

**Universidade de Brasília – UnB
Faculdade UnB Gama – FGA
Engenharia Aeroespacial**

**Estimação de atitude de satélite artificial
através de métodos não lineares**

**Autor: Lucas Moreira Martins
Orientador: Dr. William Reis Silva**

**Brasília, DF
2022**



Esta página foi intencionalmente deixada em branco.

Lucas Moreira Martins

Estimação de atitude de satélite artificial através de métodos não lineares

Monografia submetida ao curso de graduação em Engenharia Aeroespacial da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Aeroespacial.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Dr. William Reis Silva

Brasília, DF

2022

Lucas Moreira Martins

Estimação de atitude de satélite artificial através de métodos não lineares/
Lucas Moreira Martins. – Brasília, DF, 2022-
98 p. : il. ; 30 cm.

Orientador: Dr. William Reis Silva

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB
Faculdade UnB Gama – FGA , 2022.

1. Estimação. 2. Kalman. I. Dr. William Reis Silva. II. Universidade de Brasília.
III. Faculdade UnB Gama. IV. Estimação de atitude de satélite artificial através
de métodos não lineares

CDU 02:141:005.6

Lucas Moreira Martins

Estimação de atitude de satélite artificial através de métodos não lineares

Monografia submetida ao curso de graduação em Engenharia Aeroespacial da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Aeroespacial.

Trabalho aprovado. Brasília, DF, 13 de maio de 2022:

Dr. William Reis Silva
Orientador

Dr. Thiago Felipe Kurudez Cordeiro
Convidado 1

Dr. Manuel Nascimento Dias Barcelos Júnior
Convidado 2

Brasília, DF
2022

Esta página foi intencionalmente deixada em branco.

Agradecimentos

Agradeço a Deus e aos meus pais por me permitirem realizar a faculdade, aos meus diversos amigos e colegas pessoais e da faculdade, que temperaram toda a minha experiência e me deram diversos aprendizados e pontos de vista diferentes a respeito da vida, e ao meu professor orientador, por me ajudar e me guiar na realização deste trabalho. Também gostaria de agradecer ao Centro de Controle de Satélites do Instituto Nacional de Pesquisas Espaciais (CCS INPE) pelos dados dos sensores do satélite CBERS-2.

Esta página foi intencionalmente deixada em branco.

Esta página foi intencionalmente deixada em branco.

Resumo

Neste trabalho, os filtros de Kalman Estendido, H_∞ Estendido e H_∞ Estendido de Segunda Ordem foram aplicados para a estimação de atitude e viés de girômetros do satélite CBERS-2 (*China Brazil Earth Resources Satellite 2*), lançado em 21 de outubro de 2003 e inativo desde 15 de Janeiro de 2009, operado tanto pelo Brasil quanto pela China para observação da Terra. O trabalho foi realizado através de medidas simuladas e reais dos sensores de atitude do satélite, que tem como características orbitais uma órbita polar heliossíncrona com altitude de 778 km, inclinação de $98,504^\circ$, excentricidade de $1,1 \times 10^{-3}$ e argumento do periastro de 90° , cruzando a linha do Equador às 10:30 da manhã, na direção descendente. Foi feita uma introdução sobre a teoria por trás da estimação de atitude, que inclui a forma de representar a atitude, os sensores e os métodos de estimação utilizados, e os conceitos apresentados foram aplicados em forma de simulação numérica computacional, gerando resultados. Por fim, os resultados gerados pelos diferentes métodos de estimação utilizados foram analisados e comparados entre si, onde foram feitas observações a respeito de seus desempenhos. Como resultado, a filtragem H_∞ obteve melhores medidas de atitude, por ser um filtro mais robusto, porém a um custo computacional maior, devido às suas maiores quantidades de operações matemáticas.

Palavras-chave: Estimação de atitude. Filtro de Kalman Estendido. Filtro H Infinito Estendido. Filtro H Infinito Estendido de Segunda Ordem. Quatérnions. CBERS-2.

Esta página foi intencionalmente deixada em branco.

Abstract

In this work, the Extended Kalman Filter, Extended H_∞ Filter and Second-Order Extended H_∞ Filter were applied for the attitude and gyro bias estimation of the satellite CBERS-2 (China Brazil Earth Resources Satellite 2), launched in October 21, 2003, and inactive since January 15, 2009, operated by both Brazil and China for Earth observation. The work was done through simulated and real measurements of the satellite, which has as orbital characteristics a Sun-synchronous polar orbit with an altitude of 778 km, inclination of 98.504° , eccentricity of 1.1×10^{-3} and argument of periapsis of 90° , crossing the Equator line at 10:30 am, in a descending direction. An introduction about the theory behind attitude estimation was made, which includes the way of representing attitude, the used sensors and estimation methods, and the presented concepts were applied in the form of numerical computer simulation, generating results. Finally, the results generated by the different estimation methods used were analyzed and compared to each other, where remarks were made about their performance. As a result, the H_∞ filtering obtained better attitude measures, as it is a more robust filter, but at a higher computational cost, due to its greater amounts of mathematical operations.

Key-words: Attitude estimation. Extended Kalman Filter. Extended H Infinity Filter. Second-Order Extended H Infinity Filter. Quaternions. CBERS-2.

Esta página foi intencionalmente deixada em branco.

Lista de ilustrações

Figura 1 – Satélite CBERS-2. Fonte: INPE (2018a).	12
Figura 2 – Os diferentes campos de visada do CBERS-2. Fonte: INPE (2018b).	12
Figura 3 – Órbita do CBERS-2. Fonte: INPE (2018c).	13
Figura 4 – Sistema de Referência Equatorial Terrestre. Fonte: Adaptado de Fehse (2003).	14
Figura 5 – Sistema de Referência Orbital. Fonte: Adaptado de Fehse (2003).	15
Figura 6 – Longitude do nodo ascendente da órbita. Fonte: Adaptado de Fehse (2003).	16
Figura 7 – Sistema de Referência Orbital Local. Fonte: Silva (2016).	17
Figura 8 – Sistema de Referência Orbital Local, vista alternativa. Fonte: Adaptado de Silva (2016).	17
Figura 9 – Sistema de Referência de Atitude. Fonte: Adaptado de Silva, Kuga e Zanardi (2015).	18
Figura 10 – Rotação em ângulos de Euler, sequência 3-2-1. Fonte: Adaptado de Silva, Kuga e Zanardi (2014b).	19
Figura 11 – Multiplicação de quatérnion. Fonte: Wikipedia (2022).	21
Figura 12 – Integração dos filtros e sensores para os dados simulados.	35
Figura 13 – Integração dos filtros e sensores para os dados reais.	35
Figura 14 – Posição no Sistema Equatorial Terrestre simulada.	38
Figura 15 – Velocidade no Sistema Equatorial Terrestre simulada.	39
Figura 16 – Medidas simuladas dos sensores DSS e IRES.	39
Figura 17 – Medidas simuladas dos girômetros.	40
Figura 18 – Estimação de atitude utilizando dados simulados 1.	41
Figura 19 – Estimação de viés utilizando dados simulados 1.	42
Figura 20 – Resíduos referentes aos DSS, pré e pós atualização, utilizando dados simulados 1.	42
Figura 21 – Resíduos referentes aos IRES, pré e pós atualização, utilizando dados simulados 1.	43
Figura 22 – Estimação de atitude utilizando dados simulados 2.	46
Figura 23 – Estimação de viés utilizando dados simulados 2.	47
Figura 24 – Resíduos referentes aos DSS, pré e pós atualização, utilizando dados simulados 2.	47
Figura 25 – Resíduos referentes aos IRES, pré e pós atualização, utilizando dados simulados 2.	48
Figura 26 – Estimação de atitude utilizando dados simulados 3.	49
Figura 27 – Estimação de viés utilizando dados simulados 3.	50

Figura 28 – Estimação de atitude utilizando dados simulados 4.	52
Figura 29 – Estimação de viés utilizando dados simulados 4.	53
Figura 30 – Posição no Sistema Equatorial Terrestre.	55
Figura 31 – Velocidade no Sistema Equatorial Terrestre.	55
Figura 32 – Medidas reais dos sensores DSS e IRES.	56
Figura 33 – Medidas reais dos girômetros.	56
Figura 34 – Estimação de atitude utilizando dados reais.	57
Figura 35 – Estimação de viés utilizando dados reais.	58
Figura 36 – Resíduos referentes aos DSS, pré e pós atualização, utilizando dados reais.	59
Figura 37 – Resíduos referentes aos IRES, pré e pós atualização, utilizando dados reais.	59

Lista de tabelas

Tabela 1 – Características da órbita do CBERS-2.	33
Tabela 2 – Elementos orbitais aplicados.	33
Tabela 3 – Parâmetros dos sensores.	34
Tabela 4 – Parâmetros de inicialização para a estimação utilizando dados simulados 1.	38
Tabela 5 – Erros RMS 1 (100 simulações).	43
Tabela 6 – Erros RMS normalizados 1 (100 simulações).	44
Tabela 7 – Tempo médio gasto para processar os dados.	44
Tabela 8 – Tempo médio normalizado.	45
Tabela 9 – Parâmetros de inicialização para a estimação utilizando dados simulados 2.	45
Tabela 10 – Erros RMS 2 (100 simulações).	48
Tabela 11 – Erros RMS normalizados 2 (100 simulações).	48
Tabela 12 – Erros RMS 3 (100 simulações).	50
Tabela 13 – Erros RMS normalizados 3 (100 simulações).	51
Tabela 14 – Erros RMS 4 (100 simulações).	51
Tabela 15 – Erros RMS normalizados 4 (100 simulações).	52
Tabela 16 – Parâmetros de inicialização para a estimação utilizando dados reais.	54

Esta página foi intencionalmente deixada em branco.

Lista de abreviaturas e siglas

CAST	<i>China Academy of Space Technology</i> (Academia Chinesa de Tecnologia Espacial)
CBERS	<i>China Brazil Earth Resources Satellite</i> (Satélite Sino-Brasileiro de Recursos Terrestres)
CCD	<i>Charge-Coupled Device</i> (Dispositivo de Carga Acoplada)
CCS	Centro de Controle de Satélites
DSS	<i>Digital Sun Sensor</i> (Sensor Solar Digital)
$EH_{\infty}F$	<i>Extended H_{∞} Filter</i> (Filtro H_{∞} Estendido)
EKF	<i>Extended Kalman Filter</i> (Filtro de Kalman Estendido)
INPE	Instituto Nacional de Pesquisas Espaciais
IRES	<i>Infrared Earth Sensor</i> (Sensor da Terra Infravermelho)
IRMSS	<i>Infrared Multispectral Scanner</i> (Imageador por Varredura de Média Resolução)
PROPAT	<i>Attitude Propagator</i> (Propagador de Atitude)
RIGs	<i>Rate-Integrating Gyros</i> (Giroscópios de Integração de Velocidade)
$SOEH_{\infty}F$	<i>Second-Order Extended H_{∞} Filter</i> (Filtro H_{∞} Estendido de Segunda Ordem)
WFI	<i>Wide Field Imager</i> (Imageador de Amplo Campo de Visada)

Esta página foi intencionalmente deixada em branco.

Lista de símbolos

$E(\cdot)$	Esperança matemática de uma variável aleatória
\vec{e}	Eixo de rotação do quatérnion
$\tilde{\mathbf{F}}_k$	Matriz Jacobiana de $f(\cdot)$ em representação reduzida
$\tilde{\mathbf{F}}_{k-1}$	Matriz Jacobiana de $f(\cdot)$ em representação reduzida no tempo anterior
$f(\cdot)$	Função vetorial não linear referente aos processos
f_i	i -ésimo elemento de $f(\cdot)$
$\tilde{\mathbf{G}}_k$	Matriz referente aos ruídos de processo em representação reduzida
$\tilde{\mathbf{H}}_k$	Matriz Jacobiana de $h(\cdot)$ em representação reduzida
$h(\cdot)$	Função vetorial não linear referente às medidas
h_i	i -ésimo elemento de $h(\cdot)$
\mathbf{I}	Matriz identidade
$(\mathbf{i}, \mathbf{j}, \mathbf{k})$	Vetores unitários pertencentes à parte vetorial do quatérnion
i_{op}	Ângulo de inclinação da órbita
J	Função custo modificada
J_1	Função custo dos filtros H_∞
J^*	Problema de <i>minimax</i> da função custo
\mathbf{K}_k	Ganho dos filtros H_∞
$\bar{\mathbf{K}}_k$	Ganho do EKF
$\tilde{\mathbf{K}}_k$	Ganho do EKF em representação reduzida
k	Índice de tempo discreto
\mathbf{L}_k	Matriz de combinação linear do estado
$\mathbf{M}(\boldsymbol{\omega}, \Delta t)$	Função de cinemática discretizada
\mathbf{O}	Origem do sistema de coordenadas

\mathbf{P}_0	Matriz de covariância dos estados inicial
\mathbf{P}_k	Matriz de covariância dos estados
$\tilde{\mathbf{P}}_k^-$	Matriz de covariância dos estados na predição
$\tilde{\mathbf{P}}_k^+$	Matriz de covariância dos estados na atualização
\mathbf{P}_{k+1}	Matriz de covariância dos estados futuro
$\tilde{\mathbf{P}}_{k-1}^+$	Matriz de covariância dos estados na atualização anterior
$\tilde{\mathbf{Q}}_k$	Matriz de covariância do ruído do processo modificada
\mathbf{Q}_k	Matriz de covariância do ruído do processo
\mathbf{q}	Quatérnion
$\mathbf{q}(t)$	Quatérnion em tempo contínuo
\mathbf{q}_k	Quatérnion em tempo discreto
\mathbf{q}_{k+1}	Quatérnion em tempo discreto futuro
$\mathbf{q}_{1:3}$	Parte vetorial do quatérnion
$\dot{\mathbf{q}}$	Derivada do quatérnion em relação ao tempo
q_i	Componente i do quatérnion
\mathbf{R}_k	Matriz de covariância do ruído de medidas
\mathfrak{R}	Matriz de atitude
$\mathfrak{R}(i,j)$	Elemento na linha i e coluna j da matriz de atitude
$\mathfrak{R}(\mathbf{q})$	Matriz de atitude obtida através de quatérnions
\mathbf{S}_k	Matriz de ponderação da estimação dos estados
$\bar{\mathbf{S}}_k$	Matriz de ponderação da estimação dos estados modificada
(S_x, S_y, S_z)	Componentes do vetor de direção do Sol no Sistema de Atitude do satélite
$(S_{0_x}, S_{0_y}, S_{0_z})$	Componentes do vetor de direção do Sol no Sistema de Referência Orbital
\mathbf{S}_Ξ	Matriz de transformação de representação
t	Índice de tempo contínuo

v_i	Ruído branco gaussiano do girômetro ao redor do eixo i ($i = x, y, z$)
$v_{\alpha\theta}$	Ruído branco gaussiano do DSS nas medidas de arfagem.
$v_{\alpha\psi}$	Ruído branco gaussiano do DSS nas medidas de guinada.
γ	Coefficiente de performance dos filtros H_∞
v_{θ_H}	Ruído branco gaussiano do IRES nas medidas de arfagem.
v_{ϕ_H}	Ruído branco gaussiano do IRES nas medidas de rolamento.
\mathbf{v}	Vetor de ruídos de medida
\mathbf{v}_k	Vetor de ruídos de medida
\mathbf{w}	Vetor de ruídos de processo
\mathbf{w}_k	Vetor de ruídos de processo
(X, Y, Z)	Eixos do Sistema de Referência Equatorial Terrestre
(x_b, y_b, z_b)	Eixos do Sistema de Referência de Atitude
(x_o, y_o, z_o)	Eixos do Sistema de Referência Orbital Local
(x_{op}, y_{op}, z_{op})	Eixos do Sistema de Referência Orbital
\mathbf{x}	Vetor de estados
$\dot{\mathbf{x}}$	Derivada do vetor de estados em relação ao tempo
\mathbf{x}_0	Vetor de estados inicial
$\hat{\mathbf{x}}_0$	Vetor de estados estimado inicial
\mathbf{x}_k	Vetor de estados em tempo discreto
\mathbf{x}_{k+1}	Vetor de estados em tempo discreto futuro
$\hat{\mathbf{x}}_k$	Vetor de estados em tempo discreto estimado
$\hat{\mathbf{x}}_k^-$	Vetor de estados predito
$\hat{\mathbf{x}}_k^+$	Vetor de estados atualizado
$\hat{\mathbf{x}}_{k+1}$	Vetor de estados estimado futuro
$\hat{\mathbf{x}}_{k-1}^+$	Vetor de estados atualizado no tempo anterior
\mathbf{y}	Vetor de saídas

\mathbf{y}_k	Vetor de saídas em tempo discreto
$\tilde{\mathbf{y}}_k$	Resíduo das medidas no SOEH _∞ F
\mathbf{z}_k	Vetor da combinação linear do estado
$\hat{\mathbf{z}}_k$	Vetor da combinação linear do estado estimado
α_θ	Medida do DSS associada ao ângulo de arfagem
α_ψ	Medida do DSS associada ao ângulo de guinada
β	Ângulos do argumento do perigeu e anomalia verdadeira somados
Δt	Intervalo de amostragem
$\Delta\Theta_i$	Deslocamento angular medido pelos RIGs ao redor do eixo i ($i = x, y, z$)
ϵ	Vetor de vieses de girômetro
$\dot{\epsilon}$	Derivada do vetor de vieses de girômetro em relação ao tempo
ϵ_i	Viés de girômetro no eixo i ($i = x, y, z$)
η	Escalar positivo entre 0 e 1
θ	Ângulo de arfagem real
θ_H	Ângulo de arfagem medido pelo IRES
ϑ	Ângulo de rotação do quatérnion
λ_0	Multiplicador de Lagrange inicial
λ_k	Multiplicador de Lagrange
λ_{k+1}	Multiplicador de Lagrange futuro
ξ	Escalar positivo para prevenir a singularidade de $\tilde{\mathbf{F}}_k \tilde{\mathbf{F}}_k^T$
ϕ	Ângulo de rolamento real
ϕ_H	Ângulo de rolamento medido pelo IRES
φ_i^f	Vetor com 1 em seus i -ésimos elementos e 0 no resto.
φ_i^h	Vetor com 1 em seus i -ésimos elementos e 0 no resto.
ψ	Ângulo de guinada real

ω	Vetor de velocidades angulares do corpo
ω_i	Velocidade angular do corpo ao redor do eixo i ($i = x, y, z$)
$\omega(t)$	Vetor de velocidades angulares do corpo em tempo contínuo
Ω	Longitude do nodo ascendente
∞	Infinito

Esta página foi intencionalmente deixada em branco.

Sumário

1	INTRODUÇÃO	1
1.1	Justificativa	3
1.2	Objetivos	3
1.3	Metodologia	4
1.4	Organização do Trabalho	4
2	REVISÃO BIBLIOGRÁFICA	7
3	DESENVOLVIMENTO	11
3.1	CBERS-2	11
3.2	Sistemas de Referência	14
3.2.1	Sistema de Referência Equatorial Terrestre	14
3.2.2	Sistema de Referência Orbital	15
3.2.3	Sistema de Referência Orbital Local	16
3.2.4	Sistema de Referência de Atitude	17
3.3	Representação de Atitude	18
3.3.1	Representação de Atitude com Ângulos de Euler	18
3.3.2	Quatérnions	20
3.3.2.1	Representação de Atitude com Quatérnions	22
3.3.2.2	Cinemática de Atitude em Quatérnions	23
3.4	Modelos Matemáticos dos Sensores de Atitude	24
3.4.1	Modelo do Sensor Girômetro	24
3.4.2	Modelo do Sensor da Terra Infravermelho	25
3.4.3	Modelo do Sensor Solar Digital	25
3.5	Métodos de Estimação de Estados	26
3.5.1	Filtro de Kalman Estendido	26
3.5.2	Filtro H_{∞} Estendido	28
3.5.3	Filtro H_{∞} Estendido de Segunda Ordem	30
4	APLICAÇÃO	33
4.1	Órbita Simulada	33
4.2	Sensores Simulados	34
4.3	Sistema Aplicado	34
4.4	Integração dos Filtros e Sensores	34
5	RESULTADOS	37

5.1	Estimação Utilizando Dados Simulados 1	37
5.2	Estimação Utilizando Dados Simulados 2	45
5.3	Estimação Utilizando Dados Simulados 3	49
5.4	Estimação Utilizando Dados Simulados 4	51
5.5	Estimação Utilizando Dados Reais	54
6	CONCLUSÃO	61
	REFERÊNCIAS	63
	APÊNDICE A – CÓDIGOS DOS FILTROS EM LINGUAGEM MA- TLAB	67
A.1	Parâmetros de Inicialização	67
A.2	Filtro de Kalman Estendido	71
A.3	Filtro H_{∞} Estendido	77
A.4	Filtro H_{∞} Estendido de Segunda Ordem	84

1 Introdução

A atitude, que também pode ser chamada de orientação ou pose, de uma nave espacial é a forma com que esta se orienta no espaço (WERTZ, 1978), ela pode ser definida como a orientação de um objeto com relação a um sistema de coordenadas específico, de forma simplificada, é a direção para a qual um corpo aponta no espaço. O conhecimento da atitude de um veículo é fundamental em diversas áreas (CORDEIRO, 2012), especialmente nas áreas da tecnologia que envolvem veículos não tripulados, automatizados e autônomos. Nos satélites, por exemplo, que não são tripulados e são extremamente automatizados, o controle da atitude é necessário para a orientação adequada de painéis solares, antenas, dissipadores de calor, propulsores para manobra orbital, câmeras, dentre outros (SANTOS, 2008), sendo que este controle de atitude é dependente do conhecimento da mesma.

De acordo com Tsai (2005), em naves espaciais, a execução precisa de uma manobra é crítica para a estabilidade, portanto, a falha em executar manobras com acurácia causará a necessidade de manobras de correção custosas e frequentes, que aumentariam a provisão de combustível e o custo da operação. Erros na direção do impulso em manobras orbitais podem causar um aumento de mais de 100% em tempo e em consumo de combustível, ocasionalmente levando à perda da missão (QUINSAC et al., 2020). Além disso, em satélites de observação da terra, a acurácia da observação de atitude é sempre o maior contribuidor para a atuação do satélite em sua missão (SUN et al., 2016). Dessa forma, o direcionamento de suas câmeras está diretamente ligado à medida de orientação do veículo, e erros na atitude podem causar um direcionamento inadequado das câmeras. Sendo assim, em veículos espaciais, a determinação adequada da atitude é vital para o cumprimento de seus objetivos.

O “conhecimento” da atitude de um veículo é dado por um sistema de determinação de atitude, que provê estimativas através do processamento de informações obtidas a partir de sensores (CORDEIRO, 2012). Isto posto, devido a imperfeições, os sensores podem produzir medidas acrescidas de erros (SILVA, 2016), que atrapalham na aquisição da orientação verdadeira. Por isso se faz necessário processar essas informações, através de algoritmos que filtrem esses erros, permitindo assim a estimação de uma atitude mais próxima da verdadeira.

Devido à importância, e até mesmo a necessidade, da estimação de atitude em missões espaciais, este trabalho tem por objetivo principal a realização da estimação de atitude de um satélite artificial, que deve ser feita através da implementação de métodos de estimação. O satélite a ser estudado neste trabalho é o CBERS-2 (*China*

Brazil Earth Resources Satellite 2, ou Satélite Sino-Brasileiro de Recursos Terrestres 2), um satélite utilizado para a observação da Terra, operado tanto pela China quanto pelo Brasil entre os anos de 2003 e 2009. Este satélite foi escolhido para a estimação de atitude devido à grande quantidade de estudos já realizados a respeito dele, como se pode ver nos trabalhos de Garcia (2011), Silva, Kuga e Zanardi (2014a), Silva (2016), dentre outros, pois assim existe a possibilidade de se verificar os resultados obtidos com trabalhos científicos já realizados.

A estimação de atitude do CBERS-2 é feita através da leitura de medidas de sensores posicionados em seu corpo. Neste caso são utilizados: dois Sensores Solares Digitais (*Digital Sun Sensor - DSS*), que fornecem medidas indiretas de arfagem (θ) e guinada (ψ) respectivamente, através de uma função não linear da atitude, dois Sensores da Terra Infravermelhos (*Infrared Earth Sensor - IRES*), que fornecem a medida direta dos ângulos de rolamento (ϕ) e arfagem respectivamente, e três Giroscópios de Integração de Velocidade (*Rate-Integrating Gyros - RIGs*), que são usados para obter as velocidades angulares nos três eixos do sistema de referência de coordenadas do seu corpo (GARCIA, 2011).

Para o fim de estimar a atitude, o renomado Filtro de Kalman (KALMAN, 1960) é uma solução ótima, por mérito de ser um dos principais e mais utilizados métodos de estimação na atualidade. Este é o “cavalo de batalha” da estimação de estados (SIMON, 2006), e se estabeleceu como uma ferramenta fundamental para a análise e solução de uma extensa classe de problemas de estimação (MCGEE; SCHMIDT, 1985). Além do Filtro de Kalman, também considera-se o Filtro H_∞ (BANAVAR, 1992), a fim de se ter uma alternativa a ser comparada no método de estimação utilizado. Este foi considerado por ser uma alternativa ao Filtro de Kalman, sendo mais robusto com relação a erros na modelagem dos ruídos e da dinâmica do sistema (SIMON, 2006).

Devido ao fato das medidas do CBERS-2 conterem não-linearidades com relação à atitude, se faz necessário o uso de métodos de estimação não lineares. Sendo assim, serão utilizadas as versões estendidas dos filtros já mencionados, sendo eles o Filtro de Kalman Estendido (LEFFERTS; MARKLEY; SHUSTER, 1982) e o Filtro H_∞ Estendido (SILVA; KUGA; ZANARDI, 2014a), que são capazes de lidar com a estimação de sistemas não lineares através da linearização do sistema pela expansão em série de Taylor de primeira ordem. Além destes, no caso da filtragem H_∞ , também foi considerado o Filtro H_∞ Estendido de Segunda Ordem (HU; YANG, 2011), que considera a segunda ordem da expansão em série de Taylor, e por isso pode reduzir os erros procedentes da simplificação do sistema para casos de maiores graus de não linearidades.

A forma de parametrização da atitude utilizada pelos filtros para a cinemática de atitude será através dos quatérnions (HAMILTON, 1844a), que serão utilizados devido

às seguintes praticidades: a cinemática de atitude em quatérnions é tratada de forma linear, de forma que elimina a necessidade de simplificação e é computacionalmente eficaz; esta representação de atitude é livre de singularidades, de forma que o *gimbal lock*, que é a perda de um grau de liberdade da atitude, é evitado; e a matriz de mudança de atitude obtida a partir dos quatérnions é algébrica, eliminando a necessidade de funções transcendentais (LEFFERTS; MARKLEY; SHUSTER, 1982).

1.1 Justificativa

A justificativa se dá pela importância da estimação da atitude em missões não somente espaciais, mas principalmente qualquer sistema em missão não tripulada existente, como foguetes, sondas, satélites, drones, ferramentas ou veículos autônomos e automáticos, pois a falta da estimação de atitude inviabilizaria estas aplicações. Além disso, a estimação de atitude também ajuda em missões tripuladas, como no caso de aviões, onde os sistemas auxiliam os pilotos a operar a aeronave, garantindo maior segurança em voo.

1.2 Objetivos

O objetivo geral deste trabalho é a realização da estimação de atitude do satélite CBERS-2, utilizando diferentes métodos de estimação, e a comparação de seus desempenhos através dos erros de estimação de atitude.

A partir do objetivo geral, os objetivos específicos são:

- Introduzir a teoria sobre estimação de atitude, decorrendo sobre os quatérnions, os sensores de atitude e os diferentes métodos de estimação utilizados.
- Implementar os sensores e os filtros mencionados em um programa de computador e simular a estimação de atitude, baseando-se nas características do satélite CBERS-2.
- Aplicar o Filtro de Kalman Estendido, o Filtro H_{∞} Estendido e o Filtro H_{∞} Estendido de Segunda Ordem na estimação de atitude do satélite CBERS-2, utilizando ambos dados simulados e reais.
- Analisar os desempenhos dos métodos de estimação utilizados, com base no erro de estimação.

1.3 Metodologia

Os cálculos deste trabalho foram realizados na forma de simulação numérica computacional. A estimação de estados foi realizada através do Filtro de Kalman Estendido, Filtro H_∞ Estendido e Filtro H_∞ Estendido de Segunda Ordem (ver Seção 3.5), a partir de dados simulados e reais dos sensores de atitude do satélite (ver Seção 3.4 e Capítulo 4), e utilizando quatérnions para a cinemática de atitude (ver Seção 3.3). Para os resultados, ao invés de representar a atitude em quatérnions, esta foi representada em ângulos de Euler, pois os quatérnions são altamente complexos e não possuem uma interpretação física compreensível, enquanto que os ângulos de Euler são facilmente visualizáveis. No caso dos dados simulados, a dinâmica orbital, a atitude e as informações advindas dos sensores foram geradas a partir do *toolbox* PROPAT (CAR-RARA, 2015). Por fim, os desempenhos dos filtros foram comparados através do erro de estimação de atitude (ver Seções 5 e 6).

1.4 Organização do Trabalho

O Capítulo 1 se dedica a introduzir o tema, sua importância, a justificativa do trabalho, os objetivos e a metodologia do trabalho.

O Capítulo 2 apresenta a revisão bibliográfica para este trabalho, que inclui trabalhos realizados previamente a respeito da estimação de estados utilizando o Filtro de Kalman Estendido, Filtro H_∞ Estendido e Filtro H_∞ Estendido de Segunda Ordem, além dos conceitos relacionados à estimação de atitude.

O Capítulo 3 se dedica a introduzir o satélite CBERS-2 e a teoria sobre a estimação de atitude utilizada, decorrendo sobre os sistemas de referência relacionados ao movimento do satélite (Seção 3.2), a representação de atitude (Seção 3.3), os sensores de atitude, que são Giroscópios de Integração de Velocidade, Sensores Solares Digitais e Sensores da Terra Infravermelhos (Seção 3.4), e os diferentes métodos de estimação utilizados, sendo eles o Filtro de Kalman Estendido, Filtro H_∞ Estendido e Filtro H_∞ Estendido de Segunda Ordem (Seção 3.5).

O Capítulo 4 contém os dados utilizados para a aplicação do trabalho em simulação numérica computacional, que inclui os elementos da órbita do satélite, as características dos sensores, o sistema utilizado e a forma com que foi feita a integração dos filtros com os dados dos sensores.

O Capítulo 5 apresenta os dados de inicialização dos filtros juntamente com os respectivos resultados, para ambos os casos simulados e o caso das medidas reais.

O Capítulo 6 apresenta as conclusões a respeito dos resultados obtidos, onde se comenta sobre os desempenhos dos filtros utilizados.

O último capítulo seção contém as referências bibliográficas utilizadas ao longo deste trabalho.

Em apêndice, se encontram os códigos relacionados aos filtros utilizados em linguagem MATLAB.

Esta página foi intencionalmente deixada em branco.

2 Revisão Bibliográfica

Neste capítulo, encontra-se uma breve revisão dos trabalhos considerados como fontes importantes para a realização deste, os quais se dedicam ao estudo de métodos de estimação de sistemas não lineares, como o Filtro de Kalman Estendido, o Filtro H_∞ Estendido e o Filtro H_∞ Estendido de Segunda Ordem. Os estudos estão citados em ordem cronológica.

No trabalho feito por Lefferts, Markley e Shuster (1982), são avaliados diferentes métodos de estimação de atitude utilizando a filtragem de Kalman e seus desenvolvimentos nas duas décadas anteriores à sua publicação, com foco na estimação de atitude de satélites equipados com girômetros em seus três eixos, dos quais são obtidos as velocidades angulares do satélite, além de outros sensores de atitude, como sensores solares e rastreadores de estrelas, para atualização dos dados. Para a representação de atitude são utilizados quatérnions, e os vieses de girômetro são tratados como estados, juntamente com os quatérnions. O trabalho informa que a matriz de covariância para o vetor de estados com 7 elementos é singular, devido à não independência dos elementos do quatérnion, que estão relacionados pela restrição de norma unitária do quatérnion de atitude. Sendo assim, são apresentadas três formas de contornar este problema: a representação reduzida da matriz de covariância, a representação da covariância truncada, e a representação da covariância fixa no corpo. A partir deste trabalho, foi utilizado o método de estimação de atitude com quatérnions, juntamente com os vieses de girômetro no vetor de estados, de forma que também é feita a estimação destes vieses. Além disso, foi utilizada a formulação do Filtro de Kalman Estendido apresentada, com a representação reduzida da matriz de covariância, visando evitar as condições de singularidade geradas pela representação original.

O trabalho de Shaked e Berman (1995) apresenta o desenvolvimento do Filtro H_∞ , originalmente linear, para a estimação de um sistema não linear em tempo discreto, desta forma gerando a versão estendida do filtro. É utilizada uma abordagem baseada no conceito de sistemas dissipativos, e introduz-se o estimador de forma que, teoricamente, o erro estimado tem a garantia de ser limitado por um certo nível prescrito. Para atingir as condições necessárias para tal estimador, é utilizado um análogo da desigualdade de Hamilton-Jacobi. Por fim, tanto o Filtro H_∞ Estendido quanto o Filtro de Kalman Estendido são aplicados na estimação do sistema, e seus resultados são comparados, onde se encontra um desempenho substancialmente superior por parte do Filtro H_∞ Estendido. Este trabalho serviu como motivação para a realização da comparação entre os filtros H_∞ e o Filtro de Kalman que é realizada neste.

Na obra de Simon (2006), é apresentada a teoria básica por trás da estimação de estados, onde se discute os conceitos de sistemas lineares, probabilidade, estimação recursiva, e a propagação de estados e covariâncias. Neste trabalho, é feito o desenvolvido e a discussão a respeito das características do Filtro de Kalman linear e do Filtro de Kalman Estendido, além do Filtro H_∞ em sua versão linear. Além disso, são apresentadas estratégias para a configuração dos parâmetros de inicialização dos filtros, de forma a deixá-los mais robustos quanto às incertezas de modelagem. Desta forma, a partir desta obra, foram utilizadas as estratégias de configuração descritas para o fim de melhorar a acurácia da estimação de atitude, principalmente quando há a existência de erros na modelagem do sistema.

No trabalho feito por Garcia (2011), o Filtro de Kalman Estendido e o Filtro de Kalman *Unscented* são aplicados para a estimação de atitude dos satélites CBERS-2 e CBERS-2B, que possuem dinâmicas não lineares, baseando-se nos dados reais dos sensores a bordo dos satélites. Os sensores de atitude dos satélites incluem Sensores da Terra Infravermelhos, Sensores Solares Digitais, Giroscópios de Integração de Velocidade e Sensores de Estrelas, e seus modelos matemáticos foram descritos neste trabalho. Para a representação de atitude foram utilizados ângulos de Euler, quatérnions e incrementos de quatérnions. Com os resultados, concluiu-se que o Filtro de Kalman *Unscented* possui maior robustez que o Filtro de Kalman Estendido, principalmente quando há condições iniciais degradadas, porém isto a um maior custo computacional. A partir deste trabalho, os modelos matemáticos dos sensores de atitude descritos foram utilizados para a estimação da atitude baseada nos dados simulados e reais do satélite CBERS-2.

No trabalho de Hu e Yang (2011), o Filtro H_∞ Estendido de Segunda Ordem para estimação de sistemas não lineares em tempo discreto é desenvolvido, baseando-se em uma aproximação da matriz de erro quadrático, devido à alta complexidade gerada pelas derivadas de segunda ordem. A solução do filtro é obtida através de uma abordagem da teoria de jogos, comum ao desenvolvimento do Filtro H_∞ , em que o objetivo é de minimizar o erro em uma estimativa de pior caso, onde se considera que a natureza tende a maximizar o erro inicial e os ruídos dos sensores, maximizando assim o erro de estimativa dos estados. É realizada a implementação do Filtro H_∞ Estendido, Filtro H_∞ Estendido de Segunda Ordem e Filtro de Kalman Estendido de Segunda Ordem em uma simulação de rastreamento dos estados de um veículo, onde se compara os desempenhos dos mesmos. Os resultados mostram um desempenho superior por parte do Filtro H_∞ Estendido de Segunda Ordem. A partir deste trabalho, o Filtro H_∞ Estendido de Segunda Ordem desenvolvido foi utilizado para a estimação dos estados do CBERS-2.

O trabalho de Silva, Kuga e Zanardi (2014a) realiza o desenvolvimento do Filtro

de Kalman Estendido e do Filtro H_∞ Estendido, obtido através de uma abordagem da teoria de jogos, e realiza sua aplicação para a estimação de atitude e da deriva de girômetros do satélite CBERS-2, utilizando como representação de atitude os ângulos de Euler. São utilizados dados reais dos sensores do satélite, que incluem medidas de três girômetros RIGs, dois Sensores Solares Digitais e dois Sensores da Terra Infravermelhos. Os desempenhos dos filtros foram comparados a partir dos resultados, e encontrou-se uma maior acurácia por parte do Filtro H_∞ Estendido na estimação da deriva de girômetros. A partir deste trabalho, o Filtro H_∞ Estendido formulado foi utilizado para a estimação da atitude e dos vieses dos girômetros.

Na obra de Markley e Crassidis (2014), são apresentados os fundamentos de determinação de atitude e controle de veículos espaciais, onde se discute sobre sistemas de referências de coordenadas, como um Sistema Equatorial Terrestre, diferentes métodos de representação de atitude, como os ângulos de Euler e os Quatérnions, cinemática e dinâmica de atitude, e também se apresentam modelos matemáticos de sensores de atitude. Além disso, são introduzidos diferentes métodos de determinação de atitude, assim como a estimação e calibração dos sensores, e o controle da espaçonave. A partir desta obra, foram utilizados os conceitos referentes à representação de atitude e à cinemática de atitude utilizando os quatérnions.

No trabalho desenvolvido por Silva (2016), utiliza-se o Filtro de Kalman Estendido, o Filtro H_∞ Estendido e o Filtro H_∞ Estendido de Segunda Ordem, que realizam a expansão em série de Taylor para a simplificação dos sistemas de equações não lineares, e além deles, o Filtro de Partículas Regularizado e o Filtro de Kalman *Unscented*, que não passam pela linearização, para a estimação da atitude e dos vieses do satélite CBERS-2. A aplicação dos filtros foi feita utilizando a representação de atitude por ângulos de Euler, e foram realizadas estimativas considerando dados tanto simulados quanto reais dos sensores do satélite, que incluem medidas de um trio de girômetros RIGs, dois Sensores Solares Digitais e dois Sensores da Terra Infravermelhos. Além disso, o trabalho apresenta uma breve fundamentação teórica contendo os sistemas de referência de coordenadas relacionados ao satélite e a representação de atitude através dos ângulos de Euler. Por fim os resultados dos diferentes filtros foram comparados em termos de acurácia e custo computacional. A partir deste trabalho, foi utilizada a fundamentação teórica desenvolvida dos sistemas de referência e da representação de atitude, além disso, a estratégia de realizar a estimação de atitude utilizando dados simulados e reais, para fins de comparação, foi baseada neste trabalho.

Esta página foi intencionalmente deixada em branco.

3 Desenvolvimento

Neste capítulo, será apresentado o satélite CBERS-2, além de uma breve fundamentação teórica com relação aos conceitos utilizados para a realização da estimação de atitude deste trabalho. Lembrando que o objetivo do trabalho é a realização da estimação de atitude, portanto procurou-se apresentar a teoria pertinente à sua aplicação, de forma que o desenvolvimento do trabalho não teve como enfoque os desenvolvimentos matemáticos e as provas matemáticas a respeito das equações apresentadas.

3.1 CBERS-2

De acordo com o INPE (2014), o satélite CBERS-2 surgiu a partir do programa CBERS, implantado em 1988 através de um acordo de parceria envolvendo o INPE (Instituto Nacional de Pesquisas Espaciais), no Brasil, e a CAST (*China Academy of Space Technology*, ou Academia Chinesa de Tecnologia Espacial), na China, com o objetivo de desenvolver satélites de sensoriamento remoto, ou observação da Terra. Inicialmente visou-se construir dois satélites, o CBERS-1 (lançado em 1999 e inativo desde 2003) e o CBERS-2 (lançado em 2003 e inativo desde 2009), mas, com o sucesso da missão, em 2002 o programa foi continuado, com a construção de mais quatro satélites: CBERS-2B (lançado em 2007 e inativo desde 2010), CBERS-3 (perdido no lançamento em 2013), CBERS-4 (lançado em 2014 e ainda ativo) e CBERS-4A (lançado em 2019 e ainda ativo).

O CBERS-2 produziu 175 mil imagens durante seu período de operação. Este foi o primeiro satélite de observação da Terra no mundo a adotar a política de distribuição gratuita de imagens, fazendo da política de distribuição do Brasil um grande exemplo mundial e contribuindo com a disseminação do uso de imagens de satélites. O satélite CBERS-2 pode ser visualizado através da Figura 1, e seus diferentes campos de visada podem ser visualizados na Figura 2.

Para realizar sua função de observação da Terra, o satélite é equipado com as seguintes câmeras (INPE, 2018b):

- Uma câmera WFI (*Wide Field Imager*, ou Imageador de Amplo Campo de Visada), que produz imagens em uma faixa de 890 km de largura, permitindo a obtenção de imagens com resolução espacial de 260 m e obtendo uma cobertura completa do globo em um período aproximado de cinco dias;
- Uma câmera imageadora de alta resolução CCD (*Charge-Coupled Device*, ou



Figura 1 – Satélite CBERS-2. Fonte: INPE (2018a).

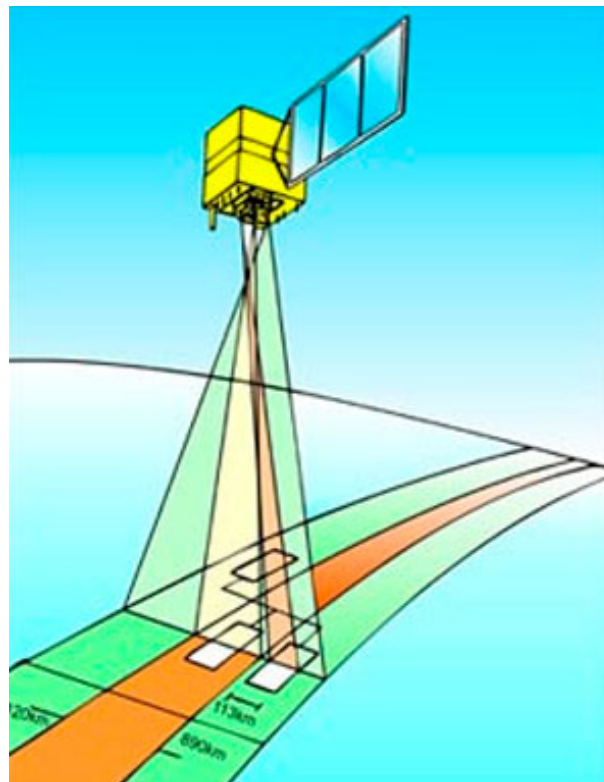


Figura 2 – Os diferentes campos de visada do CBERS-2. Fonte: INPE (2018b).

Dispositivo de Carga Acoplada), que fornece imagens a uma faixa de 113 km de largura, com uma resolução de 20 m, além disso, qualquer fenômeno detectado pela WFI pode ser focalizado pela câmera CCD, para estudos mais detalhados;

- Uma câmera IRMSS (*Infrared Multispectral Scanner*, ou Imageador por Varredura

de Média Resolução), que produz imagens a uma faixa de 120 km de largura com uma resolução de 80 m;

Além das câmeras, uma característica essencial para a missão do satélite é a sua órbita ao redor da Terra. A órbita escolhida para o CBERS-2 é uma órbita heliossíncrona, possuindo uma altitude aproximada de 778 km acima da superfície terrestre, com uma inclinação de $98,504^\circ$ e um período de 100,26 minutos. Esta órbita permite ao satélite cruzar a linha do Equador sempre na mesma hora local, às 10:30 da manhã, permitindo assim que se tenha as mesmas condições de iluminação solar para a comparação de fotos tiradas em dias diferentes. Com aproximadamente 14 revoluções por dia, o satélite consegue fazer a varredura da Terra em 26 dias, que é o tempo necessário para se obter imagens de todo o globo terrestre com suas câmeras (INPE, 2018c). A Figura 3 é uma imagem representativa do CBERS-2 em sua órbita.

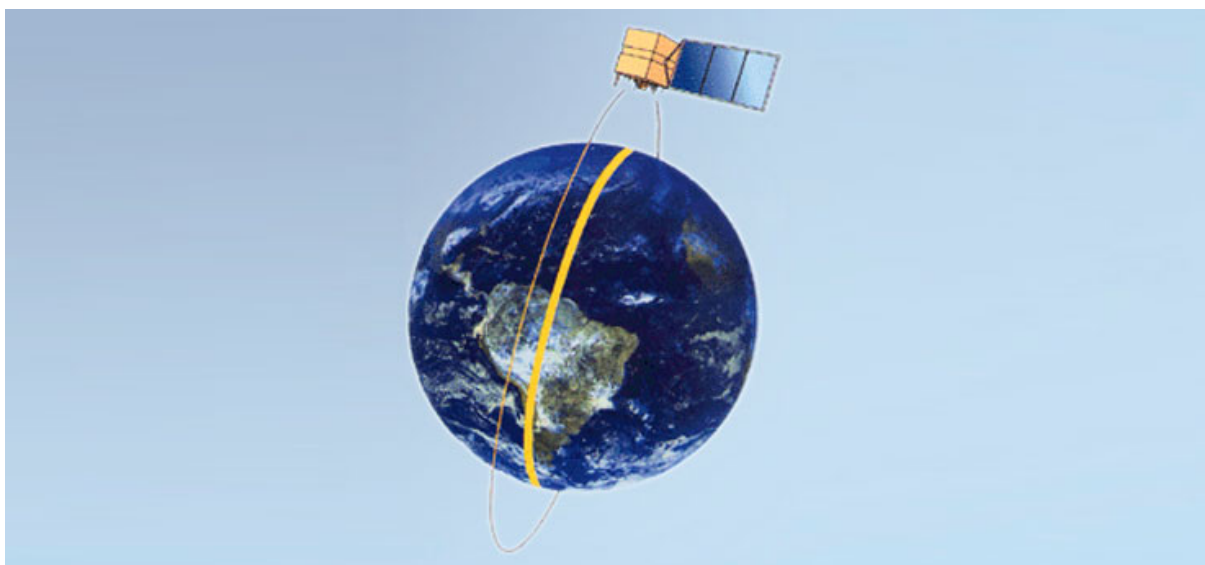


Figura 3 – Órbita do CBERS-2. Fonte: INPE (2018c).

Para realizar a estimativa de atitude, o satélite conta com os seguintes sensores (SILVA, 2016), a serem discutidos na Seção 3.4:

- Dois Sensores Solares Digitais (DSS).
- Dois Sensores da Terra Infravermelho (IRES).
- Três Giroscópios de Integração de Velocidade (RIGs).

Mais informações a respeito do satélite CBERS-2 e seu programa podem ser encontradas em seu site oficial <<http://www.cbbers.inpe.br>>.

3.2 Sistemas de Referência

Nesta seção, são apresentados os sistemas de referência de coordenadas que estão envolvidos na estimação de atitude do satélite deste trabalho.

3.2.1 Sistema de Referência Equatorial Terrestre

O Sistema de Referência Equatorial Terrestre é um sistema de referência de coordenadas usado para descrever o movimento ao redor da Terra. Neste caso, assume-se que a Terra é esférica, e que o baricentro da órbita do satélite, o centro geométrico e o centro de massa da Terra coincidem. Este sistema pode ser assumido como um sistema inercial para se analisar o movimento de satélites ao redor da Terra.

O Sistema de Referência Equatorial Terrestre pode ser visualizado através da Figura 4, e é composto por:

- Origem O : Centro de massa da Terra.
- Eixo X : Posicionado no plano equatorial e direcionado ao Ponto Vernal.
- Eixo Y : Perpendicular aos eixos X e Z .
- Eixo Z : Normal ao plano equatorial, direcionado ao norte, perpendicular aos eixos X e Y .

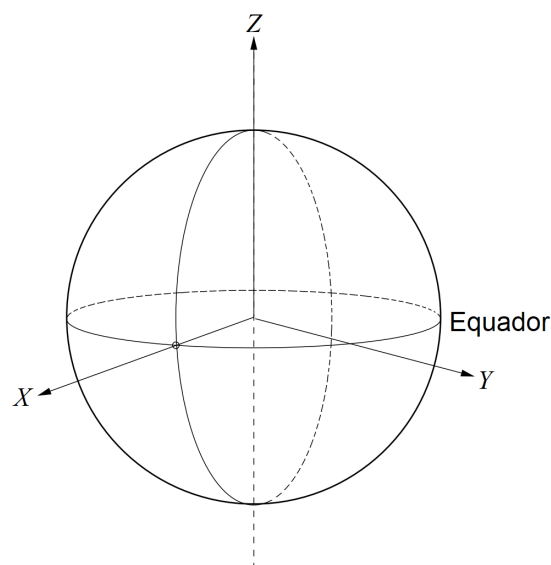


Figura 4 – Sistema de Referência Equatorial Terrestre. Fonte: Adaptado de Fehse (2003).

3.2.2 Sistema de Referência Orbital

O Sistema de Referência Orbital descreve o movimento do satélite em seu plano orbital. Este pode ser visualizado através das Figuras 5 e 6, e é composto por:

- Origem O : Centro de massa da Terra.
- Eixo x_{op} : Posicionado no plano equatorial e direcionado ao nodo ascendente da órbita do satélite.
- Eixo y_{op} : Perpendicular aos eixos x_{op} e z_{op} .
- Eixo z_{op} : Normal ao plano orbital e inclinado da direção norte pelo ângulo de inclinação da órbita, perpendicular aos eixos x_{op} e y_{op} .

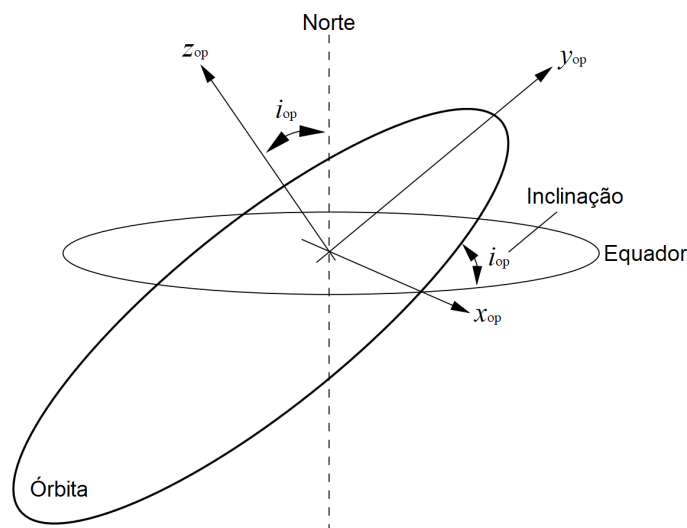


Figura 5 – Sistema de Referência Orbital. Fonte: Adaptado de Fehse (2003).

Para realizar a mudança de representação do vetor de coordenadas, partindo do Sistema de Referência Equatorial Terrestre para o Sistema de Referência Orbital, realiza-se a rotação da coordenada inicial primeiramente pelo ângulo de longitude do nodo ascendente, denominado pelo símbolo Ω , ao redor de Z , e depois a rotação pelo ângulo de inclinação da órbita, denominado por i_{op} , ao redor de x_{op} , como está descrito na Equação (3.1) (SILVA, 2016).

$$\begin{bmatrix} x_{op} \\ y_{op} \\ z_{op} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(i_{op}) & \sin(i_{op}) \\ 0 & -\sin(i_{op}) & \cos(i_{op}) \end{bmatrix} \begin{bmatrix} \cos(\Omega) & \sin(\Omega) & 0 \\ -\sin(\Omega) & \cos(\Omega) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (3.1)$$

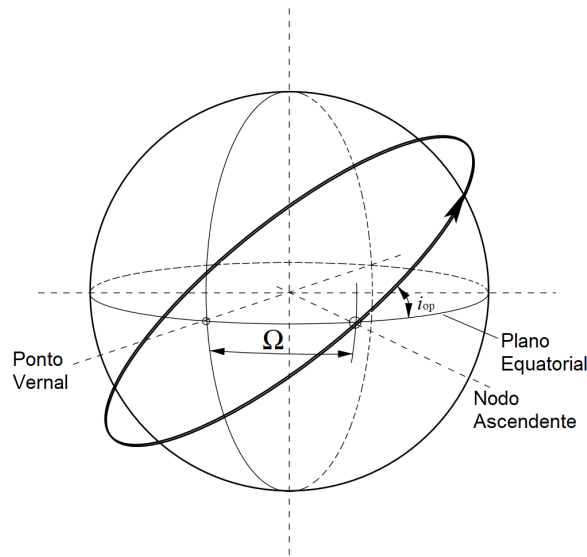


Figura 6 – Longitude do nodo ascendente da órbita. Fonte: Adaptado de Fehse (2003).

3.2.3 Sistema de Referência Orbital Local

O Sistema de Referência Orbital Local descreve o movimento do satélite ao longo de sua órbita. Diferentemente dos anteriores, este sistema é centralizado no centro de massa do próprio satélite. O sistema pode ser visualizado através das Figuras 7 e 8, e é composto por:

- Origem O : Centro de massa do satélite.
- Eixo x_o : Aponta na direção do vetor de velocidade orbital do satélite, tangente à sua órbita.
- Eixo y_o : Perpendicular aos eixos x_o e z_o e normal à órbita.
- Eixo z_o : Aponta na direção do centro de massa da Terra, direção radial, e perpendicular aos eixos x_o e y_o .

Partindo do Sistema de Referência Orbital para o Sistema de Referência Orbital Local, a mudança de representação de coordenadas é realizada a partir de uma rotação de um ângulo β somado de 90° ao redor de z_{op} (onde β se equivale à soma dos ângulos do argumento do perigeu e anomalia verdadeira, indicados na Figura 8), seguida de uma rotação de 90° ao redor de x_o , como representado na Equação (3.2) (SILVA, 2016).

$$\begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \cos(\beta + 90^\circ) & \text{sen}(\beta + 90^\circ) & 0 \\ -\text{sen}(\beta + 90^\circ) & \cos(\beta + 90^\circ) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{op} \\ y_{op} \\ z_{op} \end{bmatrix} \quad (3.2)$$

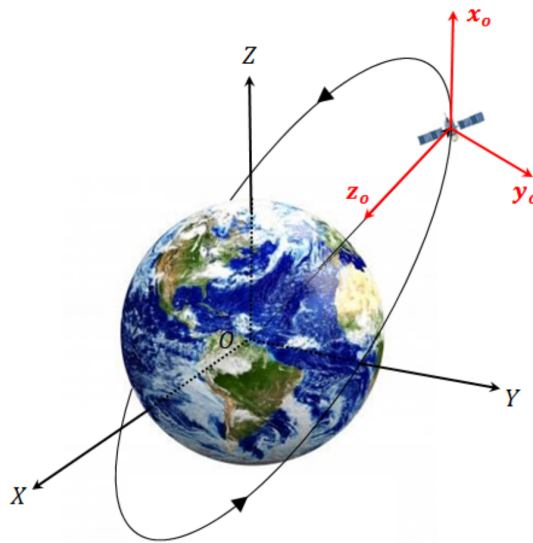


Figura 7 – Sistema de Referência Orbital Local. Fonte: Silva (2016).

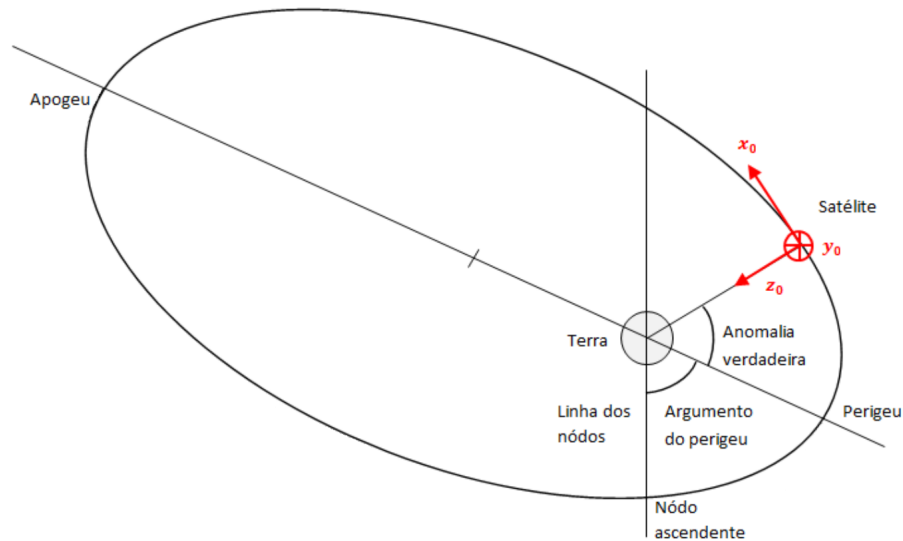


Figura 8 – Sistema de Referência Orbital Local, vista alternativa. Fonte: Adaptado de Silva (2016).

3.2.4 Sistema de Referência de Atitude

Como já dito por Wertz (1978), a atitude de uma nave espacial é sua orientação no espaço. A forma com que se representa a atitude do satélite é através da rotação ao redor do seu centro de massa. Neste caso, uma forma comum e de fácil visualização de representar a atitude do satélite é através do Sistema de Referência de Atitude, que se correlaciona com o Sistema de Referência Orbital Local através dos ângulos ϕ , θ e ψ , a serem discutidos na Seção 3.3.1. O sistema utilizado para representar a atitude pode ser visualizado através da Figura 9, e é composto por:

- Origem O : Centro de massa do satélite.

- Eixo x_b : Eixo nominalmente alinhado com x_o , para o caso dos ângulos θ e ψ iguais a zero.
- Eixo y_b : Eixo nominalmente alinhado com y_o , para o caso dos ângulos ϕ e ψ iguais a zero.
- Eixo z_b : Eixo nominalmente alinhado com z_o , para o caso dos ângulos ϕ e θ iguais a zero.

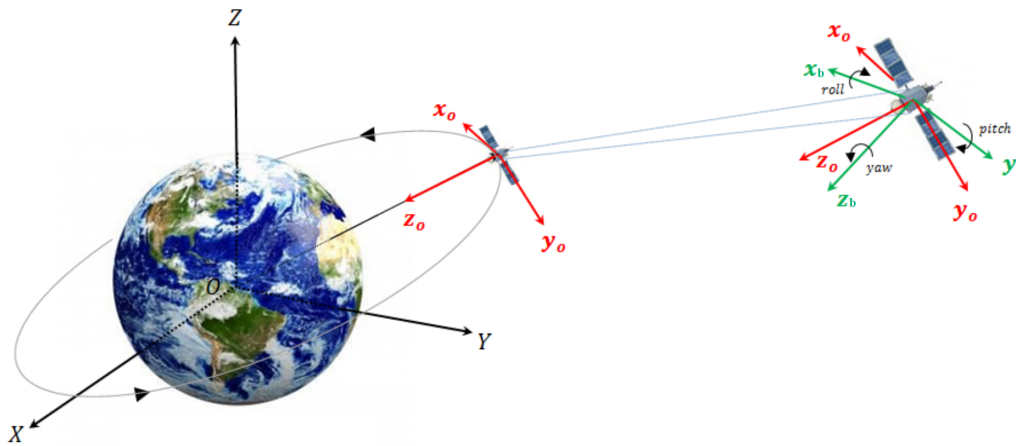


Figura 9 – Sistema de Referência de Atitude. Fonte: Adaptado de Silva, Kuga e Zanardi (2015).

3.3 Representação de Atitude

Nesta seção se encontram as duas formas de representação de atitude utilizadas no trabalho. A representação de atitude com ângulos de Euler foi utilizada para a visualização dos resultados, por ser facilmente compreendida por parte dos seres humanos, enquanto que a representação por quatérnions foi utilizada para a cinemática de atitude, por ser mais eficiente computacionalmente.

3.3.1 Representação de Atitude com Ângulos de Euler

A forma de representação de atitude com ângulos de Euler é uma forma de fácil visualização e compreensão para se representar a atitude, por isso é uma das formas mais comuns de serem utilizadas. Partindo da Seção 3.2.4, a relação entre o Sistema de Referência de Atitude e o Sistema de Referência Orbital Local, dada através dos ângulos de Euler, é definida pelos movimentos de rolamento, ou *roll* (ϕ), arfagem, ou *pitch* (θ) e guinada, ou *yaw* (ψ).

A representação de atitude com ângulos de Euler pode ser realizada utilizando diversas sequências de rotação diferentes, sendo que sequências diferentes geram

resultados diferentes. A sequência 3-2-1 (também chamada de Z-Y-X) foi adotada pelo fato de ser capaz de representar a atitude dentro das condições deste trabalho de maneira adequada, isto pois, para sistemas de referência tridimensionais, até três rotações distintas podem ser necessárias para a representação de atitude. Sendo assim, a sequência 3-2-1 pode ser vista na Figura 10, e é definida da seguinte maneira:

1. Rotação do ângulo ψ , realizada ao redor do eixo z_o .
2. Rotação do ângulo θ , realizada ao redor de um eixo intermediário y' .
3. Rotação do ângulo ϕ , realizada ao redor do eixo x_b .

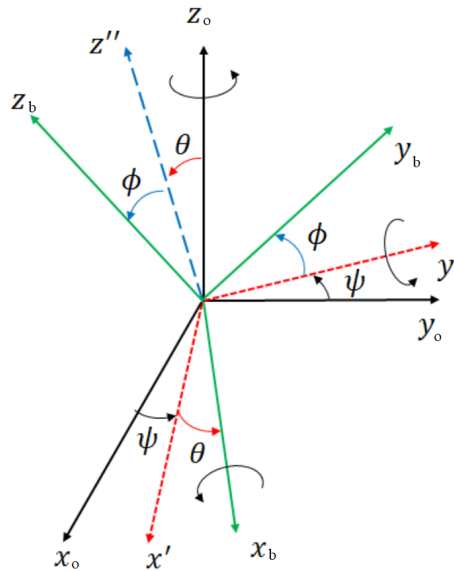


Figura 10 – Rotação em ângulos de Euler, sequência 3-2-1. Fonte: Adaptado de Silva, Kuga e Zanardi (2014b).

Dessa forma, a sequência 3-2-1 é descrita pela Equação (3.3) (STEVENS; LEWIS; JOHNSON, 2015).

$$\begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} \quad (3.3)$$

que pode ser escrita na forma de (3.4),

$$\begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} = \mathfrak{R} \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} \quad (3.4)$$

onde \mathfrak{R} é a matriz de atitude, dada por (3.5):

$$\mathfrak{R} = \begin{bmatrix} \cos(\theta)\cos(\psi) & \cos(\theta)\sin(\psi) & -\sin(\theta) \\ \sin(\phi)\sin(\theta)\cos(\psi) - \sin(\psi)\cos(\phi) & \sin(\phi)\sin(\theta)\sin(\psi) + \cos(\phi)\cos(\psi) & \sin(\phi)\cos(\theta) \\ \cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi) & \cos(\phi)\sin(\theta)\sin(\psi) - \sin(\phi)\cos(\psi) & \cos(\phi)\cos(\theta) \end{bmatrix} \quad (3.5)$$

3.3.2 Quatérnions

Os quatérnions (HAMILTON, 1844a) \mathbf{q} são números hipercomplexos, introduzidos por Sir William Rowan Hamilton, constituídos de uma parte vetorial com três números imaginários $\mathbf{q}_{1:3}$ e um escalar q_4 , como está representado nas Equações (3.6) e (3.7). Os quatérnions, diferentemente dos ângulos de Euler, por serem hipercomplexos, não possuem uma visualização física e compreensível por parte do ser humano, a Figura 11 é uma imagem que representa o quatérnion através de suas rotações. É importante notar que há divergências quanto à representação do quatérnion em diferentes literaturas, por isso, é necessário deixar claro que a representação utilizada neste trabalho é a definida abaixo, como colocado por Lefferts, Markley e Shuster (1982):

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_{1:3} \\ q_4 \end{bmatrix} \quad (3.6)$$

onde $\mathbf{q}_{1:3}$ é dado por:

$$\mathbf{q}_{1:3} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (3.7)$$

De acordo com Hamilton (1844b), a fórmula fundamental para a multiplicação de quatérnions é dada pela Equação (3.8):

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \quad (3.8)$$

onde \mathbf{i} , \mathbf{j} e \mathbf{k} são vetores unitários pertencentes à parte vetorial do quatérnion, de forma que o quatérnion também pode ser definido por (3.9):

$$\mathbf{q} = q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} + q_4 \quad (3.9)$$

Deve-se notar que o produto de quatérnions é uma operação associativa e distributiva, porém não é uma operação comutativa. Além da fórmula fundamental, as seguintes relações devem ser obedecidas:

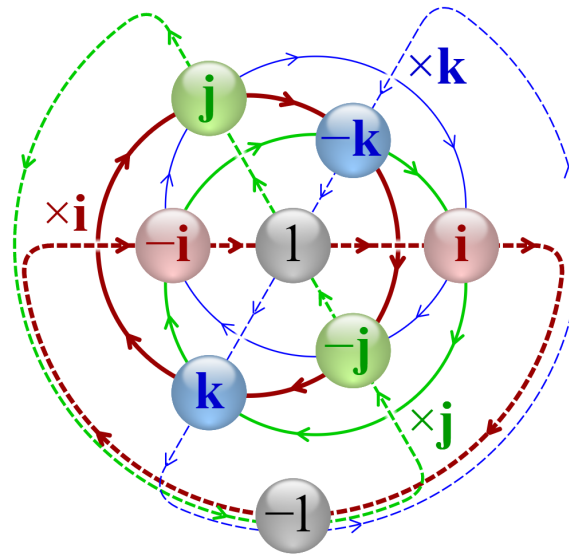


Figura 11 – Multiplicação de quatérnion. Fonte: Wikipedia (2022).

$$ij = -ji = k \quad (3.10)$$

$$jk = -kj = i \quad (3.11)$$

$$ki = -ik = j \quad (3.12)$$

Sejam dois quatérnions quaisquer \mathbf{q} e $\bar{\mathbf{q}}$. O produto de quatérnion é definido pelo símbolo \otimes e dado pela Equação (3.13) (MARKLEY; CRASSIDIS, 2014):

$$\bar{\mathbf{q}} \otimes \mathbf{q} = \begin{bmatrix} q_4 \bar{\mathbf{q}}_{1:3} + \bar{q}_4 \mathbf{q}_{1:3} - \bar{\mathbf{q}}_{1:3} \times \mathbf{q}_{1:3} \\ \bar{q}_4 q_4 - \bar{\mathbf{q}}_{1:3} \cdot \mathbf{q}_{1:3} \end{bmatrix} \quad (3.13)$$

onde \times é o símbolo do produto vetorial e \cdot o símbolo do produto escalar. Uma forma alternativa para se realizar o produto de quatérnions é através de uma multiplicação de matriz, como representado em (3.14):

$$\bar{\mathbf{q}} \otimes \mathbf{q} = [\bar{\mathbf{q}} \otimes] \mathbf{q} \quad (3.14)$$

em que $[\bar{\mathbf{q}} \otimes]$ multiplica \mathbf{q} à esquerda, assim como é feito na multiplicação de matriz. A notação $[\mathbf{q} \otimes]$ é definida por (3.15):

$$[\mathbf{q} \otimes] = \begin{bmatrix} q_4 & q_3 & -q_2 & q_1 \\ -q_3 & q_4 & q_1 & q_2 \\ q_2 & -q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix} \quad (3.15)$$

De maneira similar, o produto de quatérnion pode ser realizado entre um quatérnion e um vetor de três dimensões, desde que se adicione uma quarta dimensão de valor nulo no vetor. Seja um vetor tridimensional qualquer \mathbf{a} , o produto de quatérnion entre um quatérnion e um vetor é dado pelas Equações (3.16) e (3.17):

$$\mathbf{q} \otimes \mathbf{a} \equiv \mathbf{q} \otimes \begin{bmatrix} \mathbf{a} \\ 0 \end{bmatrix} = [\mathbf{q} \otimes] \begin{bmatrix} \mathbf{a} \\ 0 \end{bmatrix} \quad (3.16)$$

$$\mathbf{a} \otimes \mathbf{q} \equiv \begin{bmatrix} \mathbf{a} \\ 0 \end{bmatrix} \otimes \mathbf{q} = [\mathbf{a} \otimes] \mathbf{q} \quad (3.17)$$

em que, seguindo a notação definida em (3.15):

$$[\mathbf{a} \otimes] = \begin{bmatrix} 0 & a_3 & -a_2 & a_1 \\ -a_3 & 0 & a_1 & a_2 \\ a_2 & -a_1 & 0 & a_3 \\ -a_1 & -a_2 & -a_3 & 0 \end{bmatrix} \quad (3.18)$$

3.3.2.1 Representação de Atitude com Quatérnions

Diferentemente das sequências de rotações através dos ângulos de Euler, a representação de atitude com quatérnions é feita considerando uma única rotação de um ângulo ϑ ao redor de um eixo \vec{e} , desta forma, nos quatérnions, a sequência de rotação não importa, que é um benefício do quatérnion. O quatérnion de atitude é representado pela Equação (3.19):

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_{1:3} \\ q_4 \end{bmatrix} = \begin{bmatrix} \vec{e} \operatorname{sen} \left(\frac{\vartheta}{2} \right) \\ \cos \left(\frac{\vartheta}{2} \right) \end{bmatrix} \quad (3.19)$$

A representação de rotações sucessivas em quatérnions é feita da mesma forma das rotações em ângulos de Euler, onde se realiza o produto dos quatérnions constituintes de cada rotação. A matriz de atitude, como no caso da Equação (3.5), pode ser obtida através da Equação (3.20), como demonstrado por Markley e Crassidis (2014):

$$\mathfrak{R}(\mathbf{q}) = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \quad (3.20)$$

Deve-se atentar que o quatérnion utilizado para representar atitude deve possuir norma unitária, ou seja:

$$\|\mathbf{q}\| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} = 1 \quad (3.21)$$

Observando as Equações (3.5) e (3.20), percebe-se que, para se obter os ângulos de Euler, na sequência 3-2-1, partindo da matriz de atitude, realiza-se as seguintes operações:

$$\phi = \tan^{-1} \left(\frac{\mathfrak{R}(2,3)}{\mathfrak{R}(3,3)} \right) \quad (3.22)$$

$$\theta = \text{sen}^{-1} (-\mathfrak{R}(1,3)) \quad (3.23)$$

$$\psi = \tan^{-1} \left(\frac{\mathfrak{R}(1,2)}{\mathfrak{R}(1,1)} \right) \quad (3.24)$$

onde $\mathfrak{R}(i,j)$ representa o elemento na linha i e coluna j da matriz de atitude. Lembrando que, embora a rotação por quatérnions não possua sequência específica, em ângulos de Euler existe, então, para cada sequência de rotação em ângulos de Euler, haverá uma forma diferente para se obter esses ângulos a partir da matriz de atitude.

3.3.2.2 Cinemática de Atitude em Quatérnions

A cinemática de atitude descreve o movimento angular do corpo ao redor do seu centro de massa. Considerando o satélite como um corpo rígido, a cinemática de atitude em quatérnions é dada por (MARKLEY; CRASSIDIS, 2014):

$$\dot{\mathbf{q}}(t) = \lim_{\Delta t \rightarrow 0} \frac{\mathbf{q}(t + \Delta t) - \mathbf{q}(t)}{\Delta t} = \frac{1}{2} [\boldsymbol{\omega}(t) \otimes] \mathbf{q}(t) \quad (3.25)$$

Como os computadores de bordo que executam estes algoritmos são digitais, a equação de cinemática é discretizada, por isso, assumindo que os sensores possuem taxa de amostragem fixa, e que a variação de velocidade angular do satélite é pequena no intervalo de amostragem, considera-se $\boldsymbol{\omega}$ constante entre os intervalos, e a cinemática de atitude é obtida a partir de Silva et al. (2021) e Lefferts, Markley e Shuster (1982):

$$\mathbf{q}_{k+1} = \mathbf{M}(\boldsymbol{\omega}, \Delta t) \mathbf{q}_k \quad (3.26)$$

Sendo

$$\mathbf{M}(\boldsymbol{\omega}, \Delta t) = \cos\left(\frac{\|\boldsymbol{\omega}\| \Delta t}{2}\right) \mathbf{I} + \frac{\text{sen}\left(\frac{\|\boldsymbol{\omega}\| \Delta t}{2}\right)}{\|\boldsymbol{\omega}\|} [\boldsymbol{\omega} \otimes] \quad (3.27)$$

onde $\boldsymbol{\omega}$ é o vetor de velocidades angulares do corpo, k é um índice de tempo discreto, Δt é o intervalo de amostragem, \mathbf{q}_k é o valor do quatérnio no tempo atual e \mathbf{q}_{k+1} é seu valor no próximo passo.

3.4 Modelos Matemáticos dos Sensores de Atitude

Esta seção se dedica a apresentar os modelos matemáticos referentes aos sensores de atitude do satélite CBERS-2, mencionados na Seção 3.1, que são utilizados para a determinação da orientação do satélite.

3.4.1 Modelo do Sensor Girômetro

Os sensores girômetros, também chamados de giroscópios, captam variações de movimentos angulares ao redor de um eixo, e assim são utilizados para medir mudanças na orientação de um objeto. Para o objeto de estudo deste trabalho, foram utilizados os RIGs, e estes medem o deslocamento angular do objeto ao redor de um eixo de referência. Para representar a atitude do satélite em todos os eixos do sistema de referência do corpo, são utilizados três girômetros, cada um medindo os movimentos angulares ao redor de um eixo: rolamento, arfagem e guinada.

Além disso, os sensores em questão apresentam vieses e ruído branco gaussiano devido a imperfeições, e estes valores devem ser considerados na aquisição dos dados. Sendo assim, os RIGs fornecem um ângulo de saída $\Delta\theta_i$ que é idealmente proporcional à integral da velocidade angular ω_i ao longo de um intervalo de amostragem Δt (WERTZ, 1978). Assim podemos construir a relação (GARCIA, 2011):

$$\Delta\theta_i = \int_0^{\Delta t} (\omega_i + \epsilon_i) dt, \quad i = x, y, z \quad (3.28)$$

e velocidade angular do satélite pode ser obtida a partir de:

$$\omega_i = \frac{d\theta_i}{dt} - \epsilon_i - v_i \quad (3.29)$$

onde $\Delta\theta_i$ são os deslocamentos angulares em um intervalo Δt , ω_i são as velocidades angulares medidas, ϵ_i são os vieses dos sensores e v_i é o ruído branco gaussiano.

3.4.2 Modelo do Sensor da Terra Infravermelho

Os sensores IRES detectam radiação infravermelha vinda do horizonte da Terra, e assim realizam medidas diretas da atitude do satélite com relação ao centro da Terra em dois eixos: rolamento e arfagem. O modelo matemático dos sensores IRES é (FUMING; KUGA, 1999):

$$\phi_H = \phi + v_{\phi_H} \quad (3.30)$$

$$\theta_H = \theta + v_{\theta_H} \quad (3.31)$$

onde v_{ϕ_H} e v_{θ_H} são ruídos brancos gaussianos, ϕ_H e θ_H são os ângulos medidos pelo sensor e ϕ e θ são os ângulos reais de rolamento e arfagem, respectivamente.

3.4.3 Modelo do Sensor Solar Digital

Os sensores DSS utilizados detectam a posição do Sol com relação ao satélite e, a partir desta posição, fornecem o direcionamento do satélite com relação à Terra em medidas indiretas de arfagem e guinada. Sendo assim, a utilização destes sensores compensa o fato de que os IRES não possuem medidas de guinada. Para ser detectado, o Sol deve estar no campo de visada do sensor, com isso, as medidas fornecidas possuem as seguintes condições (FUMING; KUGA, 1999):

Caso $|S_x \cos(60^\circ) + S_z \cos(150^\circ)| \geq \cos(60^\circ)$:

$$\alpha_\psi = \tan^{-1} \left(\frac{-S_y}{S_x \cos(60^\circ) + S_z \cos(150^\circ)} \right) + v_{\alpha_\psi} \quad (3.32)$$

Caso $\left| 24^\circ - \tan^{-1} \left(\frac{S_x}{S_z} \right) \right| < 60^\circ$:

$$\alpha_\theta = 24^\circ - \tan^{-1} \left(\frac{S_x}{S_z} \right) + v_{\alpha_\theta} \quad (3.33)$$

onde S_x , S_y e S_z são as componentes do vetor unitário associado à posição do Sol no sistema do satélite, v_{α_θ} e v_{α_ψ} são ruídos brancos gaussianos e α_θ e α_ψ representam os ângulos acoplados de arfagem e guinada, respectivamente. S_x , S_y e S_z são dados por:

$$S_x = S_{0_x} + \psi S_{0_y} - \theta S_{0_z} \quad (3.34)$$

$$S_y = S_{0_y} - \psi S_{0_x} + \phi S_{0_z} \quad (3.35)$$

$$S_z = S_{0z} - \phi S_{0y} + \theta S_{0x} \quad (3.36)$$

em que S_{0x} , S_{0y} e S_{0z} são as componentes do vetor unitário associado à posição do Sol no Sistema de Referência Orbital.

3.5 Métodos de Estimação de Estados

Nesta seção, serão apresentados os métodos de estimação de estados não lineares utilizados para estimar a atitude e os vieses dos sensores girômetros do satélite. Estes métodos utilizam o processo de expansão em série de Taylor de primeira ou segunda ordem para simplificar os sistemas de equações não lineares.

3.5.1 Filtro de Kalman Estendido

O Filtro de Kalman (KALMAN, 1960), nomeado por Rudolf Emil Kálmán, é um observador de estados capaz de estimar os estados de um sistema linear, apesar das incertezas advindas dos sensores de atitude. O Filtro de Kalman Estendido (*Extended Kalman Filter* - EKF) é uma adaptação do Filtro de Kalman, capaz de trabalhar com sistemas não lineares, através da linearização do sistema a cada passo de seu processo. O processo de trabalho do EKF é dividido em duas etapas que se repetem a cada iteração: predição, ou propagação, e atualização. Primeiramente, na predição, os estados são estimados a partir do modelo matemático do sistema, e após isso, os estados estimados são atualizados a partir das medidas dos sensores.

Para o EKF, são feitas as seguintes considerações: os ruídos de processo \mathbf{w}_k e de medida \mathbf{v}_k são brancos, com média igual a zero e não correlacionados, e as matrizes de covariância \mathbf{Q}_k e \mathbf{R}_k são conhecidas (SIMON, 2006). Definindo valor esperado (esperança matemática) de uma variável aleatória x como $E[x]$:

$$\begin{aligned} E[\mathbf{w}_k \mathbf{w}_k^T] &= \mathbf{Q}_k \\ E[\mathbf{v}_k \mathbf{v}_k^T] &= \mathbf{R}_k \\ E[\mathbf{w}_k \mathbf{v}_k^T] &= 0 \end{aligned} \quad (3.37)$$

Este filtro tem como objetivo minimizar o erro da estimativa de forma que:

$$E[\mathbf{x}_k - \hat{\mathbf{x}}_k] = 0 \quad (3.38)$$

De acordo com a teoria disposta em Silva, Kuga e Zanardi (2014a) e Simon (2006), considere um sistema não linear discreto no tempo:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k) + \mathbf{w}_k \quad (3.39)$$

$$\mathbf{y}_k = h(\mathbf{x}_k) + \mathbf{v}_k \quad (3.40)$$

onde \mathbf{x} é o vetor de estados de dimensão n , $f(\mathbf{x}_k)$ é uma função vetorial dos estados diferenciável com respeito a \mathbf{x}_k , \mathbf{y}_k é o vetor de medidas de dimensão m , $h(\mathbf{x}_k)$ é uma função vetorial dos estados diferenciável com respeito a \mathbf{x}_k , \mathbf{w}_k é o ruído branco de processo com covariância \mathbf{Q}_k e \mathbf{v}_k é o ruído branco de medida, com covariância \mathbf{R}_k .

A solução do filtro é dada a partir da teoria obtida de Lefferts, Markley e Shuster (1982), onde realizou-se o desenvolvimento do Filtro de Kalman Estendido com foco na aplicação de veículos espaciais equipados com sensores girômetros em três eixos juntamente com outros sensores de atitude, como sensores solares e sensores de estrelas. Devido à restrição de norma unitária do quatérnio de atitude, a matriz de covariância de dimensão 7×7 é singular, sendo assim, neste trabalho, para lidar com este problema, é utilizada a representação reduzida da matriz de covariância, que reduz sua dimensão para 6×6 .

Na etapa da predição, tem-se as seguintes relações (LEFFERTS; MARKLEY; SHUSTER, 1982; SILVA, 2016):

$$\hat{\mathbf{x}}_k^- = f(\hat{\mathbf{x}}_{k-1}^+) \quad (3.41)$$

$$\tilde{\mathbf{P}}_k^- = \tilde{\mathbf{F}}_{k-1} \tilde{\mathbf{P}}_{k-1}^+ \tilde{\mathbf{F}}_{k-1}^T + \tilde{\mathbf{Q}}_{k-1} \quad (3.42)$$

sendo:

$$\tilde{\mathbf{Q}}_k = \tilde{\mathbf{F}}_k \tilde{\mathbf{G}}_k \mathbf{Q}_k \tilde{\mathbf{G}}_k^T \tilde{\mathbf{F}}_k^T \Delta t \quad (3.43)$$

$$\tilde{\mathbf{G}}_k = \begin{bmatrix} -\frac{1}{2} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \quad (3.44)$$

$$\tilde{\mathbf{F}}_k = \begin{bmatrix} \mathbf{A}_k & \mathbf{\Gamma}_k \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \quad (3.45)$$

$$\mathbf{\Gamma}_k = -\frac{1}{2} \mathbf{A}_k \Delta t \quad (3.46)$$

$$\mathbf{A}_k = \mathbf{\Xi}_{(-)}^T \mathbf{M}(\boldsymbol{\omega}, \Delta t) \mathbf{\Xi}_{(+)} \quad (3.47)$$

$$\Xi = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \quad (3.48)$$

onde $\Xi_{(+)}$ se refere ao quatérnio de atitude atualizado no tempo anterior e $\Xi_{(-)}$ se refere ao quatérnio de atitude propagado no tempo atual.

Para a etapa de atualização, tem-se:

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \bar{\mathbf{K}}_k (\mathbf{y}_k - h(\hat{\mathbf{x}}_k^-)) \quad (3.49)$$

$$\tilde{\mathbf{P}}_k^+ = (\mathbf{I} - \tilde{\mathbf{K}}_k \tilde{\mathbf{H}}_k) \tilde{\mathbf{P}}_k^- \quad (3.50)$$

sendo:

$$\bar{\mathbf{K}}_k = \mathbf{S}_\Xi \tilde{\mathbf{K}}_k \quad (3.51)$$

$$\tilde{\mathbf{K}}_k = \tilde{\mathbf{P}}_k^- \tilde{\mathbf{H}}_k^T (\tilde{\mathbf{H}}_k \tilde{\mathbf{P}}_k^- \tilde{\mathbf{H}}_k^T + \mathbf{R}_k)^{-1} \quad (3.52)$$

$$\tilde{\mathbf{H}}_k = \left. \frac{\partial h}{\partial \mathbf{x}_k} \right|_{\hat{\mathbf{x}}_k^-} \mathbf{S}_\Xi \quad (3.53)$$

$$\mathbf{S}_\Xi = \begin{bmatrix} \Xi_{(-)} & \mathbf{0}_{4 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \quad (3.54)$$

onde $\hat{\mathbf{x}}_k^-$ é o estado predito e $\hat{\mathbf{x}}_k^+$ é o estado atualizado, $\bar{\mathbf{K}}_k$ é o ganho do filtro para a atualização, $\tilde{\mathbf{P}}_k^-$ é a matriz de covariância na propagação e $\tilde{\mathbf{P}}_k^+$ é a matriz de covariância na atualização.

Visando evitar a perda da norma unitária do quatérnio de atitude, que por sua vez causaria erros na estimação, é realizada a normalização forçada do mesmo a cada passo, onde se divide o quatérnio de atitude estimado por sua norma, que aprimora a convergência e a acurácia do filtro, como visto em Bar-Itzhack e Oshman (1985). Esta estratégia também foi utilizada nos próximos métodos de estimação descritos.

3.5.2 Filtro H_∞ Estendido

O Filtro H_∞ Estendido (*Extended H_∞ Filter* - EH_∞F) foi originalmente desenvolvido por Banavar (1992) e posteriormente discutido por Shaked e Berman (1995)

e Hu e Yang (2011) em suas versões não-lineares. Possui similaridades com o EKF, por ser um filtro recursivo e linearizar o sistema a cada passo. De acordo com Simon (2006), a filtragem H_∞ é mais robusta que a filtragem de Kalman, minimizando o erro de estimação de pior caso, além disso, diferentemente do Filtro de Kalman, o Filtro H_∞ não faz suposições sobre as estatísticas dos ruídos (embora essa informação pode ser utilizada se disponível), que podem ter média diferente de zero.

De acordo com o sistema não linear discreto no tempo descrito nas Equações (3.39) e (3.40), visa-se estimar uma combinação linear do estado, dada por:

$$\mathbf{z}_k = \mathbf{L}_k \mathbf{x}_k \quad (3.55)$$

e encontrar este valor estimado ($\hat{\mathbf{z}}_k$) que minimiza o erro $\|\mathbf{z}_k - \hat{\mathbf{z}}_k\|$ para quaisquer \mathbf{w}_k , \mathbf{v}_k e \mathbf{x}_0 , em que \mathbf{x}_0 é o estado inicial e \mathbf{L}_k é uma matriz definida pelo usuário com rank completo. Para estimar diretamente \mathbf{x}_k como no Filtro de Kalman, basta definir $\mathbf{L}_k = \mathbf{I}$. Na filtragem H_∞ , define-se uma função custo J_1 :

$$J_1 = \frac{\sum_{k=0}^{N-1} \|\mathbf{z}_k - \hat{\mathbf{z}}_k\|_{\mathbf{S}_k}^2}{\|\mathbf{x}_0 - \hat{\mathbf{x}}_0\|_{\mathbf{P}_0^{-1}}^2 + \sum_{k=0}^{N-1} (\|\mathbf{w}_k\|_{\mathbf{Q}_k^{-1}}^2 + \|\mathbf{v}_k\|_{\mathbf{R}_k^{-1}}^2)} \quad (3.56)$$

onde a notação $\|\mathbf{a}\|_{\mathbf{B}}^2 = \mathbf{a}^T \mathbf{B} \mathbf{a}$, para quaisquer \mathbf{a} e \mathbf{B} . As matrizes \mathbf{P}_0 , \mathbf{Q}_k , \mathbf{R}_k e \mathbf{S}_k são matrizes de ponderação simétricas, positivas definidas, escolhidas de acordo com a particularidade do problema.

A abordagem utilizada na filtragem H_∞ considera que a natureza tenderá a maximizar J_1 através de \mathbf{w}_k , \mathbf{v}_k e \mathbf{x}_0 , maximizando então o erro de estimação $\|\mathbf{z}_k - \hat{\mathbf{z}}_k\|$, enquanto que o objetivo do projetista é minimizar o mesmo. A minimização de J_1 é feita através de um limite de desempenho especificado pelo usuário γ , onde se procura uma estimativa $\hat{\mathbf{z}}_k$ que resulta em:

$$J_1 < \frac{1}{\gamma} \quad (3.57)$$

Rearranjando a Equação (3.57), tem-se:

$$J = \frac{-1}{\gamma} \|\mathbf{x}_0 - \hat{\mathbf{x}}_0\|_{\mathbf{P}_0^{-1}}^2 + \sum_{k=0}^{N-1} \left[\|\mathbf{z}_k - \hat{\mathbf{z}}_k\|_{\mathbf{S}_k}^2 - \frac{1}{\gamma} (\|\mathbf{w}_k\|_{\mathbf{Q}_k^{-1}}^2 + \|\mathbf{v}_k\|_{\mathbf{R}_k^{-1}}^2) \right] < 1 \quad (3.58)$$

De acordo com a Equação (3.55), é perceptível que $\|\mathbf{z}_k - \hat{\mathbf{z}}_k\|_{\mathbf{S}_k}^2 = \|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_{\bar{\mathbf{S}}_k}^2$, para $\bar{\mathbf{S}}_k = \mathbf{L}_k^T \mathbf{S}_k \mathbf{L}_k$. Além disso, de acordo com a Equação (3.40), tem-se $\mathbf{v}_k = \mathbf{y}_k - h(\mathbf{x}_k)$. Assim, J pode ser obtido na forma de:

$$J = \frac{-1}{\gamma} \|\mathbf{x}_0 - \hat{\mathbf{x}}_0\|_{\mathbf{P}_0^{-1}}^2 + \sum_{k=0}^{N-1} \left[\|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_{\tilde{\mathbf{S}}_k}^2 - \frac{1}{\gamma} \left(\|\mathbf{w}_k\|_{\mathbf{Q}_k^{-1}}^2 + \|\mathbf{y}_k - h(\mathbf{x}_k)\|_{\mathbf{R}_k^{-1}}^2 \right) \right] \quad (3.59)$$

A solução do filtro é dada a partir da resolução do problema de *minimax*, definido por:

$$J^* = \min_{\hat{\mathbf{x}}_k} \max_{\mathbf{w}_k, \mathbf{y}_k, \mathbf{x}_0} J \quad (3.60)$$

e seu desenvolvimento e provas matemáticas geram uma longa e complexa discussão, que podem ser encontrados em Silva, Kuga e Zanardi (2014a), Hu e Yang (2011) e Simon (2006).

Utilizando a representação reduzida da matriz de covariância, discutida em Leferts, Markley e Shuster (1982), a solução do Filtro H_∞ Estendido é dada por (SILVA; KUGA; ZANARDI, 2014a):

$$\hat{\mathbf{x}}_{k+1} = f(\hat{\mathbf{x}}_k) + \mathbf{F}_k \mathbf{K}_k [\mathbf{y}_k - h(\hat{\mathbf{x}}_k)] \quad (3.61)$$

$$\mathbf{P}_{k+1} = \tilde{\mathbf{F}}_k \mathbf{P}_k \mathbf{G}_k^{-1} \tilde{\mathbf{F}}_k^T + \mathbf{Q}_k \quad (3.62)$$

onde

$$\mathbf{F}_k = \left. \frac{\partial f}{\partial \mathbf{x}_k} \right|_{\hat{\mathbf{x}}_k} \quad (3.63)$$

$$\mathbf{K}_k = \mathbf{S}_\Xi \mathbf{P}_k \mathbf{G}_k^{-1} \tilde{\mathbf{H}}_k^T \mathbf{R}_k^{-1} \quad (3.64)$$

$$\mathbf{G}_k = \mathbf{I} - \gamma \tilde{\mathbf{S}}_k \mathbf{P}_k + \tilde{\mathbf{H}}_k^T \mathbf{R}_k^{-1} \tilde{\mathbf{H}}_k \mathbf{P}_k \quad (3.65)$$

Para que o estimador seja uma solução para o problema, a seguinte condição deve ser satisfeita a cada passo:

$$\mathbf{P}_k^{-1} - \gamma \tilde{\mathbf{S}}_k + \tilde{\mathbf{H}}_k^T \mathbf{R}_k^{-1} \tilde{\mathbf{H}}_k > 0 \quad (3.66)$$

3.5.3 Filtro H_∞ Estendido de Segunda Ordem

O Filtro H_∞ Estendido de Segunda Ordem (*Second-Order Extended H_∞ Filter* - SOEH $_\infty$ F), desenvolvido em Hu e Yang (2011), é similar ao Filtro H_∞ Estendido, de forma que possui a mesma abordagem e o mesmo objetivo.

A construção do $\text{SOEH}_\infty\text{F}$ é quase idêntica à do EH_∞F , com a diferença de que neste caso, a aproximação do sistema não linear é feita realizando a expansão em série de Taylor de segundo grau, ou seja, para o sistema descrito nas Equações (3.39) e (3.40), (HU; YANG, 2011):

$$f(\mathbf{x}_k) = f(\hat{\mathbf{x}}_k) + \left. \frac{\partial f}{\partial \mathbf{x}_k} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k) + \frac{1}{2} \sum_{i=1}^n \varphi_i^f (\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \left. \frac{\partial^2 f_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k) \quad (3.67)$$

$$h(\mathbf{x}_k) = h(\hat{\mathbf{x}}_k) + \left. \frac{\partial h}{\partial \mathbf{x}_k} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k) + \frac{1}{2} \sum_{i=1}^m \varphi_i^h (\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \left. \frac{\partial^2 h_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k) \quad (3.68)$$

onde f_i e h_i são os i -ésimos elementos de $f(\mathbf{x}_k)$ e $h(\mathbf{x}_k)$, φ_i^f e φ_i^h são vetores com dimensão $n \times 1$ e $m \times 1$, respectivamente, com valores iguais a 1 em seus i -ésimos elementos e 0 no resto deles.

Devido à complexidade dos termos quadráticos da expansão em série de Taylor, considera-se que eles podem ser aproximados pela matriz de covariância da estimativa do erro, que simplifica o problema e facilita sua implementação em tempo real, assim:

$$(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \left. \frac{\partial^2 f_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k) = \text{tr} \left[\left. \frac{\partial^2 f_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \right] \approx \text{tr} \left[\left. \frac{\partial^2 f_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} \bar{\mathbf{P}} \right] \quad (3.69)$$

$$(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \left. \frac{\partial^2 h_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k) = \text{tr} \left[\left. \frac{\partial^2 h_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} (\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \right] \approx \text{tr} \left[\left. \frac{\partial^2 h_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} \bar{\mathbf{P}} \right] \quad (3.70)$$

onde $\text{tr}[\cdot]$ se define como a operação traço.

A partir das considerações dispostas, seguindo a mesma ideia do Filtro H_∞ Estendido, e utilizando a representação reduzida da matriz de covariância, a solução do Filtro H_∞ Estendido de Segunda Ordem é dada por Hu e Yang (2011) e Silva (2016):

$$\hat{\mathbf{x}}_{k+1} = f(\hat{\mathbf{x}}_k) + \frac{1}{2} \sum_{i=1}^n \varphi_i^f \text{tr} \left[\left. \frac{\partial^2 f_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} \mathbf{S}_\Xi \bar{\mathbf{P}}_k \mathbf{S}_\Xi^T \right] + \mathbf{F}_k \mathbf{K}_k \tilde{\mathbf{y}}_k \quad (3.71)$$

$$\mathbf{P}_{k+1} = \tilde{\mathbf{F}}_k \mathbf{P}_k \mathbf{G}_k^{-1} \tilde{\mathbf{F}}_k^T + \mathbf{Q}_k \quad (3.72)$$

onde

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - h(\hat{\mathbf{x}}_k) - \frac{1}{2} \sum_{i=1}^m \varphi_i^h \text{tr} \left[\left. \frac{\partial^2 h_i}{\partial \mathbf{x}_k^2} \right|_{\hat{\mathbf{x}}_k} \bar{\mathbf{P}}_k \right] \quad (3.73)$$

$$\bar{\mathbf{P}}_{k+1} = \eta \bar{\mathbf{P}}_k + (1 - \eta) \mathbf{P}_k \boldsymbol{\lambda}_k \boldsymbol{\lambda}_k^T \mathbf{P}_k^T \quad (3.74)$$

$$\boldsymbol{\lambda}_{k+1} = \left(\tilde{\mathbf{F}}_k \tilde{\mathbf{F}}_k^T + \xi \mathbf{I} \right)^{-1} \tilde{\mathbf{F}}_k \left(\mathbf{G}_k \boldsymbol{\lambda}_k - \tilde{\mathbf{H}}_k^T \mathbf{R}_k^{-1} \tilde{\mathbf{y}}_k \right) \quad (3.75)$$

em que $0 < \eta \leq 1$, ξ é um escalar positivo inserido para prevenir que $\tilde{\mathbf{F}}_k \tilde{\mathbf{F}}_k^T$ se torne singular e $\boldsymbol{\lambda}_k$ é o multiplicador de Lagrange usado na solução do problema.

De forma análoga ao EH_∞F , a seguinte condição deve ser satisfeita a cada passo para que o estimador seja uma solução para o problema:

$$\bar{\mathbf{S}}_k + \gamma \bar{\mathbf{S}}_k \mathbf{P}_k \mathbf{G}_k^{-1} \bar{\mathbf{S}}_k > 0 \quad (3.76)$$

4 Aplicação

Este capítulo é dedicado a apresentar os dados que foram inseridos no *toolbox* PROPAT para simular a dinâmica orbital do satélite e sua atitude, assim como as informações advindas dos sensores. Além disso, também é apresentado o sistema aplicado para a estimação e a forma com que os sensores e os filtros estão associados.

4.1 Órbita Simulada

De acordo com as informações disponibilizadas na Seção 3.1, as características referentes à órbita heliosíncrona do CBERS-2 estão dispostas na Tabela 1.

Característica orbital	Valor
Altitude	778 km
Excentricidade	$1,1 \times 10^{-3}$
Inclinação	98,504°
Argumento do perigeu	90°
Hora local no nodo descendente	10:30 a.m.
Longitude do nodo ascendente	variável
Repetição do ciclo	26 dias

Tabela 1 – Características da órbita do CBERS-2.

Partindo das informações dadas, procurou-se encontrar os elementos orbitais que, quando aplicados no PROPAT, simulassem valores semelhantes às informações dos sensores reais do satélite. Sendo assim, os elementos orbitais dispostos na Tabela 2 são os elementos encontrados que trazem as maiores semelhanças com os dados reais, em termos de atitude, velocidade angular e posição.

Elemento orbital	Valor
Semieixo maior	7149 km
Excentricidade	$1,1 \times 10^{-3}$
Inclinação	98,504°
Argumento do perigeu	90°
Longitude do nodo ascendente	183,93°
Anomalia média	66,54°

Tabela 2 – Elementos orbitais aplicados.

4.2 Sensores Simulados

Assim como na simulação da dinâmica orbital do satélite, procurou-se encontrar os parâmetros dos sensores que trouxessem as maiores semelhanças aos sensores reais, em termos de intervalo de amostragem e ruídos brancos gaussianos. Desta forma, os parâmetros utilizados para simular os sensores estão descritos na Tabela 3.

Parâmetro	Valor
Intervalo de amostragem	10 s
Desvio padrão do ruído branco gaussiano dos sensores DSS	0,2°
Desvio padrão do ruído branco gaussiano dos sensores IRES	0,02°
Desvio padrão do ruído branco gaussiano dos girômetros	$2,5 \times 10^{-4}$ °/s
Viés do girômetro no eixo x	-2°/h
Viés do girômetro no eixo y	-3°/h
Viés do girômetro no eixo z	1°/h

Tabela 3 – Parâmetros dos sensores.

4.3 Sistema Aplicado

Conforme apresentado por Lefferts, Markley e Shuster (1982) e por Silva et al. (2021), o vetor de estados a ser utilizado na estimação é composto pelo quatérnio de atitude e pelos vieses dos girômetros, como está apresentado na Equação (4.1):

$$\mathbf{x} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & \epsilon_x & \epsilon_y & \epsilon_z \end{bmatrix}^T = \begin{bmatrix} \mathbf{q}^T & \boldsymbol{\epsilon}^T \end{bmatrix}^T \quad (4.1)$$

e o sistema não linear aplicado na estimação é dado pelas Equações (4.2) e (4.3):

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\boldsymbol{\epsilon}} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} [\boldsymbol{\omega} \otimes] & 0_{4 \times 3} \\ 0_{3 \times 4} & 0_{3 \times 3} \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\epsilon} \end{bmatrix} + \mathbf{w} \quad (4.2)$$

$$\mathbf{y} = \begin{bmatrix} \tan^{-1} \left(\frac{-(S_{0y} - \psi S_{0x} + \phi S_{0z})}{(S_{0x} + \psi S_{0y} - \theta S_{0z}) \cos(60^\circ) + (S_{0z} - \phi S_{0y} + \theta S_{0x}) \cos(150^\circ)} \right) \\ 24^\circ - \tan^{-1} \left(\frac{S_{0x} + \psi S_{0y} - \theta S_{0z}}{S_{0z} - \phi S_{0y} + \theta S_{0x}} \right) \\ \phi \\ \theta \end{bmatrix} + \mathbf{v} \quad (4.3)$$

4.4 Integração dos Filtros e Sensores

Os algoritmos foram integrados de forma que todos os filtros usassem os mesmos valores provenientes dos sensores, para que, desta forma, a comparação dos

filtros não fosse prejudicada pela aleatoriedade dos erros. A Figura 12 representa a forma com que o PROPAT juntamente com os filtros foram integrados, para o caso dos dados simulados. O *toolbox* gera e propaga os dados de dinâmica orbital e atitude do satélite e, após isso, simula os dados dos sensores a partir dos modelos matemáticos dos mesmos, disponíveis na Seção 3.4, onde também são adicionados os ruídos. A Figura 13 representa a forma com que os dados dos sensores e os filtros foram agregados, para o caso dos sensores reais do satélite, onde se observa que a informação é limitada por ter somente informações advindas dos sensores do satélite, ou seja, não se sabe quais são os valores reais.

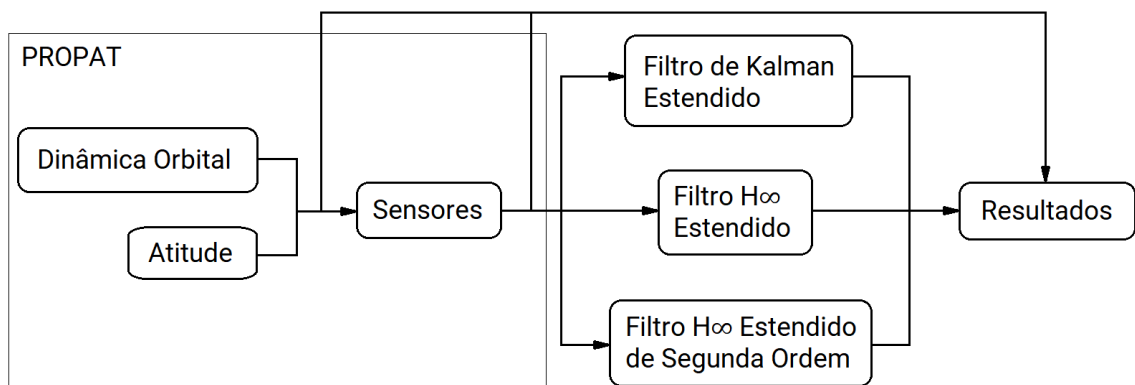


Figura 12 – Integração dos filtros e sensores para os dados simulados.

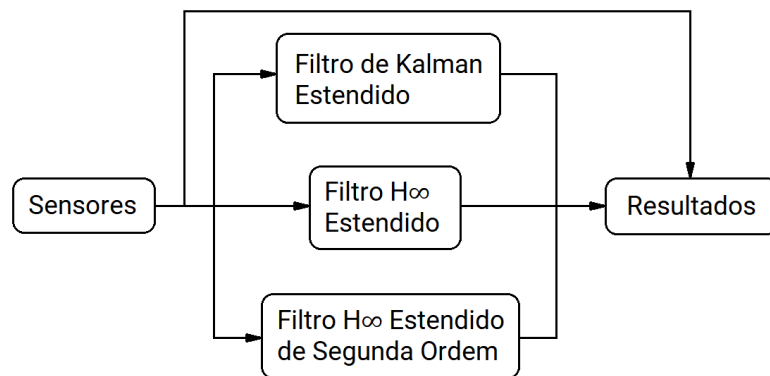


Figura 13 – Integração dos filtros e sensores para os dados reais.

Esta página foi intencionalmente deixada em branco.

5 Resultados

Neste capítulo, encontram-se os parâmetros de inicialização dos filtros juntamente com os resultados respectivos a cada aplicação. Primeiramente foi realizada a estimação de atitude simulada inserindo nas matrizes \mathbf{R}_k e \mathbf{Q}_k os valores exatos das variâncias referentes aos ruídos brancos gaussianos dos sensores simulados, os resultados desta simulação estão disponíveis na Seção 5.1. Após observar os resultados da primeira estimação, na Seção 5.2 foi feita uma segunda estimação da atitude simulada, desta vez modificando os dados de inicialização, com o objetivo de se obter melhores resultados. A partir dos parâmetros de inicialização utilizados na Seção 5.2, foram realizadas mais duas estimações, com o objetivo de se verificar o comportamento dos filtros em casos onde o erro do sistema é maior do que o previsto.

Na Seção 5.3, os desvios padrões dos ruídos brancos gaussianos de todos os sensores (girômetros, DSS e IRES), disponíveis na Tabela 3, foram multiplicados por 10, e para a Seção 5.4, além de utilizar os desvios multiplicados por 10, foram multiplicados por 10 os vieses dos girômetros. Por fim, foi realizada a estimação dos dados reais.

Note que os parâmetros de inicialização a serem utilizados são iguais para todos os filtros. Embora um filtro possa ter mais parâmetros do que outro, aqueles que se repetem entre os filtros, como as matrizes \mathbf{P}_0 , \mathbf{Q}_k e \mathbf{R}_k , são iguais para todos. Isto foi feito pois os desempenhos dos métodos de estimação estão diretamente ligados a seus dados de inicialização, sendo assim, ao igualá-los, podemos fazer uma comparação mais fidedigna.

5.1 Estimação Utilizando Dados Simulados 1

Os parâmetros de inicialização dos filtros para a primeira estimação dos dados simulados estão descritos na Tabela 4. As matrizes \mathbf{R}_k correspondem às variâncias dos sensores DSS e IRES, as matrizes \mathbf{Q}_k correspondem às variâncias dos girômetros e as matrizes \mathbf{P}_0 correspondem à variância da estimativa do erro inicial, como colocado por Simon (2006). Todas essas matrizes mencionadas são diagonais e positivas definidas.

Os resultados decorrentes da aplicação dos elementos orbitais mencionados na Tabela 2 e dos parâmetros dos sensores descritos na Tabela 3 no *toolbox* PROPAT estão disponibilizados nas Figuras 14, 15, 16 e 17, onde se observam os valores referentes à posição e à velocidade do satélite no Sistema Equatorial Terrestre, além das medidas advindas dos sensores DSS, IRES e girômetros.

Parâmetro	Valor
Tempo de simulação	530 s
Estado inicial em ϕ, θ e ψ	0°
Estado inicial (ϵ)	$[0 \ 215,4 \ 0]^T \text{ } ^\circ/\text{h}$
Variância dos DSS	$(0,2^\circ)^2$
Variância dos IRES	$(0,02^\circ)^2$
Variância dos girômetros	$(2,5 \times 10^{-4} \text{ } ^\circ/\text{s})^2$
Variância dos vieses de girômetro	$(0,05^\circ/\text{h})^2$
Variância da estimativa do erro inicial em ϕ, θ e ψ	$(0,5^\circ)^2$
Variância da estimativa do erro inicial dos vieses	$(1^\circ/\text{h})^2$
γ	5000
η	0,5
ξ	1
λ_0	$[1 \ 1 \ 1 \ 1 \ 1 \ 1]^T$
S_k	I
L_k	I

Tabela 4 – Parâmetros de inicialização para a estimação utilizando dados simulados 1.

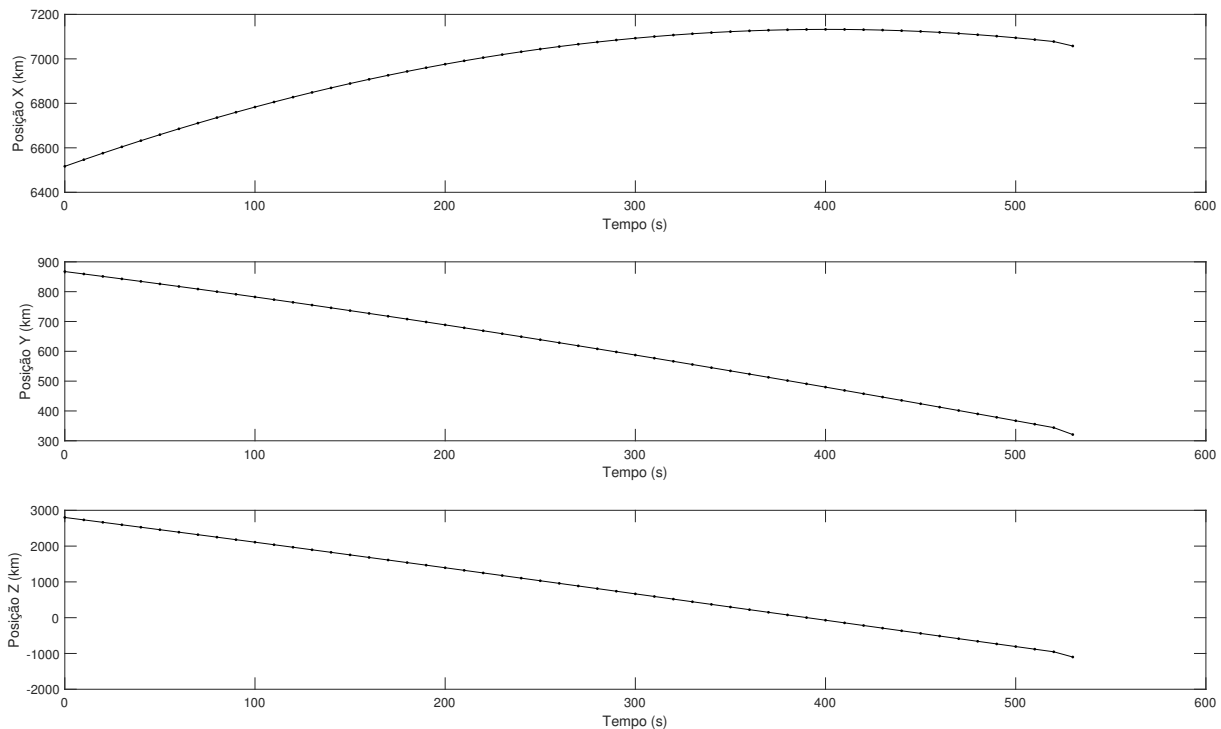


Figura 14 – Posição no Sistema Equatorial Terrestre simulada.

A partir dos dados simulados pelo PROPAT, realizou-se a estimação dos estados do sistema aplicado, os quais estão inclusos a atitude do satélite e os vieses dos girômetros. Os resultados da estimação de estados estão disponibilizados a seguir, onde a estimação de atitude pode ser encontrada na Figura 18, a estimação de viés pode ser encontrada na Figura 19, os resíduos dos filtros referentes aos sensores

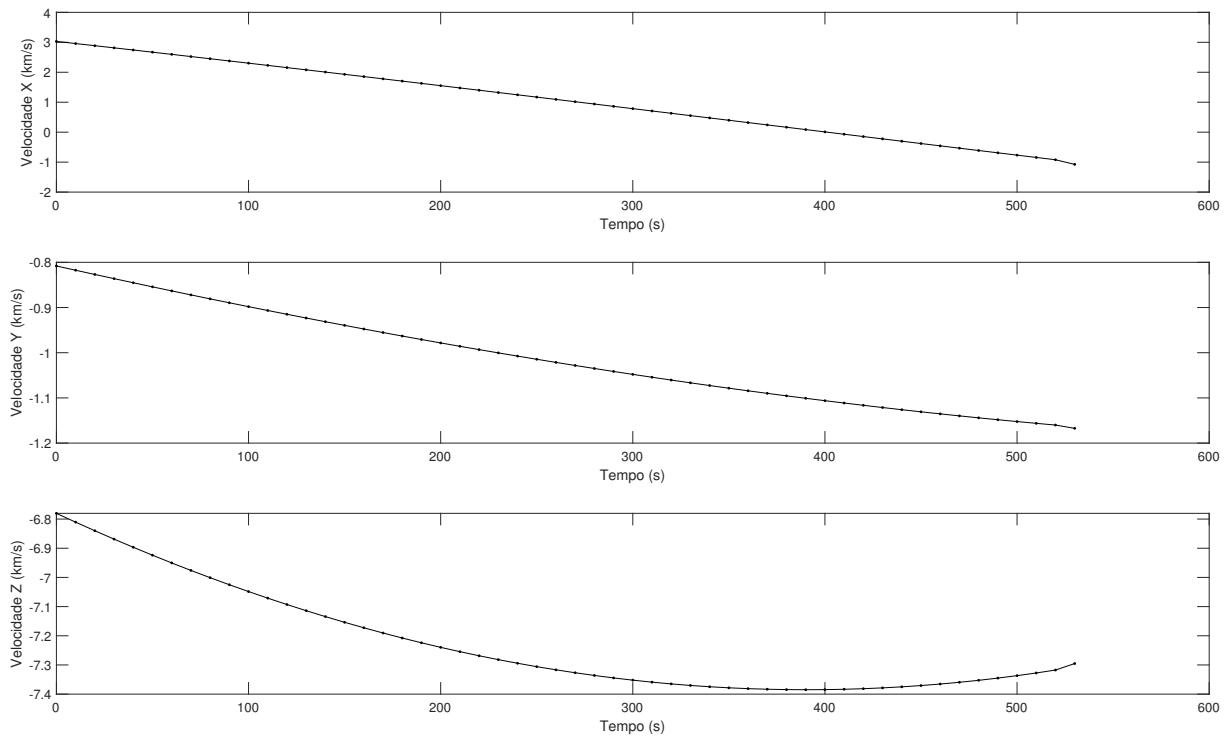


Figura 15 – Velocidade no Sistema Equatorial Terrestre simulada.

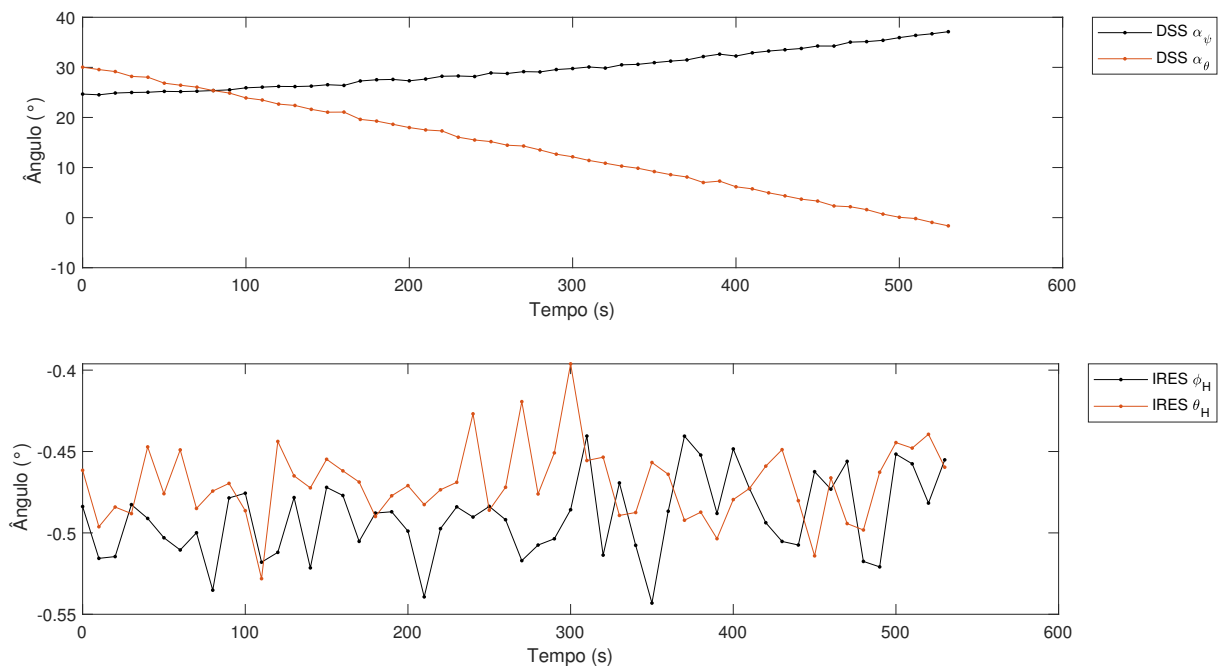


Figura 16 – Medidas simuladas dos sensores DSS e IRES.

DSS estão na Figura 20 e os resíduos dos filtros referentes aos sensores IRES estão na Figura 21.

Ao observar o resultado da estimação de atitude na Figura 18, nota-se que, mesmo inserindo nos parâmetros de inicialização dos filtros os valores corretos das variâncias dos sensores, a estimação apresenta um aparente erro sistemático. Este

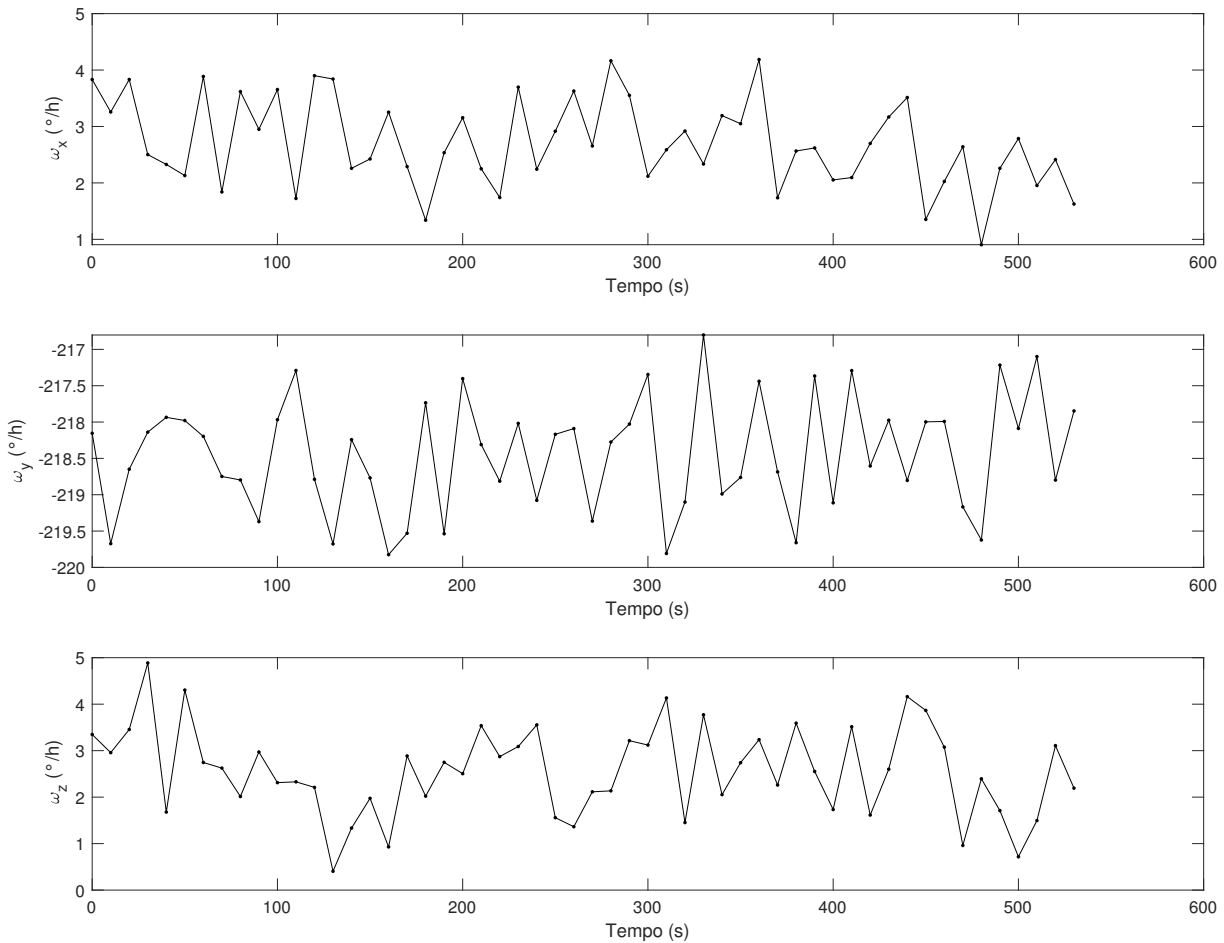


Figura 17 – Medidas simuladas dos girômetros.

erro na estimação acontece por consequência dos erros sistemáticos pertencentes aos sensores girômetros, que estão relacionados com a configuração dos parâmetros de inicialização dos filtros. Como os valores de variância respectivos aos girômetros são significativamente menores que os dos outros sensores, os filtros acabam “confiando” mais nas medidas dos girômetros, realizando uma ponderação que traz uma atitude mais influenciada pelos giroscópios, e conseqüentemente uma atitude enviesada. Além disso, como estes valores de velocidade angular são integrados para gerar valores de ângulo, os vieses são somados ao longo do tempo, podendo causar a divergência da estimação. Para o ângulo ψ , nota-se que o tempo de convergência é mais longo do que nos ângulos ϕ e θ , e isto se deve a dois fatores: a variância do sensor DSS é substancialmente maior do que no caso do IRES, sendo que os IRES não medem o ângulo ψ , além disso, nos parâmetros de inicialização, a variância da estimativa do erro inicial está com um valor abaixo do recomendável. Ademais, percebe-se que as medidas do Filtro de Kalman estendido se aproximam mais do valor real nos casos dos ângulos ϕ e θ , e os filtros H_∞ apresentam melhor desempenho para a medida do ângulo ψ .

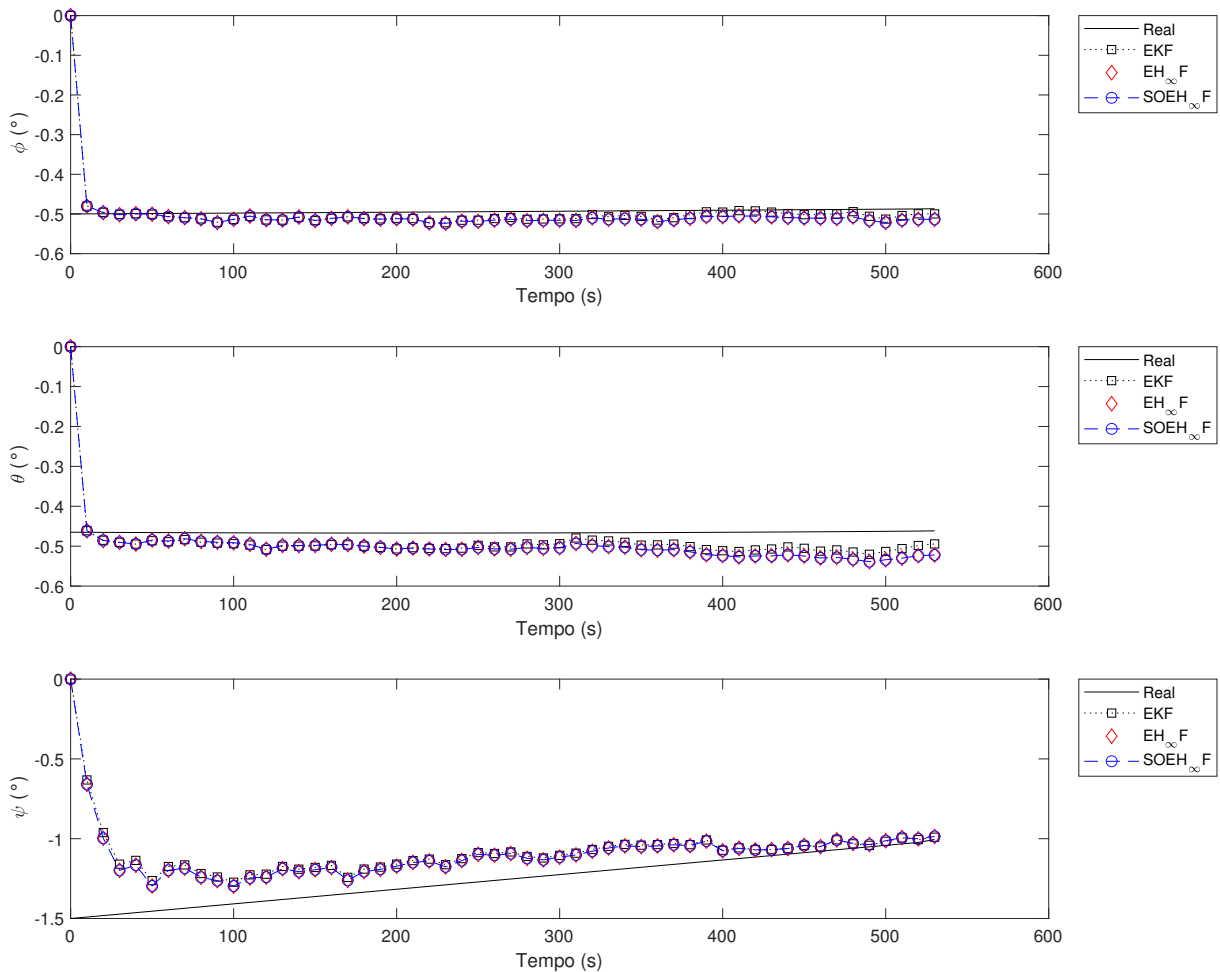


Figura 18 – Estimação de atitude utilizando dados simulados 1.

A partir do resultado da estimação de viés na Figura 19, observa-se que a estimação de viés inicialmente tende à convergência, porém, dentro do intervalo de tempo em que a simulação foi realizada, não houveram amostras suficientes para se determinar se a estimação de viés poderia convergir ou não. Este comportamento é dado devido ao fato de que os filtros não possuem dados de observação dos vieses nas medidas dos sensores, além do fato de que não há um algoritmo específico para a sua propagação. A estimação dos vieses no método utilizado neste trabalho, como apresentado por Lefferts, Markley e Shuster (1982) depende exclusivamente do ganho de atualização do filtro, que por sua vez está relacionado com a estimação da atitude. Por isso, há uma alta incerteza inerente à estimação de viés, e sua convergência acontece de forma lenta. Além disso, como a própria estimação da atitude neste caso não é ideal, consequentemente a estimação de viés não será. Quanto ao desempenho, os filtros H_{∞} apresentam um erro significativamente inferior.

As Figuras 20 e 21 trazem os resíduos dos filtros medidos antes e após a etapa de atualização. Os resíduos são os resultados da diferença entre os valores de saída medidos pelos sensores e os valores estimados pelo filtro. Neste caso, nota-se a partir

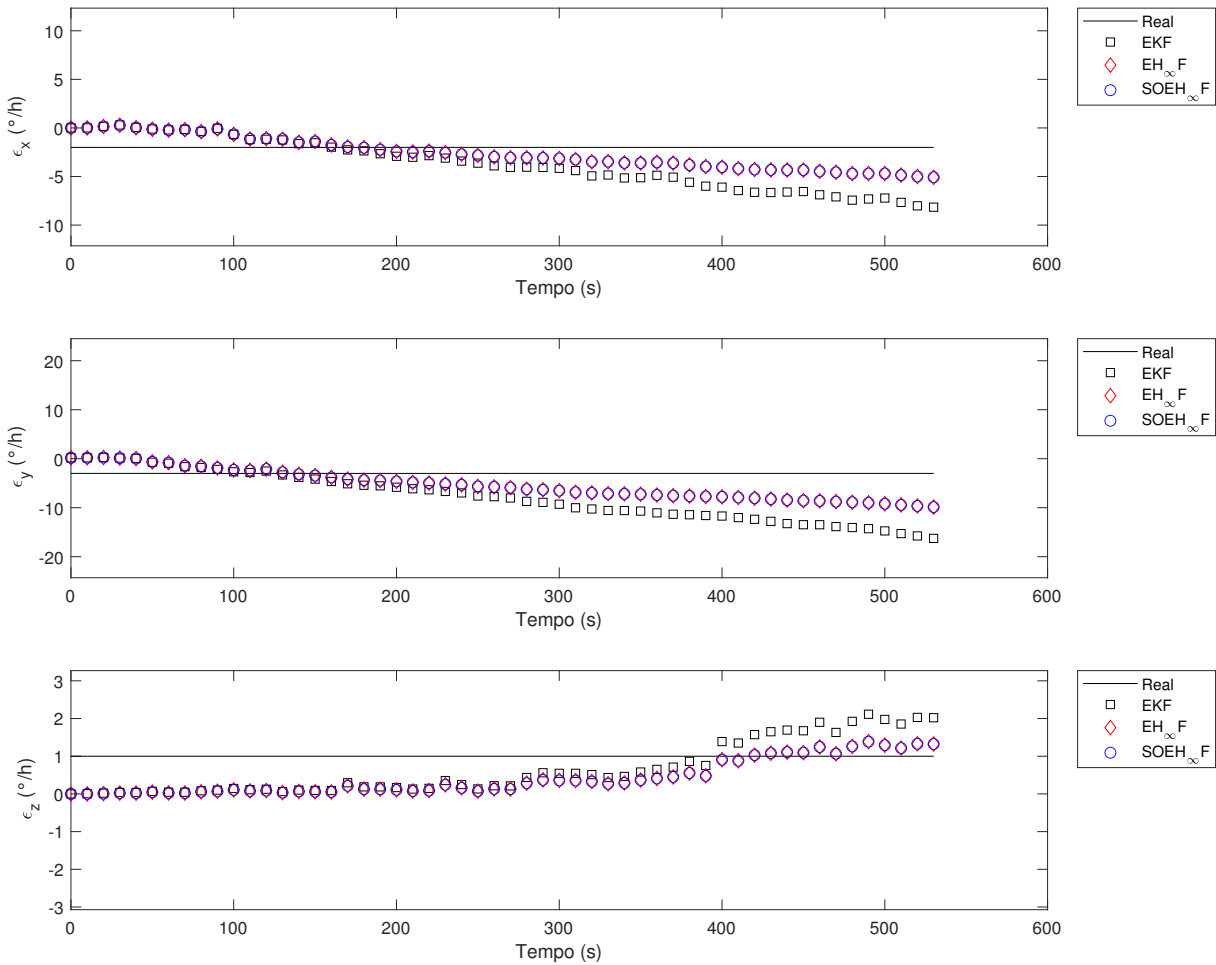


Figura 19 – Estimação de viés utilizando dados simulados 1.

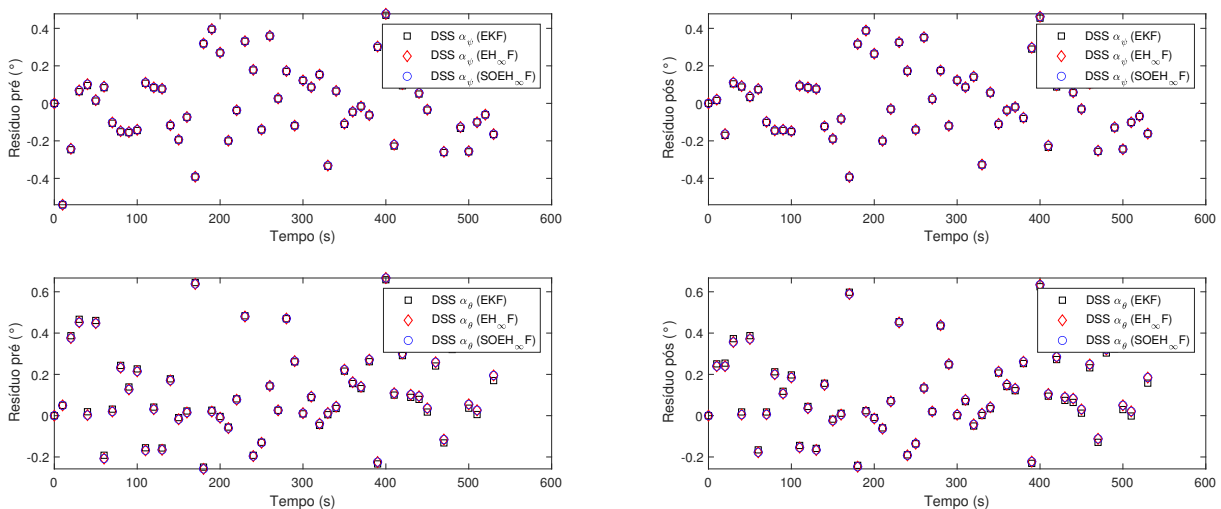


Figura 20 – Resíduos referentes aos DSS, pré e pós atualização, utilizando dados simulados 1.

das figuras que os resíduos não apresentam uma distribuição aleatória centralizada em 0. Este comportamento revela o viés existente nos girômetros, onde as saídas estimadas apresentam um erro sistemático com relação ao dado medido pelos sensores,

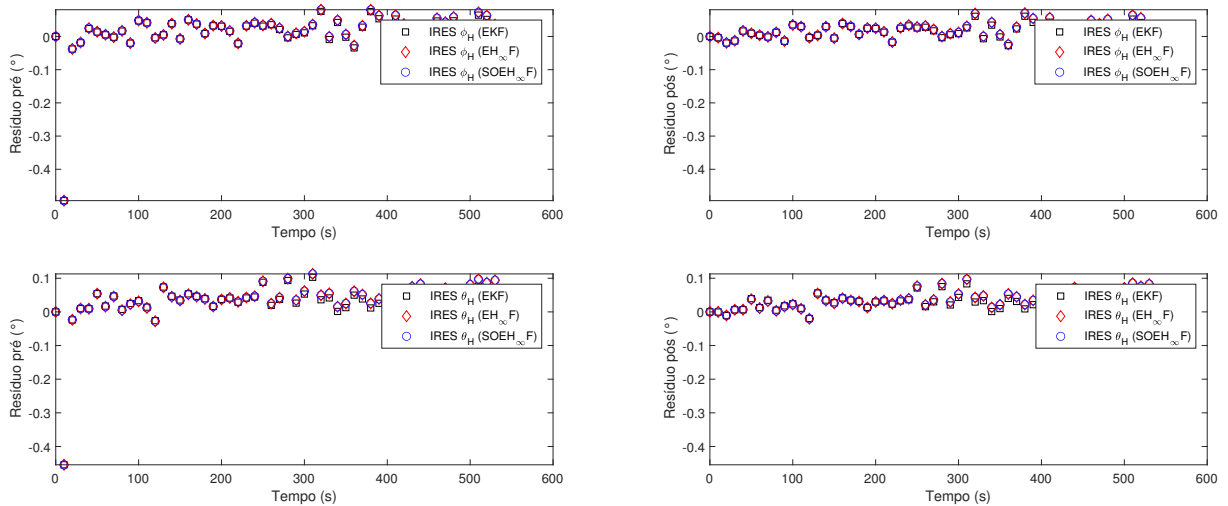


Figura 21 – Resíduos referentes aos IRES, pré e pós atualização, utilizando dados simulados 1.

por isso a diferença entre elas não está centralizada em 0.

As figuras nesta seção foram retiradas de uma única simulação, porém, devido à existência de variáveis aleatórias nas estimações, uma única simulação pode não ser capaz de apresentar resultados confiáveis, uma vez que tal resultado pode ser consequência de “sorte” ou “azar”. Sendo assim, com o objetivo de reduzir as interferências causadas pela aleatoriedade dos erros nos resultados, foi utilizado um método baseado no Método de Monte Carlo, onde várias simulações foram feitas, e foi computada a média dos resultados. Para comparar os desempenhos dos filtros, utilizando 100 simulações, foram calculados os erros RMS (*Root Mean Square*, ou Erro Quadrático Médio) de estimação, onde se realiza a raiz quadrada da média dos quadrados dos erros. Os resultados estão presentes nas Tabelas 5 e 6, onde primeiramente foram computados os erros RMS e posteriormente estes foram normalizados pelos valores referentes ao Filtro de Kalman Estendido, como forma de fácil visualização e comparação dos resultados.

	EKF	$EH_{\infty}F$	$SOEH_{\infty}F$
ϕ ($^{\circ}$)	0,070780893	0,072697801	0,072697814
θ ($^{\circ}$)	0,070126758	0,074974367	0,074974438
ψ ($^{\circ}$)	0,273513701	0,267300754	0,267299881
ϵ_x ($^{\circ}/h$)	3,186773548	1,885718083	1,885723294
ϵ_y ($^{\circ}/h$)	6,569193143	3,613012331	3,613176990
ϵ_z ($^{\circ}/h$)	0,793467907	0,811066702	0,811069201

Tabela 5 – Erros RMS 1 (100 simulações).

Observa-se, a partir das Tabelas 5 e 6, que os desempenhos dos filtros foram variados, onde o Filtro de Kalman Estendido apresenta melhores resultados para a

	EKF	EH _∞ F	SOEH _∞ F
ϕ	1	1,027082277	1,027082470
θ	1	1,069126376	1,069127395
ψ	1	0,977284695	0,977281501
ϵ_x	1	0,591732690	0,591734325
ϵ_y	1	0,549993318	0,550018383
ϵ_z	1	1,022179593	1,022182742

Tabela 6 – Erros RMS normalizados 1 (100 simulações).

estimação de ϕ , θ e ϵ_z , o Filtro H_∞ Estendido apresenta melhor desempenho para ϵ_x e ϵ_y , e o Filtro H_∞ Estendido de Segunda Ordem apresenta melhor desempenho para ψ . Nota-se que há uma grande semelhança entre os resultados dos filtros H_∞, e isto acontece pois, como dito na Seção 3.5.3, a construção destes filtros é muito similar, com a diferença de que o Filtro H_∞ Estendido de Segunda Ordem utiliza os termos de segunda ordem da expansão em série de Taylor. Como o problema analisado possui um baixo grau de não-linearidade, o filtro de segunda ordem acaba não demonstrando seu potencial, que deve trazer melhorias para problemas com maiores graus de não-linearidade. Comparando os resultados referentes à estimação de ϵ_z , observa-se na simulação da Figura 19 um melhor desempenho por parte da filtragem H_∞, porém, após realizar várias simulações, a Tabela 5 revela um melhor desempenho por parte do Filtro de Kalman Estendido. Isto revela o benefício de se utilizar o método de Monte Carlo, realizando várias simulações para evitar que casos únicos de sorte ou azar atrapalhem nos resultados.

O tempo médio gasto para realizar a estimação nos filtros também foi computado, e está disponível nas Tabelas 7 e 8, onde novamente são computados os valores brutos e normalizados, para facilitar a comparação entre os resultados. Observa-se que o filtro mais rápido, em outras palavras, que demanda menor poder computacional, é o Filtro de Kalman Estendido, seguido do Filtro H_∞ Estendido, e o mais lento é o Filtro H_∞ Estendido de Segunda Ordem, devido à sua maior quantidade de operações matemáticas. Vale notar que o mesmo programa e a mesma máquina foi utilizada na execução de todos os três filtros, para assim obter resultados comparáveis entre si. A máquina em questão possui um processador Intel® Core™ i5-8300H, memória RAM de 16 GB com 2666 MHz em *dual channel* e placa de vídeo NVIDIA® GeForce® GTX 1050 Ti (*Mobile*).

	EKF	EH _∞ F	SOEH _∞ F
Tempo (s)	0,005304387	0,009145611	0,017194737

Tabela 7 – Tempo médio gasto para processar os dados.

	EKF	$EH_{\infty}F$	$SOEH_{\infty}F$
Tempo	1	1,724159891	3,241606988

Tabela 8 – Tempo médio normalizado.

5.2 Estimação Utilizando Dados Simulados 2

Na primeira estimação, observou-se que a confiança demasiada na medida dos girômetros prejudicou os resultados dos filtros, devido à existência dos vieses. Por isso, para a segunda estimação, foi utilizada a estratégia apresentada em (SIMON, 2006), onde se aumenta a variância dos sensores como uma forma de compensar possíveis incertezas. Os parâmetros de inicialização dos filtros para a segunda estimação dos dados simulados estão descritos na Tabela 9, onde as variâncias respectivas aos girômetros foram aumentadas.

Parâmetro	Valor
Tempo de simulação	530 s
Estado inicial em ϕ, θ e ψ	0°
Estado inicial (ϵ)	$[0 \ 215,4 \ 0]^T \text{ } ^\circ/\text{h}$
Variância dos DSS	$(0,2^\circ)^2$
Variância dos IRES	$(0,02^\circ)^2$
Variância dos girômetros	$(0,015^\circ/\text{s})^2$
Variância dos vieses de girômetro	$(0,05^\circ/\text{h})^2$
Variância da estimativa do erro inicial em ϕ, θ e ψ	$(0,5^\circ)^2$
Variância da estimativa do erro inicial dos vieses	$(1^\circ/\text{h})^2$
γ	5000
η	0,5
ξ	1
λ_0	$[1 \ 1 \ 1 \ 1 \ 1 \ 1]^T$
S_k	I
L_k	I

Tabela 9 – Parâmetros de inicialização para a estimação utilizando dados simulados 2.

Para a segunda estimação, foram utilizados exatamente os mesmos dados provenientes dos sensores da primeira, disponibilizados nas Figuras 14, 15, 16 e 17, possuindo os mesmos erros, com o objetivo de se comparar as diferenças causadas pela modificação dos parâmetros de forma mais próxima e sem a interferência de outras variáveis.

Os resultados da segunda estimação utilizando dados simulados estão disponíveis a seguir, onde a estimação de atitude pode ser encontrada na Figura 22, a estimação de viés pode ser encontrada na Figura 23, os resíduos dos filtros referentes

aos sensores DSS estão na Figura 24 e os resíduos dos filtros referentes aos sensores IRES estão na Figura 25.

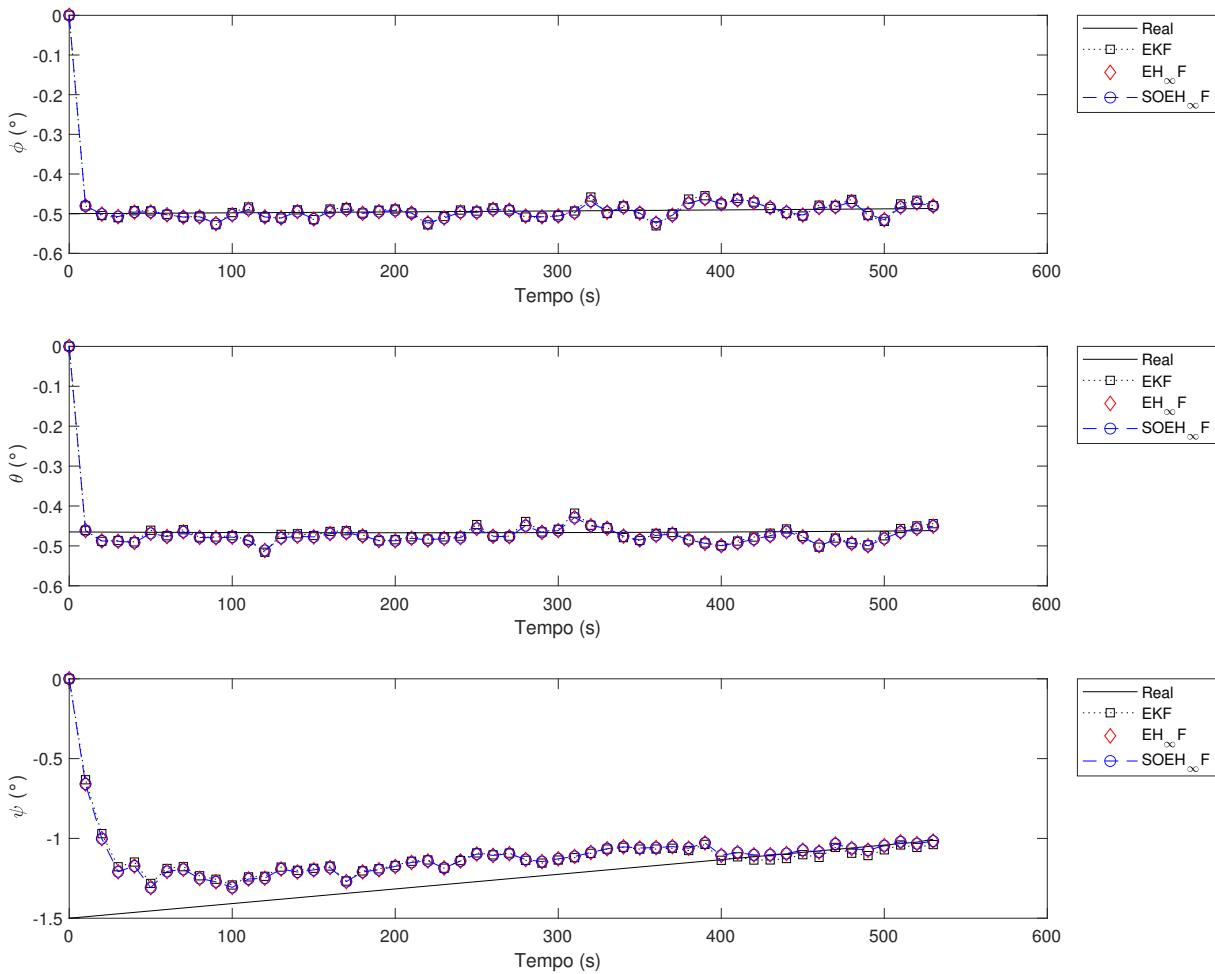


Figura 22 – Estimação de atitude utilizando dados simulados 2.

A partir da Figura 22, comparando com os resultados da Figura 18, percebe-se que houveram melhoras significativas na estimação de atitude, onde o viés já não pode mais ser notado. Este resultado comprova que a estratégia utilizada possui bastante potencial para lidar com os vieses. Além disso, observa-se um melhor desempenho por parte dos filtros H_∞ na estimação da atitude.

Analisando a Figura 23 e comparando com a Figura 19, percebe-se que, como houveram melhorias na estimação da atitude, também foram obtidas melhoras significativas na estimação dos vieses, onde há maior tendência à convergência, embora, como explicado na seção anterior, por questões de limitações do método utilizado, o tempo necessário para convergir continua demorado. A filtragem H_∞ apresenta melhor desempenho para as medidas de ϵ_x e ϵ_y , e o Filtro de Kalman Estendido apresenta menores erros para ϵ_z .

Observando as Figuras 24 e 25 e comparando-as com os resultados anteriores,

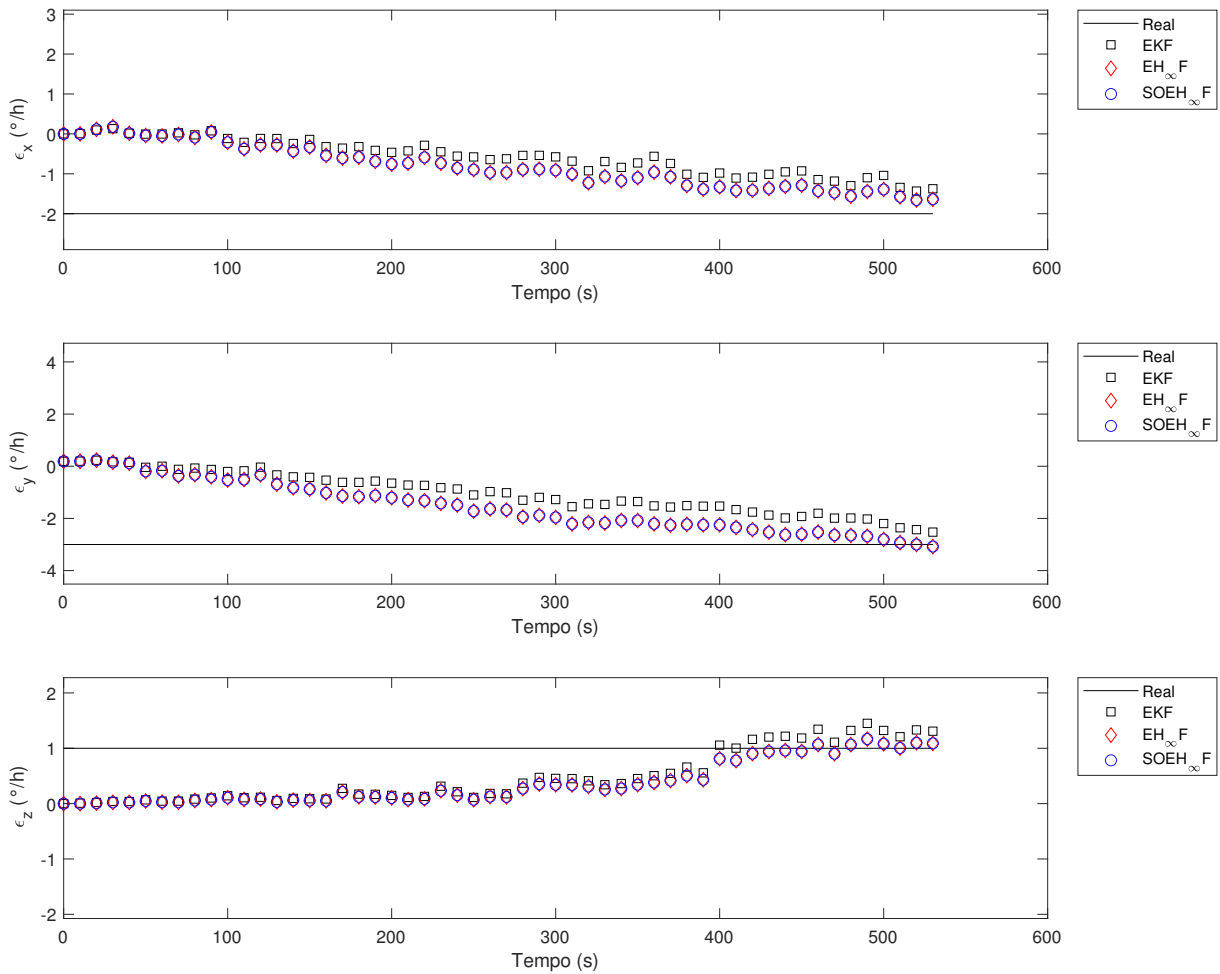


Figura 23 – Estimação de viés utilizando dados simulados 2.

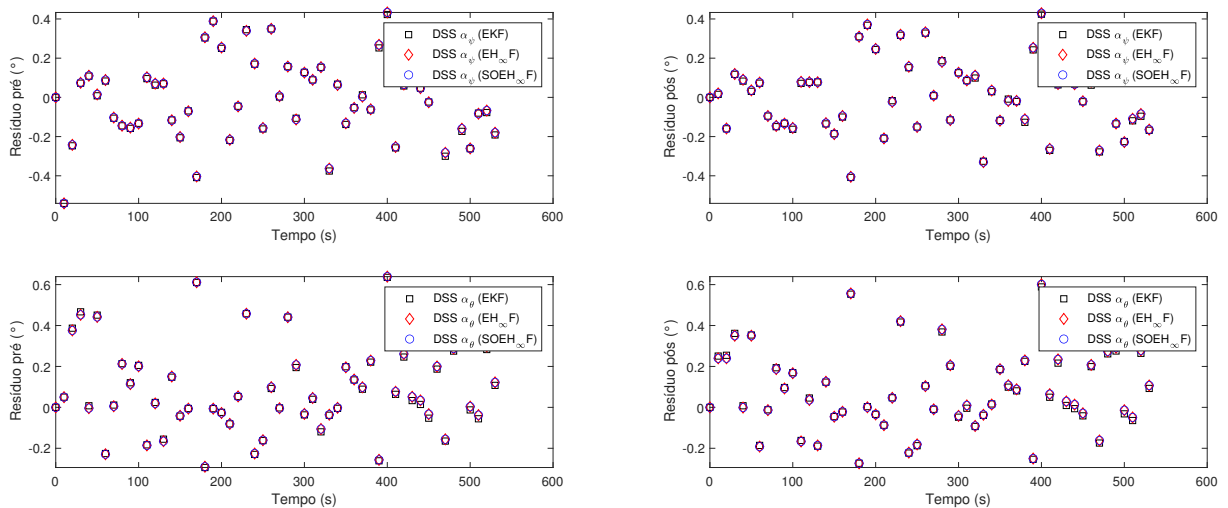


Figura 24 – Resíduos referentes aos DSS, pré e pós atualização, utilizando dados simulados 2.

nas Figuras 20 e 21, percebe-se uma maior centralização dos resíduos em 0, principalmente por parte dos sensores IRES. Isto acontece pois, com o aumento da variância dos girômetros, o filtro leva mais em conta as medidas dos sensores para a atualização

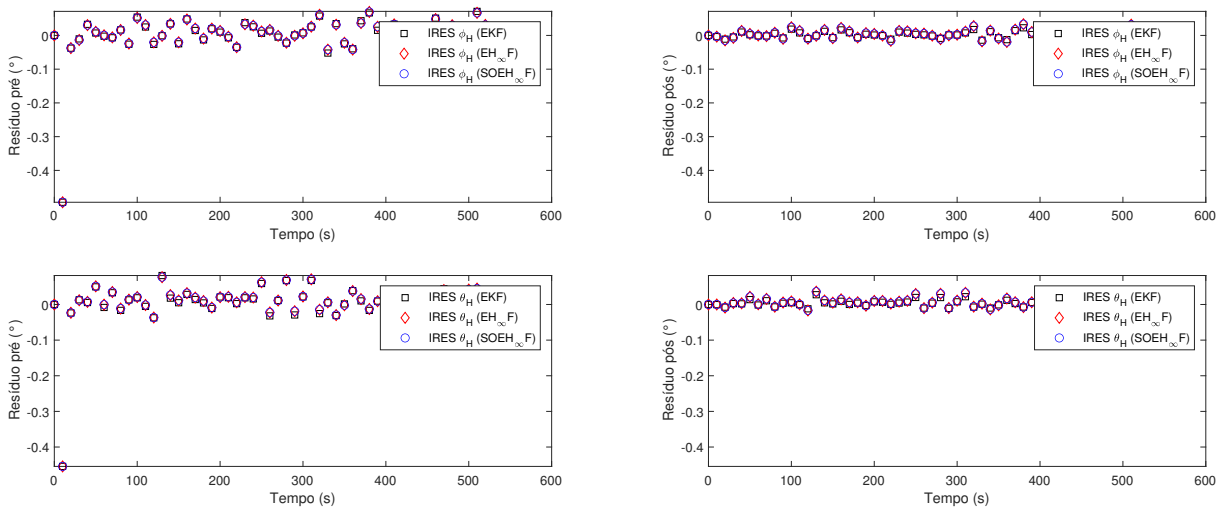


Figura 25 – Resíduos referentes aos IRES, pré e pós atualização, utilizando dados simulados 2.

das medidas. Não se observa mudanças significativas por parte dos sensores DSS, pois, mesmo aumentando a variância dos girômetros, a variância referente aos DSS continua sendo significativamente maior.

Novamente, para se analisar os desempenhos dos filtros, foi utilizado um método baseado no Método de Monte Carlo para calcular os erros RMS de estimação. Os resultados referentes a 100 simulações estão presentes nas Tabelas 10 e 11, onde se observa um desempenho superior por parte dos filtros H_∞ na estimação de todos os estados, com exceção de ϵ_z .

	EKF	$EH_\infty F$	$SOEH_\infty F$
ϕ ($^\circ$)	0,069613890	0,069310865	0,069310866
θ ($^\circ$)	0,065043074	0,064864859	0,064864878
ψ ($^\circ$)	0,264523756	0,259896179	0,259895147
ϵ_x ($^\circ/h$)	1,488529390	1,325332101	1,325327816
ϵ_y ($^\circ/h$)	2,142793014	1,780684594	1,780635037
ϵ_z ($^\circ/h$)	0,756172373	0,800408919	0,800412462

Tabela 10 – Erros RMS 2 (100 simulações).

	EKF	$EH_\infty F$	$SOEH_\infty F$
ϕ	1	0,995647061	0,995647072
θ	1	0,997260037	0,997260335
ψ	1	0,982506005	0,982502105
ϵ_x	1	0,890363408	0,890360530
ϵ_y	1	0,831011013	0,830987886
ϵ_z	1	1,058500611	1,058505296

Tabela 11 – Erros RMS normalizados 2 (100 simulações).

5.3 Estimação Utilizando Dados Simulados 3

Com o objetivo de comparar a robustez dos filtros utilizados, os desvios padrões dos ruídos dos sensores girômetro, DSS e IRES, disponíveis na Tabela 3, foram multiplicados por 10, e os dados dos sensores foram inseridos nos filtros. Os parâmetros de inicialização não foram modificados, pois a intenção é verificar o desempenho dos métodos de estimação onde há erros maiores do que o esperado, desta forma, foram utilizados os mesmos parâmetros da Seção 5.2, disponíveis na Tabela 9.

Para a terceira estimação utilizando dados simulados, a estimação de atitude pode ser encontrada na Figura 26 e a estimação de viés pode ser encontrada na Figura 27.

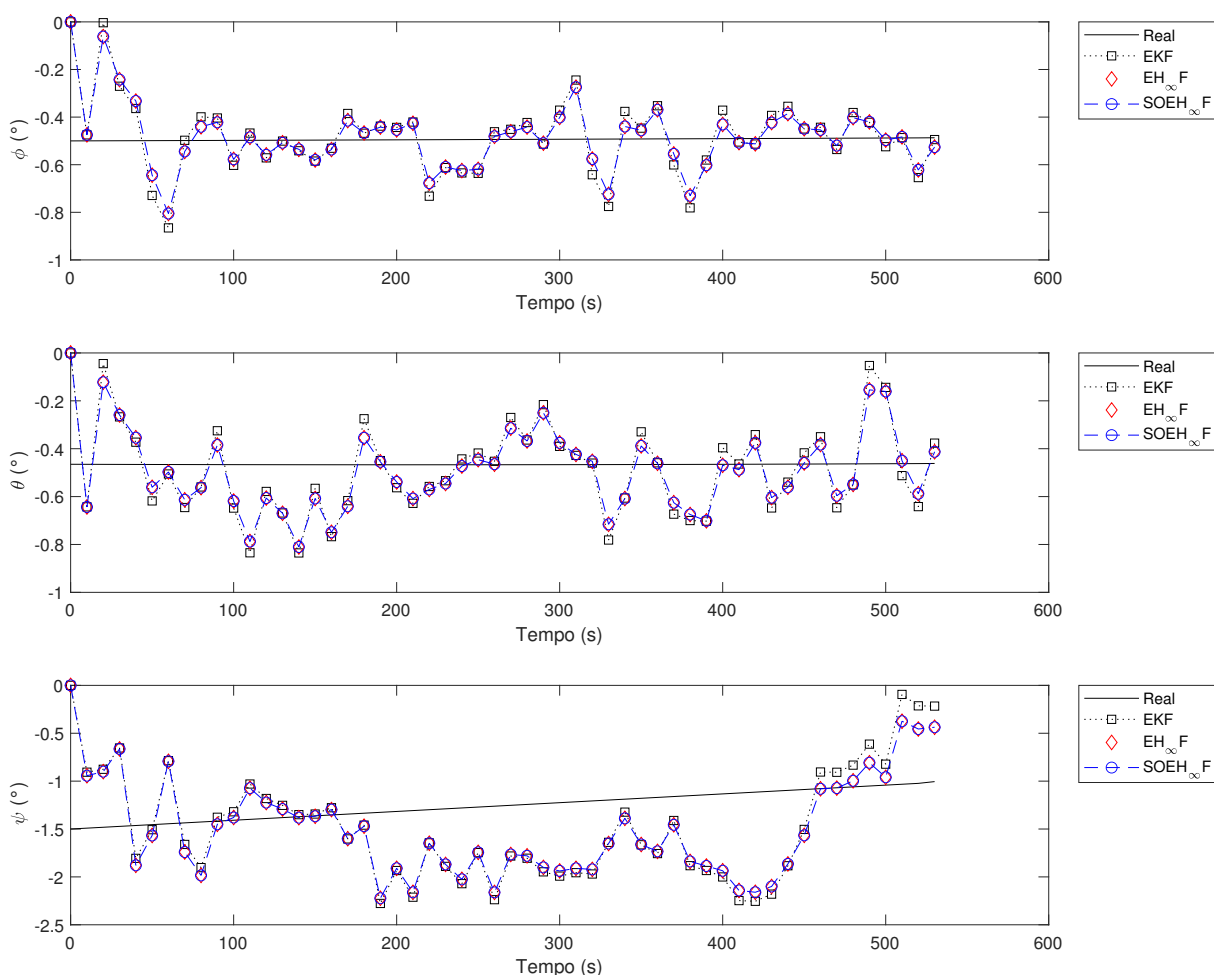


Figura 26 – Estimação de atitude utilizando dados simulados 3.

Comparando os resultados obtidos na Figura 26 com os da Figura 22, observa-se que o aumento nos desvios padrões dos ruídos dos sensores, sem que haja conhecimento prévio por parte do filtro, aumenta a ordem dos erros de estimação significativamente. Também é notável um melhor desempenho por parte da filtragem H_{∞} em todas as medidas de atitude.

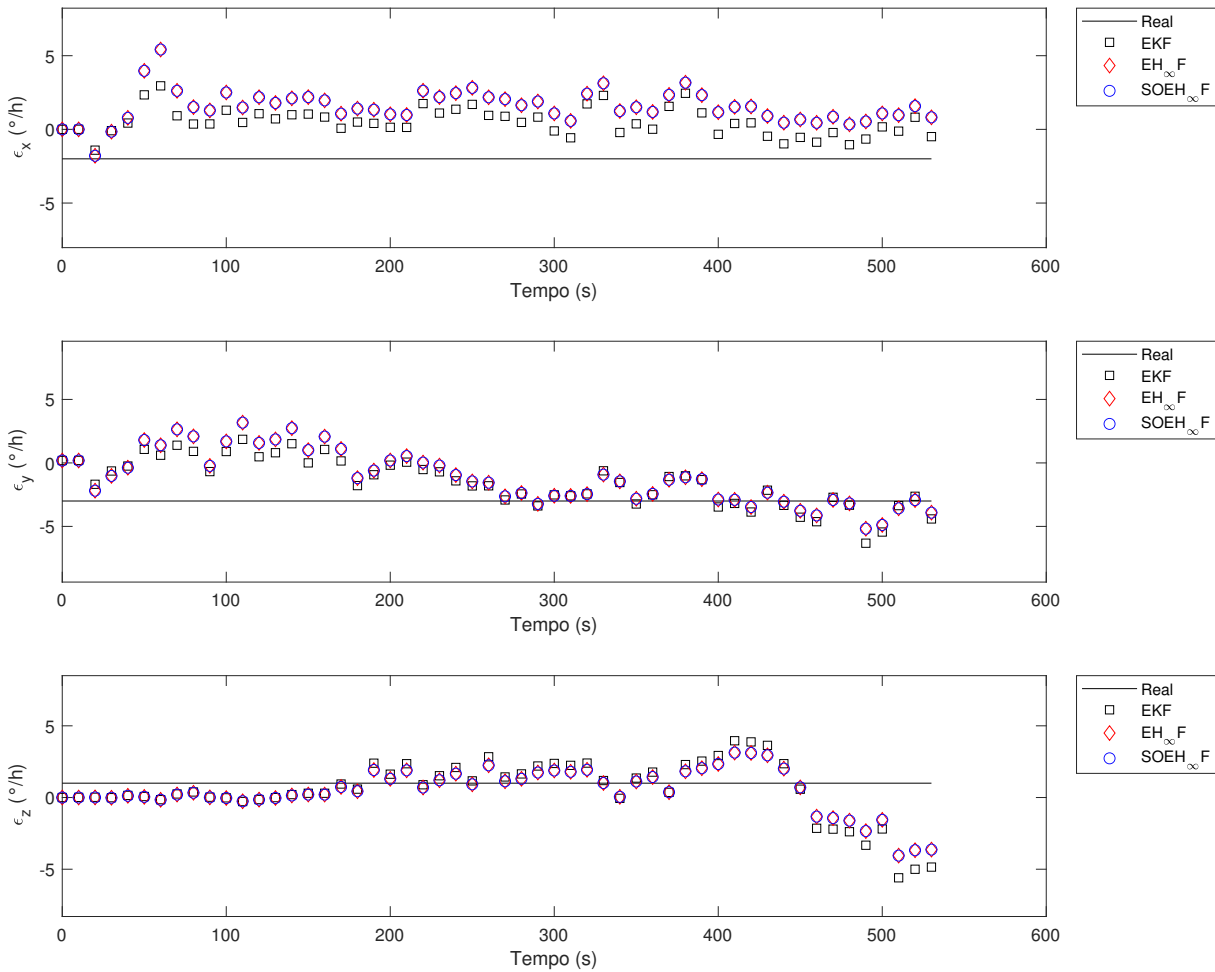


Figura 27 – Estimação de viés utilizando dados simulados 3.

Comparando a Figura 27 com as Figuras 23 e 19, observa-se, novamente, que o erro na estimação de atitude deteriora o erro da estimação de viés, para o método utilizado. Além disso, nota-se um desempenho superior por parte da filtragem de Kalman para as medidas de ϵ_x e ϵ_y , e menores erros por parte da filtragem H_∞ para ϵ_z .

Os resultados dos erros RMS de estimação a partir do Método de Monte Carlo estão presentes nas Tabelas 12 e 13, onde se nota um melhor desempenho na estimação de atitude e na estimação do ϵ_z por parte dos filtros H_∞ , comprovando os resultados obtidos nas Figuras 26 e 27.

	EKF	$EH_\infty F$	$SOEH_\infty F$
ϕ ($^\circ$)	0,159521368	0,142080738	0,142080738
θ ($^\circ$)	0,157362250	0,139175558	0,139175557
ψ ($^\circ$)	1,162864262	1,149033893	1,149029687
ϵ_x ($^\circ/h$)	2,338039177	2,961035757	2,961023080
ϵ_y ($^\circ/h$)	2,390239122	2,426574994	2,426546208
ϵ_z ($^\circ/h$)	2,941455660	2,482411737	2,482403623

Tabela 12 – Erros RMS 3 (100 simulações).

	EKF	EH _∞ F	SOEH _∞ F
ϕ	1	0,890669008	0,890669003
θ	1	0,884427860	0,884427849
ψ	1	0,988106635	0,988103018
ϵ_x	1	1,266461138	1,266455716
ϵ_y	1	1,015201773	1,015189730
ϵ_z	1	0,843939880	0,843937122

Tabela 13 – Erros RMS normalizados 3 (100 simulações).

5.4 Estimação Utilizando Dados Simulados 4

Na Seção 5.3, a robustez dos filtros foi analisada considerando-se somente o aumento por parte dos ruídos brancos gaussianos dos sensores, porém, como os vieses também afetam a estimação de forma significativa, conforme visto na Seção 5.1, esta seção visa comparar os desempenhos dos filtros multiplicando não somente os desvios padrões dos ruídos dos sensores, mas também multiplicando os vieses dos girômetros, disponíveis na Tabela 3, por 10.

Para a quarta estimação utilizando dados simulados, a estimação de atitude pode ser encontrada na Figura 28 e a estimação de viés pode ser encontrada na Figura 29.

A partir da Figura 28, nota-se um pequeno aumento na ordem dos erros, devido à multiplicação dos vieses, porém, comparando-a com a Figura 18 observa-se que, mesmo com a multiplicação os erros sistemáticos, a estratégia apresentada por Simon (2006), produz bons resultados, evitando a divergência dos filtros. Novamente nota-se um desempenho superior por parte da filtragem H_∞ para a estimação de atitude.

Como se pode observar na Figura 29, os erros na estimação de viés aumentam ainda mais, que é um resultado esperado, e nota-se um desempenho superior por parte da filtragem H_∞.

Os resultados dos erros RMS de estimação a partir do Método de Monte Carlo estão presentes nas Tabelas 14 e 15, onde se observa um melhor desempenho por parte dos filtros H_∞ em todos os estados, com exceção de ϵ_z (°/h).

	EKF	EH _∞ F	SOEH _∞ F
ϕ (°)	0,160036347	0,146651360	0,146651366
θ (°)	0,160237937	0,152957936	0,152958020
ψ (°)	1,111703381	1,109394605	1,109391181
ϵ_x (°/h)	13,37997943	11,25653774	11,25653151
ϵ_y (°/h)	19,97454407	16,63590014	16,63585034
ϵ_z (°/h)	9,323813892	9,379149618	9,379154660

Tabela 14 – Erros RMS 4 (100 simulações).

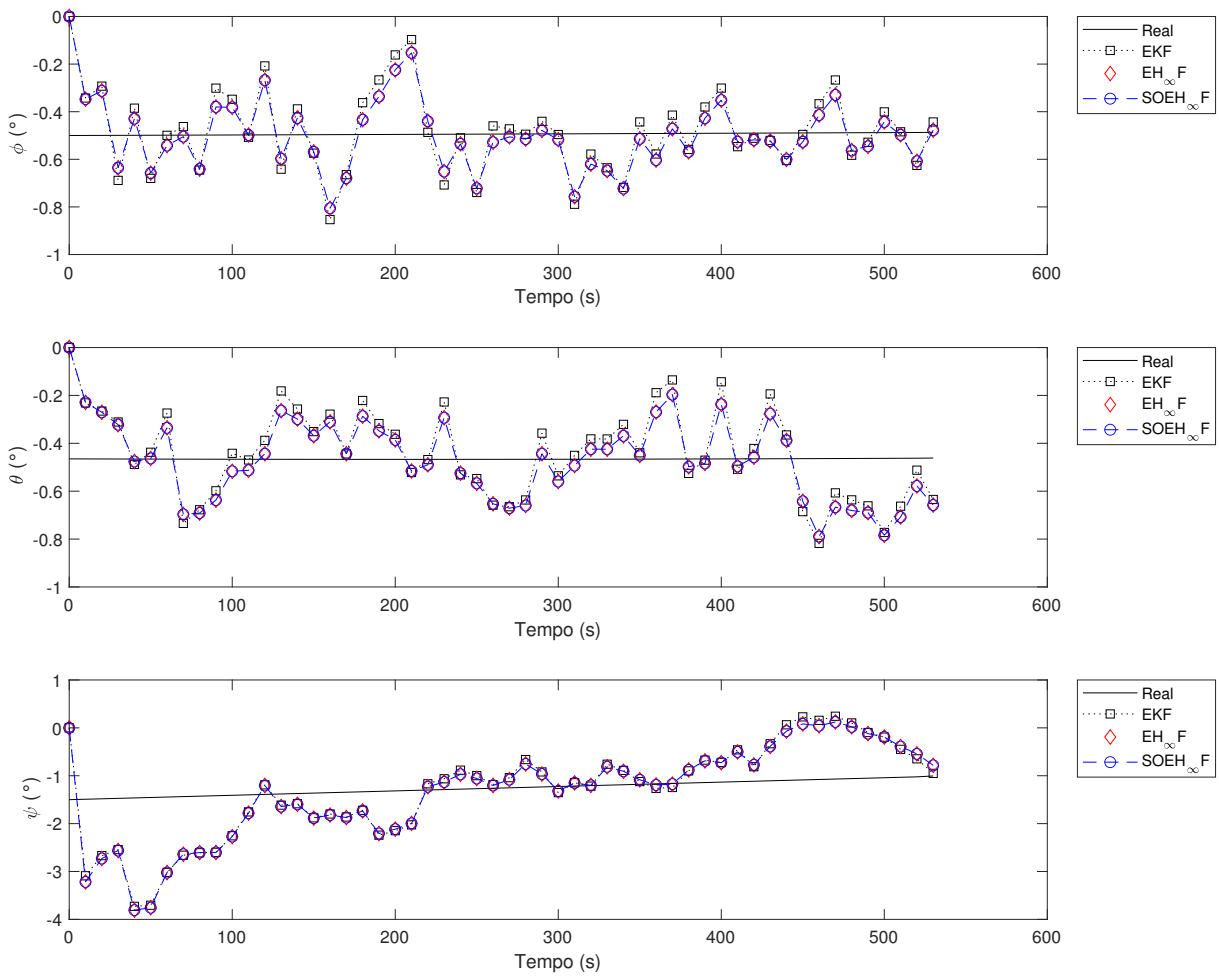


Figura 28 – Estimação de atitude utilizando dados simulados 4.

	EKF	$EH_{\infty}F$	$SOEH_{\infty}F$
ϕ	1	0,916362833	0,916362874
θ	1	0,954567551	0,954568080
ψ	1	0,997923209	0,997920128
ϵ_x	1	0,841297089	0,841296623
ϵ_y	1	0,832855062	0,832852569
ϵ_z	1	1,005934881	1,005935422

Tabela 15 – Erros RMS normalizados 4 (100 simulações).

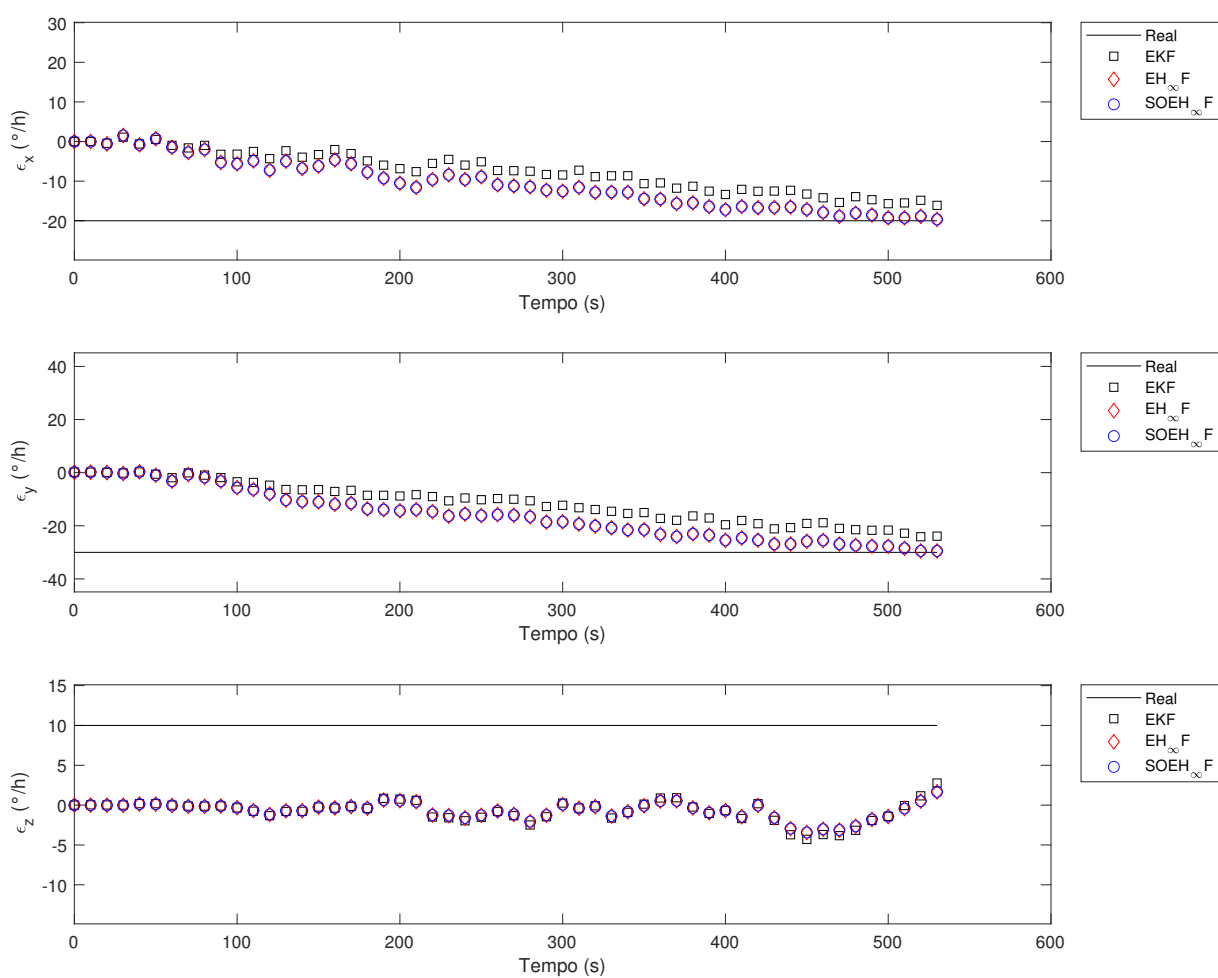


Figura 29 – Estimação de viés utilizando dados simulados 4.

5.5 Estimação Utilizando Dados Reais

A aplicação dos filtros e a estimação dos estados utilizando os dados simulados serviram como uma forma de confirmação que os filtros foram implementados corretamente e que realizam a estimativa de estados como esperado. Os resultados a partir das simulações são vitais para a estimação dos dados reais porque, se os valores aos quais os filtros estão estimando já são conhecidos, pode-se avaliar se os filtros estão funcionando corretamente. Já no caso da estimação prática, em que se utiliza os dados reais, não se sabe quais são os valores reais dos estados, por isso é importante que os filtros já tenham sido testados e ajustados de forma adequada antes da aplicação prática.

A partir dos conhecimentos adquiridos nas simulações, utilizou-se a mesma estratégia descrita na Seção 5.2, para evitar que a estimação seja demasiadamente deteriorada por vieses. Além disso, aumentou-se ainda mais as variâncias inseridas nos parâmetros de inicialização porque, nos sensores reais, há uma limitação de precisão, onde as medidas possuem Algarismos Significativos Limitados, como se pode observar nas Figuras 32 e 33. Isto não é problema no caso dos dados gerados em simulação computacional moderna, devido à alta quantidade de Algarismos Significativos disponíveis. Os parâmetros de inicialização dos filtros para a estimação utilizando dados reais estão descritos na Tabela 16.

Parâmetro	Valor
Tempo de simulação	530 s
Estado inicial em ϕ, θ e ψ	0°
Estado inicial (ϵ)	$[0 \ 215,4 \ 0]^T \text{ } ^\circ/\text{h}$
Variância dos DSS	$(0,4^\circ)^2$
Variância dos IRES	$(0,04^\circ)^2$
Variância dos girômetros	$(0,03^\circ/\text{s})^2$
Variância dos vieses de girômetro	$(0,05^\circ/\text{h})^2$
Variância da estimativa do erro inicial em ϕ, θ e ψ	$(0,5^\circ)^2$
Variância da estimativa do erro inicial dos vieses	$(1^\circ/\text{h})^2$
γ	5000
η	0,5
ξ	1
λ_0	$[1 \ 1 \ 1 \ 1 \ 1 \ 1]^T$
S_k	I
L_k	I

Tabela 16 – Parâmetros de inicialização para a estimação utilizando dados reais.

Os dados adquiridos dos sensores do satélite referentes à posição e velocidade no Sistema Equatorial Terrestre estão disponibilizados nas Figuras 30 e 31, que podem ser comparadas com as Figuras 14 e 15. As medidas advindas dos sensores DSS,

IRES e girômetros estão dispostas nas Figuras 32 e 33, que podem ser comparadas com as Figuras 16 e 17. Observando as imagens mencionadas, nota-se grande similaridade entre os dados gerados em simulação e os dados obtidos a partir dos sensores, e isto traz maior confiabilidade para as simulações.

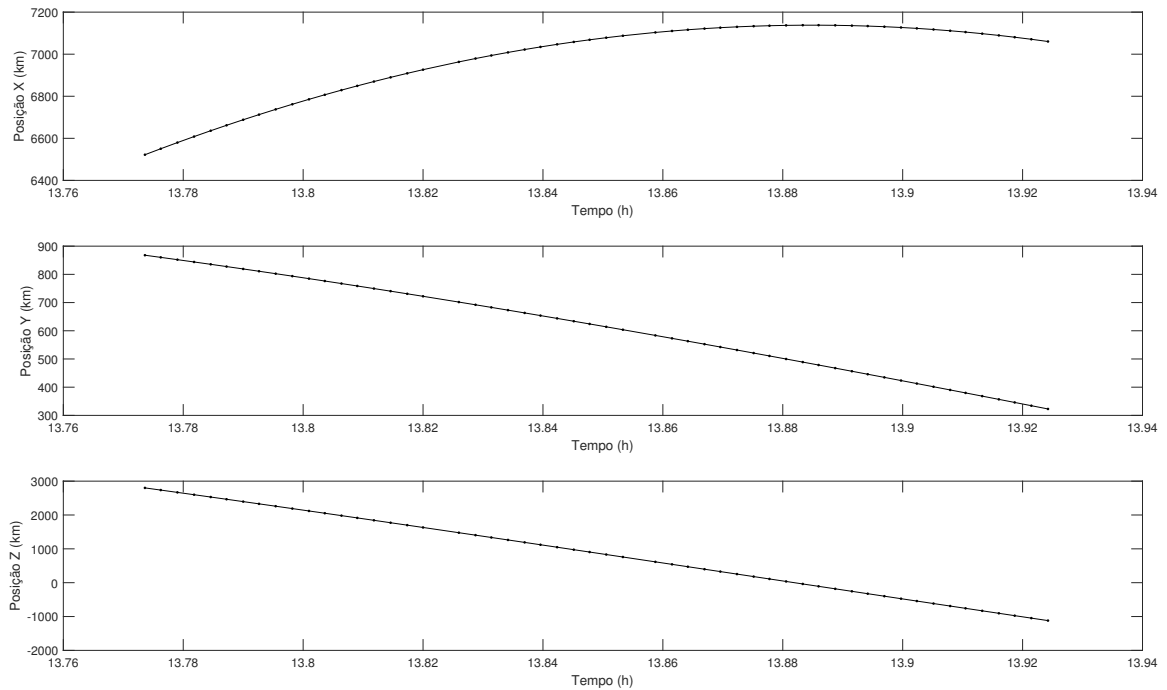


Figura 30 – Posição no Sistema Equatorial Terrestre.

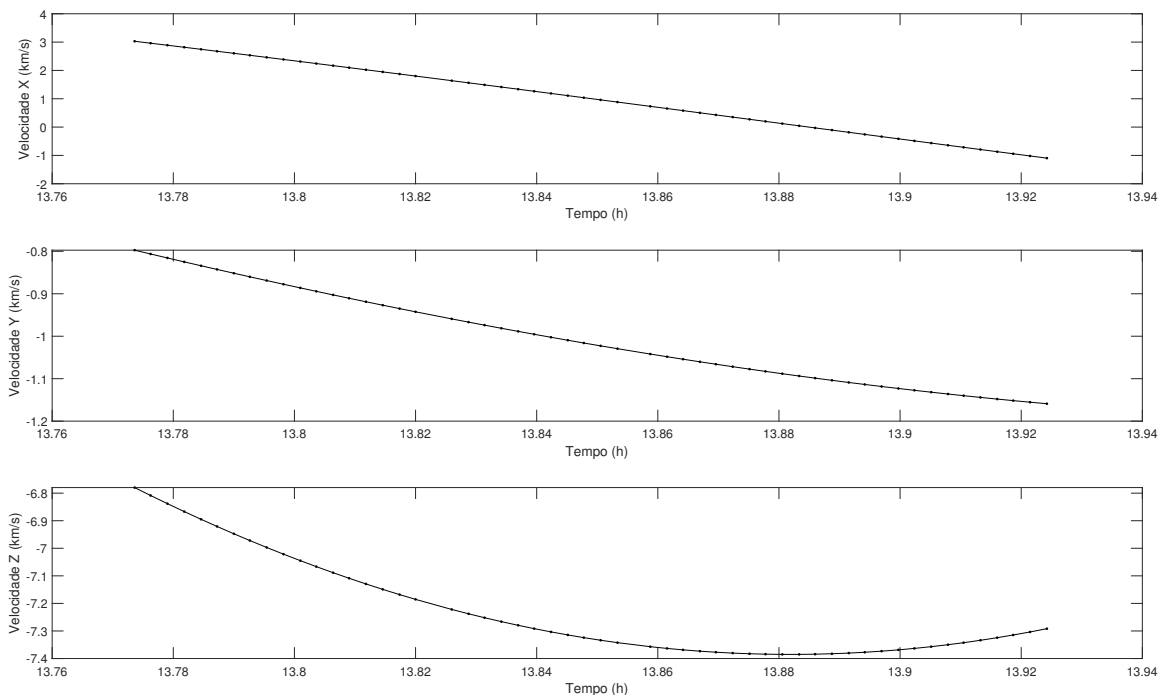


Figura 31 – Velocidade no Sistema Equatorial Terrestre.

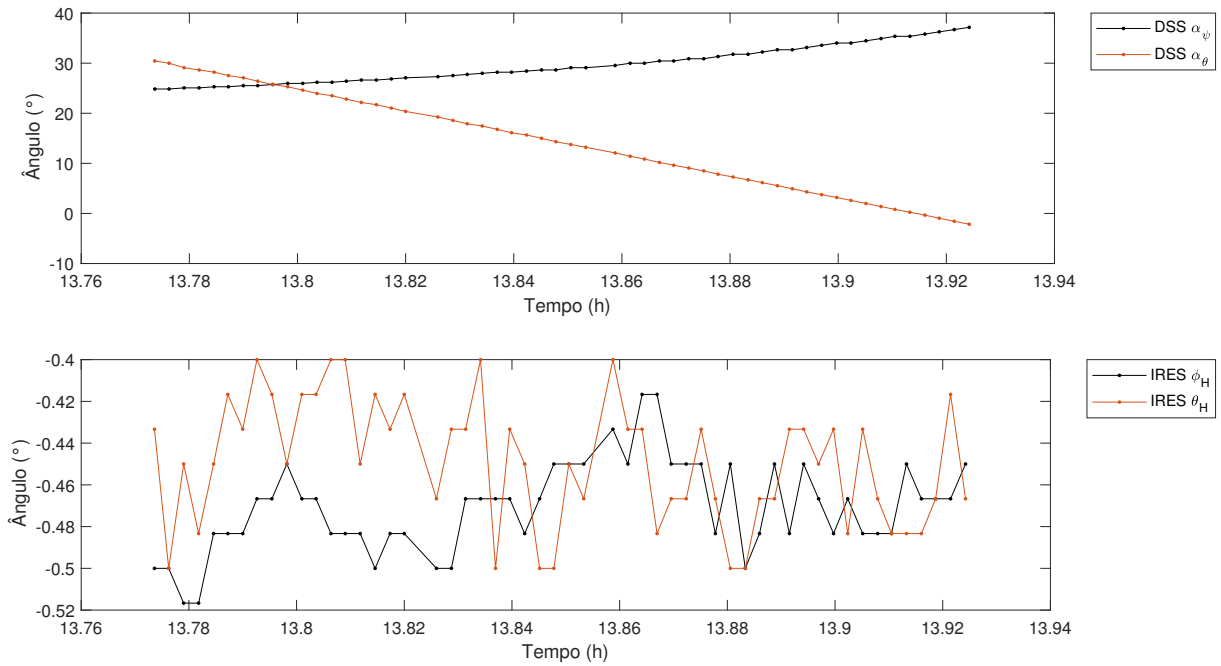


Figura 32 – Medidas reais dos sensores DSS e IRES.

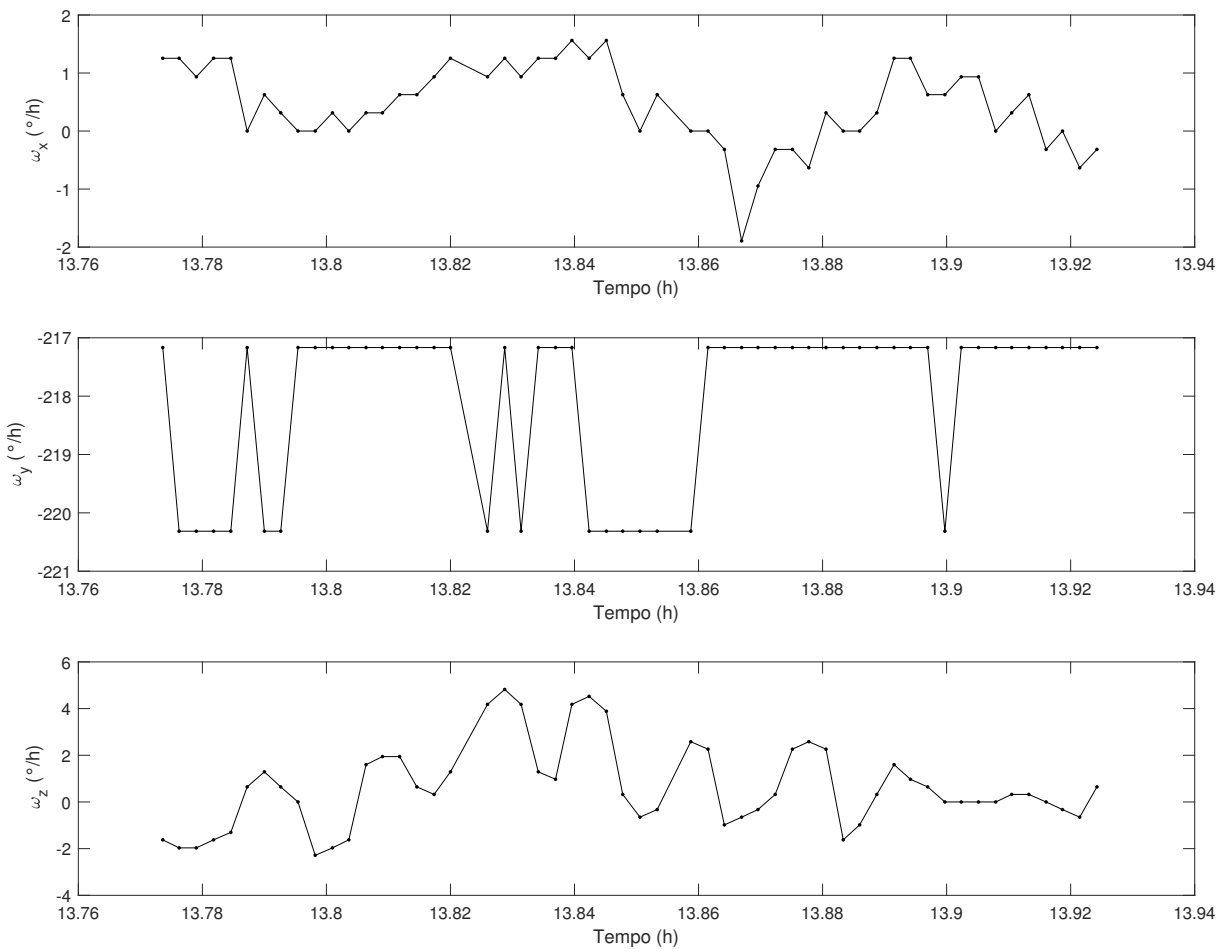


Figura 33 – Medidas reais dos girômetros.

Tendo obtido os dados dos sensores, realizou-se a estimação dos estados. Os resultados da estimação de atitude podem ser encontrados na Figura 34, onde σ_ϕ , σ_θ e σ_ψ são os desvios padrões da covariância do erro. A estimação de viés pode ser encontrada na Figura 35, os resíduos dos filtros referentes aos sensores DSS estão na Figura 36 e os resíduos dos filtros referentes aos sensores IRES estão na Figura 37.

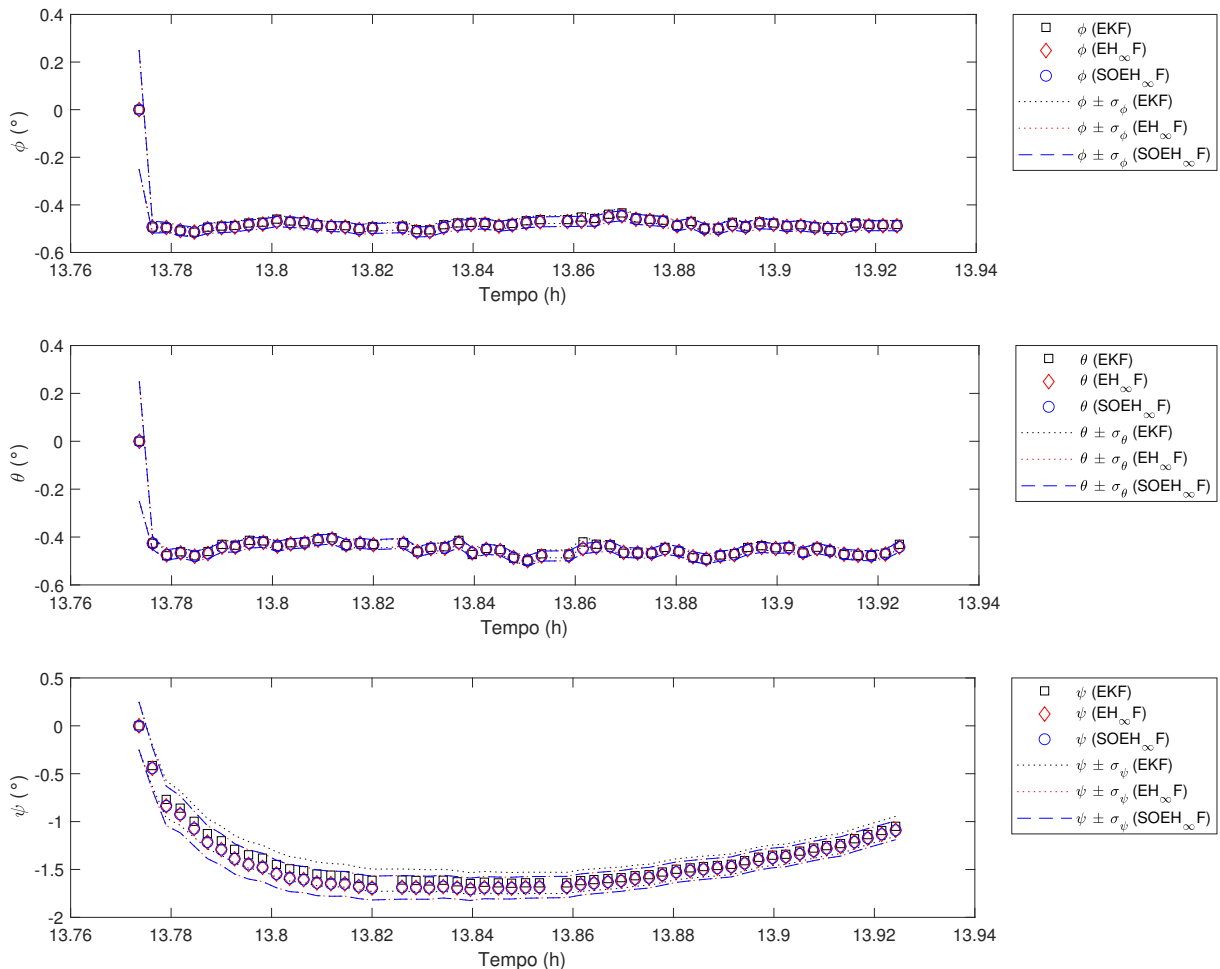


Figura 34 – Estimação de atitude utilizando dados reais.

A análise dos resultados obtidos na estimação dos dados reais foi baseada nos resultados referentes aos dados simulados, isto porque, como dito, não há como saber ao certo qual seria o estado verdadeiro. Sendo assim, comparando a Figura 34 com a Figura 22, nota-se que há uma rápida convergência na estimação dos ângulos ϕ e θ e uma convergência mais demorada para o ângulo ψ , que acontece devido à maior incerteza associada ao sensor.

Com base nas simulações, existe uma maior confiabilidade relacionada às medidas de atitude dos filtros H_∞ , isto porque, a partir do reajuste na variância dos ruídos dos girômetros, todas as medidas de atitude por parte da filtragem H_∞ obtiveram maior acurácia do que o Filtro de Kalman Estendido. Desta forma, espera-se que os ângulos reais estejam mais próximos das medidas dos filtros H_∞ .

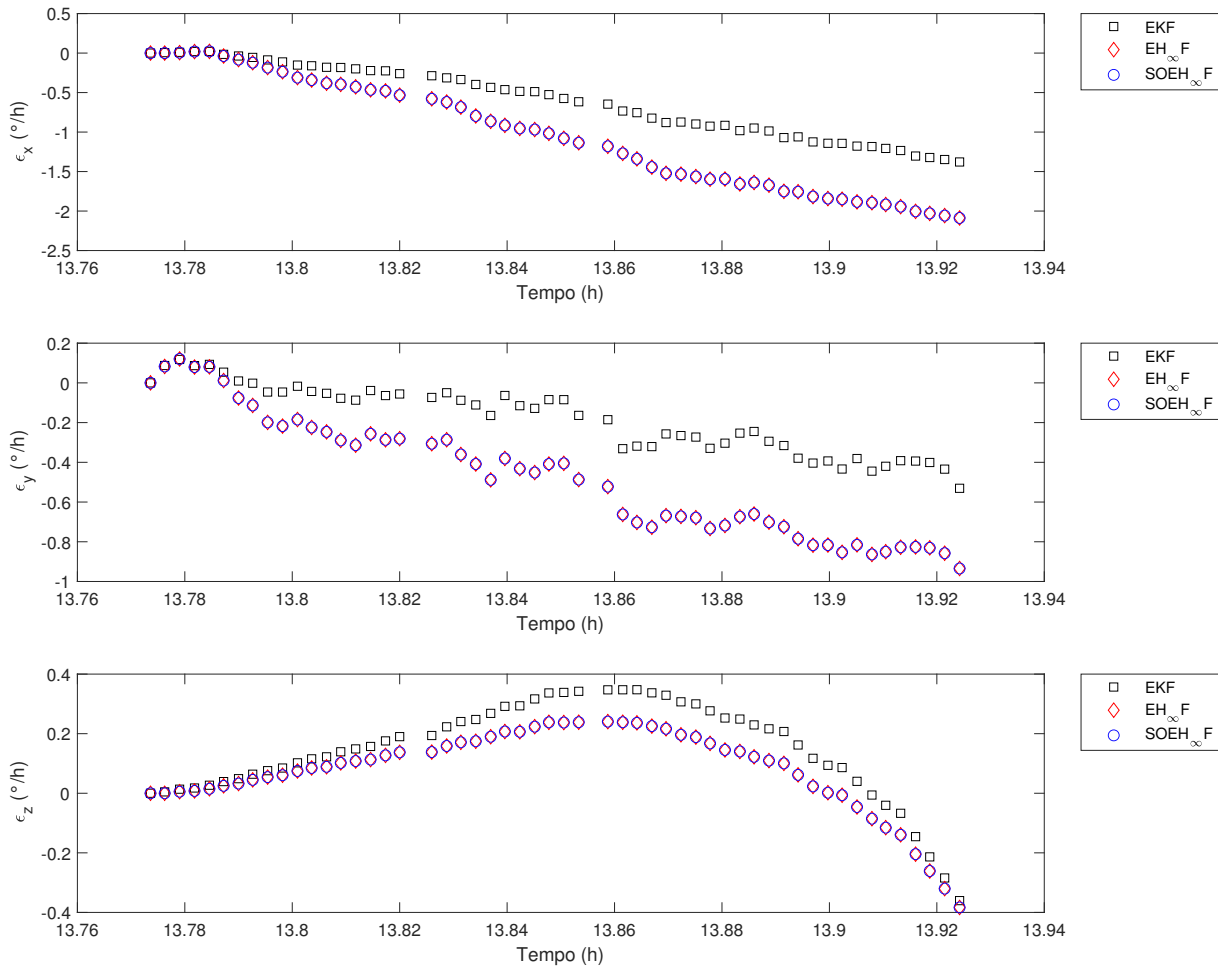


Figura 35 – Estimação de viés utilizando dados reais.

A partir da Figura 35, percebe-se um comportamento semelhante aos casos simulados, porém, deve-se ter em mente que, no caso do *toolbox* PROPAT, o viés foi constante, sendo que um sensor real pode apresentar diversos tipos de vieses, como constantes, lineares, e não-lineares. Além disso, a Figura 29 mostra que, para alguns casos, não há convergência dentro do intervalo estimado. Por isso, devido à grande incerteza inerente à estimação do viés neste trabalho, e a falta de um valor real, não há garantias de que os vieses estimados tenham convergido. Ademais, nas simulações, os desempenhos dos filtros com relação à estimação do viés foram variados, com isso, não há como deduzir qual das medidas se aproxima mais da real.

Analisando as Figuras 36 e 37, percebe-se que suas distribuições não estão centralizadas em 0, revelando que há a existência de viés por parte dos sensores. Nota-se através das Figuras 37 e 25 um comportamento similar por parte dos resíduos simulados e reais dos IRES, porém, observa-se uma diferença significativa entre os resíduos simulados e reais referentes aos DSS, a partir das Figuras 36 e 24, que demonstram que provavelmente há viés tanto nos sensores girômetros, quanto nos sensores de observação.

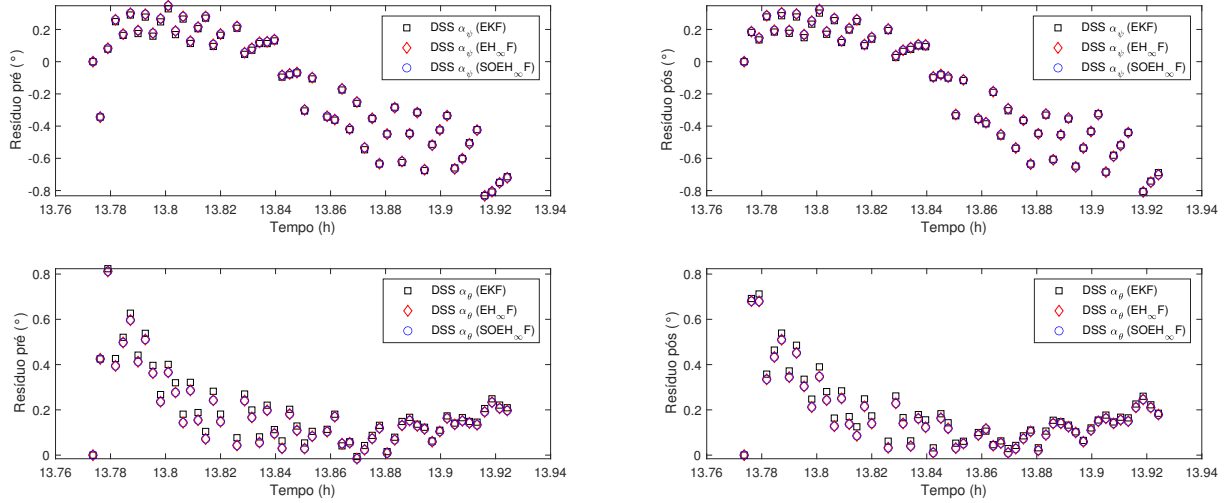


Figura 36 – Resíduos referentes aos DSS, pré e pós atualização, utilizando dados reais.

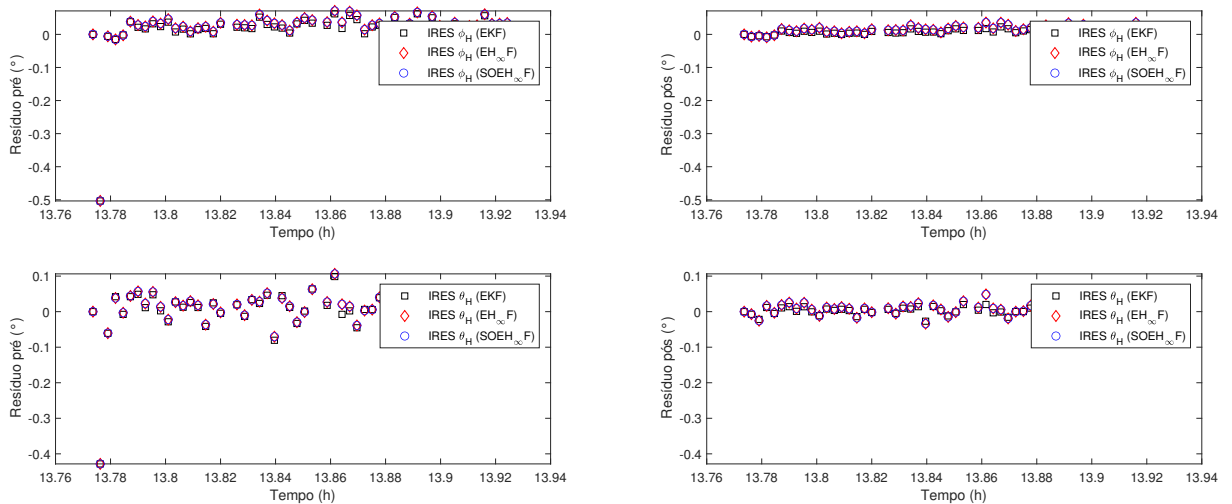


Figura 37 – Resíduos referentes aos IRES, pré e pós atualização, utilizando dados reais.

Esta página foi intencionalmente deixada em branco.

6 Conclusão

O objetivo principal deste trabalho foi a realização da estimação de atitude, baseada em ambos dados simulados e reais, do satélite CBERS-2, com diferentes métodos de estimação não-lineares, e a comparação de seus desempenhos. A partir dos resultados obtidos, dispostos no Capítulo 5, observa-se que o objetivo foi concluído para ambos os casos, visto que todos os filtros implementados foram capazes de estimar a atitude, observada através das Figuras 18, 22, 26, 28 e 34.

A partir das Tabelas 10, 12 e 14, observa-se que a Filtragem H_∞ obtém um desempenho claramente superior à Filtragem de Kalman para a estimação de atitude, tanto na situação em que procurou-se obter os melhores resultados (Seção 5.2), quanto nas situações onde houveram incertezas de modelagem (Seções 5.3 e 5.4). As estimativas de atitude provenientes dos filtros H_∞ chegaram a ser 13,1% mais precisas do que na filtragem de Kalman, para os casos onde houveram incertezas de modelagem. Porém, no caso onde os filtros foram devidamente calibrados para se obter as melhores medidas, a melhoria na acurácia foi menor do que 2%. Além disso, embora os métodos H_∞ tenham obtido melhores resultados, isto foi dado com a consequência de um custo computacional maior, como pode ser observado pela Tabela 7.

De acordo com Simon (2006), o Filtro de Kalman assume que o modelo do sistema e as estatísticas dos ruídos são conhecidas, e também assume que os ruídos sejam brancos gaussianos de média zero, assim, se qualquer uma dessas suposições são violadas, então as estimativas do filtro podem se degradar, por outro lado, o Filtro H_∞ não depende dessas suposições, embora possa se beneficiar delas. Isto explica o motivo pelo qual os filtros H_∞ obtiveram melhores resultados, pois, até mesmo na melhor calibração, os vieses continuaram a infringir as suposições da filtragem de Kalman.

O Filtro H_∞ Estendido de Segunda Ordem, para o caso estudado, produziu resultados muito similares aos do Filtro H_∞ Estendido, com uma diferença máxima percentual da ordem de 10^{-4} . Isto foi dado ao mesmo tempo em que se necessitou de praticamente o dobro do poder computacional do filtro de primeira ordem, como visto na Tabela 7.

Embora o Filtro de Kalman Estendido tenha obtido os maiores erros por parte da estimação de atitude, este foi o filtro que demandou menor poder computacional, de forma que o Filtro H_∞ Estendido precisou, em média, de 72,4% a mais de tempo, e o Filtro H_∞ Estendido de Segunda Ordem precisou de mais do que o triplo do tempo gasto para realizar o processamento dos dados.

No caso da estimação de atitude de satélites em tempo real, o poder computacional é um fator importante, pois a quantidade de potência disponível é limitada para o sistema embarcado, sendo assim, mesmo com piores medidas, a filtragem de Kalman pode ser utilizada, desde que seja precisa o suficiente para a aplicação.

De forma geral, para a aplicação real do caso estudado, a escolha entre um método de estimação ou outro vai depender dos requisitos de poder computacional, assim como os requisitos de precisão do filtro. Além disso, um fator importante a ser considerado, como visto nos resultados, é a relação de incerteza de modelagem. Dessa forma, tanto o Filtro de Kalman Estendido quanto o Filtro H_∞ Estendido podem ser utilizados em situações diferentes. O filtro de segunda ordem, para o caso estudado, não apresentou grandes vantagens, devido à seu baixo custo-benefício em termos de precisão e poder computacional requerido, quando comparado aos outros métodos aplicados.

Quanto à estimação do viés dos sensores girômetros, os resultados revelaram um desempenho variado entre os filtros, onde não se obtiveram resultados consistentes, como foi no caso da estimação de atitude, porém, de forma geral, menores erros foram obtidos por parte da filtragem H_∞ .

Referências

- BANAVAR, R. A game theoretic approach to linear dynamics estimation. *Doctoral Dissertation*, University of Texas at Austin, 1992. Citado 2 vezes nas páginas 2 e 28.
- BAR-ITZHACK, I. Y.; OSHMAN, Y. Attitude determination from vector observations: Quaternion estimation. *IEEE Transactions on Aerospace and Electronic Systems*, IEEE, AES-21, p. 128–135, 1985. Citado na página 28.
- CARRARA, V. An open source satellite attitude and orbit simulator toolbox for matlab. *Proceedings of the XVII International Symposium on Dynamic Problems of Mechanics*, ABCM, Natal, RN, 2015. Citado na página 4.
- CORDEIRO, T. F. K. Fusão descentralizada com filtragem de kalman estendida para estimação de atitude e deriva com rede de sensores inerciais e magnetômetro de baixo custo. *Dissertação de Mestrado em Sistemas e Controle*, Instituto Tecnológico de Aeronáutica, São José dos Campos, 2012. Citado na página 1.
- FEHSE, W. Automated rendezvous and docking of spacecraft. *Cambridge University Press*, 2003. Citado 4 vezes nas páginas xiii, 14, 15 e 16.
- FUMING, H.; KUGA, H. K. CBERS simulator mathematical models. CBTT Project, CBTT/2000/MM/001, São José dos Campos, 1999. Citado na página 25.
- GARCIA, R. V. Filtro não-linear de kalman sigma-ponto com algoritmo unscented aplicado a estimativa dinâmica da atitude de satélites artificiais. *Tese de Doutorado*, Instituto Nacional de Pesquisas Espaciais, São José dos Campos, São Paulo, Brasil, 2011. Citado 3 vezes nas páginas 2, 8 e 24.
- HAMILTON, W. R. On a new species of imaginary quantities connected with a theory of quaternions. *Proceedings of the Royal Irish Academy*, v. 2, p. 424–434, 1844. Citado 2 vezes nas páginas 2 e 20.
- HAMILTON, W. R. On quaternions, or on a new system of imaginaries in algebra. *Philosophical Magazine*, v. 25, n. 3, p. 489–495, 1844. Citado na página 20.
- HU, J. S.; YANG, C. H. Second-order extended H_∞ filter for nonlinear discrete-time systems using quadratic error matrix approximation. *IEEE Transactions on Signal Processing*, v. 59, n. 7, p. 3110–3119, 2011. Citado 5 vezes nas páginas 2, 8, 29, 30 e 31.
- INPE. *CBERS - China-Brazil Earth Resources Satellite ou Satélite Sino-Brasileiro de Recursos Terrestres*. 2014. Disponível em: <<http://www.dgi.inpe.br/documentacao/satelites/cbers#wrapper>>. Acesso em: 24/04/2022. Citado na página 11.
- INPE. *CBERS-1, 2 e 2B*. 2018. Disponível em: <<http://www.cbers.inpe.br/sobre/cbers1-2-2b.php>>. Acesso em: 24/04/2022. Citado 2 vezes nas páginas xiii e 12.
- INPE. *Câmeras Imageadoras CBERS-1, 2 e 2B*. 2018. Disponível em: <<http://www.cbers.inpe.br/sobre/cameras/cbers1-2-2b.php>>. Acesso em: 24/04/2022. Citado 3 vezes nas páginas xiii, 11 e 12.

INPE. *Órbita do CBERS-1, 2 e 2B*. 2018. Disponível em: <<http://www.cbbers.inpe.br/sobre/orbita/cbers1-2-2b.php>>. Acesso em: 24/04/2022. Citado 2 vezes nas páginas xiii e 13.

KALMAN, R. E. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, v. 82, n. 1, p. 35–45, 03 1960. ISSN 0021-9223. Disponível em: <<https://doi.org/10.1115/1.3662552>>. Citado 2 vezes nas páginas 2 e 26.

LEFFERTS, E. J.; MARKLEY, F. L.; SHUSTER, M. D. Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, v. 5, n. 5, p. 417–429, 1982. Citado 9 vezes nas páginas 2, 3, 7, 20, 23, 27, 30, 34 e 41.

MARKLEY, F. L.; CRASSIDIS, J. L. *Fundamentals of spacecraft attitude determination and control*. Springer, 2014. Citado 4 vezes nas páginas 9, 21, 22 e 23.

MCGEE, L. A.; SCHMIDT, S. F. Discovery of the kalman filter as a practical tool for aerospace and industry. NASA Ames Research Center Moffett Field, CA 94035, 1985. Citado na página 2.

QUINSAC, G. et al. Attitude control: A key factor during the design of low-thrust propulsion for cubesats. *Acta Astronautica*, v. 176, p. 40–51, 2020. ISSN 0094-5765. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0094576520301818>>. Citado na página 1.

SANTOS, D. A. Estimação de atitude e velocidade angular de satélites utilizando medidas do campo geomagnético e da direção do sol. *Dissertação de Mestrado em Sistemas e Controle*, Instituto Tecnológico de Aeronáutica, São José dos Campos, 2008. Citado na página 1.

SHAKED, U.; BERMAN, N. H infinity nonlinear filtering of discrete-time processes. *IEEE Trans. Signal Process.*, v. 43, n. 9, p. 2205–2209, 1995. Citado 2 vezes nas páginas 7 e 28.

SILVA, W. R. Filtro H_{∞} estendido de segunda ordem e filtro de partículas regularizado com roughening aplicados na estimação de atitude de satélites artificiais. *Tese de Doutorado*, Instituto Nacional de Pesquisas Espaciais, São José dos Campos, São Paulo, Brasil, 2016. Citado 10 vezes nas páginas xiii, 1, 2, 9, 13, 15, 16, 17, 27 e 31.

SILVA, W. R. et al. Rao-blackwellized particle filter for the CBERS-4 attitude and gyros bias estimation. *Acta Astronautica*, 2021. ISSN 0094-5765. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0094576521004148>>. Citado 2 vezes nas páginas 23 e 34.

SILVA, W. R.; KUGA, H. K.; ZANARDI, M. C. Application of the extended H_{∞} filter for attitude determination and gyro calibration. *Advances in the Astronautical Sciences*, v. 152, p. 3848, 01 2014. Citado 4 vezes nas páginas 2, 8, 26 e 30.

SILVA, W. R.; KUGA, H. K.; ZANARDI, M. C. The extended H_{∞} filter for calibration and attitude determination using real data of cbers-2 satellite. *XXIV International Symposium on Space Flight Dynamics*, Laurel, 2014. Citado 2 vezes nas páginas xiii e 19.

SILVA, W. R.; KUGA, H. K.; ZANARDI, M. C. O filtro H_{∞} estendido de segunda ordem para estimação de atitude e bias de giros. *Proceeding Series of the Brazilian Society of Applied and Computational Mathematics*, v. 3, n. 1, p. 010166–1–010166–7, 2015. Citado 2 vezes nas páginas xiii e 18.

SIMON, D. Optimal state estimation: Kalman, H_{∞} , and nonlinear approaches. John Wiley & Sons, 2006. Citado 9 vezes nas páginas 2, 8, 26, 29, 30, 37, 45, 51 e 61.

STEVENS, B. L.; LEWIS, F. L.; JOHNSON, E. N. Aircraft control and simulation. dynamics, controls design, and autonomous systems. *John Wiley and Sons*, 2015. Citado na página 19.

SUN, T. et al. A new method for attitude refinement of earth observation satellite with asynchronous images. *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, p. 1221–1224, 2016. Citado na página 1.

TSAI, D. C. The effects of thrust uncertainty and attitude knowledge errors on the mms formation maintenance maneuver. NASA Goddard Space Flight Center Greenbelt, MD, United States, 2005. Citado na página 1.

WERTZ, J. R. Spacecraft attitude determination and control. Springer Netherlands, v. 73, 1978. Citado 3 vezes nas páginas 1, 17 e 24.

WIKIPEDIA. *Quaternion*. 2022. Disponível em: <<https://en.wikipedia.org/wiki/Quaternion>>. Acesso em: 24/04/2022. Citado 2 vezes nas páginas xiii e 21.

Esta página foi intencionalmente deixada em branco.

APÊNDICE A – Códigos dos Filtros em Linguagem MATLAB

A.1 Parâmetros de Inicialização

```
% Condições Iniciais: Filtro de Kalman Estendido, Filtro Hinf
e FHinfSO em Quaternions

% Transformações de unidades:

rad = pi/180;
deg = 180/pi;

% Tabelas referentes aos dados de efemerides e telemetrias:

Telemetria1_CBERS2;
Efemerides1_CBERS2;

% Número de loops:

m=length(T_M);

% Tempo inicial em seg:

TS0 = Efm(1,2)-10;

% Dado de entrada: vetor de estado inicial

% Entrada: ângulos de Euler(rad) e bias do gyro(rad/s)

x0=zeros(7,1);

fi=0*rad;
teta=0*rad;
psi=0*rad;
```

```

[ q ] = convetEuler_Q(fi,teta,psi);

x0(1)=q(1);
x0(2)=q(2);
x0(3)=q(3);
x0(4)=q(4);

x0(5)=0;
x0(6)=1/sqrt((rr0)^3/3.986004418e14); % Simulacao
% x0(6)=sqrt(3.986004418e14/(6371000 + 778000)^3); % Real
x0(7)=0;

% velocidade angular estimada pelo gyro:

West = zeros(3,1);

%=====Cálculo das Matrizes Transformadas=====

[ H_dqdE ] = deriv_parcial_dQdE(fi,teta,psi);

% Componentes da Covariancia (P) dos ângulos de Euler e dos
  bias:

SigAtt1=0.5*rad;
SigAtt2=0.5*rad;
SigAtt3=0.5*rad;
SigAttx=SigAtt1;
SigAtty=SigAtt2;
SigAttz=SigAtt3;

SigBiasx=1.0*rad/3600;
SigBiasy=1.0*rad/3600;
SigBiasz=1.0*rad/3600;

%Transformação da matriz P em ângulos de Euler (Pfi) para
  quaternions (Pq):

Pfi=zeros(3,3);
Pfi(1,1)=(SigAtt1)^2;

```



```
Pfi(2,2)=(SigAtt2)^2;
Pfi(3,3)=(SigAtt3)^2;

PQ=zeros(4,4);
PQ=H_dqdE*Pfi*H_dqdE';

P=zeros(7,7);
P(1,1)=PQ(1,1);
P(2,2)=PQ(2,2);
P(3,3)=PQ(3,3);
P(4,4)=PQ(4,4);

P(5,5)=(SigBiasx)^2;
P(6,6)=(SigBiasy)^2;
P(7,7)=(SigBiasz)^2;

% Matriz P reduzida:

P0=zeros(6,6);

MS=matriz_S( q );
P0=MS'*P*MS;

%Desvio-padrão do ruído do processo (Q):

% SigRoll=2.5e-04*rad; % Simulacao 1
% SigPitch=2.5e-04*rad;
% SigYaw=2.5e-04*rad;

SigRoll=0.015*rad; % Simulacao 2
SigPitch=0.015*rad;
SigYaw=0.015*rad;

% SigRoll=0.03*rad; % Real
% SigPitch=0.03*rad;
% SigYaw=0.03*rad;

SigDGyrox=0.05*rad/3600; % Desvio-padrão da deriva do gyro
SigDGyroy=0.05*rad/3600;
```

```

SigDGyroz=0.05*rad/3600;

Qfi=zeros(3,3);
Qfi(1,1)=(SigRoll)^2;
Qfi(2,2)=(SigPitch)^2;
Qfi(3,3)=(SigYaw)^2;

Qq=zeros(4,4);
Qq=H_dqdE*Qfi*H_dqdE';

Qaum=zeros(7,7);
Qaum(1,1)=Qq(1,1);
Qaum(2,2)=Qq(2,2);
Qaum(3,3)=Qq(3,3);
Qaum(4,4)=Qq(4,4);

Qaum(5,5)=(SigDGyrox)^2;
Qaum(6,6)=(SigDGyroy)^2;
Qaum(7,7)=(SigDGyroz)^2;

% Matriz Q reduzida em quatérnions:

Q=zeros(6,6);
Q=MS'*Qaum*MS;

% Inicialização da matriz de ruído das medidas (Rk):

% Desvio-padrão dos sensores (R):

SigDSS = 0.2*rad; % Simulacao
SigIRES = 0.02*rad;
% SigDSS = 0.4*rad; % Real
% SigIRES = 0.04*rad;

Rk = [SigDSS^2;SigDSS^2;SigIRES^2;SigIRES^2];
Rx = [(SigDSS^2), 0, 0, 0; 0, (SigDSS^2), 0, 0; 0, 0, (SigIRES^2), 0; 0, 0, 0, (SigIRES^2)];

% Matriz Gquat:

```

```

Gquat = [-(1/2)*eye(3) zeros(3,3);
          zeros(3,3) eye(3)];

% Dados utilizados no filtro H infinito

gamma = 5000;

Shinf = eye(6);
L = eye(6);
Sbar = L'*Shinf*L;

% Para o filtro H infinito de sunda ordem

eta = 0.5;

epsilon = 1;

lambda0 = [0.1; 0.1; 0.1; 0.1; 0.1; 0.1]*10;

PhiF = eye(6);
PhiFtil = eye(7);
PhiH = eye(4);

Pbar0 = zeros(6,6);
Pbar0(1,1) = (SigAtt1)^2;
Pbar0(2,2) = (SigAtt2)^2;
Pbar0(3,3) = (SigAtt3)^2;
Pbar0(4,4) = (SigBiasx)^2;
Pbar0(5,5) = (SigBiasy)^2;
Pbar0(6,6) = (SigBiasz)^2;

```

A.2 Filtro de Kalman Estendido

```

% Filtro de Kalman Estendido

% Objetivo: Estimar a atitude de satélites artificiais
           utilizando Quaternions.

% Comentários:

```

```
% vetor de estado inicial
% x0=[fi teta psi bp bq br]’
% (fi,teta,psi)=angulos de euler [rad]
% (bp,bq,br)=bias do giroscópio [rad/s]

% x_prop,x_est==> estado propagado e atualizado para instante
    k

% Sensores:

%...DSS_psi==>sensor solar digital [rad]
%...DSS_teta==> sensor solar digital [rad]
%...IRES_fi==> sensor de horizonte [deg]
%...IRES_teta==> sensor de horizonte [deg]

% Entrada: x0, P0,

% Saída: x_est, P_est

%.....
% Atualização: 09/01/2014
% Revisado por: William R. S. ; Hélio K. K.
%
% Ultima Atualização: 03/05/2022
% Revisado por: Lucas Moreira Martins - Unb - FGA
%.....

% Inicialização dos filtros de das constantes:

close all;

format long

tic

% Inicialização das Condições Iniciais:

condicoes_iniciais_quat;
```

```

% Taxa de amostragem do giro (0.5s), Gyros em ângulo [rad];

Dg = 1/0.5;

% Inicializa primeiro valor

q_est=x0(1:4);
q_estn=q_est/norm(q_est);

x_est = [q_estn;x0(5:7)];
P_est = P0;

[fi,teta,psi]=convertQ_Euler(x0(1:4)/norm(x0(1:4)));
result_x(:,1) = [fi;teta;psi;x0(5:7)];
result_x(5,1) = result_x(5,1) - x0(6);

sigma1(1) = sqrt(P_est(1,1))*deg;
sigma2(1) = sqrt(P_est(2,2))*deg;
sigma3(1) = sqrt(P_est(3,3))*deg;
sigma4(1) = sqrt(P_est(4,4))*deg;
sigma5(1) = sqrt(P_est(5,5))*deg;
sigma6(1) = sqrt(P_est(6,6))*deg;

for i = 1:m-1

Wgyro = T_M(i,11:13) '*Dg; % medida do giro

% Calculo do intervalo de tempo

DJM = Efm(i,1);
TS = Efm(i,2);
dt = TS - TS0;

% Velocidade angular orbital [rad/s]

rr = sqrt(Efm(i,3)^2 + Efm(i,4)^2 + Efm(i,5)^2); % vetor
      posição
w0 = sqrt(3.986005e14/rr^3); % movimento médio orbital (n^2=

```

```

    mi/r^3)

%% Filtro de Kalman Estendido

% Gyros information

Worbt=[2*(q_estn(1)*q_estn(2)+q_estn(3)*q_estn(4));
-q_estn(1)^2+q_estn(2)^2-q_estn(3)^2+q_estn(4)^2;
2*(q_estn(2)*q_estn(3)-q_estn(1)*q_estn(4))] * w0;

West = Wgyro + Worbt;

% propagação dos estados usando a matriz de transição

[Aq]= matriz_transicao_Q(West,dt);

q_prop=Aq*q_estn;
q_propn=q_prop/norm(q_prop);

b_prop=x_est(5:7);

x_prop=[q_propn;b_prop];

% Calculo dos versores orbitais (Vp, Vq, Vw)

[Vp, Vq, Vw] = orbPQW(i, Efm);

% Calculo do vetor posição do sol no sistema inercial
geocentrico:

S = SUN_50(DJM, TS);

% Calculo do vetor solar unitario no sistema orbital (xo,yo,
zo)

S0 = SUN_BF(S, Vp, Vq, Vw);

% Transformar quaternions em euler

```

```

[AT(1),AT(2),AT(3)]=convertQ_Euler(x_prop(1:4));

% Sensores (DSS & IRES):

[DSS,HPSIdss,HTETdss,HDSS] = CBERS_DSS(S0,AT(1:3));
[IRES,HFIires,HTETires,HIRES] = CBERS_IRES(AT(1),AT(2),0,0);

% Jacobiana H

Heuler=[HDSS(:,1:3);HIRES(:,1:3)];
[Hq]=deriv_parcial_dEdQ(q_propn);

Haux=Heuler*Hq;
H=[Haux zeros(4,3)];

% Cálculo do Resíduo do Filro de Kalman Estendido

z = [DSS IRES]'; % vetor de observação (4x1)

%.....

% Calculo das matrizes jacobianas F;

% Propagação da covariancia reduzida:

[Fi] = matriz_transicao_sistema(West,dt,q_estn,q_propn,Aq);
P_prop = Fi*P_est*Fi'+(Fi*Gquat*Q*Gquat'*Fi')*dt;

% Cálculo do Resíduo do Filro de Kalman Estendido

res = T_M(i,7:10)' - z(:);

% Linhas de comando do Filtro de Kalman Estendido

Sq = matriz_S( q_propn );
Htil = H*Sq;

Ktil = P_prop*Htil'*inv(Htil*P_prop*Htil' + Rx);
P_est = (eye(6) - Ktil*Htil)*P_prop;

```

```

K = Sq*Ktil;
x_est = x_prop + K*(res);

% Normaliza o quatérnion atualizado:

q_estn = x_est(1:4)/norm(x_est(1:4));

% Reconstitui o estado estimado em quaternions normalizados:

x_est = [q_estn;x_est(5:7)];

% Vetor de estado estimado com atitude em angulos de Euler:

[fi,teta,psi]=convertQ_Euler(q_estn);

% Reconstitui o estado estimado em angulos de Euler:

x_estE = [fi;teta;psi;x_est(5:7)];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

TS0 = TS;

% Desvio-Padrão da Covariância

sigma1(i+1) = sqrt(P_est(1,1))*deg; % [deg/s]
sigma2(i+1) = sqrt(P_est(2,2))*deg;
sigma3(i+1) = sqrt(P_est(3,3))*deg;
sigma4(i+1) = sqrt(P_est(4,4))*deg;
sigma5(i+1) = sqrt(P_est(5,5))*deg;
sigma6(i+1) = sqrt(P_est(6,6))*deg;

% Armazenamento de resultados por colunas

result_x(:,i+1) = x_estE;
result_x(5,i+1) = result_x(5,i+1)-w0;
res_prop(:,i+1) = res;

```



```

%.....Resíduos Pós Atualização.....

[DSSa ,HPSIdssa ,HTETdssa ,H_DSSa] = CBERS_DSS(S0 ,x_estE(1:3));

[IRESa ,HFIiresa ,HTETiresa ,H_IRESa] = CBERS_IRES(x_estE(1) ,
    x_estE(2) ,0,0);

z_atual = [DSSa IRESa]';

for ik = 1:4

res_atual(ik,i+1) = T_M(i,ik+6)'-z_atual(ik);

if (res_atual(ik,i+1)^2/Rk(ik)>1000)

res_atual(ik,i+1) = 0;

end

end

%.....

end

tempo_proc = toc;

x_estg = result_x*deg; % [deg deg/s]
inov = res_prop*deg; % resíduo pós-propagação (inovação)
residuo_atual = res_atual*deg; % resíduo pós atualização

```

A.3 Filtro H_∞ Estendido

```

% Filtro de Kalman Estendido

% Objetivo: Estimar a atitude de satélites artificiais
    utilizando Quaternions.

% Comentários:

```

```
% vetor de estado inicial
% x0=[fi teta psi bp bq br]'
% (fi,teta,psi)=angulos de euler [rad]
% (bp,bq,br)=bias do giroscópio [rad/s]

% x_prop,x_est==> estado propagado e atualizado para instante
    k

% Sensores:

%...DSS_psi==>sensor solar digital [rad]
%...DSS_teta==> sensor solar digital [rad]
%...IRES_fi==> sensor de horizonte [deg]
%...IRES_teta==> sensor de horizonte [deg]

% Entrada: x0, P0,

% Saída: x_est, P_est

%.....
% Atualização: 09/01/2014
% Revisado por: William R. S. ; Hélio K. K.
%
% Ultima Atualização: 03/05/2022
% Revisado por: Lucas Moreira Martins - Unb - FGA
%.....

% Inicialização dos filtros de das constantes:

close all;

format long

tic

% Inicialização das Condições Iniciais:

condicoes_iniciais_quat;
```

```

% Taxa de amostragem do giro (0.5s), Gyros em ângulo [rad];

Dg = 1/0.5;

%Inicializa primeiro valor

q_estH=x0(1:4);
q_estnH=q_estH/norm(q_estH);

x_estH = [q_estnH;x0(5:7)];
P_estH = P0;

[fi,teta,psi]=convertQ_Euler(x0(1:4)/norm(x0(1:4)));
result_xH(:,1) = [fi;teta;psi;x0(5:7)];
result_xH(5,1) = result_xH(5,1) - x0(6);

sigma1H(1) = sqrt(P_estH(1,1))*deg;
sigma2H(1) = sqrt(P_estH(2,2))*deg;
sigma3H(1) = sqrt(P_estH(3,3))*deg;
sigma4H(1) = sqrt(P_estH(4,4))*deg;
sigma5H(1) = sqrt(P_estH(5,5))*deg;
sigma6H(1) = sqrt(P_estH(6,6))*deg;

for i = 1:m-1

Wgyro = T_M(i,11:13) '*Dg; % medida do giro

% Calculo do intervalo de tempo

DJM = Efm(i,1);
TS = Efm(i,2);
dt = TS - TS0;

% Velocidade angular orbital [rad/s]

rr = sqrt(Efm(i,3)^2 + Efm(i,4)^2 + Efm(i,5)^2); % vetor
      posição
w0 = sqrt(3.986005e14/rr^3); % movimento médio orbital (n^2=

```

```

    mi/r^3)

%% Filtro de H inf Estendido

% Gyros information

Worbt=[2*(q_estnH(1)*q_estnH(2)+q_estnH(3)*q_estnH(4));
-q_estnH(1)^2+q_estnH(2)^2-q_estnH(3)^2+q_estnH(4)^2;
2*(q_estnH(2)*q_estnH(3)-q_estnH(1)*q_estnH(4))] * w0;

WestH = Wgyro + Worbt;

% propagação dos estados usando a matriz de transição

[Aq]= matriz_transicao_Q(WestH,dt);

q_propH=Aq*q_estnH;
q_propnH=q_propH/norm(q_propH);

b_propH=x_estH(5:7);

x_propH=[q_propnH;b_propH];

% Calculo dos versores orbitais (Vp, Vq, Vw)

[Vp, Vq, Vw] = orbPQW(i, Efm);

% Calculo do vetor posição do sol no sistema inercial
geocentrico:

S = SUN_50(DJM, TS);

% Calculo do vetor solar unitario no sistema orbital (xo,yo,
zo)

S0 = SUN_BF(S, Vp, Vq, Vw);

% Transformar quaternions em euler

```

```

[AT(1),AT(2),AT(3)]=convertQ_Euler(x_propH(1:4));

% Sensores (DSS & IRES):

[DSSH,HPSIdssH,HTETdssH,HDSSH] = CBERS_DSS(S0,AT(1:3));
[IRESH,HFIiresH,HTETiresH,HIRESH] = CBERS_IRES(AT(1),AT(2)
,0,0);

% Jacobiana H

Heuler=[HDSSH(:,1:3);HIRESH(:,1:3)];
[Hq]=deriv_parcial_dEdQ(q_propnH);

Haux=Heuler*Hq;
Hh=[Haux zeros(4,3)];

% Cálculo do Resíduo do Filtro H inf estendido

zH = [DSSH IRESH]'; % vetor de observação (4x1)

%.....

% Calculo das matrizes jacobianas F;

[FiH] = matriz_transicao_sistema(WestH,dt,q_estnH,q_propnH,Aq
);
P_propH = P_estH;

% Cálculo do Resíduo do Filro H Infinito Estendido

resH = T_M(i,7:10)' - zH(:);

Sq = matriz_S( q_propnH );
Htilh = Hh*Sq;

temp = eye(6) - gamma*Sbar*P_propH + Htilh'*inv(Rx)*Htilh*
P_propH;

if cond(temp) > 1e10

```

```

disp('ill conditioned H-infinity estimation problem')
return;
end

G = inv(temp);

KtilH = P_propH*G*Htilh'*inv(Rx);

KH = Sq*KtilH;

FitilH = Sq*FiH*Sq';

x_estH = x_propH + FitilH*KH*(resH);
P_estH = FiH*P_propH*G*FiH' + Q;

Gtest = eig(inv(P_propH) - gamma*Sbar + Htilh'*inv(Rx)*Htilh)
;

if min(real(Gtest)) <= 0
disp('H-infinity performance bound out of range')
return;
end

% Normaliza o quatérnion atualizado:

q_estnH = x_estH(1:4)/norm(x_estH(1:4));

% Reconstitui o estado estimado em quaternions normalizados:

x_estH = [q_estnH; x_estH(5:7)];

% Vetor de estado estimado com atitude em angulos de Euler:

[fi,teta,psi]=convertQ_Euler(q_estnH);

% Reconstitui o estado estimado em angulos de Euler:

x_estEH = [fi;teta;psi;x_estH(5:7)];

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
TS0 = TS;
```

```
% Desvio-Padrão da Covariância
```

```
sigma1H(i+1) = sqrt(P_estH(1,1))*deg; % [deg/s]
```

```
sigma2H(i+1) = sqrt(P_estH(2,2))*deg;
```

```
sigma3H(i+1) = sqrt(P_estH(3,3))*deg;
```

```
sigma4H(i+1) = sqrt(P_estH(4,4))*deg;
```

```
sigma5H(i+1) = sqrt(P_estH(5,5))*deg;
```

```
sigma6H(i+1) = sqrt(P_estH(6,6))*deg;
```

```
% Armazenamento de resultados por colunas
```

```
result_xH(:,i+1) = x_estEH;
```

```
result_xH(5,i+1) = result_xH(5,i+1)-w0;
```

```
res_propH(:,i+1) = resH;
```

```
%.....Resíduos Pós Atualização
```

```
.....
```

```
[DSSaH,HPSIdssaH,HTETdssaH,H_DSSaH] = CBERS_DSS(S0,x_estEH  
(1:3));
```

```
[IRESaH,HFIiresaH,HTETiresaH,H_IRESaH] = CBERS_IRES(x_estEH  
(1),x_estEH(2),0,0);
```

```
z_atualH = [DSSaH IRESaH]';
```

```
res_atualH(:,i+1) = T_M(i,7:10)'- z_atualH(:);
```

```
end
```

```
tempo_proc = toc;
```

```
x_estgH = result_xH*deg;
```

```
inovH = res_propH*deg;
```

```
residuo_atualH = res_atualH*deg;
```

A.4 Filtro H_∞ Estendido de Segunda Ordem

```

% Filtro H Infinito Estendido de Segunda Ordem.

% Objetivo: Estimar a atitude de satélites artificiais
    utilizando Quaternions.

% Comentários:

% vetor de estado inicial
% x0=[fi teta psi bp bq br]'
% (fi,teta,psi)=angulos de euler [rad]
% (bp,bq,br)=bias do giroscópio [rad/s]

% x_prop,x_est==> estado propagado e atualizado para instante
    k

% Sensores:

%...DSS_psi==>sensor solar digital [rad]
%...DSS_teta==> sensor solar digital [rad]
%...IRES_fi==> sensor de horizonte [deg]
%...IRES_teta==> sensor de horizonte [deg]

% Entrada: x0, P0,

% Saída: x_est, P_est

%.....
% Atualização: 09/01/2014
% Revisado por: William R. S. ; Hélio K. K.
%
% Ultima Atualização: 03/05/2022
% Revisado por: Lucas Moreira Martins - Unb - FGA
%.....

% Inicialização dos filtros de das constantes:

close all;

```



```

format long

tic

% Inicialização das Condições Iniciais:

condicoes_iniciais_quat;

% Taxa de amostragem do giro (0.5s), Gyros em ângulo [rad];

Dg = 1/0.5;

%Inicializa primeiro valor

q_estSOH=x0(1:4);
q_estnSOH=q_estSOH/norm(q_estSOH);

x_estSOH = [q_estnSOH;x0(5:7)];
P_estSOH = P0;

lambda = lambda0;
Pbar_estH = Pbar0;

[fi,teta,psi]=convertQ_Euler(x0(1:4)/norm(x0(1:4)));
result_xSOH(:,1) = [fi;teta;psi;x0(5:7)];
result_xSOH(5,1) = result_xSOH(5,1) - x0(6);

sigma1SOH(1) = sqrt(P_estSOH(1,1))*deg;
sigma2SOH(1) = sqrt(P_estSOH(2,2))*deg;
sigma3SOH(1) = sqrt(P_estSOH(3,3))*deg;
sigma4SOH(1) = sqrt(P_estSOH(4,4))*deg;
sigma5SOH(1) = sqrt(P_estSOH(5,5))*deg;
sigma6SOH(1) = sqrt(P_estSOH(6,6))*deg;

for i = 1:m-1

Wgyro = T_M(i,11:13) '*Dg; % medida do giro

% Calculo do intervalo de tempo

```

```

DJM = Efm(i,1);
TS = Efm(i,2);
dt = TS - TS0;

% Velocidade angular orbital [rad/s]

rr = sqrt(Efm(i,3)^2 + Efm(i,4)^2 + Efm(i,5)^2); %vetor
      posição
w0 = sqrt(3.986005e14/rr^3); % movimento médio orbital (n^2=
      mi/r^3)

%% Filtro de H inf Estendido de Segunda Ordem

% Gyros information

Worbt=[2*(q_estnSOH(1)*q_estnSOH(2)+q_estnSOH(3)*q_estnSOH(4)
      );
      -q_estnSOH(1)^2+q_estnSOH(2)^2-q_estnSOH(3)^2+q_estnSOH(4)^2;
      2*(q_estnSOH(2)*q_estnSOH(3)-q_estnSOH(1)*q_estnSOH(4))] * w0
      ;

WestSOH = Wgyro + Worbt;

% propagação dos estados usando a matriz de transição

[Aq]= matriz_transicao_Q(WestSOH,dt);

q_propSOH=Aq*q_estnSOH;
q_propnSOH=q_propSOH/norm(q_propSOH);

b_propSOH=x_estSOH(5:7);

x_propSOH=[q_propnSOH;b_propSOH];

% Calculo dos versores orbitais (Vp, Vq, Vw)

[Vp, Vq, Vw] = orbPQW(i, Efm);

```

```

% Calculo do vetor posição do sol no sistema inercial
geocentrico:

S = SUN_50(DJM, TS);

% Calculo do vetor solar unitario no sistema orbital (xo,yo,
zo)

SO = SUN_BF(S, Vp, Vq, Vw);

% Transformar quaternions em euler

[AT(1),AT(2),AT(3)]=convertQ_Euler(x_propSOH(1:4));

% Sensores (DSS & IRES):

[DSS_SOH,HPSIdss_SOH,HTETdss_SOH,HDSS_SOH] = CBERS_DSS(SO,AT
(1:3));
[IRES_SOH,HFIires_SOH,HTETires_SOH,HIRES_SOH] = CBERS_IRES(AT
(1),AT(2),0,0);

% Jacobiana H

Heuler=[HDSS_SOH(:,1:3);HIRES_SOH(:,1:3)];
[Hq]=deriv_parcial_dEdQ(q_propnSOH);

Haux=Heuler*Hq;
HSOH=[Haux zeros(4,3)];

% Cálculo do Resíduo do Filtro H inf estendido

zSOH = [DSS_SOH IRES_SOH]'; % vetor de observação (4x1)

%.....

% Calculo das matriz jacobiana f(x);

[FiSOH] = matriz_transicao_sistema(WestSOH,dt,q_estnSOH,
q_propnSOH,Aq);

```

```

P_propSOH = P_estSOH;

% Calculo das matrizes Hessiana de f(x);

[HessF1 ,HessF2 ,HessF3 ,HessF4 ,HessF5 ,HessF6] = HESSI_F(w0 ,
    WestSOH ,AT(1:3));

% Calculo das matrizes Hessiana de h(x);

[HessH1 ,HessH2 ,HessH3 ,HessH4] = HESSI_H(S0 ,AT(1:3));

% Resíduo

resSOH = T_M(i,7:10) '- zSOH(:)...
- (1/2)*(PhiH(:,1)*trace(HessH1*Pbar_estH)...
+ PhiH(:,2)*trace(HessH2*Pbar_estH) + PhiH(:,3)*trace(HessH3*
    Pbar_estH)...
+ PhiH(:,4)*trace(HessH4*Pbar_estH));

Sq = matriz_S( q_propnSOH );
HtilSOH = HSOH*Sq;

temp = eye(6) - gamma*Sbar*P_propSOH + HtilSOH'*inv(Rx)*
    HtilSOH*P_propSOH;

if cond(temp) > 1e10
disp('ill conditioned H-infinity estimation problem')
return;
end

G = inv(temp);

KtilSOH = P_propSOH*G*HtilSOH'*inv(Rx);

KSOH = Sq*KtilSOH;

FtilSOH = Sq*FiSOH*Sq';
HessF1til = Sq*HessF1*Sq';
HessF2til = Sq*HessF2*Sq';

```

```

HessF3til = Sq*HessF3*Sq';
HessF4til = Sq*HessF4*Sq';
HessF5til = Sq*HessF5*Sq';
HessF6til = Sq*HessF6*Sq';
Pbar_estHtil = Sq*Pbar_estH*Sq';

x_estSOH = x_propSOH + FtilSOH*KSOH*(resSOH)...
+ (1/2)*(PhiFtil(:,1)*trace(HessF1til*Pbar_estHtil)...
+ PhiFtil(:,2)*trace(HessF2til*Pbar_estHtil) + PhiFtil(:,3)*
    trace(HessF3til*Pbar_estHtil)...
+ PhiFtil(:,4)*trace(HessF4til*Pbar_estHtil) + PhiFtil(:,5)*
    trace(HessF5til*Pbar_estHtil)...
+ PhiFtil(:,6)*trace(HessF6til*Pbar_estHtil) + PhiFtil(:,7)*
    trace(HessF6til*Pbar_estHtil));

P_estSOH = FiSOH*P_propSOH*G*FiSOH' + Q;

Pbar_estH = eta*Pbar_estH + (1 - eta)*P_propSOH*lambda*lambda
    '*P_propSOH';
lambda = inv(FiSOH*FiSOH' + epsilon*eye(6))*FiSOH*(G*lambda -
    HtilSOH'*inv(Rx)*resSOH);

% Verificar se G é positivo definido

Gtest = eig(inv(P_propSOH) - gamma*Sbar + HtilSOH'*inv(Rx)*
    HtilSOH);

if min(real(Gtest)) <= 0
disp('H-infinity performance bound out of range')
return;
end

% Normaliza o quatérnion atualizado:

q_estnSOH = x_estSOH(1:4)/norm(x_estSOH(1:4));

% Reconstitui o estado estimado em quaternions normalizados:

x_estSOH = [q_estnSOH; x_estSOH(5:7)];

```

```

% Vetor de estado estimado com atitude em angulos de Euler:

[fi,teta,psi]=convertQ_Euler(q_estnSOH);

% Reconstitui o estado estimado em angulos de Euler:

x_estESOH = [fi;teta;psi;x_estSOH(5:7)];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

TSO = TS;

% Desvio-Padrão da Covariância

sigma1SOH(i+1) = sqrt(P_estSOH(1,1))*deg; % [deg/s]
sigma2SOH(i+1) = sqrt(P_estSOH(2,2))*deg;
sigma3SOH(i+1) = sqrt(P_estSOH(3,3))*deg;
sigma4SOH(i+1) = sqrt(P_estSOH(4,4))*deg;
sigma5SOH(i+1) = sqrt(P_estSOH(5,5))*deg;
sigma6SOH(i+1) = sqrt(P_estSOH(6,6))*deg;

% Armazenamento de resultados por colunas

result_xSOH(:,i+1) = x_estESOH;
result_xSOH(5,i+1) = result_xSOH(5,i+1) - w0;
res_propSOH(:,i+1) = resSOH;

%.....Resíduos Pós Atualização
.....

[DSSaH,HPSIdssaH,HTETdssaH,H_DSSaH] = CBERS_DSS(S0,x_estESOH
(1:3));

[IRESaH,HFIiresaH,HTETiresaH,H_IRESaH] = CBERS_IRES(x_estESOH
(1),x_estESOH(2),0,0);

z_atualSOH = [DSSaH IRESaH]';

```

```

[HessH1 ,HessH2 ,HessH3 ,HessH4] = HESSI_H(S0 ,x_estESOH(1:3));

res_atualSOH(:,i+1) = T_M(i,7:10) '- z_atualSOH(:)...
- (1/2)*(PhiH(:,1)*trace(HessH1*Pbar_estH)...
+ PhiH(:,2)*trace(HessH2*Pbar_estH) + PhiH(:,3)*trace(HessH3*
    Pbar_estH)...
+ PhiH(:,4)*trace(HessH4*Pbar_estH));

end

tempo_proc = toc;

x_estgSOH = result_xSOH*deg;
inovSOH = res_propSOH*deg;
residuo_atualSOH = res_atualSOH*deg;

% Formação da Matriz Time.

Time = [Efm(:,2)]/3600;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Plot
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure(1)
set(gcf, 'Position', [0, 0, 1080, 1080/4*2])

subplot(2,1,1)
plot(Time,T_M(:,7)*deg, '-k. ');
hold on
plot(Time,T_M(:,8)*deg, '-. ');
ylabel(['\u00c3ngulo (' char(176) ')']);
xlabel('Tempo (h)');
legend('DSS \alpha_\psi', 'DSS \alpha_\theta', 'location', '
    bestoutside')

subplot(2,1,2)
plot(Time,T_M(:,9)*deg, '-k. ');

```

```

hold on
plot(Time,T_M(:,10)*deg, '-');
ylabel(['\u00c3ngulo (' char(176) ')']);
xlabel('Tempo (h)');
legend('IRES \phi_H', 'IRES \theta_H', 'location', 'bestoutside'
)

print(gcf, 'fig_real_sensores', '-depsc');

figure(2)
set(gcf, 'Position', [0, 0, 1080, 1080/4*3])

subplot(3,1,1)
plot(Time,T_M(:,11)*Dg*deg*3600, '-k. ');
ylabel(['\u03c9_{x} (' char(176) '/h)']);
xlabel('Tempo (h)');

subplot(3,1,2)
plot(Time,T_M(:,12)*Dg*deg*3600, '-k. ');
ylabel(['\u03c9_{y} (' char(176) '/h)']);
xlabel('Tempo (h)');

subplot(3,1,3)
plot(Time,T_M(:,13)*Dg*deg*3600, '-k. ');
ylabel(['\u03c9_{z} (' char(176) '/h)']);
xlabel('Tempo (h)');

print(gcf, 'fig_real_omeg', '-depsc');

figure(3);
set(gcf, 'Position', [0, 0, 1080, 1080/4*3])

subplot(3,1,1)
plot(Time, x_estg(1,:), 'ks', Time, x_estgH(1,:), 'rd', Time,
x_estgSOH(1,:), 'bo', ...
Time, x_estg(1,:) + sigma1, 'k:', Time, x_estgH(1,:) + sigma1H
, 'r:', Time, x_estgSOH(1,:) + sigma1SOH, 'b--', ...

```



```

Time, x_estg(1,:) - sigma1, 'k:', Time, x_estgH(1,:) - sigma1H,
    'r:', Time, x_estgSOH(1,:) - sigma1SOH, 'b--');
legend('\phi (EKF)', '\phi (EH_{\infty}F)', '\phi (SOEH_{\infty}
    F)', '\phi \pm \sigma_{\phi} (EKF)', ...
'\phi \pm \sigma_{\phi} (EH_{\infty}F)', '\phi \pm \sigma_{\phi} (
    SOEH_{\infty}F)', 'location', 'bestoutside');
ylabel(['\phi (' char(176) ')']);
xlabel('Tempo (h)');

subplot(3,1,2)
plot(Time, x_estg(2,:), 'ks', Time, x_estgH(2,:), 'rd', Time,
    x_estgSOH(2,:), 'bo', ...
Time, x_estg(2,:) + sigma2, 'k:', Time, x_estgH(2,:) +
    sigma2H, 'r:', Time, x_estgSOH(2,:) + sigma2SOH, 'b--', ...
Time, x_estg(2,:) - sigma2, 'k:', Time, x_estgH(2,:) -
    sigma2H, 'r:', Time, x_estgSOH(2,:) - sigma2SOH, 'b--');
legend('\theta (EKF)', '\theta (EH_{\infty}F)', '\theta (SOEH_
    {\infty}F)', '\theta \pm \sigma_{\theta} (EKF)', ...
'\theta \pm \sigma_{\theta} (EH_{\infty}F)', '\theta \pm \
    sigma_{\theta} (SOEH_{\infty}F)', 'location', 'bestoutside')
ylabel(['\theta (' char(176) ')']);
xlabel('Tempo (h)');

subplot(3,1,3)
plot(Time, x_estg(3,:), 'ks', Time, x_estgH(3,:), 'rd', Time,
    x_estgSOH(3,:), 'bo', ...
Time, x_estg(3,:) + sigma3, 'k:', Time, x_estgH(3,:) + sigma3H
    , 'r:', Time, x_estgSOH(3,:) + sigma3SOH, 'b--', ...
Time, x_estg(3,:) - sigma3, 'k:', Time, x_estgH(3,:) - sigma3H
    , 'r:', Time, x_estgSOH(3,:) - sigma3SOH, 'b--');
legend('\psi (EKF)', '\psi (EH_{\infty}F)', '\psi (SOEH_{\infty}
    F)', '\psi \pm \sigma_{\psi} (EKF)', ...
'\psi \pm \sigma_{\psi} (EH_{\infty}F)', '\psi \pm \sigma_{\psi} (
    SOEH_{\infty}F)', 'location', 'bestoutside')
ylabel(['\psi (' char(176) ')']);
xlabel('Tempo (h)');

print(gcf, 'fig_real_estados', '-depsc');

```

```

figure(4);
set(gcf, 'Position', [0, 0, 1080, 1080/4*3])

subplot(3,1,1)
plot(Time, x_estg(4,:)*3600, 'ks', Time, x_estgH(4,:)*3600, 'rd',
      ',Time, x_estgSOH(4,:)*3600, 'bo');
ylabel(['\epsilon_{x} (' char(176) '/h)']);
xlabel('Tempo (h)');
legend('EKF', 'EH_{\infty}F', 'SOEH_{\infty}F', 'location', 'bestoutside');

subplot(3,1,2)
plot(Time, x_estg(5,:)*3600, 'ks', Time, x_estgH(5,:)*3600, 'rd',
      ',Time, x_estgSOH(5,:)*3600, 'bo');
ylabel(['\epsilon_{y} (' char(176) '/h)']);
xlabel('Tempo (h)');
legend('EKF', 'EH_{\infty}F', 'SOEH_{\infty}F', 'location', 'bestoutside');

subplot(3,1,3)
plot(Time, x_estg(6,:)*3600, 'ks', Time, x_estgH(6,:)*3600, 'rd',
      ',Time, x_estgSOH(6,:)*3600, 'bo');
ylabel(['\epsilon_{z} (' char(176) '/h)']);
xlabel('Tempo (h)');
legend('EKF', 'EH_{\infty}F', 'SOEH_{\infty}F', 'location', 'bestoutside');

print(gcf, 'fig_real_vies', '-depsc');

figure(5)
set(gcf, 'Position', [0, 0, 1440, 1080/4*2])

subplot(2,2,1);
plot(Time, inov(1,:), 'ks', Time, inovH(1,:), 'rd', Time, inovSOH(1,:), 'bo');
ylabel(['Resíduo pré (' char(176) ')']);
xlabel('Tempo (h)');

```

```

legend('DSS \alpha_\psi (EKF)', 'DSS \alpha_\psi (EH_{\infty}F)')', 'DSS \alpha_\psi (SOEH_{\infty}F)');
ylim([min([ min(inov(1,:)) min(inovH(1,:)) min(inovSOH(1,:))
            ]), ...
      max([ max(inov(1,:)) max(inovH(1,:)) max(inovSOH(1,:)) ])]])

subplot(2,2,2);
plot(Time, residuo_atual(1,:), 'ks', Time, residuo_atualH(1,:),
      'rd', Time, residuo_atualSOH(1,:), 'bo');
ylabel(['Resíduo pós (' char(176) ')']);
xlabel('Tempo (h)');
legend('DSS \alpha_\psi (EKF)', 'DSS \alpha_\psi (EH_{\infty}F)')', 'DSS \alpha_\psi (SOEH_{\infty}F)');
ylim([min([ min(inov(1,:)) min(inovH(1,:)) min(inovSOH(1,:))
            ]), ...
      max([ max(inov(1,:)) max(inovH(1,:)) max(inovSOH(1,:)) ])]])

subplot(2,2,3);
plot(Time, inov(2,:), 'ks', Time, inovH(2,:), 'rd', Time, inovSOH
      (2,:), 'bo');
ylabel(['Resíduo pré (' char(176) ')']);
xlabel('Tempo (h)');
legend('DSS \alpha_\theta (EKF)', 'DSS \alpha_\theta (EH_{\infty}F)')', 'DSS \alpha_\theta (SOEH_{\infty}F)');
ylim([min([ min(inov(2,:)) min(inovH(2,:)) min(inovSOH(2,:))
            ]), ...
      max([ max(inov(2,:)) max(inovH(2,:)) max(inovSOH(2,:)) ])]])

subplot(2,2,4);
plot(Time, residuo_atual(2,:), 'ks', Time, residuo_atualH(2,:),
      'rd', Time, residuo_atualSOH(2,:), 'bo');
ylabel(['Resíduo pós (' char(176) ')']);
xlabel('Tempo (h)');
legend('DSS \alpha_\theta (EKF)', 'DSS \alpha_\theta (EH_{\infty}F)')', 'DSS \alpha_\theta (SOEH_{\infty}F)');
ylim([min([ min(inov(2,:)) min(inovH(2,:)) min(inovSOH(2,:))
            ]), ...
      max([ max(inov(2,:)) max(inovH(2,:)) max(inovSOH(2,:)) ])]])

```

```

print(gcf, 'fig_real_res_dss', '-depsc');

figure(6)
set(gcf, 'Position', [0, 0, 1440, 1080/4*2])

subplot(2,2,1);
plot(Time, inov(3,:), 'ks', Time, inovH(3,:), 'rd', Time, inovSOH(3,:), 'bo');
ylabel(['Resíduo pré (' char(176) ')']);
xlabel('Tempo (h)');
legend('IRES \phi_H (EKF)', 'IRES \phi_H (EH_{\infty}F)', 'IRES \phi_H (SOEH_{\infty}F)');
ylim([min([ min(inov(3,:)) min(inovH(3,:)) min(inovSOH(3,:)) ]), ...
max([ max(inov(3,:)) max(inovH(3,:)) max(inovSOH(3,:)) ])]])

subplot(2,2,2);
plot(Time, residuo_atual(3,:), 'ks', Time, residuo_atualH(3,:), 'rd', Time, residuo_atualSOH(3,:), 'bo');
ylabel(['Resíduo pós (' char(176) ')']);
xlabel('Tempo (h)');
legend('IRES \phi_H (EKF)', 'IRES \phi_H (EH_{\infty}F)', 'IRES \phi_H (SOEH_{\infty}F)');
ylim([min([ min(inov(3,:)) min(inovH(3,:)) min(inovSOH(3,:)) ]), ...
max([ max(inov(3,:)) max(inovH(3,:)) max(inovSOH(3,:)) ])]])

subplot(2,2,3);
plot(Time, inov(4,:), 'ks', Time, inovH(4,:), 'rd', Time, inovSOH(4,:), 'bo');
ylabel(['Resíduo pré (' char(176) ')']);
xlabel('Tempo (h)');
legend('IRES \theta_H (EKF)', 'IRES \theta_H (EH_{\infty}F)', 'IRES \theta_H (SOEH_{\infty}F)');
ylim([min([ min(inov(4,:)) min(inovH(4,:)) min(inovSOH(4,:)) ]), ...
max([ max(inov(4,:)) max(inovH(4,:)) max(inovSOH(4,:)) ])]])

```

```

subplot(2,2,4);
plot(Time, residuo_atual(4,:), 'ks', Time, residuo_atualH(4,:),
      'rd', Time, residuo_atualSOH(4,:), 'bo');
ylabel(['Resíduo pós (' char(176) ')']);
xlabel('Tempo (h)');
legend('IRES \theta_H (EKF)', 'IRES \theta_H (EH_{\infty}F)',
       'IRES \theta_H (SOEH_{\infty}F)');
ylim([min([ min(inov(4,:)) min(inovH(4,:)) min(inovSOH(4,:))
           ]), ...
      max([ max(inov(4,:)) max(inovH(4,:)) max(inovSOH(4,:)) ])]])

print(gcf, 'fig_real_res_ires', '-depsc');

figure(7)
set(gcf, 'Position', [0, 0, 1440, 1080/4*3])

subplot(3,1,1);
plot(Time, Efm(:,3)/1000, '-k. ');
ylabel(['Posição X (km)']);
xlabel('Tempo (h)');

subplot(3,1,2);
plot(Time, Efm(:,4)/1000, '-k. ');
ylabel(['Posição Y (km)']);
xlabel('Tempo (h)');

subplot(3,1,3);
plot(Time, Efm(:,5)/1000, '-k. ');
ylabel(['Posição Z (km)']);
xlabel('Tempo (h)');

print(gcf, 'fig_real_efmpos', '-depsc');

figure(8)
set(gcf, 'Position', [0, 0, 1440, 1080/4*3])

subplot(3,1,1);

```

```
plot(Time, Efm(:,6)/1000, '-k.');
```

ylabel(['Velocidade X (km/s)']);

xlabel('Tempo (h)');


```
subplot(3,1,2);
```

plot(Time, Efm(:,7)/1000, '-k.');

ylabel(['Velocidade Y (km/s)']);

xlabel('Tempo (h)');


```
subplot(3,1,3);
```

plot(Time, Efm(:,8)/1000, '-k.');

ylabel(['Velocidade Z (km/s)']);

xlabel('Tempo (h)');


```
print(gcf, 'fig_real_efmvel', '-depsc');
```