{x} = CAx(t) + CBu(t)$$

Deste modo, juntando as duas equações na equação da saída, passa-se a considerar o sistema com as mesmas matrizes A e B, mas com novas matrizes C' e D':

$$C' = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 2.8065 & 7.0915 \\ -7.5304 & -47.6249 \end{bmatrix} \quad D' = \begin{bmatrix} 0 \\ CB \end{bmatrix} = \begin{bmatrix} 0 \\ 2.8065 \end{bmatrix}$$

Estão assim definidas as equações do sistema em espaço de estados a utilizar para obter a matriz de ganho L , do filtro de Kalman em modo estacionário e o respetivo filtro de Kalman.

Para tal, foi utilizada a função *kalman()* do MATLAB. De notar que esta função retorna um novo sistema em espaço de estados que corresponde já à implementação do filtro de Kalman. Este sistema recebe como entradas as grandezas medidas pelos sensores (θ_{raw} e $\omega_y = q$) e a entrada do sistema (θ_{ref}), e como saídas os estados estimados e as saídas do sistema estimadas ($\hat{\theta}$ e $\dot{\hat{\theta}}$). Para isso, é necessário descrever alguns parâmetros da função. A função utiliza o seguinte modelo:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) + G\mathbf{w}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t) + H\mathbf{w}(t) + \mathbf{v}(t) \end{aligned}$$

Neste modelo, $\mathbf{w}(t)$ e $\mathbf{v}(t)$ representam ruído branco. O ruído representado por $\mathbf{w}(t)$, denominado de ruído de processo, foi aqui utilizado para representar a não idealidade do modelo para a equação de estado. Deste modo, podemos indicar qual seria a matriz da covariância associada a este ruído de processo, designada por matriz Q . Já o ruído branco representado por $\mathbf{v}(t)$, tem como finalidade representar o ruído presente nos sensores. Deverá também ser indicada uma covariância para este ruído, que serve para indicar quão bons são os valores obtidos pelos sensores. É ainda necessário definir as matrizes G e H. Para simplificar os cálculos, considerou-se que $G = I_{2x2}$ e que $H = 0_{2x2}$, ou seja, os ruídos representados em $\mathbf{w}(t)$ entram diretamente para a equação de estado e considera-se que o ruído de processo não tem influência direta na saída, havendo apenas ruído associado aos sensores. Considerou-se ainda que os ruídos $\mathbf{w}(t)$ e $\mathbf{v}(t)$ são não correlacionados.

De seguida, foi necessário definir as matrizes Q e R . Estas matrizes deverão ser ajustadas experimentalmente, e servirão de *tuning knobs* para o ajuste do filtro de Kalman. Consideraram-se ambas as matrizes como sendo diagonais e da seguinte forma:

$$\begin{aligned} Q &= qI_{2x2} \\ R &= \begin{bmatrix} r_\theta & 0 \\ 0 & r_{\dot{\theta}} \end{bmatrix} \end{aligned}$$

Assim, é possível ajustar q para descrever quão válido é o modelo, e os valores de r_θ e $r_{\dot{\theta}}$, para descrever a qualidade dos dados dos sensores, nomeadamente, a variância do ruído inerente a estes (que se assume ser ruído branco). Assim, inseriu-se na matriz R os valores das covariâncias obtidas para os respetivos sensores e ajustou-se depois a matriz Q , de forma a afinar o filtro. É no entanto importante notar que não foi obtido ainda um valor para a covariância da medida de θ_{raw} . Procedendo de maneira semelhante à descrita na secção anterior, e utilizando agora apenas os dados da experiência C, obtém-se $\text{Cov}[\theta_{raw}] = 1.9932 \times 10^{-5} \text{ rad}^2$. Note-se ainda que o valor para a covariância obtido na secção anterior para a medida do giroscópio vem expressa em $(^\circ/\text{s})^2$, e é necessário converter para $(\text{rad}/\text{s})^2$, para estar coerente com os valores utilizados. Para isto basta multiplicar a covariância obtida por $(\pi/180)^2$. Note-se que será necessário converter também a medida proveniente do giroscópio para rad/s .

Foi assim possível obter a respetiva matriz de ganhos L , para o filtro de Kalman. Implementando o filtro de Kalman, este fornece a solução ótima para o problema de estimação, obtendo-se um novo modelo em espaço de estados. O novo sistema tem a forma:

$$\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= A\hat{\mathbf{x}}(t) + B\mathbf{u}(t) + L(\mathbf{y}(t) - C\hat{\mathbf{x}}(t) - D\mathbf{u}(t)) \\ &= (A - LC)\hat{\mathbf{x}}(t) + (B - LD)\mathbf{u}(t) + Ly(t) \\ \begin{bmatrix} \hat{\mathbf{y}}(t) \\ \hat{\mathbf{x}}(t) \end{bmatrix} &= \begin{bmatrix} C \\ I \end{bmatrix} \hat{\mathbf{x}}(t) + \begin{bmatrix} D \\ 0 \end{bmatrix} \mathbf{u}(t)\end{aligned}$$

Partindo do sistema como descrito acima, a sua implementação em SIMULINK é bastante simples. No entanto, para podermos testar o filtro em ambiente de simulação, é necessário fazer algumas alterações ao modelo de SIMULINK, para obter as grandezas medidas pelos sensores, e modelar os mesmos. Após obter o que seriam os valores exatos das grandezas que os sensores iriam medir, estas serão corrompidas com ruído branco, para simular o ruído dos sensores. É também adicionado ruído branco nos blocos correspondentes às equações de estado, para simular o ruído de processo. Foi então testado o filtro descrito para o ângulo de picada.

Na figura 2 é ilustrado o resultado obtido ao aplicar o filtro de Kalman descrito. Como podemos observar, há uma zona com uma “chicotada”, e verifica-se que a zona da sobre-elevação é mal estimada. Após alguma análise do comportamento do filtro, podemos verificar que isto se deve ao atraso do sistema devido à comunicação Wi-Fi, descrito anteriormente neste relatório, e também no relatório anterior. Acontece que o modelo dinâmico implementado no nosso filtro, responde de imediato à excitação introduzida pela entrada de referência, enquanto que o sistema apenas reage após um certo período de tempo. Para contornar este problema, foi inserido um atraso na entrada de referência antes de aplicar ao filtro de Kalman. Após isto, foi obtido o resultado ilustrado na figura 3.

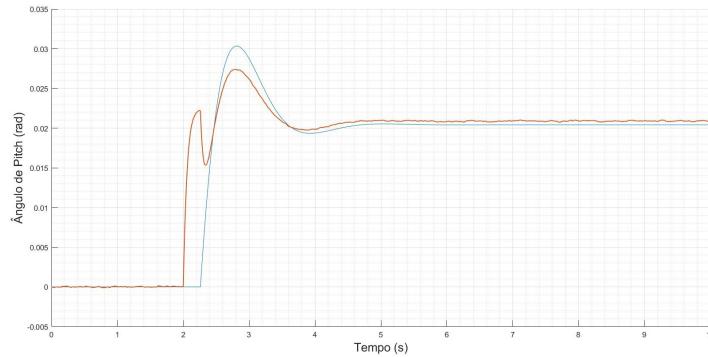


Figura 2: Comparação entre o ângulo de pitch estimado e o real

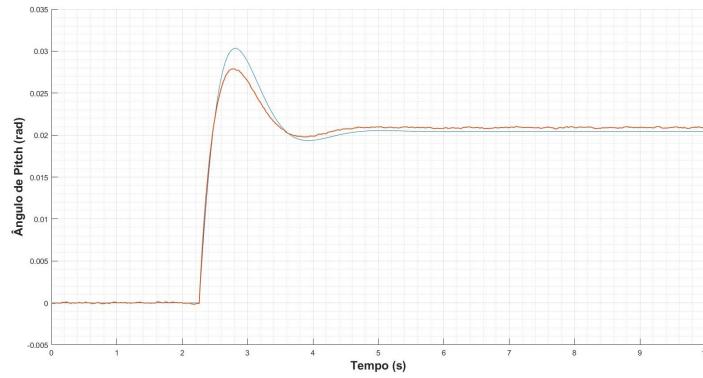


Figura 3: Comparação entre o ângulo de pitch estimado e o real, com delay na entrada

Como podemos observar, o efeito de chicotada não é agora visível, e a variável estimada tem agora um melhor comportamento, apresentando um valor mais próximo do real. É ainda de referir que o valor do ângulo estimado, em estado estacionário, difere do valor real. Isto deve-se ao facto de o modelo utilizado para projetar o filtro de Kalman, não ser exato. Como estamos a trabalhar em ambiente de simulação, para melhor aproximar da realidade, o modelo utilizado no filtro de Kalman é ligeiramente diferente do modelo que rege a simulação, ou seja, como foi já referido, o modelo da dinâmica utilizado foi o modelo identificado na sessão de laboratório anterior. Uma vez que estes modelos são diferentes, os resultados obtidos em regime estacionário diferem também ligeiramente. Posto isto, podemos concluir que, após inserir o atraso na entrada, o filtro apresenta um bom comportamento, em ambiente de simulação.

Foi ainda possível tirar algumas conclusões relativamente à afinação do filtro. Para o valor a inserir na matriz R , para a covariância do ruído do giroscópio, o valor obtido experimentalmente é uma boa escolha. Já para o valor da covariância do ruído do *raw pitch*, a inserir no filtro, a covariância do ruído parece não ser uma boa escolha. Isto porque este valor é obtido assumindo que o acelerómetro mede apenas a aceleração da gravidade. Isto não é sempre verdade, e num regime transitório, esta medida é pouco válida, sendo mais

válida em regime estacionário. Posto isto, ao afinar o filtro, optou-se por aumentar o valor da covariância do ruído deste sensor introduzida na matriz R , de forma a dar menos importância a este “sensor” e mais importância à medida do giroscópio.

Apesar de este filtro poder já ser testado com os dados experimentais, optou-se por apenas aplicar os estimadores obtidos aos dados experimentais, e analisar os resultados, após acrescentar a estimativa de bias, testando assim a solução final.

3.1.2 Ângulo de rolamento

Primeiramente, considerou-se que a função de transferência de Φ/Φ_{ref} é a mesma que se obteve para o ângulo de picada Θ/Θ_{ref} :

$$\frac{\Phi(s)}{\Phi_{ref}(s)} = \frac{2.8065s + 7.0915}{s^2 + 5.21s + 16.9695}$$

Deste modo, o sistema descrito aqui em espaço de estados é o mesmo que obtido na subsecção anterior, e sendo os sensores utilizados também semelhantes, podemos concluir que, a parte de valores diferentes das matrizes Q e R , após afinação do filtro, a implementação é exatamente a mesma. A saída $y(t)$ do sistema é o ângulo de rolamento, ϕ , e a respetiva entrada ϕ_{ref} . Para construir o filtro de Kalman foi também necessário obter $\dot{\phi}$, de forma a utilizar os dados medidos pelos 2 sensores disponíveis.

Recorrendo novamente à função *kalman()* que retorna o sistema em espaço de estados correspondente à implementação do filtro referido, obtém-se um sistema cujas entradas são as grandezas medidas pelos sensores (ϕ_{raw} e $\omega_x = p$) e ϕ_{ref} . As saídas retornadas pelo filtro são os correspondentes estados estimados e as saídas estimadas, $\hat{\phi}$ e $\dot{\hat{\phi}}$.

Mais uma vez, considerou-se que as matrizes Q e R eram da seguinte forma:

Consideraram-se ambas as matrizes como sendo diagonais e da seguinte forma:

$$Q = qI_{2x2}$$

$$R = \begin{bmatrix} r_\phi & 0 \\ 0 & r_{\dot{\phi}} \end{bmatrix}$$

Assim, pode-se ajustar q para descrever quão válido é o modelo, e os valores de r_ϕ e $r_{\dot{\phi}}$ para descrever a qualidade dos dados dos nossos sensores. Experimentalmente, os valores obtidos para a variância do ruído destes sensores, utilizando os dados da experiência C, são $\text{Cov}[\phi_{raw}] = 2.4310 \times 10^{-5} \text{ rad}^2$ e $\text{Cov}[\omega_x] = 2.3018(\text{°}/\text{s})^2 = 7.0117 \times 10^{-4} (\text{rad}/\text{s})^2$.

Mais uma vez, a covariância utilizada para a medida ϕ_{raw} foi aumentada, de forma a afinar o filtro para melhor funcionamento, à semelhança ao que foi feito para o ângulo de pitch. Os resultados obtidos são também bastante semelhantes, uma vez que foi utilizado o mesmo modelo.

É no entanto importante referir que foi necessário ajustar o ganho estático do modelo,

para este estar de acordo com os valores obtidos para o ângulo de roll. Isto porque, apesar da simetria do drone, e de este apresentar o mesmo tipo de comportamento relativamente a este ângulo, o modelo da dinâmica irá variar, visto que este modelo entra já em consideração com o controlo implementado pelo fabricante do drone, que poderá ser diferente entre o ângulo de rolamento e picada, o que irá levar a diferentes valores para o ganho estático do sistema controlado. Isto leva-nos a um outro problema que surge do uso do modelo dinâmico, uma vez que não procedemos à identificação da dinâmica para o rolamento, sendo necessário aproximar o modelo com base no modelo obtido para o ângulo de picada.

3.1.3 Altura

Na parte experimental do trabalho laboratorial anterior determinou-se a função de transferência de segunda ordem de H_{ref} para H , tendo obtido:

$$\frac{H(s)}{H_{ref}(s)} = \frac{2.169}{s^2 + 2.633s + 2.206}$$

Com recurso ao Matlab, facilmente se representou o sistema em espaço de estados, utilizando a função `tf2ss()`. No entanto, estes resultados foram confirmados utilizando métodos matemáticos. O sistema é da forma:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t)\end{aligned}$$

Em que as matrizes da representação em espaço de estados obtidas são agora:

$$A = \begin{bmatrix} -2.633 & -2.206 \\ 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 2.169 \end{bmatrix} \quad D = 0$$

Como saída $y(t)$, tem-se a altura h , como já referido. No entanto, pretende-se também observar $a_v = \ddot{h}$, para posteriormente se poder fazer o design de um filtro de Kalman que utilize tanto os dados medidos pelo altímetro, como os que podem ser obtidos do acelerómetro. Devido às características do modelo em espaço de estados, podemos escrever:

$$\ddot{h} = \ddot{y}(t) = CA^2x(t) + CABu(t)$$

Deste modo, juntando as duas equações na equação da saída, passou-se a considerar o sistema com as mesmas matrizes A e B, mas com novas matrizes C' e D':

$$C' = \begin{bmatrix} C \\ CA^2 \end{bmatrix} = \begin{bmatrix} 0 & 2.169 \\ -5.711 & -4.785 \end{bmatrix} \quad D' = \begin{bmatrix} 0 \\ CAB \end{bmatrix} = \begin{bmatrix} 0 \\ 2.169 \end{bmatrix}$$

Mais uma vez, foi utilizada a função `kalman()` do MATLAB. Esta função retorna um novo sistema em espaço de estados que corresponde já à implementação do filtro de Kalman. Este

sistema recebe como entradas as grandezas medidas pelos sensores (h_m e \ddot{h}_m) e a entrada do sistema (h_{ref}) e como saídas os estados estimados e as saídas do sistema estimadas (\hat{h} e $\ddot{\hat{h}}$).

De seguida, foi necessário definir as matrizes Q e R . A matriz Q foi ajustada experimentalmente, e serviu de parâmetro para a afinação do filtro. Para a matriz R , considerou-se ser da forma:

$$R = \begin{bmatrix} r_{h_m} & 0 \\ 0 & r_{\ddot{h}_m} \end{bmatrix}$$

Em que para r_{h_m} e $r_{\ddot{h}_m}$ se irá utilizar as covariâncias obtidas para o altímetro e acelerômetro, respetivamente. É necessário converter estas covariâncias para as unidades desejadas. Para o altímetro, obtém-se $Cov[h_m] = 4.7262 \times 10^{-89} / (1.4024 \times 10^{-42})^2 = 2.4031 \times 10^{-5}$. Já para a medida do acelerômetro, iremos considerar a covariância do acelerômetro segundo z, obtendo assim $Cov[\ddot{h}_m] = 39.3808 / 100^2 = 3.939 \times 10^{-3} m^2$.

Note-se agora que é necessário obter \ddot{h}_m dos sensores disponíveis. Para tal, iremos transformar as acelerações medidas pelo acelerômetro, do referencial do corpo, para o referencial inercial utilizando a matriz $R(\lambda)$ e remover a aceleração da gravidade. Note-se no entanto que a matriz $R(\lambda)$ depende do ângulo de guinada. No entanto, uma vez que pretendemos determinar apenas a componente da aceleração medida, segundo o eixo z do referencial inercial, apenas irá ser necessário o valor dos ângulos de picada e de rolamento. Os valores para estes ângulos que irão ser aqui utilizados serão os valores estimados pelos filtros de Kalman para estas grandezas, descritos nas subsecções anteriores. Assim, podemos escrever:

$$\ddot{h}_m = -(-\sin(\hat{\theta})a_x + \sin(\hat{\phi})\cos(\hat{\theta})a_y + \cos(\hat{\phi})\cos(\hat{\theta})a_z + g)$$

É também necessário proceder a mais algum tratamento para o altímetro. Com base nos valores anteriormente obtidos, podemos apurar o valor da medida de altura. Apurar no sentido em que, consoante a inclinação do drone, as ondas sonoras emitidas pelo drone para medir a altura, vão demorar mais ou menos a atingirem a superfície, obtendo diferentes valores de altura, para aquilo que deveria ser a mesma. Tendo em conta os ângulos $\hat{\theta}$ e $\hat{\phi}$, escrevemos:

$$h_m = h_{sensor} \cos(\hat{\phi}) \cos(\hat{\theta})$$

O filtro descrito foi então implementado em ambiente de simulação para averiguar o seu funcionamento. Mais uma vez, foi necessário introduzir um atraso na referência, devido ao atraso da comunicação Wi-Fi. Para melhor modelar o altímetro, foi introduzido um bloco de forma a que este só comece a medir valores a partir dos 25cm, como verificado experimentalmente.

Quando implementar-mos o filtro, poderíamos considerar que, se o altímetro mede um valor inferior a 25cm, a sua medida não é válida, e utilizar apenas o acelerômetro no filtro de Kalman. No entanto, uma vez que haverá um bias no acelerômetro, e ruído que terá provavelmente média não nula, o uso apenas do acelerômetro poderia levar a estimativa a

desviar-se para valores elevados (visto que não sabemos quanto tempo o drone iria permanecer no chão após inicializar o drone). Decidiu-se portanto utilizar o valor nulo do altímetro, até este começar a fornecer valores. No entanto, caso quiséssemos utilizar apenas o valor do acelerómetro até obter medidas do altímetro, poderíamos simplesmente alimentar a entrada do filtro correspondente à medida de altura com a estimativa de altura do filtro, o que faria com que $L(y - C\hat{x}) = L[0 \quad \ddot{h}_m - \hat{h}]^T$, isto é, a correção viria apenas do acelerómetro.

Procedendo sempre com as mesmas medidas, e após implementação do filtro, obtivemos os resultados ilustrados na figura 4.

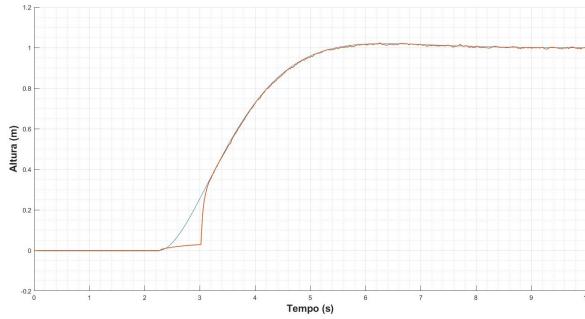


Figura 4: Comparaçāo entre a altura estimada e a real, com delay na entrada

Na figura podemos observar o efeito de a medição do altímetro se manter nula até este atingir os 25cm. Para atenuar este efeito, poderá ser aumentado a valor da matriz R correspondente à covariância da medição de altitude. Também sabemos que não é possível ter alturas negativas, por isso, ao obter a estimativa do filtro, se este estimar uma altura negativa, este valor será posto a zero após o filtro.

3.2 Funções transferência

3.2.1 Ângulo de picada

Partindo da matriz de ganhos obtida na subsecção anterior, pode-se obter as funções de transferência pedidas, procedendo da seguinte forma:

$$T(s) = C[sI - (A - LC)]^{-1}L + D$$

Com isto podemos obter as seguintes funções de transferência:

$$\begin{aligned}\frac{\hat{\Theta}(s)}{\Theta_{raw}(s)} &= \frac{0.02472s + 1.59}{s^2 + 120.3s + 317.9} \\ \frac{\hat{\Theta}(s)}{\Omega_y(s)} &= \frac{-17.64s + 23.21}{s^2 + 120.3s + 317.9}\end{aligned}$$

Como podemos verifica por observação direta das funções de transferência, estas são do tipo passa-baixo, no entanto, é de notar uma diferença razoável no ganho estático, e na

colocação do zero. Por observação do diagrama de Bode de ambas as funções de transferência, vemos que a função de transferência associada à medição do giroscópio deixa passar mais frequências. Isto era já de esperar visto que neste sensor são esperadas maiores variações, a maiores frequências, e é também nas frequências mais altas que a influência deste sensor é importante. Já a influência do sensor do *pitch inclinometer* é mais relevante a baixas frequências. O facto de ambas as funções serem passa-baixo era também esperado, visto que o filtro de Kalman vai tentar remover parte do ruído de alta frequência das medições ao estimar a saída, utilizando para isso o modelo para “suavizar” a estimativa.

3.2.2 Ângulo de rolamento

Procedendo da mesma forma para determinar as funções de transferência pedidas, obtém-se:

$$\begin{aligned}\hat{\Phi}(s) &= \frac{0.01079s + 0.4568}{s^2 + 90.21s + 236.2} \\ \hat{\Omega}_x(s) &= \frac{-12.89s + 17.84}{s^2 + 90.21s + 236.2}\end{aligned}$$

Devido à semelhança das condições entre a estimação para o ângulo de picada e rolamento, os resultados obtidos são semelhantes, tratando-se as funções de transferência novamente de funções de transferência de segunda ordem, do tipo passa-baixo.

3.2.3 Altura

Após termos o sistema bem definido, podemos obter as funções de transferência pedidas:

$$\begin{aligned}\hat{H}(s) &= \frac{7.605s + 22.84}{s^2 + 10.89s + 25.42} \\ \ddot{H}_m(s) &= \frac{-0.09054s - 0.4091}{s^2 + 10.89s + 25.42}\end{aligned}$$

Podemos aqui verificar novamente a existência de funções de transferência do tipo passa-baixo. Verifica-se também aqui uma maior influência da medida do altímetro, principalmente em estado estacionário, proveniente da baixa covariância do sensor de altitude. Já para a medida da aceleração, o ganho estático é bastante reduzido, como seria já de esperar.

3.3 Tuning Parameters Q e R

Como já foi abordado, estes parâmetros estão relacionados com a qualidade dos sensores e do modelo. Comecemos por analisar Q : como foi definido, apenas precisamos de nos preocupar com o valor de q . A matriz Q está relacionada com a qualidade do nosso modelo, nomeadamente, representa a covariância do ruído de processo, que podemos utilizar para

dizer que o nosso modelo é melhor ou pior. Deste modo, se q apresentar um valor bastante reduzido, significa que consideramos o modelo bastante bom. No limite, se considerássemos $q = 0$, estaríamos a dizer que o nosso modelo não apresenta erros, e portanto, não precisa de ser corrigido pelos sensores, ou seja, o ganho L seria nulo e ficaríamos portanto com o nosso modelo inalterado. Com o aumento deste valor, o modelo passa a ter cada vez menos importância na estimação, visto que isto é como dizer que o nosso modelo é pouco fiável.

Uma análise semelhante pode ser feita para a matriz R . Da maneira como foi aqui definida, é uma matriz diagonal, o que significa que o erro num sensor não tem relação com o erro de outro sensor. Deste modo, os elementos da diagonal de R estão relacionados com a qualidade das medições de cada sensor. Aumentando o elemento da diagonal associado a um sensor, está-se a indicar que esse sensor irá ter dados mais ruidosos, e portanto deverá ter menos influência nos valores estimados.

Como já foi feita uma devida caracterização dos sensores, utilizamos os valores das covariâncias obtidas para substituição na matriz R . No entanto, temos antes de aplicar o fator de conversão entre as unidades do sensor, e as unidades do nosso sistema em espaço de estado (valores esses que se encontram definidos na secção 2 deste relatório), como já referido na secção anterior. Neste caso em específico, o fator de conversão tem de ser elevado ao quadrado e multiplicado pelo valor de covariância obtido para cada sensor.

3.4 Com *bias*

3.4.1 Ângulo de picada

Nesta subsecção pretende-se implementar um estimador para determinar o *bias* do *rate gyro*, aumentando o nosso sistema com um estado, que representa o *bias*, e assumindo que:

$$\dot{b}_y = 0 \quad e \quad \omega_y = \dot{\theta} + b_y$$

Com este propósito, alterou-se as matrizes do espaço de estados, obtendo o seguinte:

$$A = \begin{bmatrix} -5.21 & -16.9695 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 2.8065 & 7.0915 & 0 \\ -7.5304 & -47.6249 & 1 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ 2.8065 \end{bmatrix}$$

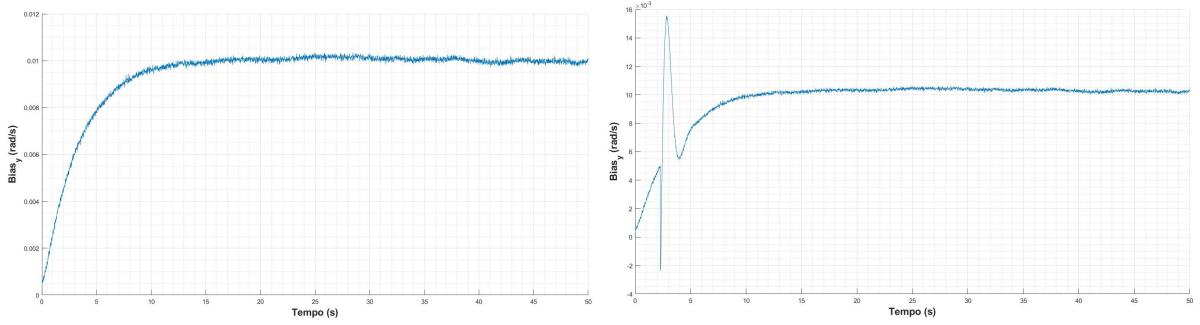
Para implementar o filtro de Kalman, procedeu-se de modo semelhante ao referido anteriormente, no entanto é agora necessário modificar a matriz Q da covariância do ruído de processo. Esta vai então tomar a seguinte forma:

$$Q = \begin{bmatrix} q & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & q_b \end{bmatrix}$$

Neste caso, pela maneira como decidimos representar o ruído de processo, q_b deverá

representar a covariância do ruído de processo associado à equação de estado do bias. Ora, esta equação, se assumirmos o bias constante, é exata, o que levaria a dizer que $q_b = 0$. No entanto, se considerássemos este valor para q_b , o bias não poderia ser corrigido, visto que fica forçadamente constante. Assim, q_b servirá de *tuning knob* para afinar a evolução do bias. Pretende-se que o bias evolua de forma lenta, para não estar sujeito a variações elevadas, e convergir para um valor constante.

Após implementar as alterações aqui descritas ao filtro, para estimação do bias, foi adicionado um bias à medida de ω_y em simulação, para depois afinar o filtro. Após a afinação deste, foram obtidos os seguintes resultados para a estimativa do bias em simulação:



(a) Evolução da estimativa do bias no tempo com
entrada nula (b) Evolução da estimativa do bias no tempo com
um escalão de 0.6 à entrada

Figura 5: Resultados e comparação da estimação do bias do giroscópio, utilizando um bias artificial de 0.01 rad/s em simulação

Como podemos verificar na pela figura 5, o nosso estimador tem a capacidade de identificar o bias e, lentamente, convergir para o seu valor. É no entanto de fazer notar que existe uma ligeira influência da entrada na estimativa do bias, isto é, no erro de estimativa, devido ao facto de o modelo não ser exato. Esta diferença é mais notória no regime transitório, sendo pouco visível em estado estacionário (note-se que os gráficos não têm a mesma escala). Este efeito pode ser attenuado ajustando q_b , para tornar a estimativa mais lenta, no entanto, não desaparece completamente, e implicaria um acréscimo considerável no tempo de estimativa no mesmo.

3.4.2 Ângulo de rolamento

Para determinar o *bias* do giroscópio segundo o eixo de rolamento, procedeu-se do mesmo modo que para o do eixo de picada, visto que é um caso bastante semelhante. Assumiu-se assim que:

$$\dot{b}_x = 0 \quad e \quad \omega_x = \dot{\phi} + b_x$$

Modificando novamente o sistema, obtém-se:

$$A = \begin{bmatrix} -5.21 & -16.9695 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 2.8065 & 7.0915 & 0 \\ -7.5304 & -47.6249 & 1 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ 2.8065 \end{bmatrix}$$

A matriz Q voltará a tomar a forma descrita para o caso da picada. Mais uma vez, q_b servirá como um afinador na evolução do bias, e verifica-se, como esperado, que este caso se processa de igual forma ao da picada.

3.4.3 Altura

Nesta subsecção pretende-se implementar um estimador para determinar o *bias* da medida de aceleração vertical, \ddot{h}_m , aumentando o nosso sistema com um estado, que representa o *bias*, e assumindo que:

$$\dot{b} = 0 \quad e \quad \ddot{h}_m = \ddot{h} + b$$

Com este propósito, alterou-se as matrizes do espaço de estados, obtendo o seguinte:

$$A = \begin{bmatrix} -2.633 & -2.206 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 2.1690 & 0 \\ -5.711 & -4.7848 & 1 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ 2.169 \end{bmatrix}$$

Para implementar o filtro de Kalman, procedeu-se de modo semelhante ao referido anteriormente. Para afinar o filtro, à semelhança dos casos anteriores, iremos inserir um bias artificial nesta medição, em simulação. Os resultados encontram-se ilustrados na figura 6.

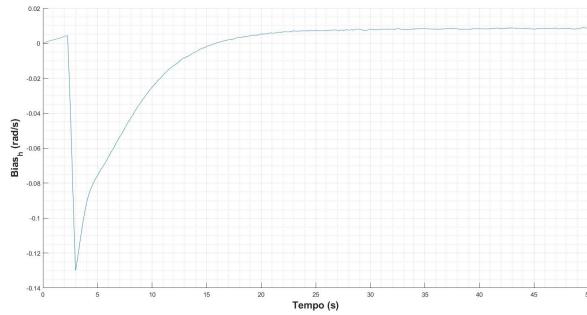


Figura 6: Estimação do bias de \ddot{h}_m , para uma entrada de referência do tipo escalão de 1 metro, e um bias de $0.01 m/s^2$

Mais uma vez, à uma forte oscilação quando se insere uma entrada no sistema. Quando o sistema tende para regime estacionário, o estimador aproxima-se de facto do valor do bias, com uma ligeira diferença que provém do facto de o modelo não ser exato.

3.5 Filtros complementares

Para averiguar se os filtros são complementares, foram obtidas a funções de transferência, de ambos os sensores para a saída do nosso sistema. Se se tratasse de um filtro complementar, sabíamos que somando as funções de transferência seria suposto somarem 1. Ainda assim, visto que este tipo de estimador apresenta uma entrada de referência, para o modelo utilizado, este tipo de filtro não poderia ser complementar, visto que haverá uma parte da estimação correspondente à entrada de referência. Verifiquemos agora as vantagens e desvantagens de um filtro complementar face ao tipo de filtro obtido:

- Vantagens
 - Não necessita de um modelo para o sistema
 - É de simples implementação
- Limitações
 - Apenas tem em conta as medidas dos sensores, não podendo tomar partido de algum conhecimento da dinâmica do sistema.

4 Filtro de Kalman com modelo cinemático

Como explicado na secção anterior, alguns problemas surgiram pela implementação do filtro de Kalman utilizando o modelo dinâmico. Optou-se então por implementar o modelo cinemático para verificar as vantagens e desvantagens de implementação. Uma grande vantagem deste modelo, é não necessitar da entrada de referência do sistema. Outra vantagem é a simplicidade e facilidade de implementação deste modelo, e ainda, o facto de este ser exato.

4.1 Design, implementação e avaliação do filtro de Kalman

4.1.1 Ângulo de picada

Comecemos então por deduzir escrever o modelo para o ângulo de picada. Este é bastante simples. Assumindo que $\dot{\theta} = \omega_y$, o nosso modelo está descrito, e podemos assim escrever com facilidade:

$$A = [0] \quad B = [1] \quad C = [1] \quad D = [0]$$

Aqui, a entrada do nosso modelo é a medida do giroscópio, e a saída, θ , corresponde a uma integração dessa entrada. Temos ainda a medida do *pitch inclinometer* que será utilizada, para corrigir este valor para o ângulo de picada, para evitar que a integração de erro durante um longo período de tempo afaste a estimativa do valor real. Assim, utilizando a função *kalman()*, é possível projetar um filtro de Kalman, que faz a fusão sensorial destas medições.

A matriz Q será aqui um escalar, e a matriz R também. Para estes valores utilizou-se a covariância da medição do giroscópio obtida experimentalmente, e ajustou-se a matriz R da covariância do *pitch inclinometer* para afinação do filtro. Os resultados obtidos encontram-se ilustrados na figura 7.

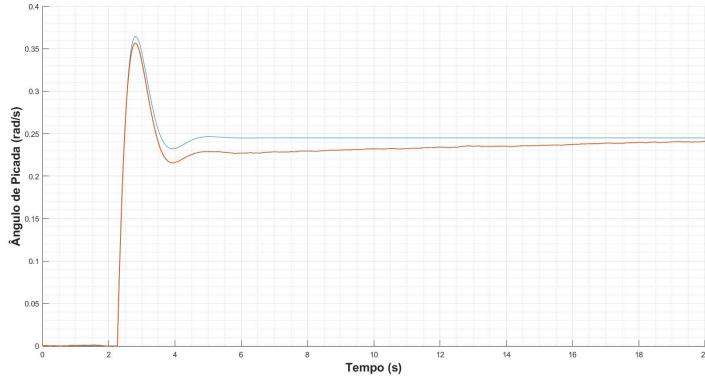


Figura 7: A vermelho, a estimativa do ângulo de picada obtida e a azul, o valor real

Podemos verificar que o filtro produziu bons resultados, utilizando este modelo mais simples, que faz apenas a fusão sensorial. Este modelo não tem problemas também com o atraso do Wi-Fi, o que torna a sua implementação mais direta. Note-se ainda que para utilizar o modelo dinâmico experimentalmente, seria necessário também obter um bom valor experimental do atraso do sistema.

4.1.2 Ângulo de rolamento

Neste caso, o modelo fica portanto $\dot{\phi} = \omega_x$, produzindo uma representação em espaço de estados com as mesmas matrizes que o caso anterior, como já era de esperar, em que o que se torna diferente é a entrada do sistema. Procedendo-se à semelhança do caso anterior, para a escolha das matrizes Q e R , foi bastante simples a implementação do filtro. Em simulação, os resultados são praticamente iguais, visto que estes dois casos têm praticamente as mesmas características.

4.1.3 Altura

O caso da altura é agora ligeiramente diferente. O modelo é agora $\ddot{h} = \ddot{h}_m$, que é um sistema de segunda ordem. A passagem para uma representação em espaço de estados é bastante simples e intuitiva, obtendo-se as matrizes:

$$A = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad D = [0]$$

Para este caso considerou-se a matriz Q como sendo da forma:

$$Q = \begin{bmatrix} q_{\dot{h}} & 0 \\ 0 & q_h \end{bmatrix}$$

Considerou-se assim que $q_{\dot{h}}$ é a covariância obtida experimentalmente para o acelerômetro segundo o eixo z, nas unidades associadas, e a matriz R , sendo um escalar, a covariância do altímetro obtida experimentalmente, sendo então q_h um parâmetro de afinação do filtro. Para melhor afinar o filtro, foi também alterado o valor de $q_{\dot{h}}$. Note-se que, a segunda equação de estado, correspondente à integração de \dot{h} , é exata, e portanto, podemos impôr o valor de q_h igual a zero. Recorrendo agora à função *kalman()* podemos obter um representação em espaço de estados do nosso filtro de Kalman, cujas entradas serão \ddot{h}_m e h_m , e a saída será \hat{h} . Este filtro foi implementado em ambiente de simulação e testado. Os resultados obtidos encontram-se ilustrados na figura 8.

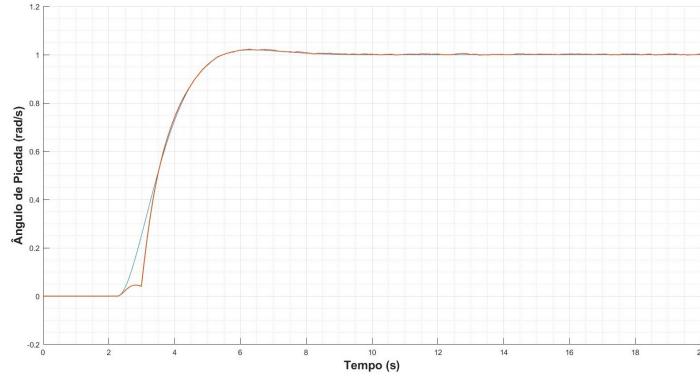


Figura 8: A vermelho, a estimativa da altura obtida e a azul, o valor real

Como podemos observar, a estimativa da nossa variável e o seu valor real estão praticamente sobrepostos, excepto na região em que, devido às características do altímetro, há uma falha na estimativa. Apesar disso, os resultados são favoráveis, obtendo-se resultados tão bons ou melhores que para o modelo dinâmico, de forma bastante mais simples, não tendo os problemas envolvidos no uso do modelo dinâmico.

4.2 Funções de Transferência

4.2.1 Ângulo de Picada

Seguindo o mesmo raciocínio já descrito para o modelo dinâmico, podemos obter as funções de transferência das medidas dos sensores, para a estimativa do ângulo de picada. Visto que temos duas entradas do filtro, correspondentes aos sensores, é de esperar que as funções de transferência dos sensores para a estimativa sejam complementares. As funções de

transferência obtidas são:

$$\begin{aligned}\frac{\hat{\Theta}(s)}{\Theta_{raw}(s)} &= \frac{0.05876}{s + 0.05876} \\ \frac{\hat{\Theta}(s)}{\Omega_y(s)} &= \frac{1}{s + 0.05876}\end{aligned}$$

Como podemos observar, as funções de transferência das medidas dos sensores para a variável a estimar são ambas do tipo passa-baixo de primeira ordem, como seria de esperar. O nosso sistema é de primeira ordem, tendo a nossa representação em espaço de estados apenas um estado, que corresponde à variável a estimar. Uma vez que pretendemos utilizar o filtro de Kalman para obter estimativas mais corretas das variáveis e eliminar o efeito do ruído, era já de esperar que estas funções de transferência fossem do tipo passa-baixo. A escolha das bandas de passagem é feita pelo filtro de Kalman, com base nas matriz Q e R definidas, tendo neste caso frequências de corte baixas.

4.2.2 Ângulo de Rolamento

Mais uma vez, procedendo do mesmo modo, podem ser obtidas as funções de transferência pedidas para o caso do ângulo de rolamento. Note-se que uma vez que o tipo de filtro é bastante semelhante, as funções de transferência serão do mesmo tipo, tendo no entanto diferentes bandas de passagem, devido aos diferentes valores para as matrizes Q e R definidos. As conclusões a tirar para estas funções de transferência serão portanto as mesmas que para o caso anterior.

$$\begin{aligned}\frac{\hat{\Phi}(s)}{\Phi_{raw}(s)} &= \frac{0.05296}{s + 0.05296} \\ \frac{\hat{\Phi}(s)}{\Omega_x(s)} &= \frac{1}{s + 0.05296}\end{aligned}$$

4.2.3 Altura

Será agora utilizado o mesmo raciocínio para a determinação das funções de transferência para o caso da altura, sendo agora esperadas funções de transferência de segunda ordem, uma vez que o modelo apresenta agora mais um estado, e uma das medidas dos sensores representa agora a segunda derivada da variável a estimar.

$$\begin{aligned}\frac{\hat{H}(s)}{H_m(s)} &= \frac{5.048s + 12.74}{s^2 + 5.048s + 12.74} \\ \frac{\hat{H}(s)}{\ddot{H}_m(s)} &= \frac{1}{s^2 + 5.048s + 12.74}\end{aligned}$$

Como podemos observar, foram de facto obtidas funções de transferência de segunda

ordem, como teria de ser, e novamente, estas tratam-se de funções de transferência do tipo passa-baixo, pelas mesmas razões enunciadas para os casos anteriores. Note-se agora que, por comparação com as funções de transferência obtidas para os casos dos ângulos, estas apresentam frequências de corte um pouco mais elevadas. Isto deve-se ao facto de a covariância do sensor de altitude ser bastante reduzida, podendo portanto utilizar-se informação de mais alta frequência proveniente dos sensores. No entanto, a escolha destas frequências é feita pelo cálculo dos ganhos do filtro, e é feita ao afinar as matrizes Q e R das covariâncias dos ruídos, pelo que se obtém o estimador ótimo, sendo por isso uma boa maneira de definir estas frequências com base nos dados experimentais.

4.3 Impacto das matrizes Q e R

Esta questão foi já abordada para a versão do modelo dinâmico, no entanto aqui pode ser dada uma interpretação um pouco diferente. Devido à forma como foram agora escritos os sistemas, e as matrizes Q e R , podemos verificar, para o caso das medidas dos ângulos de picada e rolamento, que à matriz Q está associada a covariância do ruído na equação correspondente à medida do osciloscópico, e a matriz R está associada ao ruído da medição do *raw angle*, como descrito anteriormente. Posto isto, e uma vez que Q e R são escalares, ao aumentar o valor de Q , estamos a dar menos importância à medida do osciloscópio e isto irá implicar que os ganhos de Kalman tomarão valores mais elevados. Ao aumentar o valor de R , estamos a desvalorizar a medida do *raw angle*, e a dar mais valor à medida do osciloscópico, e portanto, os ganhos de Kalman serão agora mais baixos.

Já para o caso da altura, temos um estado extra, visto o sistema tratar-se de um sistema de segunda ordem. No entanto, se reparar-mos, a equação do segundo estado é $\dot{h} = \ddot{h}$, isto é, é uma equação exata. O ruído de processo terá influência na estimação desta variável a partir da primeira equação de estado, correspondente ao nosso modelo, $\ddot{h} = \ddot{h}_m$. Assim, podemos descrever a matriz Q como:

$$Q = \begin{bmatrix} q_{\ddot{h}_m} & 0 \\ 0 & 0 \end{bmatrix} = q_{\ddot{h}_m} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

Assim, teremos apenas um grau de liberdade para variar a matriz Q , e podemos analisar da mesma forma que para os ângulos de picada e rolamento, tirando as mesmas conclusões quanto ao impacto da variação das matrizes Q e R .

4.4 Com bias

4.4.1 Ângulo de Picada

Vamos agora aplicar o caso da estimação de bias ao modelo cinemático. Procedendo da mesma forma, podemos adicionar um novo estado, correspondente ao bias ao nosso sistema.

Para o nosso modelo, passamos agora a assumir que:

$$\dot{\theta} = \omega_y - b_y \quad \dot{b}_y = 0$$

Passando este modelo para uma representação em espaço de estados, vamos agora obter um modelo de segunda ordem, devido aos dois estados, que será então:

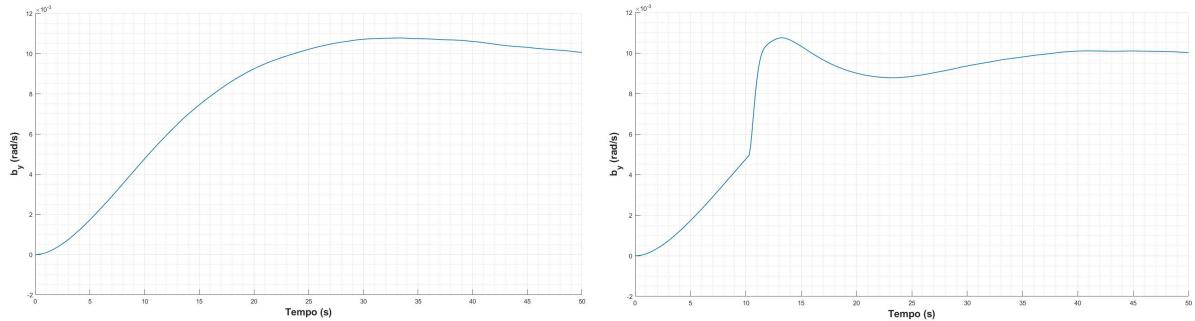
$$\begin{bmatrix} \dot{\theta} \\ \dot{b}_y \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ b_y \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \omega_y$$

$$y = \theta = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ b_y \end{bmatrix}$$

Podemos agora implementar um filtro de Kalman, para o novo sistema um mais um estado. Para isto, a matriz Q deixou agora de ser um escalar, e passou a ser da seguinte forma:

$$Q = \begin{bmatrix} q_\theta & 0 \\ 0 & q_b \end{bmatrix}$$

Optou-se por manter o valor da matriz R escolhido antes de acrescentar este novo estado correspondente ao bias, e o valor de q_θ igual ao valor que a matriz Q tinha antes de adicionar este novo estado. Deste modo, precisamos agora apenas de ajustar o valor de q_b para afinar o filtro, e a evolução da estimativa desta variável. Para tal, à semelhança do que foi feito para o modelo dinâmico, testou-se o filtro em simulação, acrescentando um bias artificial de valor conhecido à medida de ω_y . Após o ajuste de q_b , os resultados obtidos encontram-se ilustrados na figura 9



(a) Evolução da estimativa do bias no tempo com (b) Evolução da estimativa do bias no tempo com
entrada nula um escalão de 0.6 à entrada

Figura 9: Resultados e comparação da estimativa do bias do giroscópio, utilizando um bias artificial de 0.01 rad/s em simulação

Como podemos observar, o estimador implementado converge para o valor do bias artificial, de forma suficientemente lenta de forma a ser pouco influenciado pelas variações de alta

frequência das medições, onde estas poderão ser menos exatas, especialmente no caso da medição do *raw angle*. Podemos também ver que, apesar de haver uma pequena variação na estimativa do bias quando é aplicado o escalão, em regime estacionário o valor estimado converge exatamente para o valor real, ao contrário do que sucedia com o modelo dinâmico, sendo esta portanto outra vantagem do modelo cinemático.

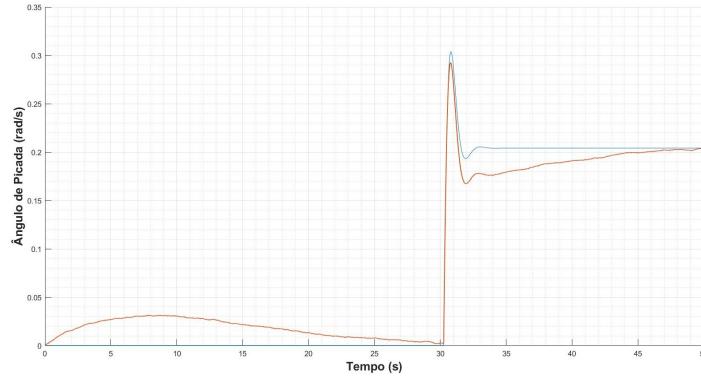


Figura 10: A vermelho, a estimativa de θ obtida e a azul, o valor real, para uma entrada escalão de 0.6

Observando ainda a figura 10, vemos que a nossa estimativa do ângulo de picada começa por se afastar ligeiramente do valor nulo nos instantes iniciais, devido ao bias da medição do giroscópio, sendo depois corrigida pela medição do *raw angle*, e o bias estimado.

4.4.2 Ângulo de Rolamento

À semelhança do que foi já explicado anteriormente, este caso é bastante idêntico ao caso da estimativa do ângulo de picada, sendo o modelo cinemático o mesmo, mas com entradas e saídas diferentes. A entrada será agora ω_x e a saída será ϕ , a variável a estimar. Para a implementação do filtro de Kalman serão utilizadas as medidas do *raw angle* e do giroscópio segundo o eixo de rolamento. O que irá diferir ligeiramente na implementação do filtro será as matrizes Q e R utilizadas nestes casos, que são no entanto, semelhantes. Como tal, os resultados obtidos são bastante semelhantes, podendo as mesmas conclusões ser retiradas.

4.4.3 Altura

Para o caso da altura, há agora alguma diferença, visto que este se tratava de um sistema de segunda ordem, e irá tornar-se agora num sistema de terceira ordem. O modelo fica agora:

$$\begin{bmatrix} \ddot{h} \\ \dot{h} \\ \dot{b}_h \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} h \\ \dot{h} \\ b_h \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \ddot{h}_m$$

$$y = h = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{h} \\ h \\ b_h \end{bmatrix}$$

E a matriz Q ficará agora da forma:

$$Q = \begin{bmatrix} q_h & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & q_b \end{bmatrix}$$

Obtendo o filtro de Kalman correspondente ao novo modelo e implementando em simulação, com um bias artificial adicionado à medida \ddot{h}_m , podemos obter os resultados ilustrados na figura 11

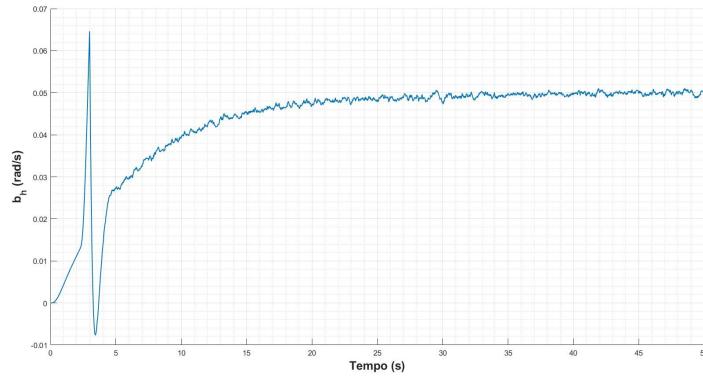


Figura 11: Estimativa do bias da medida de \ddot{h}_m , para um bias de 0.05 e um escalão de 1 metro à entrada

Por observação da figura, podemos concluir que a estimativa do bias converge para o valor pretendido. Há no entanto aquela variação nos instantes iniciais, que se deve ao facto de o altímetro apenas apresentar medidas após 25cm de altura, devolvendo zero se a medida fôr inferior a 25cm. Estando esta situação explicada, podemos concluir que o estimador para o bias do sensor \ddot{h}_m funciona de acordo com o esperado.

4.5 Filtros complementares

Queremos agora verificar se os filtros obtidos para o modelo cinemático são complementares. Comecemos por notar que o nosso filtro possui como entra as medidas de dois sensores relacionados com a variável que pretendemos estimar, e são relacionados a partir de um modelo bastante simples. É de esperar aqui, para este tipo de modelo, que os filtro sejam complementares. Comecemos por determinar este resultados para o caso do ângulo de picada. Para tal, começou por se determinar as funções de transferência das entradas para a saída, agora com o estado adicional do bias. Segundo o raciocínio explicado anteriormente, chegamos a:

$$\frac{\hat{\Theta}(s)}{\Theta_{raw}(s)} = \frac{0.2098s + 0.02028}{s^2 + 0.2098s + 0.02028}$$

$$\frac{\hat{\Theta}(s)}{\Omega_y(s)} = \frac{s}{s^2 + 0.2098s + 0.02028}$$

Para verificarmos se se trata de um filtro complementar, podemos verificar que, se tivermos duas medições da mesma variável, para estimar essa variável, a soma das funções de transferência é um *all-pass filter*. Neste caso não temos duas medições da variável a estimar, mas podemos escrever que $\Omega_y(s) = s\Theta(s)$. Deste modo, substituindo e somando as funções de transferência, obtemos:

$$\frac{s^2}{s^2 + 0.2098s + 0.02028} + \frac{0.2098s + 0.02028}{s^2 + 0.2098s + 0.02028} = 1$$

Assim se conclui que se trata de um filtro complementar. O mesmo raciocínio pode ser conduzido para as restantes variáveis a estimar, chegando à conclusão que todos são filtro complementares. As vantagens e desvantagens deste tipo de filtro foram já abordadas para o caso do modelo dinâmico.

5 Resultados Experimentais

De acordo com os resultados obtidos após a análise do modelo dinâmico e cinemático, verificámos que há inúmeras vantagens no uso do modelo cinemático, visto este ser um modelo mais exato e de implementação mais simples, não necessitando do valor da entrada do sistema para a estimação das variáveis. Outra vantagem ainda é uma melhor isenção à existência de perturbações, que por não poderem ser previstas pelo modelo dinâmico, poderiam levar a maus resultados. Já o modelo cinemático, ao usar apenas as medidas dos sensores, não tem este problema. É ainda importante referir as limitações deste modelo. Comecemos por referir que, o modelo exato para as medidas dos ângulos seria:

$$\dot{\lambda} = Q(\lambda)\omega$$

Isto implica que a simplificação utilizada no nosso modelo cinemático, que equivale a $\dot{\lambda} = \omega$ levará a problemas para valores de ângulos elevados. Também o cálculo das medidas θ_{raw} e ϕ_{raw} não é exato, e o mesmo acontece com a medida \ddot{h}_m , o que implica que sobre certas condições, a estimativa poderá ter valores errados. No entanto, isto acontece tanto no caso do modelo dinâmico como cinemático. Tendo todas estas considerações em mente, optou-se pelo uso do modelo cinemático para implementação experimental e análise dos seus resultados, por comparação com as estimativas obtidas pelo drone.

Começou então por se aplicar o filtro obtido no modelo dinâmico aos dados retirados da experiência D.

5.1 Experiência D

5.1.1 Ângulo de Picada

Na figura 12 estão ilustrados os resultados obtidos ao implementar o estimador para o ângulo de picada.

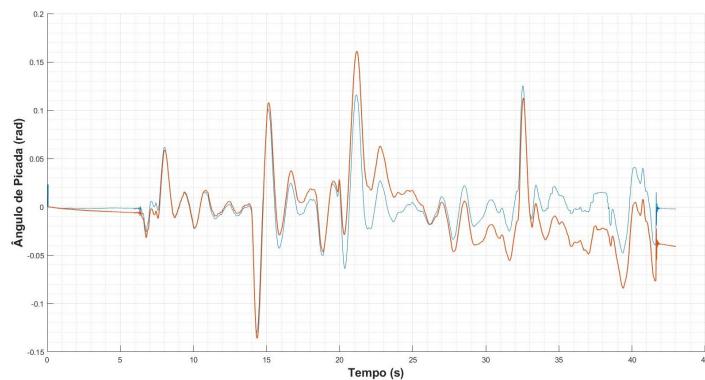


Figura 12: A vermelho, a estimativa obtida e a azul a estimativa fornecida pelo drone

Como podemos observar, as estimativas tomam valores bastante próximos, tendo o mesmo formato. Apresenta-se ainda a diferença entre a nossa estimativa e a fornecida pelo drone, na figura 13.

É fácil verificar que a diferença entre as estimativas toma valores bastante reduzidos. É bastante provável que a estimativa fornecida tenha sido obtida com base num método bastante semelhante ao usado por nós.

Foi ainda verificado o valor estimado para o bias da medida do giroscópio, como verificado na figura 14.

Como podemos verificar, o bias toma valores bastante reduzidos, da ordem de 10^{-3} rad. Note-se ainda pela figura 14, que parece haver uma variação no valor do bias, apartir dos 20 segundos. É também nesta altura que, na experiência D, se fez variar o ângulo de guinada do drone.

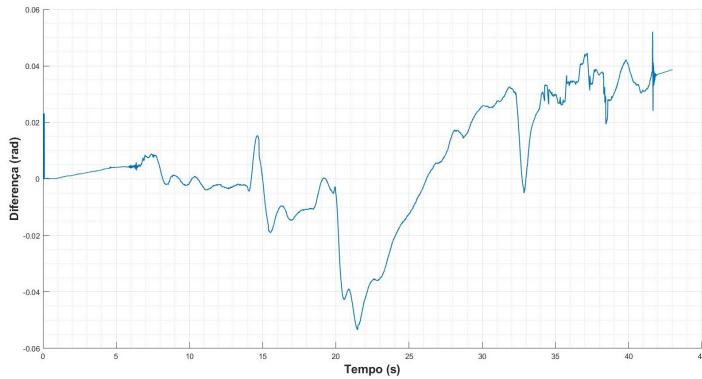
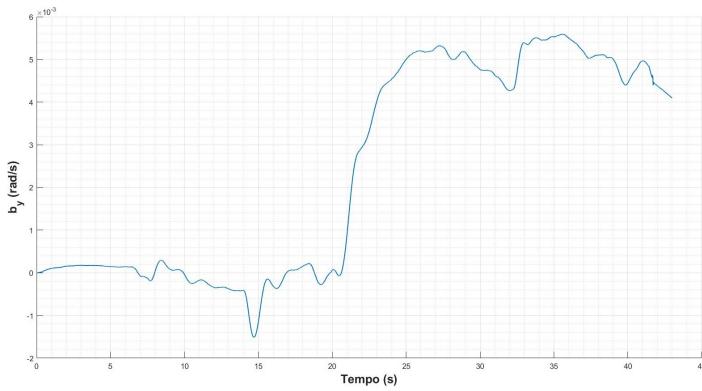


Figura 13: Diferença entre as estimativas

Figura 14: Bias estimado da medida ω_y

5.1.2 Ângulo de Rolamento

Implementando e aplicando a solução sugerida aos dados experimentais, podemos obter os resultados que se encontram ilustrados na figura 15.

Mais uma vez, são observáveis as semelhanças entre a estimativa obtida e a fornecida pelo drone, havendo no entanto algumas diferenças.

Na figura 16 encontra-se ilustrada a diferença entre a estimativa obtida e fornecida. Podemos verificar que a diferença apresenta, por norma, um valor bastante reduzido. Apresenta-se ainda na figura 17 a estimativa do bias da medida do giroscópio. Como podemos verificar, este apresenta um valor bastante reduzido, como no caso o ângulo de picada, havendo uma variação a cerca de 32 segundos, correspondente ao momento em que há uma maior variação na figura 16. Este bias aparece como uma maneira de explicar uma diferença da informação fornecida pelos dois sensores.

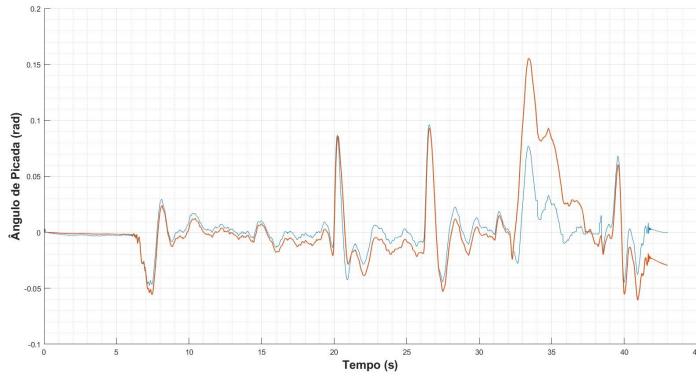


Figura 15: A vermelho, a estimativa obtida e a azul a estimativa fornecida pelo drone

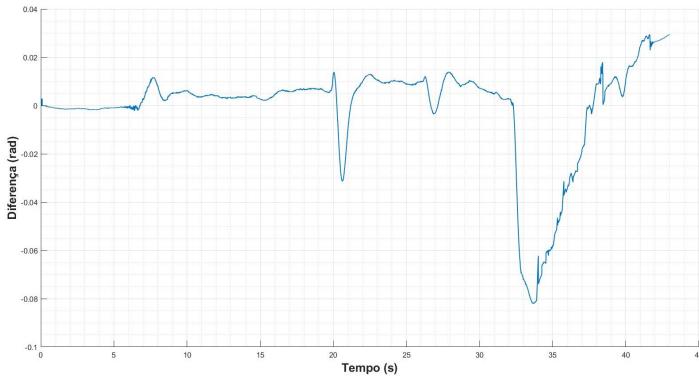


Figura 16: Diferença entre as estimativas

5.1.3 Altura

Implementando agora a solução para a estimativa da altura do drone, obtém-se os resultados apresentados na figura 18.

Podemos aqui observar uma diferença razoável entre a estimativa fornecida e a obtida. Em primeiro lugar, podemos notar que a estimativa fornecida parece ter por base apenas os dados do altímetro, apresentando ainda os outliers descritos anteriormente no relatório. Neste aspecto, a estimativa obtida por nós apresenta um resultado bastante superior. Seria ainda importante fazer notar as oscilações presentes na parte final da nossa estimativa. Esta oscilação, provém da medida de \ddot{h}_m , que por sua vez está fortemente relacionada com a medida do acelerômetro segundo z. Quando o drone aterra, esta aterragem não é muito suave. Isto faz com que haja uma força com um valor razoável a ser exercida sobre o drone quase instantaneamente. Quando isto acontece, o acelerômetro irá medir uma forte aceleração, provocando o efeito descrito. É ainda apresentado na figura 19, a evolução da estimativa do bias para a medida \ddot{h}_m .

Como podemos verificar, a estimativa do bias parece convergir para um valor não muito reduzido. Isto era já de esperar, devido à diferença entre a média da medida do acelerômetro

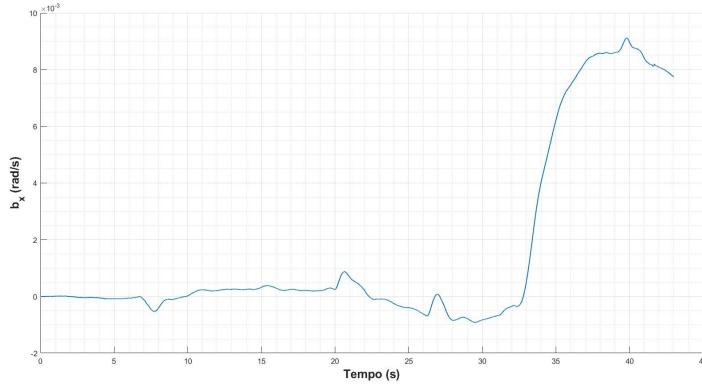
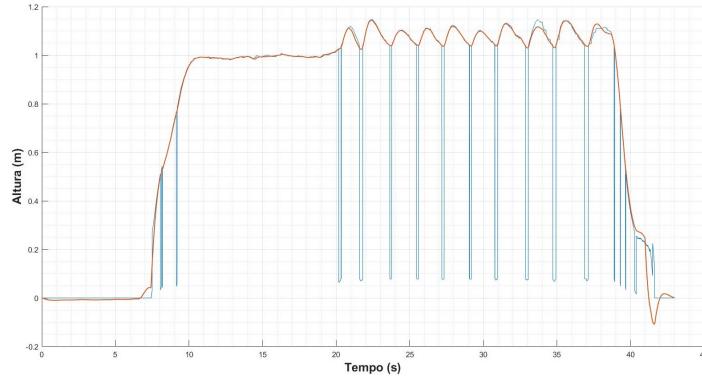
Figura 17: Bias estimado da medida ω_x 

Figura 18: A vermelho, a estimativa obtida e a azul a estimativa fornecida pelo drone

segundo o eixo z do drone na experiência A e o valor da aceleração da gravidade. Há ainda algumas variações, devendo-se algumas ao facto já explicado de o altímetro apenas começar a debitar valores a partir dos 25cm. As últimas variações correspondem ao inicio da descida e ao impacto no chão.

5.1.4 Conclusões

Após verificar os resultados obtidos pelos nossos filtros quando aplicados à experiência D, podemos já tirar algumas conclusões. No que diz respeito à estimação dos ângulos de picada e rolamento, a solução implementada apresenta valores bastante semelhantes aos valores fornecidos pelo drone, e parecem estar de acordo com o movimento feito pelo drone no decorrer desta experiência. Já para a estimação da altura, por comparação com os dados fornecidos pelo drone, notam-se também fortes semelhanças nos resultados, mas a estimativa que obtivemos, ao contrário do valor fornecido pelo drone, não apresenta as falhas do altímetro, sendo portanto uma grande melhoria.

O drone não deverá voar abaixo dos 25cm em voo normal. O erro obtido pela estimação da altura na região final não parece ser portanto importante, uma vez que, durante a aterragem,

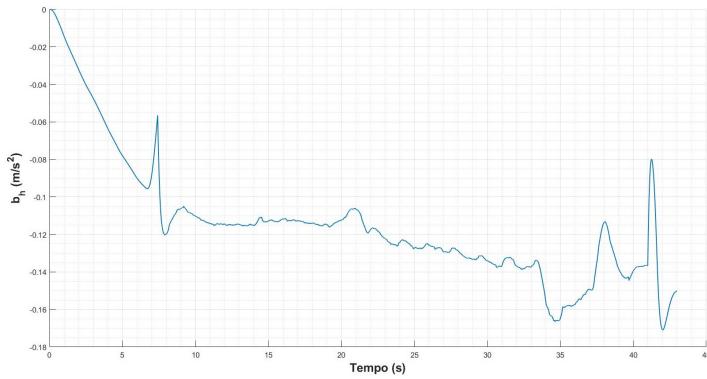


Figura 19: Bias estimado para a medida \ddot{h}_m

o drone parece primeiro voar até uma determinada altura fixa, cerca de 30 cm, e depois, gradualmente desligar os motores. Se isto fôr feito independentemente do valor estimado para a altura, isto é, de forma programada e não por controlo, aquele “overshoot” não teria efeito na aterragem do drone.

5.2 Experiência E

Para melhor testar a solução de estimação implementada, desenhou-se uma experiência E para verificar os resultados de estimação obtidos. Como já referimos, existem algumas simplificações inerentes ao modelo considerado, como por exemplo, o modelo considera que $\dot{\theta} = \omega_y$. Isto só se aplica para ângulos de rolamento e picada nulos, e é uma boa aproximação para ângulos pequenos. Também os valores utilizados para o modelo da altura não são exatos, e dependem das estimativas dos ângulos. Na experiência E, tentou-se variar todas as variáveis a estimar em simultâneo, para verificar o funcionamento da solução implementada. Os resultados são de seguida apresentados.

5.2.1 Resultados

Como podemos observar pela figura 20, a estimativa obtida pela solução implementada e a estimativa fornecida pelo drone, são semelhantes, e a nossa estimativa parece estar de acordo com o movimento do drone durante a experiência E. Apresenta-se na figura 21, a estimativa fornecida pelo drone e a obtida com a solução implementada, para o ângulo de rolamento. Mais uma vez, estimativas são semelhantes, e a estimativa obtida parece estar de acordo com o movimento do drone durante a experiência.

Na figura 22, apresenta-se também as estimativas obtidas e fornecida para a altura estimada.

As estimativas obtidas para a altura parece estar também de acordo com a estimativa fornecida pelo drone, a parte dos outliers dos dados fornecidos pelo drone.

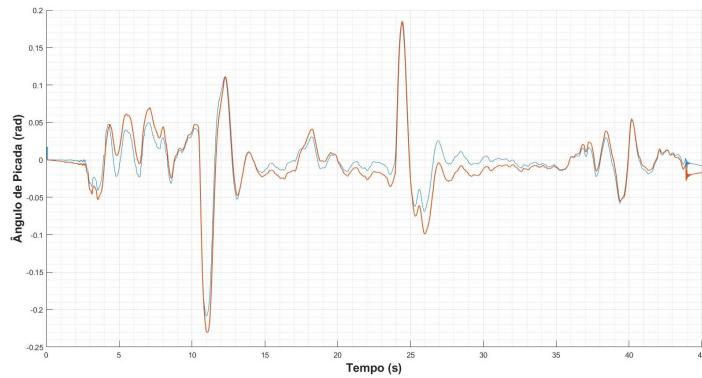


Figura 20: A vermelho, a estimativa obtida e a azul a estimativa fornecida pelo drone

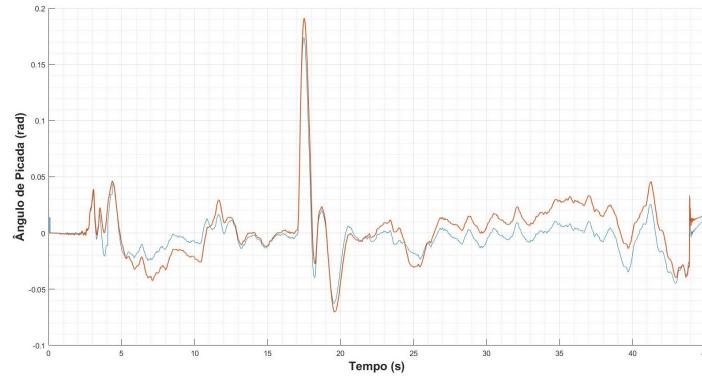


Figura 21: A vermelho, a estimativa obtida e a azul a estimativa fornecida pelo drone

5.2.2 Conclusões

Analisando os resultados obtidos para a experiência E, concluímos que as estimativas obtidas para as variáveis em estudo, utilizando a solução desenvolvida, vão de encontro ao esperado, parecendo fornecer boas estimativas, das variáveis em estudo, e resultados semelhantes aos fornecidos pelo drone. Uma clara melhoria é observada na estimativa da altura, em que sem obtém agora uma estimativa livre de outliers e ligeiramente menos ruidosa. É no entanto impossível verificar quão boas são as nossas estimativas, e determinar os seus erros, visto que não temos acesso ao *ground truth*.

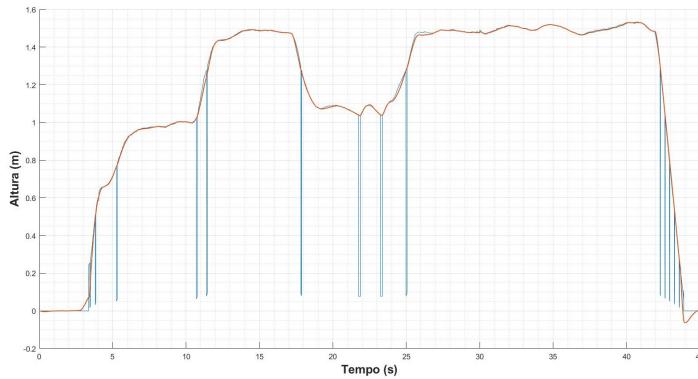


Figura 22: A vermelho, a estimativa obtida e a azul a estimativa fornecida pelo drone

6 Estimação integrada de *roll* e *pitch*

Nesta parte do laboratório foi implementado um filtro para a estimação parcial de atitude (rolamento e picada) e do bias do rate gyro baseado num vector único de observação, o vector que representa a gravidade. É uma observação de um vector num referencial fixado ao corpo de um vector constante de referência em coordenadas inerciais.

Como já foi demonstrado é possível obter *pitch* e *roll* a partir das componentes da aceleração:

$$\theta_{raw} = \arctan \left(\frac{y_{accel,x}}{\sqrt{y_{accel,y}^2 + y_{accel,z}^2}} \right)$$

$$\phi_{raw} = \arctan \left(\frac{y_{accel,y}}{y_{accel,z}} \right)$$

Estas componentes da aceleração são as que se pretendem estimar com o filtro e as medições de velocidade angular "biased" (w_m), estão disponíveis para seu propósito, como também as acelerações no referencial fixo ao corpo medidas pelo acelerômetro, $ya(t)$.

6.1 Design e implementação do filtro

6.1.1 Observabilidade do sistema em espaço de estados

Como foi demonstrado no paper referenciado pelo enunciado, a observabilidade do sistema em espaço de estados fica:

$$\dot{x}(t) = A(ya(t), w_m(t))x(t)$$

$$y(t) = Cx(t)$$

Através da comparação com os filtros de Kalman implementados nas secções anteriores, é notada uma diferença no que respeita à matriz A, que agora dos valores medidos no

acelerómetro, $ya(t)$, e das velocidades angulares medidas pelo giroscópio, $w_m(t)$. Outro aspecto relevante é a inexistência de qualquer entrada, $u(t)$, e portanto o desaparecimento da matriz B no sistema de espaço de estados.

Seguindo as demonstrações do 'paper', a matriz A fica com a seguinte forma:

$$A(ya(t), w_m(t)) = \begin{bmatrix} -S[w_m(t)] & -S[y1(t)] \\ 0 & 0 \end{bmatrix}$$

Sendo esta uma matriz $\in \mathbb{R}^{6x6}$, sendo que cada 0 representa uma matriz $\in \mathbb{R}^{3x3}$ de zeros. $S[]$ representa uma operação de produto externo entre 2 vectores, já demonstrado no 'paper'. As restantes matrizes encontram-se representadas em baixo:

$$S[w_m(t)] = \begin{bmatrix} 0 & -w_m(z) & w_m(y) \\ w_m(z) & 0 & -w_m(x) \\ -w_m(y) & w_m(x) & 0 \end{bmatrix} \quad S[y(t)] = \begin{bmatrix} 0 & -ya_z & ya_y \\ ya_z & 0 & -ya_x \\ -ya_y & ya_x & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Para posteriormente implementar o filtro de estimativa dos estados pedidos, aceleração (vector de 3 dimensões) e o bias da velocidade angular (vector de 3 dimensões), foi desenhado o esquema em Simulink, na figura 23, da observação da Matriz A implementada.

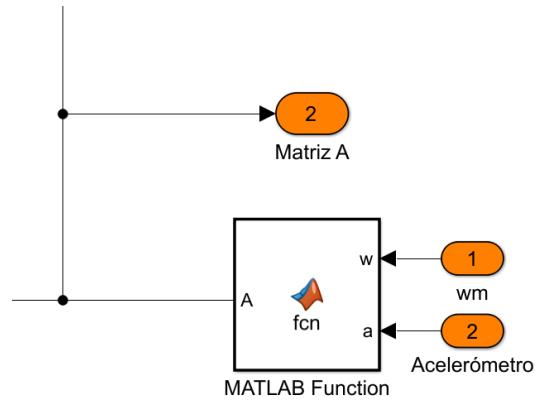


Figura 23: Simulink da Observação da Matriz A

6.1.2 Matriz do Ganho de Kalman

Nesta fase do projecto do filtro, foram adicionadas as perturbações inerentes ao sistema e o ruído correspondente aos sensores de velocidade angular. Podendo-se então, descrever o sistema em espaço de estados com a soma de dois novos vectores:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + w(t) \\ y(t) &= Cx(t) + n(t)\end{aligned}$$

Nestas equações que, representam o sistema em espaço de estados, $w(t) \in \mathbb{R}^6$ e $n(t) \in \mathbb{R}^3$ representam as perturbações do sistema e o ruído dos sensores respectivamente. Ambos são considerados processos Gaussianos Brancos de média nula.

Para obtenção da matriz K, primeiro foi determinada a Propagação da Matriz Covariância dos Erros, descrita em baixo:

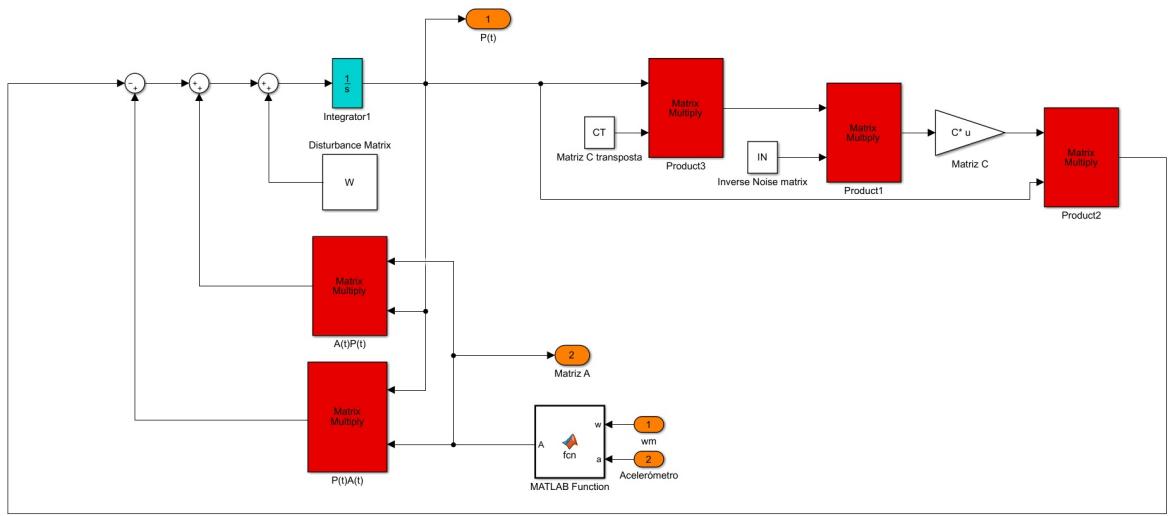
$$\dot{P}(t) = A(t)P(t) + P(t)A(t) + \Xi - P(t)C^T\Theta^{-1}CP(t)$$

Nesta fórmula são acrescentadas duas novas matrizes. A matriz Ξ é calculada por $E[w(t)w^T(t-\tau)] = \Xi\delta(\tau)$ e a matriz Θ por $E[n(t)n^T(t-\tau)] = \Theta\delta(\tau)$, em que os valores que constituem as matrizes Ξ e Θ são todos positivos. Estas matrizes ficam então representadas por:

$$\Xi = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2/5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2.5/5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.5/5 \end{bmatrix} * \left(\frac{\pi}{180}\right)^2 \quad \Theta = \begin{bmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 40 \end{bmatrix} / 100^2$$

Para mostrar a obtenção da matriz $P(t)$, é mostrado na figura 24 a sua implementação com recurso ao Simulink. Para se perceber melhor este diagrama, chama-se à atenção que W corresponde à matriz Ξ , CT à matriz transposta de C e IN corresponde à inversa da matriz Θ . As duas últimas encontram-se representadas aqui:

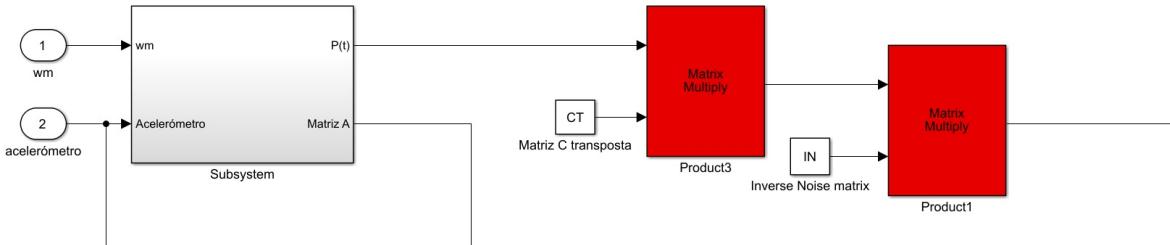
$$C^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \Theta^{-1} = \begin{bmatrix} 500 & 0 & 0 \\ 0 & 500 & 0 \\ 0 & 0 & 250 \end{bmatrix}$$

Figura 24: Esquema em Simulink da obtenção da matriz $P(t)$

Agora é possível calcular o ganho do filtro, $K(t)$, pela seguinte forma:

$$K(t) = P(t)C^T\Theta^{-1} \quad (1)$$

É de notar que a matriz K varia com o tempo, não tendo um valor fixo como nos casos anteriores deste laboratório. Isto acontece, porque o cálculo de K depende de $P(t)$, que por sua vez depende de $A(t)$, que como foi referido na observabilidade do sistema, é variante no tempo.

Figura 25: Simulink da obtenção da matriz de ganho de Kalman, K

Portanto, para usar estas duas matrizes na implementação do filtro, usamos o simulink para obtermos o seu valor directamente em qualquer instante depois das suas funções estarem implementadas no programa como é mostrado na figura 25.

6.1.3 Estimação dos estados

Já foram obtidas as matrizes necessárias para a estimação dos estados, visto que as matrizes K e A são acedidas directamente do Simulink. As acelerações nos eixos do referencial

fixo ao corpo e os respectivos bias das velocidades angulares (os nossos estados - $x(t)$), podem ser estimados da seguinte forma:

$$\hat{x}(t) = A(t)x(t) + K(t)[ya(t) - Cx(t)] \quad (2)$$

Através da equação acima referida é possível finalmente completar a implementação do filtro de Kalman no Simulink e estimar os estados referidos. Essa implementação está na figura 26.

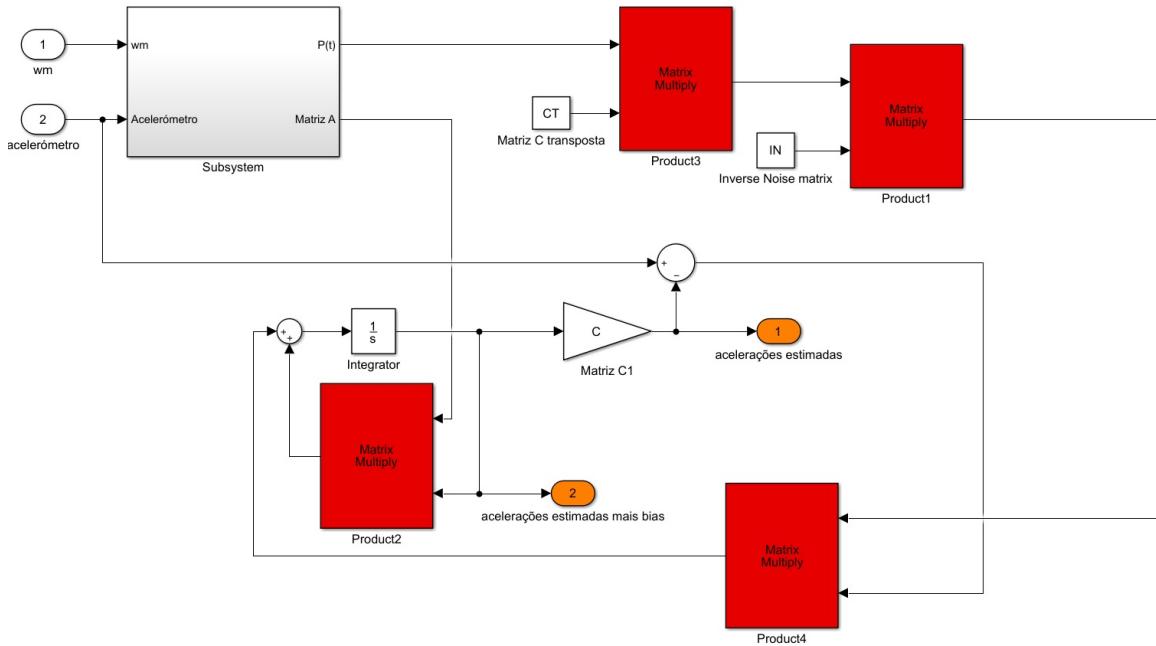


Figura 26: Simulink do estimador de estados completo

É relembrado que o estimador não recebe nenhuma entrada como referência, apenas as grandezas medidas no giroscópio correspondentes à decomposição do vector da velocidade angular nos 3 eixos do referencial fixo ao corpo e as acelerações nos 3 eixos medidas pelo acelerómetro.

6.2 Simulações

Para simular e testar o novo filtro de estimação parcial de atitude, foram usados os dados obtidos pela experiência E. Nas figuras 27 e 28, estão representados as estimativas para os ângulos de picada, θ e rolamento, ϕ . Para complementar esta análise, também estão sobrepostos nos gráficos as estimativas respectivas destes dois ângulos de Euler, pelo modelo cinemático. Pode-se concluir, que os resultados são semelhantes, embora seja notado uma diferença maior entre os dois processos de estimação no ângulo de rolamento, figura 28. O ϕ obtido pelo modelo cinemático toma valores inferiores entre 5 e 10 segundos quando comparado

com o ϕ estimado pelo método referenciado pelo 'paper'.

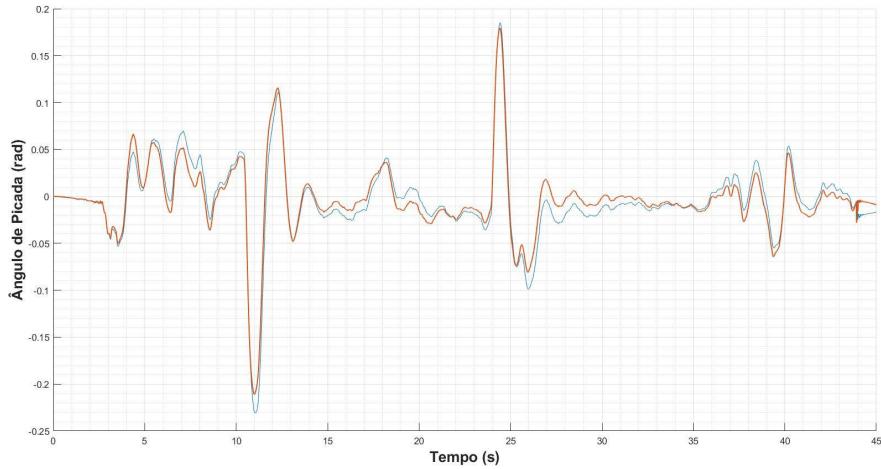


Figura 27: Gráfico com os dados obtidos pela realização da experiência E. Linha vermelha para o θ estimado por este método e linha azul para o θ estimado pelo modelo cinemático

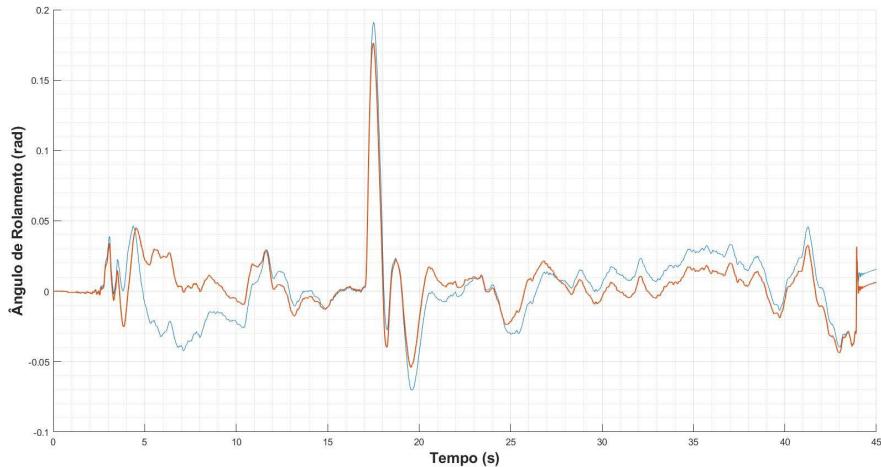


Figura 28: Gráfico com os dados obtidos pela realização da experiência E. Linha vermelha para o ϕ estimado por este método e linha azul para o ϕ estimado pelo modelo cinemático