



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa



Simulador de Voo de Investigação

Desenvolvimento e integração do módulo de controlo e de simulação da
dinâmica de uma aeronave

João Pedro Santos Jorge

Dissertação para obtenção do Grau de Mestre em
Engenharia Aeroespacial

Júri

Presidente: Prof. Fernando José Parracho Lau (DEM)
Orientação: Prof. José Raul Carreira Azinheira (DEM)
Prof. Agostinho Rui Alves da Fonseca (DEM)
Vogais: Prof. Filipe Szolnoky Ramos Pinto Cunha (DEM)

Setembro de 2008

“Aos meus pais, padrinhos e irmãs.”

Agradecimentos

A todos os que de alguma forma me ajudaram e encorajaram durante a elaboração deste trabalho, os meus reconhecidos agradecimentos.

A todos os professores que me ajudaram ao longo do presente trabalho e durante todo o curso, pela motivação e pelo conhecimento transmitido.

Um especial agradecimento aos meus orientadores Prof. José Azinheira e Prof. Agostinho Alves da Fonseca pela dedicação, encorajamento, disponibilidade e ajuda na resolução de problemas ao longo do trabalho. Pela tolerância e compreensão nos períodos de algum desânimo, e pelo apoio, profissionalismo e empenho que sempre manifestaram.

A toda a minha família o apoio dado e a coragem transmitida para conseguir ultrapassar os momentos mais difíceis.

Agradeço aos meus amigos, que me acompanharam ao longo do presente trabalho, o conhecimento transmitido assim como a sua amizade.

Agradeço ao meu amigo João Tavares Carreiro o esforço e a dedicação para me ajudar a encontrar as soluções para certos problemas.

A minha irmã, Ana Jorge, por ser a melhor irmã do Mundo, que me deu sempre ânimo e coragem para não desistir e concluir este trabalho.

Resumo

O Simulador de Voo de Investigação (SVI) é um projecto do Instituto Superior Técnico (IST), de características modulares e flexíveis, o qual está a ser desenvolvido por um grupo de alunos de Engenharia Aeroespacial. Esta dissertação tem como objectivo a implementação e integração do módulo de controlo e de simulação da dinâmica de uma aeronave.

A realização e desenvolvimento deste módulo tem como base a dinâmica de uma aeronave pré-definida, Navion, de fabrico norte-americano e inspirada no P-51 Mustang. Com base neste modelo implementaram-se pilotos automáticos de velocidade, altitude e rumo, o que torna possível um sistema controlável e autónomo, através de um controlo óptimo, denominado LQR. Todo o software é realizado através do programa MATLAB, sendo de destacar a ferramenta Simulink para criação de diagramas de blocos. O trabalho dispõe de um controlo de aterragem automática, com o qual é possível testar a operacionalidade do modelo.

Para além deste módulo, desenvolveu-se um módulo de criação de imagem. Este permite ao piloto a visualização de um ambiente de voo real, durante as simulações. O seu desenvolvimento foi realizado com base no FlightGear, dado ser um programa *open source*, permitindo assim modificações para melhoramento do cenário a simular.

A execução do módulo da dinâmica torna possível a interligação e interacção entre todos os módulos já desenvolvidos. Desta forma é possível estabelecer uma perspectiva final do simulador, atingindo-se o objectivo inicial do projecto.

O simulador encontra-se operacional, sendo já possível a sua utilização por outros investigadores. No entanto, sendo esta uma área de estudo em constante desenvolvimento, esperamos que o SVI continue a ser objecto de estudo e investigação num futuro próximo.

Palavras-chave: Simulador de Voo de Investigação; Dinâmica; Pilotos Automáticos; Regulador Quadrático Linear; FlightGear.

Abstract

This essay is part of a joint project for the initial development of a modular flexible Research Flight Simulator (SVI – *Simulador de Voo de Investigação*). The main goal of the work is the development and integration of the control and aircraft dynamics simulation module.

The production of the module was based on the dynamics of a predefined aircraft – the North American-made Navion inspired by the P-51 Mustang. On the basis of this model speed, altitude and heading automatic pilots were implemented, making it possible to create a controllable autonomous system through an optimal control named LQR. All the software was developed using the MATLAB program, the Simulink tool for block-diagram creation being a noteworthy element. The project was enriched with an automatic landing control device, with which it was possible to test the operational ability of the model.

An image creation module was also developed. This allows the pilot to visualise a real flight environment during simulation. It was developed on the basis of FlightGear, an open-source program that allows modifications designed to improve the scenario to be simulated.

Taking into account the fact that this model was the culmination of a whole project, it allowed interconnection and interaction between all the models already developed. Thus was it possible to establish a final, global perspective of the simulator and to attain the final goal of the SVI.

The simulator is fully operational and its use by other researchers is already possible. However, as this field of study is in constant development, we hope the SVI will be the subject of further study and research in the near future.

Keywords: Research Flight Simulator; Dynamics; Automatic Pilots; Linear Quadratic Regulator; FlightGear.

Índice

Agradecimentos	iii
Resumo	iv
Abstract	v
Lista de Figuras	viii
Lista de Tabelas	x
Acrónimos	xi
Lista de Símbolos	xii
1. Introdução	1
1.1. Enquadramento	1
1.2. Objectivo do trabalho	2
1.3. Estrutura da tese	2
2. Simulação de Voo	4
2.1. Estado da arte	4
2.2. Tipos de simuladores	6
2.3. Vantagens dos simuladores	7
2.4. Estrutura de um Simulador de Voo	8
2.5. Comunicação entre os módulos	10
3. Dinâmica da Aeronave	11
3.1. Referenciais	11
3.2. Equações do Movimento	13
3.3. Equações do modelo	14
3.3.1. Aerodinâmica	15
3.3.2. Propulsão	18
3.3.3. Atmosfera	20
3.3.4. Terra	22
3.3.5. Inércia	24
3.3.6. Trem de Aterragem	24
3.3.7. Aceleração Total	25
3.3.8. Momento Total	26
3.3.9. Equações do Movimento	26
3.3.10. Sensores e Actuadores	29
3.4. Controlo de Voo	31
3.5. Pilotos Automáticos	34
3.5.1. Controlo Longitudinal	34
3.5.2. Controlo Lateral	35
3.5.3. Aterragem Automática	36
4. Integração da Dinâmica no SVI	37
4.1. Modo de Comunicação	37

4.1.1.	Módulo do movimento	38
4.1.2.	Módulo de simulação de imagem.....	40
4.1.3.	Módulo de interface com o piloto.....	40
4.2.	Módulo de Simulação de Imagem.....	41
4.3.	Comandos	42
5.	Avaliação Operacional do módulo.....	44
5.1.	Testes dos Pilotos Automáticos	44
5.1.1	Seguimento de Altitude	45
5.1.2	Seguimento de Velocidade.....	48
5.1.3	Seguimento de Velocidade e de Altitude	50
5.1.4	Controlo de Rumo	51
5.1.5	Aterragem Automática.....	52
5.2.	Testes dos modos de Comunicação	54
5.3.	Perspectiva Final do Simulador.....	56
6.	Conclusões	58
6.1.	Conclusões.....	58
6.2.	Sugestões para trabalho futuro	59
7.	Bibliografia	60
Anexos.....		63
Anexo A.....		64
A.1.	Coeficientes de Inércia	64
A.2.	Produto Externo.....	65
Anexo B.....		66
B.1.	Eixos de Referência e Coordenadas	66
B.1.1.	Sistema Roll Pitch Yaw	66
B.1.2.	Sistema North East Down	67
B.1.3.	Sistema WGS84	67
B.1.4.	Conversão de referenciais ECEF para NED	68
Anexo C.....		70
C.1.	Ficheiros utilizados	70
Anexo D.....		75
D.1.	Aeronave Simulada	75
Anexo E.....		79
E.1.	Comando utilizado e parametrização.....	79
Anexo F.....		80
F.1.	Manual de utilização do SVI.....	80
Anexo G		81
G.1.	Parâmetros utilizados pelo bloco <i>Send</i>	81
G.2.	Parâmetros utilizados pelo bloco <i>Receive</i>	81
G.3.	Parâmetros utilizados pelo bloco <i>To Instrument</i>	82

Lista de Figuras

Figura 1.1 – Plataforma de movimento do SVI.....	1
Figura 2.1 – Instalações do VMS.....	4
Figura 2.2 – Interior de uma I-CAB.....	4
Figura 2.3 – SIMONA.....	5
Figura 2.4 – Interior da Cabine do SIMONA.....	5
Figura 2.5 – Estrutura modular do SVI.....	9
Figura 3.1 – Sistemas de coordenadas.....	11
Figura 3.2 – <i>Ângulos de Euler</i>	13
Figura 3.3 – Modelo da Dinâmica da Aeronave.....	15
Figura 3.4 – Bloco da Aerodinâmica.....	15
Figura 3.5 – Bloco da Aerodinâmica interior.....	16
Figura 3.6 – Bloco do Sistema de Propulsão.....	19
Figura 3.7 – Bloco do Sistema de Propulsão interior.....	19
Figura 3.8 – Bloco da Atmosfera.....	20
Figura 3.9 – Bloco da Atmosfera interior.....	21
Figura 3.10 – Bloco da Terra.....	22
Figura 3.11 – Bloco da Terra interior.....	22
Figura 3.12 – Bloco da Inércia da Aeronave.....	24
Figura 3.13 – Bloco do Trem de Aterragem da Aeronave.....	24
Figura 3.14 – Bloco da Aceleração Total da Aeronave.....	25
Figura 3.15 – Bloco do Momento Total da Aeronave.....	26
Figura 3.16 – Bloco das Equações do Movimento.....	26
Figura 3.17 – Bloco das Equações do Movimento interior.....	27
Figura 3.18 – Bloco dos Actuadores.....	29
Figura 3.19 – Actuadores para as superfícies de controlo.....	30
Figura 3.20 – Aeronave como sistema.....	31
Figura 3.21 – Sistema de controlo genérico para os pilotos automáticos.....	33
Figura 3.22 – Piloto automático de controlo longitudinal.....	34
Figura 3.23 – Piloto automático de controlo lateral.....	35
Figura 4.1 – Esquema do sistema de interligação entre os computadores do SVI.....	37
Figura 4.2 – Módulo Central do SVI.....	38
Figura 4.3 – Módulo de envio de dados para o módulo do Movimento.....	39
Figura 4.4 – Módulo de recepção de dados para o módulo do Movimento.....	39
Figura 4.5 – Módulo de interface com o piloto.....	40
Figura 4.6 – Esquema responsável pelo envio de dados para o <i>FlightGear</i>	41
Figura 4.7 – Superfícies de controlo de uma aeronave.....	42
Figura 4.8 – Blocos de parametrização do <i>Joystick</i>	43

Figura 5.1 – Estabilização de altitude.....	45
Figura 5.2 – Estabilização de velocidade.....	45
Figura 5.3 – Simulação de voo a diferentes altitudes.....	45
Figura 5.4 – Variação da velocidade ao longo do voo a diferentes altitudes.....	46
Figura 5.5 – Deflexão das superfícies de controlo ao longo do voo a diferentes altitudes.....	46
Figura 5.6 – Variação do <i>Throttle</i> ao longo do voo a diferentes altitudes.....	47
Figura 5.7 – Simulação de voo a diferentes velocidades.....	48
Figura 5.8 – Variação da altitude ao longo do voo a diferentes velocidades.....	49
Figura 5.9 – Deflexão das superfícies de controlo ao longo do voo a diferentes velocidades.....	49
Figura 5.10 – Variação do <i>Throttle</i> ao longo do voo a diferentes velocidades.....	50
Figura 5.11 – Simulação de voo a diferentes altitudes.....	50
Figura 5.12 – Simulação de voo a diferentes velocidades.....	50
Figura 5.13 – Deflexão das superfícies de controlo.....	50
Figura 5.14 – Variação do <i>Throttle</i>	50
Figura 5.15 – Simulação de voo a diferentes rumos.....	51
Figura 5.16 – Deflexão das superfícies de controlo ao longo do voo a diferentes rumos.....	51
Figura 5.17 – Aterragem automática.....	52
Figura 5.18 – Variação da velocidade ao longo da aterragem automática.....	53
Figura 5.19 – Variação da deflexão dos amortecedores do trem.....	53
Figura 5.20 – <i>Hub</i> utilizado no SVI.....	54
Figura 5.21 – Sistema EFIS do SVI.....	55
Figura 5.22 – Modo simulação de imagem.....	55
Figura 5.23 – Interface gráfica para a escolha dos pilotos automáticos e comando.....	56
Figura 5.24 – Perspectiva do piloto no SVI.....	57
Figura 5.25 – Módulo final do SVI em simulação.....	57
Figura B.1. – Referencial da aeronave RPY.....	66
Figura B.2. – Ângulos de ataque e derrapagem.....	66
Figura B.3. – Referencial NED.....	67
Figura B.4. – Modelo elipsóide WGS84.....	68
Figura E.1. – <i>Joystick</i> utilizado.....	79
Figura G.1 – Parâmetros do bloco <i>Send</i>	81
Figura G.2 – Parâmetros do bloco <i>Receive</i>	81
Figura G.3 – Parâmetros do bloco <i>To Instrument</i>	82

Lista de Tabelas

Tabela 4.1 – Intervalo de variação das superfícies de controlo e <i>Throttle</i>	43
Tabela 5.1 – Parâmetros iniciais de voo da aeronave.....	44
Tabela B.1 – Parâmetros WGS84.....	67
Tabela E.1 – Funcionalidades dos botões do <i>Joystick</i>	79

Acrónimos

SVI	Simulador de Voo de Investigação
SCC	System Control Computer
UDP	User Datagram Protocol
MDA	Motion Drive Algorithm
DOF	Degree of Freedom
LQR	Regulador Quadrático Linear
EFIS	Electronic Flight Instrument System
ILS	Instrument Landing System
NED	Referencial North East Down
MSL	Mean Sea Level
IP	Internet Protocol
USB	Universal Serial Bus
VMS	Vertical Motion Simulator
PIO	Pilot Induced Oscillation
ZFT	Zero Flight Time
JAA	Joint Aviation Authority
CG	Centro de Gravidade

Lista de Símbolos

δ_E	Deflexão do leme de profundidade da aeronave
δ_A	Deflexão dos ailerons da aeronave
δ_R	Deflexão do leme de direção da aeronave
δ_T	Comando propulsivo da aeronave
θ	Ângulo de picada da aeronave
ϕ	Ângulo de rolamento da aeronave
ψ	Ângulo de rumo da aeronave
Lat	Latitude Geodésica da aeronave
Lon	Longitude Geodésica da aeronave
Alt	Altitude referente ao nível médio das águas do mar MSL da aeronave
x_e	Componente Norte da posição da aeronave
y_e	Componente Este da posição da aeronave
H	Altitude da aeronave MSL
F	Força externa aplicada no centro de gravidade
M	Momento externo aplicado em torno do centro de gravidade
V	Velocidade no ar do centro de gravidade
h	Momento angular
I	Tensor de inércia
ω	Velocidade angular
P	Velocidade angular de rolamento
Q	Velocidade angular de picada
R	Velocidade angular de guinada
L	Momento de rolamento
M	Momento de picada
N	Momento de guinada
V_a	Velocidade de ar no referencial da aeronave
D_a	Força de resistência
Y_a	Força lateral
L_a	Força de sustentação
q	Pressão dinâmica
ρ	Densidade do ar
α	Ângulo de ataque
β	Ângulo de derrapagem
M	Número de Mach
a	Velocidade do som
C_L	Coefficiente de sustentação
C_D	Coefficiente de resistência

C_Y	Coeficiente de força lateral
C_m	Coeficiente de momento de picada
C_l	Coeficiente de momento de rolamento
C_n	Coeficiente de momento de guinada
TAS	Velocidade verdadeira (<i>True Airspeed</i>) da aeronave
OAT	Temperatura exterior (<i>Outside Air Temperature</i>)
p_{stat}	Pressão estática
MAP	Pressão interna do colector de admissão do motor (<i>Manifold Absolute Pressure</i>)
Ω_{eng}	Velocidade do motor
m_{fuel}	Quantidade de combustível da aeronave
J	Razão de avanço
C_T	<i>Coefficient of Thrust</i> do motor
C_P	<i>Coefficient of Power</i> do motor
F_{prop}	Força gerada pela hélice do motor
M_{prop}	Momento criado pela hélice do motor

1. Introdução

1.1. Enquadramento

O Simulador de Voo de Investigação (SVI) é um projecto do Instituto Superior Técnico (IST) sugerido como Tese de Mestrado Integrado a um grupo de alunos de Engenharia Aeroespacial. O simulador está a ser desenvolvido de raiz nas instalações do Pavilhão de Mecânica III do IST e o seu desenvolvimento assenta em quatro áreas distintas, também conhecidas como módulos do simulador.

Esta tese desenvolve uma dessas áreas, mais concretamente o módulo de controlo da dinâmica da aeronave. Actualmente, já se encontram desenvolvidos dois módulos, os quais foram objecto de estudo de duas dissertações de mestrado [9] e [10]. Estes módulos estão em estado operacional, de forma a serem integrados no módulo que é apresentado ao longo desta dissertação.

Através de um algoritmo de sugestão de movimento implementado e com um protocolo de comunicação específico, é já possível reproduzir diversos movimentos na plataforma do SVI, a qual se pode observar na figura. 1.1.



Figura 1.1 – Plataforma de movimento do SVI.

As duas áreas aqui desenvolvidas consistem na implementação do modelo da dinâmica da aeronave a simular e na geração do ambiente exterior ao cockpit.

A dinâmica da aeronave é gerada através das suas equações de movimento, tendo em conta fenómenos aerodinâmicos, propulsivos e condições atmosféricas a simular. O ambiente exterior é recriado utilizando um software *open source* para gerar ambientes gráficos 3D. Pretende-se que após todas as áreas estarem concluídas o simulador esteja operacional.

A presente tese tem como objectivo o desenvolvimento e integração do Módulo de Controlo e Simulação da Dinâmica da uma Aeronave no SVI.

1.2. Objectivo do trabalho

O SVI encontra-se numa fase inicial de desenvolvimento e está a ser estruturado de um modo modular e flexível, permitindo que futuramente sejam realizados melhoramentos e actualizações. Além da vertente de engenharia e investigação associadas a este simulador, também se pretende que este contribua para que alunos do Mestrado em Engenharia Aeroespacial do IST possam dar continuidade ao projecto, aplicando de um modo prático os conceitos e conhecimentos adquiridos ao longo do curso.

O principal objectivo deste projecto é simular a performance de uma aeronave, pelo que se torna necessário projectar a sua dinâmica. A dinâmica de uma aeronave é o bloco central de SVI, dado que é responsável pela recepção de informações do piloto e a sua transmissão para os restantes módulos, de forma a tornar possível a simulação de um voo real.

Tendo em conta o referido, esta tese tem como principal finalidade a implementação e finalização do SVI, estabelecendo a ligação entre os módulos anteriormente desenvolvidos nas teses [9] e [10].

Para além do desenvolvimento da dinâmica de voo é também construído o controlo da dinâmica. Para atingir este objectivo são implementados pilotos automáticos capazes de controlar a aeronave.

A realização deste trabalho permite a interacção entre os quatro módulos, o que torna possível atingir todos os objectivos do projecto SVI. A conclusão desta fase representa o ponto de partida para uma nova etapa do SVI, um projecto de utilidade para novos alunos e investigadores do IST.

Para o desenvolvimento de software utiliza-se o programa MATLAB [8] devido à sua interface gráfica intuitiva, à familiaridade existente com a sua utilização, à sua potencialidade de trabalho e adequação ao problema.

1.3. Estrutura da tese

A presente dissertação está estruturada em seis capítulos.

O primeiro capítulo – Introdução – tem como finalidade contextualizar e apresentar o estudo a desenvolver. Para além disso, enunciam-se os objectivos, bem como a importância do estudo.

O segundo capítulo – Simulação de Voo – trata da simulação de voo em geral. Apresenta um conjunto de informação acerca da simulação de voo, começando por algumas considerações sobre o “estado da arte” da simulação, no âmbito da investigação. Nas secções seguintes mostram-se os tipos de simuladores que podem ser distinguidos de acordo com as suas especificidades e as vantagens que apresentam. Expõe-se a estrutura geral de um simulador de voo e, em particular, a estrutura adoptada para o SVI, onde são introduzidos os vários módulos independentes a serem desenvolvidos. Na última secção mostram-se os protocolos e as metodologias utilizadas na comunicação entre estes mesmos módulos.

O terceiro capítulo – Dinâmica da Aeronave – aborda o desenvolvimento do Módulo da Dinâmica de uma aeronave no SVI. São descritos e explicados os referenciais existentes para posteriormente, na secção seguinte, se desenvolverem as equações da dinâmica de uma aeronave de um ponto de vista geral. São descritos todos os blocos inerentes ao modelo da dinâmica da aeronave simulada assim como todas as equações que estão implícitas a cada bloco. Realiza-se a descrição do conceito de controlo de voo, fazendo referência ao tipo de controlo utilizado para a implementação dos pilotos automáticos. Concluimos este capítulo com a implementação dos diferentes pilotos automáticos, realizando a análise de cada um deles.

O quarto capítulo – Integração da Dinâmica do SVI – descreve como o módulo da dinâmica é integrado com os restantes módulos. É realizada a explicação do modo de comunicação entre os vários módulos, enunciando todos os procedimentos efectuados. Descreve-se a implementação do módulo de simulação de imagem. Finaliza-se com a explicação do modo como foi implementado os comandos para controlar e simular a aeronave.

No quinto capítulo – Avaliação Operacional do Módulo – são realizados testes para testar a validade do módulo implementado. Inicia-se com o estudo sobre os vários pilotos automáticos implementados, fazendo referência aos principais parâmetros de voo. Em seguida são realizados testes de comunicação entre os módulos, avaliando-se a recepção dos dados em cada módulo. Terminamos com a apresentação do SVI, destacando o resultado final do ambiente implementado.

O sexto capítulo – Conclusões – são expostas as conclusões e apontadas algumas sugestões para futuros trabalhos, na sequência deste estudo.

2. Simulação de Voo

A simulação de voo, mais especificamente a simulação de pilotagem, é hoje largamente utilizada na área da aeronáutica. No entanto, o facto de a criação de um ambiente virtual fidedigno não ser ainda completamente compreendido faz com que a simulação continue a ser frequentemente considerada como uma arte e não como uma ciência [13] e [16].

2.1. Estado da arte

A referência actual no domínio da investigação e simulação de voo é dada por dois simuladores. O primeiro deles é o *Vertical Motion Simulator* (VMS) da NASA [19], localizado no centro de investigação Ames, na Califórnia.

O VMS, um dos maiores e mais avançados simuladores do mundo, tornou-se operacional por volta de 1970. É possível, através das figuras 2.1 e 2.2, ter uma noção das suas enormes dimensões.

Possuidor de inúmeras particularidades, destaca-se como uma das suas principais características a capacidade de movimento em seis graus de liberdade, com cursos de dezoito metros em termos verticais e de doze metros em termos horizontais. Esta liberdade de movimento permite o estudo de casos mais complexos, tais como helicópteros, aeronaves *Vertical / Short Take-off and Landing* (V/STOL) e ainda como uma ferramenta útil no estudo das fases mais críticas de voo, aterragens e aproximações.

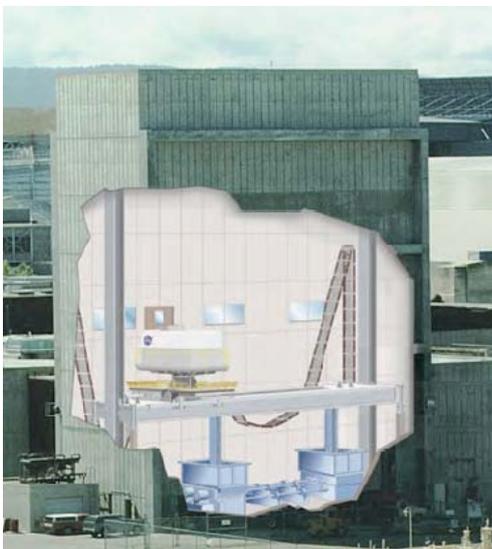


Figura 2.1 – Instalações do VMS.



Figura 2.2 – Interior de uma I-CAB.

Outra particularidade a realçar neste simulador são as cabines comutáveis utilizadas, as I-CABS [21]. Para promover a rapidez na transição de simulações, existem cinco cabines, as quais conseguem simular diversos interiores, de diferentes aeronaves. A construção das cabines é padronizada, cumprindo requisitos a nível das interfaces eléctricas, mecânicas e hidráulicas. Os seus

pesos são variados, tendo a mais pesada cerca de setenta toneladas. No interior destas cabines (figura 2.2), tem-se a visualização gráfica do exterior e a simulação de qualquer tipo de instrumentação, recorrendo a ecrãs de instrumentos presentes na aviação moderna.

O segundo simulador, conhecido por SIMONA, encontra-se na *Delft University of Technology*, tendo começado a ser desenvolvido por volta de 1992 [20]. Esta sigla consiste na abreviatura das três principais áreas que este simulador pretende investigar, a simulação, o movimento e a navegação.

SIMONA, apresentado na figura 2.3, é essencialmente constituído por seis actuadores servo hidráulicos responsáveis pelo movimento, com seis graus de liberdade. É constituído por uma cabine com peso na ordem das quatro toneladas, consideravelmente inferior quando comparado com os simuladores convencionais, de doze a quinze toneladas, referidos em [22] e [23]. Esta diminuição de peso deve-se aos materiais utilizados na sua construção, TWARON e carbono. Este tipo de materiais confere ao simulador um baixo centro de gravidade, obtendo diversas vantagens no seu movimento.

No interior da cabine, como se pode observar na figura 2.4, o ambiente exterior é simulado recorrendo a projectores e a instrumentação é visualizada pelo piloto usando vários ecrãs, que permitem simular qualquer tipo de instrumento.



Figura 2.3 – SIMONA.



Figura 2.4 – Interior da Cabine do SIMONA.

Em termos de referência para a construção do SVI, o SIMONA constitui uma melhor referência comparativamente ao VMS, dado que foi desenvolvido num ambiente universitário, onde vários alunos participaram no seu desenvolvimento. Para além deste facto, destaca-se o seu desenvolvimento de um modo altamente modular, tal como se pretende fazer no SVI.

2.2. Tipos de simuladores

Segundo Advani [24], podem ser reconhecidas diversas áreas de aplicação de simuladores:

Simuladores de engenharia – Utilizados para avaliação das características de veículos. No desenvolvimento de uma nova aeronave, o simulador é utilizado durante todo o processo de design, devido à sua capacidade de prever possíveis problemas e possibilitar uma passagem mais segura para a fase de voo. O recurso à simulação possibilita a redução do número de testes de voo e, conseqüentemente a redução de custos e celeridade na obtenção de certificações [16]. A maioria dos simuladores de engenharia não têm plataforma de movimento, são de base fixa, uma vez que as sensações de movimento nem sempre são importantes e necessárias.

No entanto, o sistema de movimento é essencial para prever qualidades de manobrabilidade, tais como por exemplo o *Pilot Induced Oscillation* (PIO), devido à dinâmica da aeronave, de forma adequada [25]. Os modos de baixa frequência de uma aeronave podem ser também apropriadamente amortecidos com a utilização de um sistema de movimento, enquanto as vibrações de alta frequência amplificadas pelos comandos do piloto (frequentemente ligadas à aeroelasticidade da aeronave) podem, normalmente, ser detectadas apenas se o algoritmo de produção de movimento não alterar esses mesmos modos.

Simuladores de treino – Usados para treino de pilotos tanto em termos de procedimentos (gestão de voo) como em tarefas básicas de destreza (controlo manual). Em aplicações militares o treino de procedimentos é normalmente realizado em simuladores de base fixa. O treino de destreza é maioritariamente realizado em voo, uma vez que a discrepância entre o simulador e a aeronave é considerável em manobras extremas. Na aviação comercial, os simuladores de base móvel são economicamente mais rentáveis comparativamente às horas em voo efectivo, pelo que muitas horas são realizadas em treino de simulador. Revalidações de competência e treino de recorrência, isto é, exames e transferências para outro tipo de aeronave, são permitidos pelas autoridades reguladoras sem ser necessário voo efectivo, para pilotos com mais de 500 horas de experiência em aeronaves da mesma classe ou 1500 (Europa) a 2500 horas (USA) no total [13]. Há uma tendência para a redução do número de horas exigidas [26], o que permite o chamado treino *Zero Flight Time* (ZFT), mas o exacto conhecimento da fidelidade requerida para o simulador é ainda uma incógnita.

Em termos genéricos, os simuladores comerciais de treino de tripulações na Europa são qualificados pela *Joint Aviation Authority* (JAA) em nível A, B, C ou D, de acordo com as suas especificações técnicas [27].

Simuladores de investigação – Utilizados no estudo fundamental da interacção piloto/veículo e na investigação da percepção humana. Este tipo de simuladores, na maioria das vezes, requer um elevado nível de desempenho do sistema de movimento, uma vez que as sensações de movimento são essenciais na tarefa de controlo manual [28].

Para além desta aplicação científica fundamental, este tipo de simulador de voo tem também demonstrado valor no que toca à investigação de acidentes [16]. Simuladores de investigação têm

sido utilizados para estudar as circunstâncias em que ocorreram incidentes devidos a turbulência em ar limpo (*clear air turbulence*) e outros tipos de turbulência severa tal como rajadas, e têm fornecido mais estratégias benéficas de controlo e melhorias em termos de procedimentos operacionais. Neste tipo de estudos, o sistema de movimento tem de ser capaz, por exemplo, de produzir vibrações que impossibilitem o piloto de ler os instrumentos.

2.3. Vantagens dos simuladores

Tendo em conta as diversas áreas de aplicação e as diferentes qualidades específicas do simulador de voo, várias são as vantagens que podem ser mencionadas quando comparado com o voo real [16].

Custo/benefício – Apesar do investimento inicial e tendo em conta o custo de operação, um simulador de voo tem custos de ordem de grandeza muito menor do que a maioria das aeronaves reais. Para além disso, treinar e efectuar estudos num simulador é, em alguns casos, mais vantajoso.

Requisitos de segurança – O simulador permite realizar treinos de condições muito diversas e perigosas, possíveis em voo real, de uma forma segura. Permitem confrontar os pilotos com os melhores procedimentos operacionais para circunstâncias anormais, aumentando a segurança dos voos. Pilotos inexperientes podem adquirir um conjunto de competências motoras e procedimentos necessários, evitando o risco de o fazerem no ar. Diversos e potenciais erros de desenho podem ser detectados em antecipação. Torna possível a análise de acidentes em detalhe. Um último aspecto de segurança indirecta é o bom compromisso custo/benefício, que permite um muito completo e ainda assim praticável programa de treino de tripulações.

Considerações ambientais – Tendo em conta as emissões de poluentes resultantes da queima de combustível e a quantidade de querosene consumida, um simulador de voo tem um impacto ambiental muito menor, tornando-se ambientalmente mais viável. Indirectamente, operações de voo menos nocivas para o ambiente podem ser treinadas num simulador.

Dado que o objectivo não é duplicar, mas sim simular um ambiente que crie os melhores e mais efectivos resultados em treino e testes [13] e [29], o simulador de voo deve ser também considerado como uma ferramenta de treino e teste por si só, em vez de ser apenas visto como um substituto de uma aeronave [26]. Tendo em conta esta visão, o simulador é mais efectivo.

A possibilidade de dividir o treino em partes e cada uma delas poder ser tratada separadamente, numa abordagem passo-a-passo. Um simulador representa uma ferramenta objectiva de avaliar diferentes indivíduos perante condições semelhantes. É muito flexível quando se quer tratar de um teste específico, como por exemplo, treinar sobre determinadas condições atmosféricas ou num determinado aeroporto.

2.4. Estrutura de um Simulador de Voo

Num simulador de voo é necessária a criação de um ambiente complexo, cujo objectivo é criar a impressão de movimento e deslocação do veículo simulado, com especial ênfase no realismo conferido à tarefa ou investigação que se pretende fazer.

O ambiente do simulador pode ser dividido em vários subsistemas:

Interior – O ambiente estático, na perspectiva do piloto, deve ser semelhante ao que encontraria na cabine da aeronave simulada.

Sistema visual – O exterior, ou seja, a perspectiva de vista de janela, é simulado através de um sistema de geração de imagem, no caso de aeronaves civis. A imagem é geralmente projectada por um espelho convexo situado na parte da frente da cabine de simulação, por sistemas de projectores/telas ou por sistemas de ecrãs.

Instrumentos – Os instrumentos da cabine dão informação acerca do estado do veículo simulado e essa informação deve ser activamente actualizada, tal como acontece no voo real.

Sistema de movimento – As acelerações efectivas do sistema são criadas recorrendo a uma plataforma de movimento, provocando de forma limitada o movimento da cabine de simulação.

Carga de controlo – Os controlos do veículo respondem a uma determinada entrada do piloto através de retorno de forças, o que reflecte algumas das forças exteriores que actuam no veículo, como por exemplo, a força aerodinâmica no leme de direcção. Estas forças têm de ser geradas artificialmente tendo a designação de carga de controlo (*control loading*) num simulador ou em veículos onde os controlos funcionam electricamente (*fly-by-wire*) [7].

Sistema de som – A realidade criada por um simulador, para além do ambiente envolvente, é substancialmente melhorada pela geração do som de um veículo, sendo este usado na acção de controlo de potência pelo piloto. Por vezes, o sistema de som pode dissimular o ruído incomodativo causado por outros sistemas do simulador [30]

A grande questão reside em saber até que ponto o ambiente simulado reflecte as condições de voo reais e de que modo é requerido para atingir certos objectivos de treino ou de teste.

A qualidade do simulador depende dos dados que contém, ou seja, da qualidade do modelo que representa a aeronave e o ambiente de voo que é simulado. As limitações inerentes ao sistema de simulação podem ser a causa de incompatibilidades. O sistema de movimento é o factor mais limitante de um simulador. No entanto, o piloto pode ser levado a sentir o movimento desejado se as sensações principais estiverem bem simuladas, permitindo alguns desvios nas restantes.

Relativamente ao desenvolvimento inicial do SVI, este é dividido em quatro módulos, cada um deles com a sua função e objectivo. Estes módulos, em conjunto, vão constituir a base de funcionamento de todo o simulador.

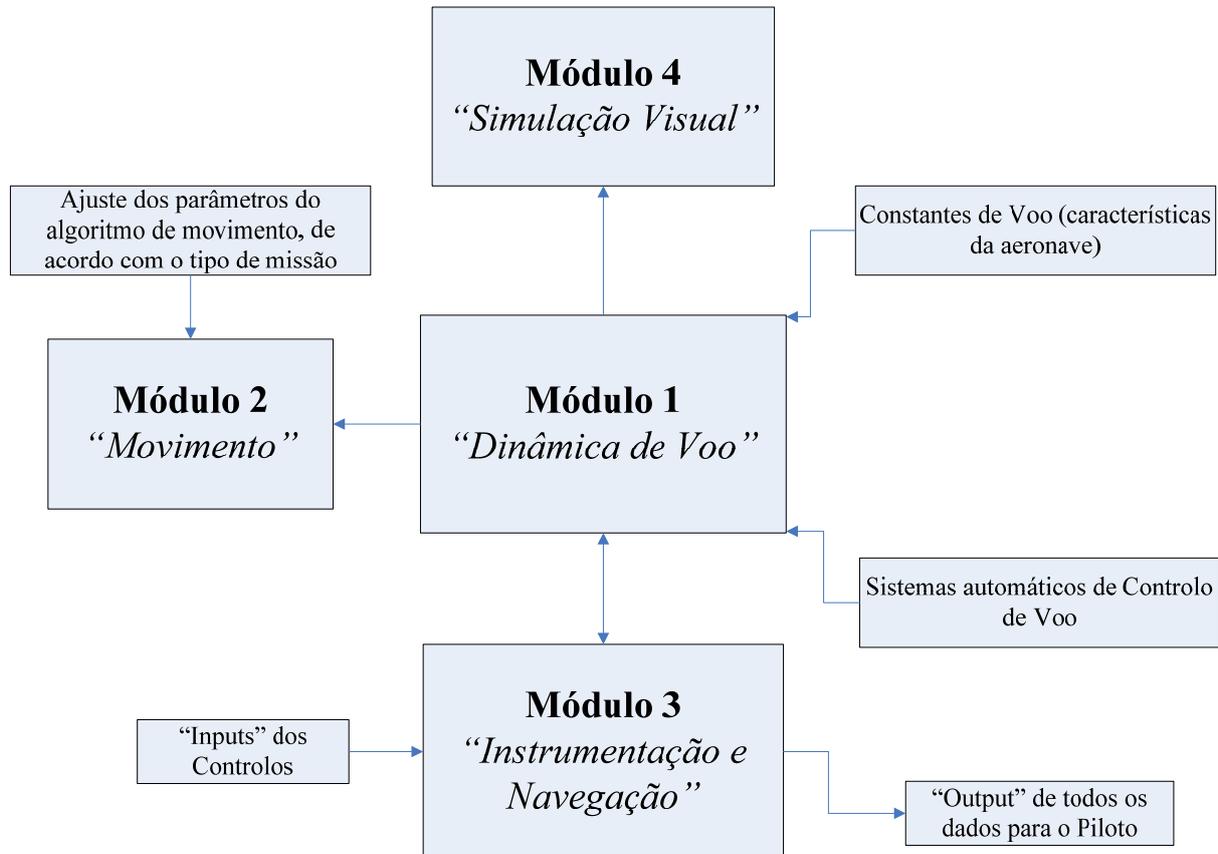


Figura 2.5 – Estrutura modular do SVI.

De acordo com a figura 2.5, onde pode ser observada a organização estrutural básica, assim como o sentido do fluxo de informação, os módulos existentes são os seguintes: (1) módulo da dinâmica de voo, (2) módulo de movimento, (3) módulo da interface com o piloto e (4) módulo de simulação visual. O protocolo de comunicações usado entre eles é descrito na secção 2.5.

No módulo da dinâmica de voo é desenvolvido todo o modelo da dinâmica de uma aeronave. Este módulo pretende ser no futuro o mais abrangente possível em relação às aeronaves pretendidas e funciona como o elo de ligação entre todos os outros módulos do simulador.

O módulo do movimento recria, da forma o mais fiel possível, as sensações reais de voo devidas ao movimento da aeronave, nomeadamente sensações de atitude e aceleração.

Os dois últimos módulos, módulo da interface com o piloto e módulo de simulação visual, são os responsáveis pela geração do “ambiente” no interior da cabine do piloto (*cockpit* da aeronave), isto é, painéis de instrumentação e cenário exterior. No primeiro caso, com base nos dados obtidos da dinâmica, é actualizada a informação disponível nos instrumentos construídos em OpenGL. No segundo caso, para a geração e actualização da imagem adequada à posição da aeronave, recorre-se ao simulador de voo *open source* FlightGear [40].

O objecto de estudo da presente tese é o módulo de simulação da dinâmica de uma aeronave. Os restantes módulos foram já desenvolvidos paralelamente a este, constituindo cada um deles a temática de estudo de outras teses [9] e [10].

2.5. Comunicação entre os módulos

A comunicação entre os módulos é estabelecida através de ligação *Ethernet* entre todos os computadores utilizados, fazendo uso de um *Hub* de rede. Utiliza-se o *User Datagram Protocol* (UDP). O protocolo UDP é normalmente utilizado por aplicações que exigem um transporte rápido e contínuo de dados entre equipamentos. Enquanto no protocolo TCP é dada prioridade à conexão e à chegada correcta dos dados ao ponto de destino, o UDP não verifica a recepção e integridade dos dados enviados. No entanto, a maior simplicidade do UDP faz com que este protocolo apresente ganhos na velocidade de transmissão e recepção de dados, algo que no presente caso tem extrema importância.

3. Dinâmica da Aeronave

Para escrever as equações do movimento é necessário primeiro definir os referenciais utilizados assim como a nomenclatura associada.

3.1. Referenciais

Para situar a aeronave é necessário considerar os diferentes sistemas de coordenadas usados para referenciar geograficamente um corpo ou um veículo. Como se pode observar na figura 3.1, os sistemas de coordenadas consistem em três eixos ortogonais entre si [6].

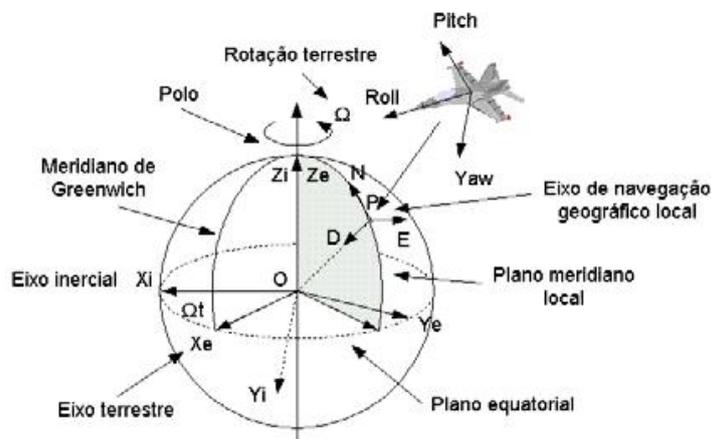


Figura 3.1 – Sistemas de coordenadas.

Segundo [6] os referenciais mais usados são:

- **Sistema ECEF (*Earth-Centered, Earth-Fixed*)** – A sua origem encontra-se no centro de massa da Terra e o seu eixo X passa na intersecção entre a linha do equador e o meridiano de Greenwich. O eixo Z passa pelo pólo norte e o seu eixo Y é orientado seguindo a regra da mão direita.
- **Sistema LLA (*Latitude, Longitude, Altitude*)** – A origem do sistema é o centro de massa da Terra. O Primeiro Meridiano e o Equador são as referências a partir das quais se definem a latitude e a longitude. A latitude Geodésica é definida como sendo o ângulo entre o plano equatorial e a normal à superfície do elipsóide de referência. A longitude geodésica corresponde ao ângulo entre o plano definido pelo Primeiro Meridiano e outro meridiano. Por fim a altitude geodésica é dada pela distância desde o elipsóide de referência até à altitude de um ponto numa direcção normal ao elipsóide.
- **Sistema RPY (*Roll-Pitch-Yaw*)** – Sistema cujos eixos estão fixos num veículo. O eixo X (eixo de rolamento) do sistema RPY aponta na direcção frontal do movimento do veículo, o eixo Y (eixo de picada) deve ser ortogonal a X estando direccionado para a direita enquanto o eixo Z

(eixo de guinada) está orientado para baixo (regra da mão direita). Este sistema de eixos é também designado como sistema de eixos do veículo.

- **Sistema NED (North-East-Down)** – Conhecido também como sistema de navegação local, a sua origem pode ser estabelecida em qualquer ponto do globo terrestre, tendo em conta que o seu eixo X aponta para o Norte geográfico, o seu eixo Z está direccionado para o centro da Terra e seu eixo Y forma um triedro directo estando orientado para Este.

Dos sistemas anteriormente referidos o que melhor se ajusta a um referencial de navegação é o sistema NED, facilitando a análise local da posição da aeronave. Contudo o sistema RPY também assume uma certa importância quando é pretendido utilizar-se o referencial da aeronave no cálculo das equações de movimento. No Anexo B estão descritos os eixos de referência usados e as transformações necessárias para se utilizarem os dois referenciais convenientemente.

Pressupondo que os dados relativos à posição da aeronave estão no formato *Lat, Long, Alt*, e assumindo o modelo do elipsóide da Terra WGS84 (latitude, longitude e altitude), torna-se necessário converter estes dados para o sistema de coordenadas NED. Realizando primeiro uma transformação para coordenadas cartesianas ECEF, é possível através de algumas transformações chegar às coordenadas desejadas [6]. Para tal implementa-se através do ficheiro *LLA2NED.m* esta transformação. No Anexo C apresenta-se este ficheiro sendo usadas a posição da aeronave e o valor de coordenadas que se assume para a origem do referencial NED.

O referencial chamado “global” é um qualquer referencial ortogonal considerado adjacente à superfície terrestre. Para o presente estudo, este referencial pode ser considerado inercial, onde são válidas as leis de Newton, dado que a velocidade rotacional da Terra pode ser desprezada [39].

O referencial local da aeronave é um referencial rígido com a estrutura da aeronave, cuja origem é o centro de gravidade da aeronave e que segue a orientação do sistema de eixos RPY.

A posição de um corpo rígido pode ser definida por uma combinação de translações e rotações, a partir de uma posição de referência. A posição de referência considerada coincide com o referencial global definido anteriormente. Assim, as velocidades e acelerações da aeronave em relação ao referencial global, tanto lineares como angulares, são expressas no seu sistema de coordenadas local. Também a orientação angular da aeronave é definida por um conjunto de rotações a partir do referencial global, por exemplo os Ângulos de Euler.

Na Figura 3.2 podem observar-se as transformações de coordenadas necessárias para a definição dos ângulos ϕ, θ e ψ , respectivamente os ângulos de rolamento, picada e guinada.

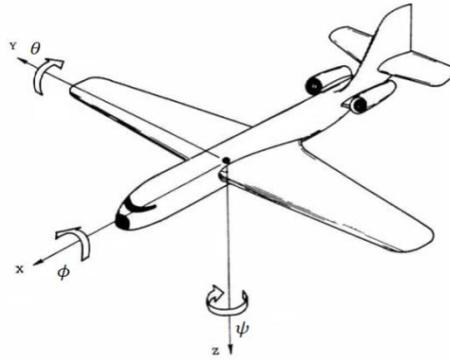


Figura 3.2 – Ângulos de Euler (só para pequenos ângulos).

Após esta introdução acerca dos referenciais existentes pode agora começar-se a desenvolver as equações do movimento de uma aeronave.

3.2. Equações do Movimento

As equações de movimento para um corpo rígido, de acordo com a 2ª Lei de Newton, considerando massa e inércias constantes, são dadas por:

$$\mathbf{F} = m \frac{d\mathbf{V}}{dt} \quad (3.1)$$

$$\mathbf{M} = \frac{d\mathbf{h}}{dt} \quad (3.2)$$

Onde \mathbf{F} é a força aplicada, \mathbf{M} é o momento, \mathbf{V} é o vector velocidade ($\mathbf{V} = \{U, V, W\}$) e \mathbf{h} é o momento angular. Pode escrever-se \mathbf{h} como o produto entre o tensor de inércia com a velocidade angular:

$$\mathbf{h} = [\mathbf{I}]\boldsymbol{\omega} \quad (3.3)$$

Onde $\boldsymbol{\omega}$ é o vector velocidade angular ($\boldsymbol{\omega} = \{P, Q, R\}$) e $[\mathbf{I}]$ é tensor de inércia, sendo o tensor de inércia definido por:

$$\mathbf{I} = \begin{bmatrix} I_{XX} & -I_{XY} & -I_{XZ} \\ -I_{YX} & I_{YY} & -I_{YZ} \\ -I_{ZX} & -I_{ZY} & I_{ZZ} \end{bmatrix} \quad (3.4)$$

No Anexo A encontram-se as fórmulas para o cálculo de cada componente do tensor de inércia.

Uma vez que queremos escrever as equações do movimento para o referencial do corpo, ou seja, referencial local, podemos então definir a derivada em ordem ao tempo como d/dt e reescrever a equação do movimento (3.1) da seguinte forma:

$$\mathbf{F} = m \frac{d\mathbf{V}}{dt} + m\boldsymbol{\omega} \times \mathbf{V} \quad (3.5)$$

Que em termos de componentes no referencial local fica definida como:

$$X = m[dU/dt + QW - RV] \quad (3.6a)$$

$$Y = m[dV/dt + RU - PW] \quad (3.6b)$$

$$Z = m[dW/dt + PV - QU] \quad (3.6c)$$

A equação do momento fica assim definida por:

$$\mathbf{M} = d\mathbf{h}/dt + \boldsymbol{\omega} \times \mathbf{h} \quad (3.7)$$

Podendo ser expandida, desprezando termos usualmente pequenos na matriz $[\mathbf{I}]$, em:

$$L = I_{xx}(dP/dt) - I_{xz}(dR/dt + PQ) + QR(I_{zz} - I_{yy}) \quad (3.8a)$$

$$M = I_{yy}(dQ/dt) + I_{xz}(P^2 - R^2) + PR(I_{xx} - I_{zz}) \quad (3.8b)$$

$$N = I_{zz}(dR/dt) + I_{xz}(QR - dP/dt) + PQ(I_{yy} - I_{xx}) \quad (3.8c)$$

A nomenclatura usada nas equações anteriores é a tradicional (americana), segundo a qual são atribuídas letras sequenciais para as três componentes dos vectores, correspondentes às diferentes grandezas.

Esta breve secção descreveu as equações de movimento de uma aeronave. De seguida será relatado como é construído todo o modelo da dinâmica da aeronave estudada.

3.3. Equações do modelo

O modelo utilizado para simular a dinâmica da aeronave foi o Navion. Esta aeronave de fabrico norte-americano, tem capacidade para quatro passageiros, possui uma envergadura de dez metros e dispõe de uma motorização com 260 CV.

O modelo é desenvolvido através de uma biblioteca do Simulink [31], chamada AeroSim [38], com a qual é possível simular a dinâmica de uma aeronave para a sua integração no SVI.

Este modelo é constituído por vários blocos, sendo cada bloco responsável pelas respectivas equações de dinâmica que lhes estão inerentes, como se pode constatar na figura 3.3.

De seguida, são apresentados, detalhadamente, cada um dos blocos do modelo e as equações que lhe estão associadas.

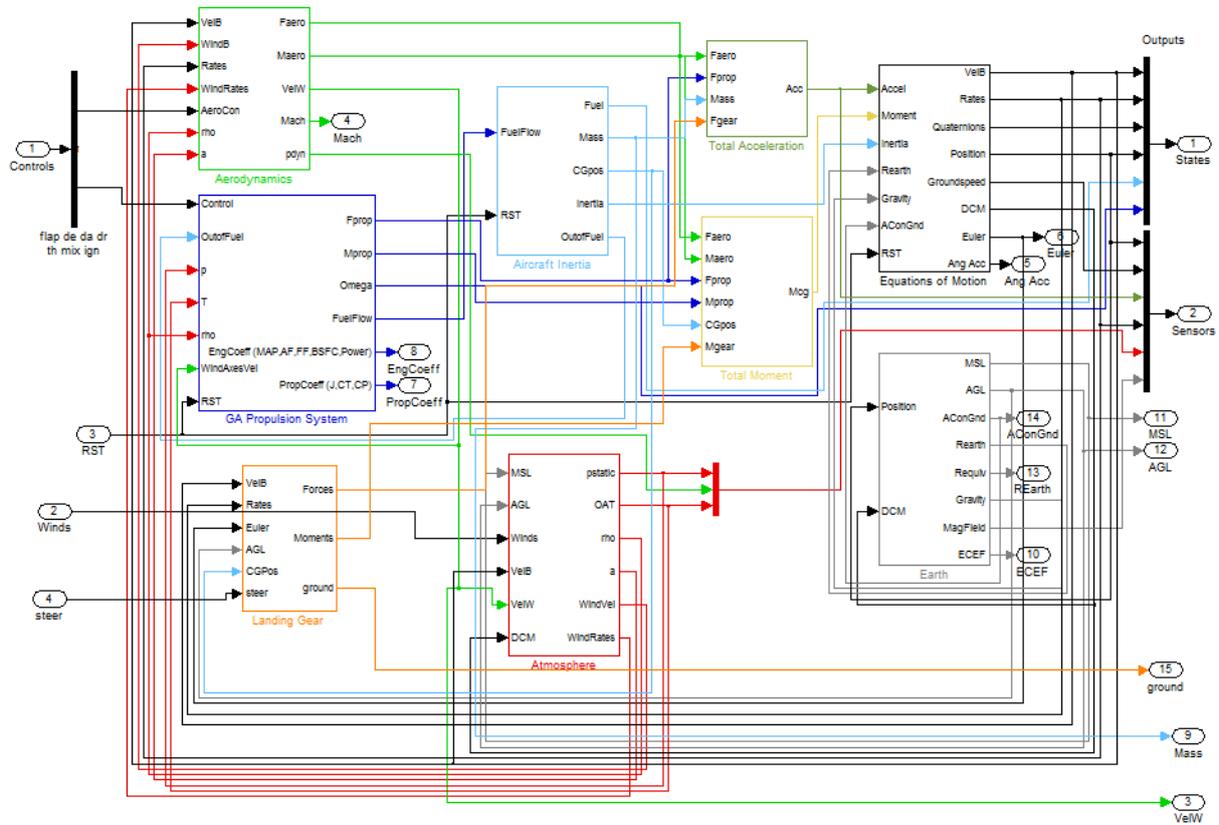


Figura 3.3 – Modelo da Dinâmica da Aeronave.

3.3.1. Aerodinâmica

O primeiro bloco a ser descrito é o bloco da Aerodinâmica. Este bloco está apresentado na figura 3.4, e contém todos os blocos simples para a descrição completa do modelo da aerodinâmica de uma aeronave.



Figura 3.4 – Bloco da Aerodinâmica.

Este bloco é uma aproximação linear, no qual as forças e os momentos aerodinâmicos são calculados usando combinações lineares das derivadas aerodinâmicas.

Os parâmetros da aerodinâmica da aeronave são configurados num *Script* inicial, e dependem do tipo de aeronave. Estes parâmetros são:

- **Ponto de referência** – Especifica a localização do ponto de aplicação das forças aerodinâmicas, em relação à origem do referencial local da aeronave.
- **Limites dos parâmetros da aerodinâmica** – Indica os valores máximos e mínimos que os parâmetros da aerodinâmica podem ter, para que os resultados do modelo da aerodinâmica estejam dentro da região linear, ou seja, a aeronave tenha velocidades que estejam dentro do envelope de voo. Estes parâmetros são simples saturações.
- **Parâmetros da aerodinâmica normalizados** – Parâmetros de referencia para os quais os coeficientes aerodinâmicos foram normalizados. Como por exemplo corda, envergadura e área da asa da aeronave.

O diagrama de blocos da aerodinâmica pode ser observado através da figura 3.5. A cada bloco corresponde uma série de equações que serão descritas seguidamente.

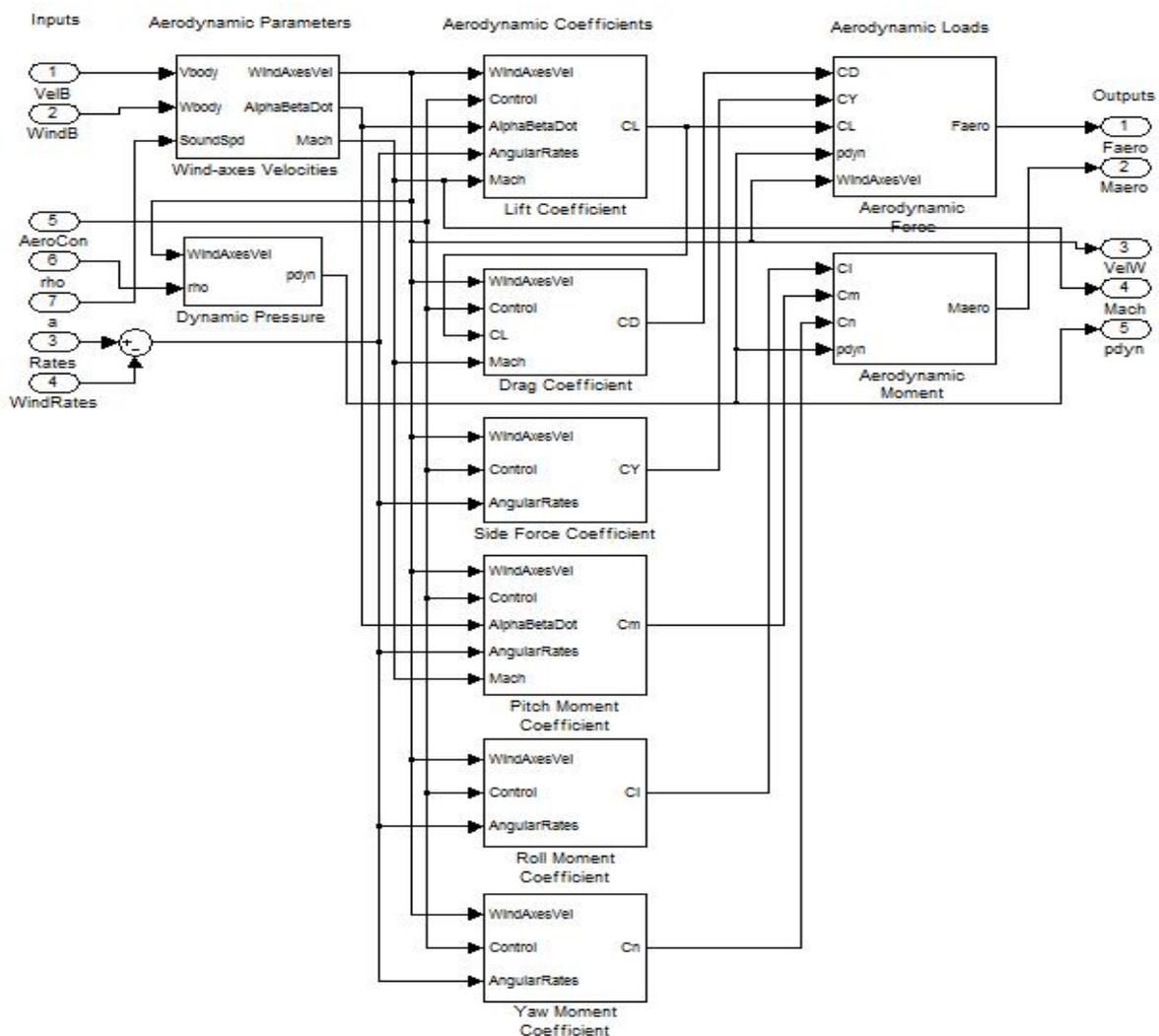


Figura 3.5 – Bloco da Aerodinâmica interior.

- **Aerodynamic Force** – Este bloco calcula a força aerodinâmica gerada, através dos coeficientes aerodinâmicos, das velocidades no referencial do ar assim como a pressão dinâmica. O único parâmetro utilizado é a área de referência da asa, S . As equações utilizadas por este bloco são:

$$D_a = \frac{1}{2} \rho V_a^2 S C_D \quad (3.9)$$

$$Y_a = \frac{1}{2} \rho V_a^2 S C_Y \quad (3.10)$$

$$L_a = \frac{1}{2} \rho V_a^2 S C_L \quad (3.11)$$

Em que D_a é a força de resistência, Y_a é a força lateral e L_a é a força de Sustentação. As forças são convertidas para o referencial da aeronave.

A entrada do bloco *Wind-axes Velocities* corresponde a um vector 3×1 contendo a velocidade do ar V_a , o ângulo de ataque α e o ângulo de derrapagem β .

- **Aerodynamic Moment** – É o Bloco responsável pelo cálculo do momento aerodinâmico criado na aeronave. As equações que regem este modelo são:

$$L = \frac{1}{2} \rho V_a^2 S b C_l \quad (3.12)$$

$$M = \frac{1}{2} \rho V_a^2 S c C_m \quad (3.13)$$

$$N = \frac{1}{2} \rho V_a^2 S b C_n \quad (3.14)$$

Onde L , M e N são respectivamente o momento de rolamento, de picada e guinada¹, c é a corda média e b é a envergadura da asa.

- **Dynamic Pressure** – Este bloco permite calcular a pressão dinâmica para uma determinada altitude. A equação que rege o bloco é:

$$q = \frac{1}{2} \rho V_a^2 \quad (3.15)$$

Em que ρ é a densidade do ar para um determinado valor de altitude.

- **Wind-Axes Velocities** – Este bloco calcula as velocidades no sistema de eixos do ar através da velocidade no chão, *Ground-speed*, da velocidade do ar, *Wind Speed*, e da velocidade do som à altitude da aeronave. A velocidade do ar no referencial da aeronave $[u \ v \ w]$ é calculada subtraindo a velocidade do vento à velocidade no chão. As equações principais inerentes a este bloco são:

$$\alpha = \text{atan} \left(\frac{w}{u} \right) \quad (3.16)$$

$$\beta = \text{asin} \left(\frac{v}{V_a} \right) \quad (3.17)$$

$$M = \frac{V_a}{a} \quad (3.18)$$

¹ Na notação usual apresentada aqui existe uma ambiguidade entre sustentação e momento de rolamento (ambos denotados L) mas o contexto permite normalmente diferenciá-los.

Onde M é o número de Mach.

- **Lift Coefficient** – Bloco responsável pelo cálculo do coeficiente de sustentação através da combinação linear da contribuição de vários parâmetros de voo, que pode ser traduzido através da seguinte equação:

$$C_L = C_{L0} + C_L^\alpha \alpha + C_L^{\delta_f} \delta_f + C_L^{\delta_e} \delta_e + \frac{c}{2V_a} (C_L^\alpha \dot{\alpha} + C_L^q q) + C_L^M M \quad (3.19)$$

- **Drag Coefficient** – Bloco que calcula o coeficiente de resistência tal como o bloco anterior e a equação que o determina é:

$$C_D = C_{D0} + \frac{(C_L - C_{L0})^2}{\pi e AR} + C_D^{\delta_f} \delta_f + C_D^{\delta_e} \delta_e + C_D^{\delta_a} \delta_a + C_D^{\delta_r} \delta_r + C_D^M M \quad (3.20)$$

Onde e é o factor de eficiência de Oswald e AR é a alongamento (*Aspect Ratio*).

- **Side Force Coefficient** – Bloco para o cálculo do coeficiente de força aerodinâmica lateral e segue a seguinte equação:

$$C_Y = C_Y^\beta \beta + C_Y^{\delta_a} \delta_a + C_Y^{\delta_r} \delta_r + \frac{b}{2V_a} (C_Y^p p + C_Y^r r) \quad (3.21)$$

- **Pitch Moment Coefficient** – Bloco para o cálculo do coeficiente de momento de picada e é caracterizado pela equação:

$$C_m = C_{m0} + C_m^\alpha \alpha + C_m^{\delta_f} \delta_f + C_m^{\delta_e} \delta_e + \frac{c}{2V_a} (C_m^\alpha \dot{\alpha} + C_m^q q) + C_m^M M \quad (3.22)$$

- **Roll Moment Coefficient** – Este bloco calcula o coeficiente de momento de rolamento e segue a equação:

$$C_l = C_l^\beta \beta + C_l^{\delta_a} \delta_a + C_l^{\delta_r} \delta_r + \frac{b}{2V_a} (C_l^p p + C_l^r r) \quad (3.23)$$

- **Yaw Moment Coefficient** – Bloco para o cálculo do coeficiente de momento de guinada, que pode ser descrito por:

$$C_n = C_n^\beta \beta + C_n^{\delta_a} \delta_a + C_n^{\delta_r} \delta_r + \frac{b}{2V_a} (C_n^p p + C_n^r r) \quad (3.24)$$

3.3.2. Propulsão

Outro bloco muito importante para o desenvolvimento da dinâmica de uma aeronave é o bloco da propulsão. Este modelo corresponde a um motor de avião de turismo com hélice de passo constante.

Este bloco encontra-se representado na figura 3.6, o qual efectua o cálculo do binário da hélice, para uma dada rotação e com determinadas condições atmosféricas e de velocidade.

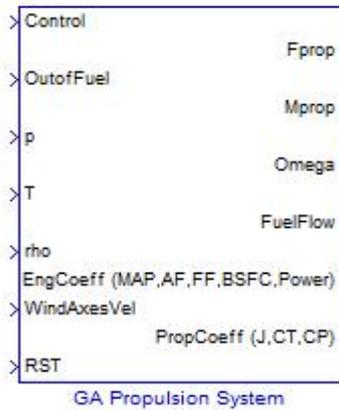


Figura 3.6 – Bloco do Sistema de Propulsão.

A equação diferencial que descreve a dinâmica do sistema de propulsão é:

$$(I_{eng} + I_{prop})\dot{\Omega} = M_{eng} + M_{prop} \quad (3.25)$$

Esta equação é integrada computacionalmente e calcula a velocidade de rotação da hélice do motor, Ω . Onde I_{eng} é o momento de inércia do motor, I_{prop} é o momento de inércia da hélice, M_{eng} é o momento do motor e M_{prop} momento da hélice da aeronave.

Este sistema de propulsão é usado para modelos não lineares de aeronaves para o cálculo das forças e momentos que estão a ser aplicadas. Este modelo de propulsão depende dos controlos do motor, das condições atmosféricas, altitude e velocidade.

Os parâmetros do bloco do sistema de propulsão são definidos no *Script* inicial e vão depender do tipo de motor assim como do tipo de hélice.

A propulsão é definida por dois blocos principais: o **Fixed-Pitch Propeller**, o qual está relacionado com as características da hélice, e o **Piston Engine** onde se definem todas as características do motor. Estes blocos podem ser observados na figura 3.7.

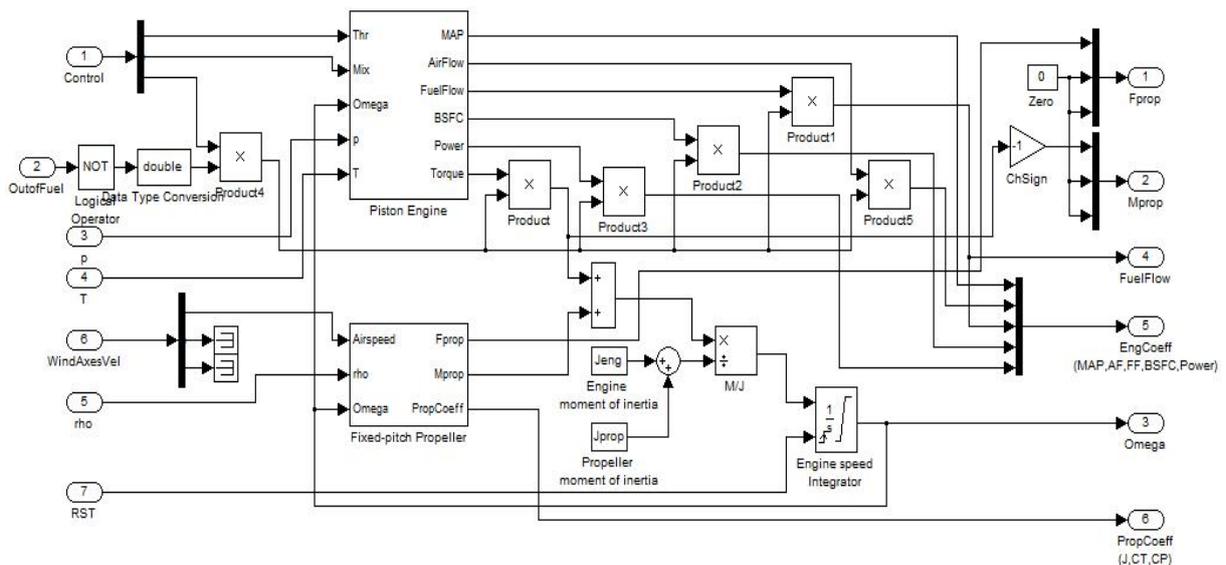


Figura 3.7 – Bloco do Sistema de Propulsão interior.

- **Fixed-Pitch Propeller** – Este bloco é responsável pelo cálculo dos coeficientes de propulsão, da razão de avanço (J), coeficiente de impulso (C_T) e coeficiente de potência (C_P), assim como a força da hélice (F_{PROP}) e o momento ou binário criado pela hélice (M_{PROP}). As equações que regem este modelo são:

$$J = \frac{\pi V_a}{\Omega R} \quad (3.26)$$

$$F_{PROP} = \frac{4}{\pi^2} \rho R^4 \Omega^2 C_T \quad (3.27)$$

$$M_{PROP} = \frac{4}{\pi^3} \rho R^5 \Omega^2 C_P \quad (3.28)$$

O cálculo de C_T e C_P é feito por interpolação linear através das tabelas destes coeficientes em função dos valores de J , definidas no *Script* inicial.

- **Piston Engine** – Bloco que contém o modelo de combustão do motor, tendo como base as tabelas dos parâmetros do motor. As tabelas necessárias para este modelo são tabelas de duas dimensões, ou seja, matrizes de fluxo de combustível e potência do motor ao nível do mar, em função da velocidade de rotação motor, RPM, e da pressão de trabalho do motor, MAP *Manifold Pressure*. Este bloco recebe duas entradas, muito importantes para o controlo da propulsão, o *Throttle*, *Thr*, e a razão de mistura ar/combustível, *Mix*. O *Throttle* será uma entrada no joystick do piloto e representa a fracção entre a pressão de trabalho do motor e a pressão atmosférica. Normalmente para motores genéricos a pressão de trabalho do motor é menor ou igual à pressão atmosférica a uma dada altitude e por isso o valor do *Throttle* varia de 0 a 1.

3.3.3. Atmosfera

O bloco da Atmosfera, representado na figura 3.8, é responsável pela estimativa dos parâmetros do ar assim como pelas perturbações atmosféricas resultante dos efeitos do vento.



Figura 3.8 – Bloco da Atmosfera.

O modelo da atmosfera é constituído por quatro blocos. A figura 3.9 mostra estes blocos.

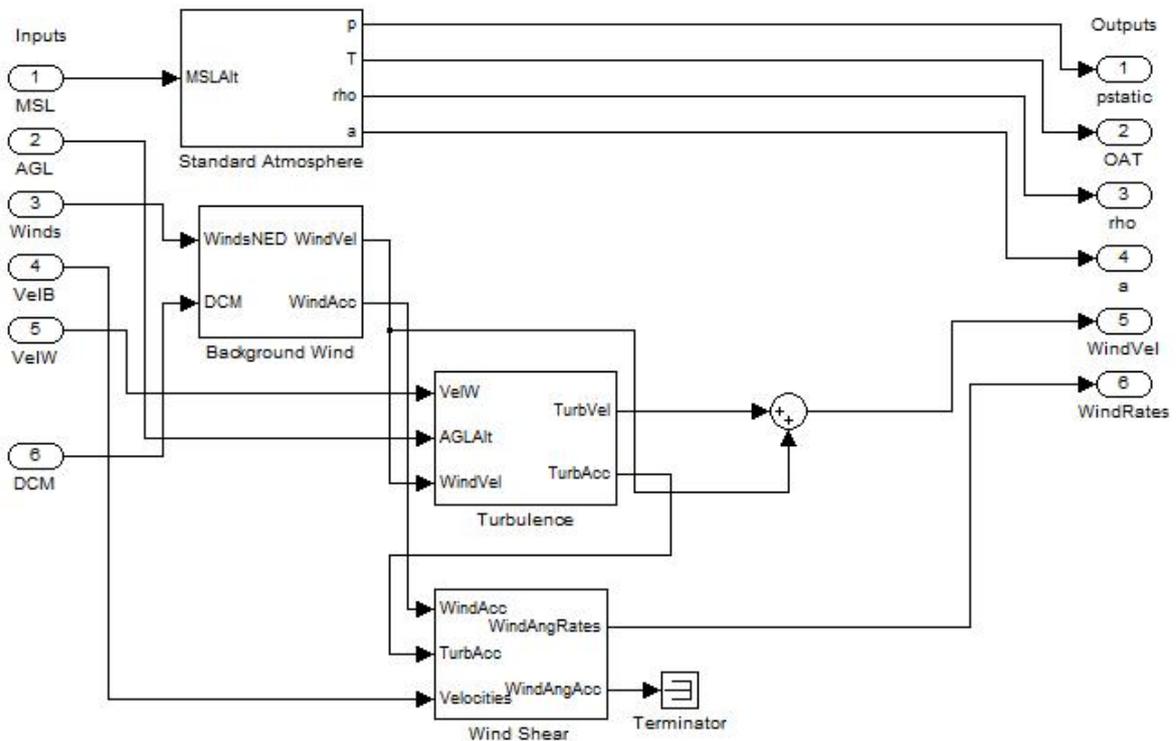


Figura 3.9 – Bloco da Atmosfera interior.

- Standard Atmosphere** – Bloco para o cálculo dos parâmetros do ar para a altitude da aeronave. A única entrada do bloco é a altitude medida a partir no nível médio da água do mar, MSL. As saídas do bloco são: a pressão estática, a temperatura exterior, a densidade do ar e a velocidade do som. Estes parâmetros são calculados através de interpolações das tabelas para uma Atmosfera Standard 1976.
- Background Wind** – Este bloco é responsável pelo cálculo das componentes da velocidade e da aceleração do vento no referencial local da aeronave. As entradas do bloco são as componentes da velocidade do vento no referencial NED e a matriz dos co-senos directores (DCM). O bloco aplica a transformação inercial ou geográfica para o referencial da aeronave através da matriz de rotação.
- Turbulence** – Bloco que define o modelo de turbulência de *Von Karman*. Neste bloco é aplicada turbulência de *Von Karman* através de filtros de 3 fontes de ruído branco para as componentes longitudinal, lateral e vertical. Os parâmetros do filtro dependem da intensidade do vento e da altitude da aeronave. O bloco devolve os valores da velocidade e da aceleração de turbulência no referencial da aeronave.
- Wind Shear** – Bloco para o cálculo das velocidades e acelerações angulares do vento no referencial local da aeronave causadas pela variação, no tempo e espaço, da velocidade do

vento e turbulência. As variações de velocidade e aceleração consideradas são para a picada e guinada. A variação da picada e da guinada devido ao vento pode ser calculada, respectivamente, através das equações:

$$q_{wind} = \frac{1}{u_{aeronave}} \frac{dw_{wind}}{dt} \quad (3.29)$$

$$r_{wind} = \frac{1}{u_{aeronave}} \frac{dv_{wind}}{dt} \quad (3.30)$$

As acelerações para a picada e guinada são calculadas computacionalmente através da derivada em ordem ao tempo das velocidades angulares.

3.3.4. Terra

De seguida é explicado o modelo que determina a forma da terra, a gravidade e o campo criado pela mesma. Este modelo é designado e representado pelo bloco *Earth* (Terra), pois como o próprio nome designa calcula e define todos os parâmetros da Terra. Este bloco está representado na figura 3.10. Este modelo é constituído por quatro blocos, responsáveis pelo cálculo de todos os parâmetros enunciados atrás, como se pode observar na figura 3.11.

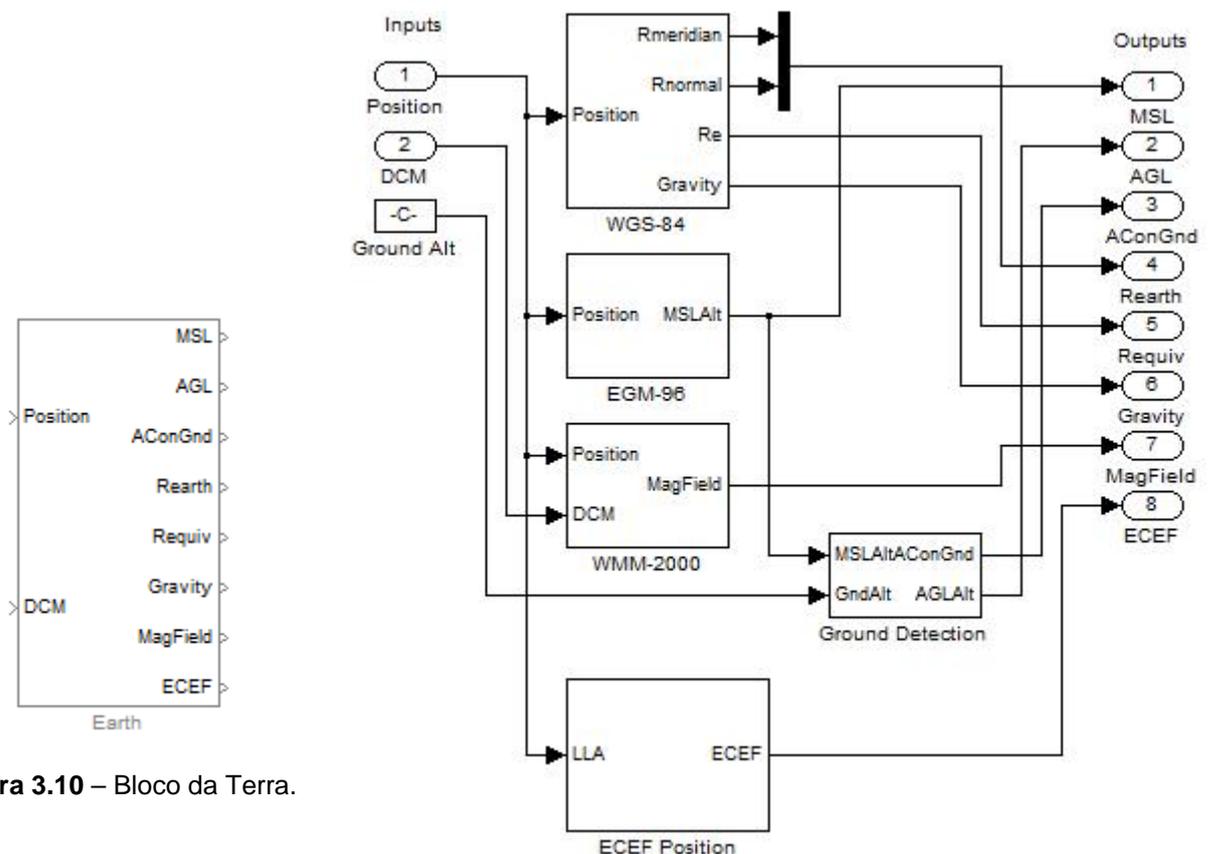


Figura 3.10 – Bloco da Terra.

Figura 3.11 – Bloco da Terra interior.

- **WGS-84** – Bloco que calcula o raio da Terra e a gravidade para a posição da aeronave. Este bloco usa as equações e os coeficientes do modelo da terra WGS-84 [32]:

$$R_{meridiano} = \frac{r_e(1-\varepsilon^2)}{(1-\varepsilon^2(\sin \phi)^2)^{\frac{3}{2}}} \quad (3.31)$$

$$R_{normal} = \frac{r_e}{(1-\varepsilon^2(\sin \phi)^2)^{\frac{1}{2}}} \quad (3.32)$$

$$R_{equiv} = \sqrt{R_{meridiano}R_{normal}} \quad (3.33)$$

$$g = g_{WGS_0} \frac{1+g_{WGS_1}(\sin \phi)^2}{(1-\varepsilon^2(\sin \phi)^2)^{\frac{1}{2}}} \quad (3.34)$$

Onde r_e é o raio da Terra no Equador e vale 6378137 (m), ε é a excentricidade da Terra e tem o valor de 0.0818191908426, g_{WGS_0} é a gravidade no Equador e vale 9.7803267714 (m/s²), g_{WGS_1} é a constante da fórmula da gravidade e tem o valor de 0.00193185138639 e ϕ é a latitude da posição da aeronave.

- **EGM-96** – Bloco responsável pelo cálculo da altitude da aeronave acima do nível do mar. Este valor é calculado através da diferença da altitude dada pela elipsóide teórica de referência e a actual altitude acima do nível do mar, através da superfície equipotencial da gravidade da Terra que está associada com a média da superfície oceânica. Esta diferença é causada pela não uniformidade do potencial gravitacional da Terra [33].
- **Ground Detection** – Bloco usado para detectar se a aeronave está no solo ou se está a uma determinada altitude. A altitude do solo deverá ser introduzida através de uma tabela, que indicará a elevação do solo, no *Script* inicial.
- **WMM-2000** – Bloco que calcula o campo magnético da Terra através da posição da aeronave usado pelo Modelo Magnético 2005 do Departamento de Defesa dos Estados Unidos da América. Devido à variação, no tempo, do campo geomagnético da Terra, os dados são fiáveis apenas em intervalos de cinco anos. Depois disso, terá que se fazer uma actualização do ficheiro wmm.cof, [34]. Ou seja, este modelo será válido até 31 de Dezembro de 2009.
- **ECEF Position** – Bloco que calcula as coordenadas no Sistema ECEF através das coordenadas geográficas da aeronave, ou seja, Latitude, Longitude e Altitude. As equações para o cálculo desta posição [32] são:

$$X_{ECEF} = \left(\frac{r_e}{\sqrt{1-\varepsilon^2(\sin Lat)^2}} + Alt \right) \cos(Lat) \cos(Lon) \quad (3.35)$$

$$Y_{ECEF} = \left(\frac{r_e}{\sqrt{1-\varepsilon^2(\sin Lat)^2}} + Alt \right) \cos(Lat) \sin(Lon) \quad (3.36)$$

$$Z_{ECEF} = \left(\frac{r_e(1-\varepsilon^2)}{\sqrt{1-\varepsilon^2(\sin Lat)^2}} + Alt \right) \sin(Lat) \quad (3.37)$$

3.3.5. Inércia

A inércia da aeronave está desenvolvida num bloco, que modela a evolução dos parâmetros da aeronave, tais como a massa, a posição do centro de gravidade (CG) e os momentos de inércia. Este bloco está representado na figura 3.12.

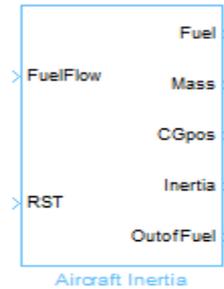


Figura 3.12 – Bloco da Inércia da Aeronave.

Este bloco integra o consumo de combustível para obter os parâmetros de inércia enunciados atrás. As entradas do bloco são: a percentagem de consumo de combustível e o *Reset* (comando responsável pelo reiniciar da simulação).

As condições iniciais para a integração são obtidas através dos parâmetros definidos no *Script* inicial. Os momentos de inércia assim como a posição do centro de gravidade são calculados através de interpolações lineares entre os parâmetros de inércia da aeronave vazia e cheia (com passageiros, cargas e combustível).

3.3.6. Trem de Aterragem

O trem de aterragem desenvolve forças e momentos que devem ser somados às equações do movimento para uma simulação mais coerente de um modelo de uma aeronave quando se pretende incluir as fases de aterragem ou descolagem. Por conseguinte, existe um bloco para modelar o trem de aterragem da aeronave e assim calcular as forças e momentos causados pelo próprio. Este bloco não existe originalmente na biblioteca Aerosim, no entanto foi adicionado permitindo uma simulação mais próxima da realidade da dinâmica da aeronave, nomeadamente para a descolagem e aterragem.

Na figura 3.13 está representado o bloco responsável pela geração das forças e momentos do trem de aterragem.

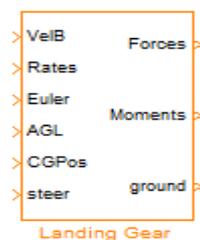


Figura 3.13 – Bloco do Trem de Aterragem da Aeronave.

Este modelo é desenvolvido de modo semelhante para as três rodas da aeronave, ou seja, para um trem convencional, sendo a força e o momento total a soma de cada uma das componentes do trem. Os parâmetros definidos no *Script* inicial são: a posição de cada roda do trem, a espessura da superfície de contacto, o coeficiente de amortecimento, o coeficiente de atrito ao longo dos eixos X e Y e a força normal máxima.

A força de atrito pode ser calculada da seguinte forma:

$$F_{atrito} = \mu \cdot R_N \quad (3.38)$$

Em que R_N é a força normal ao solo e μ é o coeficiente de atrito do pneu na superfície de contacto.

O sistema foi desenvolvido com base na teoria de um sistema massa mola, onde são calculadas as forças laterais, longitudinais e perpendiculares de cada roda. Após o cálculo da respectiva força calcula-se o respectivo momento causado pela força de atrito. Este momento é dado pela multiplicação da força pelo braço, sendo este braço a distância perpendicular do centro de gravidade da aeronave ao ponto de aplicação da força.

A roda do trem do nariz da aeronave pode rodar em torno do eixo vertical, de modo a possibilitar a aeronave a mudar de direcção no chão. A direcção da roda depende então, de uma entrada específica do modelo.

3.3.7. Aceleração Total

O bloco responsável pelo cálculo total das acelerações da aeronave está representado na figura 3.14.

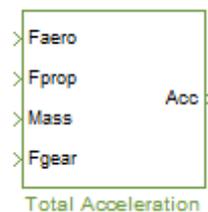


Figura 3.14 – Bloco da Aceleração Total da Aeronave.

Este bloco determina a aceleração total da aeronave devido às forças totais aplicadas, ou seja, forças aerodinâmicas, forças propulsivas e forças causadas pelo trem. Para este cálculo são incluídas todas as forças excepto a força gravítica. A equação que rege este modelo é:

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}_{Total} = \frac{1}{M_{aeronave}} \left(\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}_{aero} + \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}_{prop} + \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}_{gear} \right) \quad (3.39)$$

3.3.8. Momento Total

O bloco da figura 3.15 é responsável pelo cálculo dos momentos totais da aeronave. O momento total é calculado através da soma dos momentos aerodinâmicos, propulsivos e do trem e está medido em relação ao centro de gravidade da aeronave.

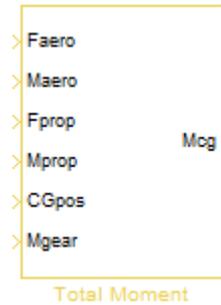


Figura 3.15 – Bloco do Momento Total da Aeronave.

A equação que rege este modelo é:

$$\mathbf{M}_{CG} = (\mathbf{r}_{aero} - \mathbf{r}_{CG}) \times \mathbf{F}_{aero} + \mathbf{M}_{aero} + (\mathbf{r}_{prop} - \mathbf{r}_{CG}) \times \mathbf{F}_{prop} + \mathbf{M}_{prop} + \mathbf{M}_{gear} \quad (3.40)$$

Em que \mathbf{r}_{aero} é a posição do ponto de aplicação das forças aerodinâmicas e momentos no referencial da aeronave, \mathbf{r}_{prop} é a posição onde as forças e momentos propulsivos são aplicados e \mathbf{r}_{CG} é a posição do centro de gravidade no referencial da aeronave. O produto vectorial (ou produto externo) é representado por \times . No Anexo A está definida a forma como se calcula o produto externo.

3.3.9. Equações do Movimento

Para finalizar é descrito o bloco das Equações do Movimento (figura 3.16). Este contém os blocos de todas as equações diferenciais para descrever a dinâmica da aeronave. Este bloco poderá ser considerado o principal de todo o modelo da dinâmica da aeronave, pois contém as equações da dinâmica (3.5) e da cinemática (3.7).

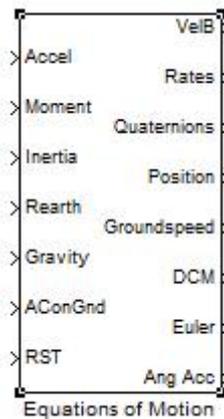


Figura 3.16 – Bloco das Equações do Movimento.

Existem duas formulações possíveis que podem ser usadas por este modelo. Por conseguinte, podemos obter as velocidades da aeronave no referencial da mesma, ou no referencial NED. Estas formulações têm as mesmas entradas e saídas sendo por isso de fácil implementação. No entanto, o modelo aqui construído é o que permite obter as velocidades no referencial local da aeronave.

Este modelo é composto por vários blocos, como se pode observar na figura 3.17. Blocos estes que contêm todas as equações diferenciais do modelo. Seguidamente será descrito cada bloco deste modelo, indicando as equações que admitem o modelo.

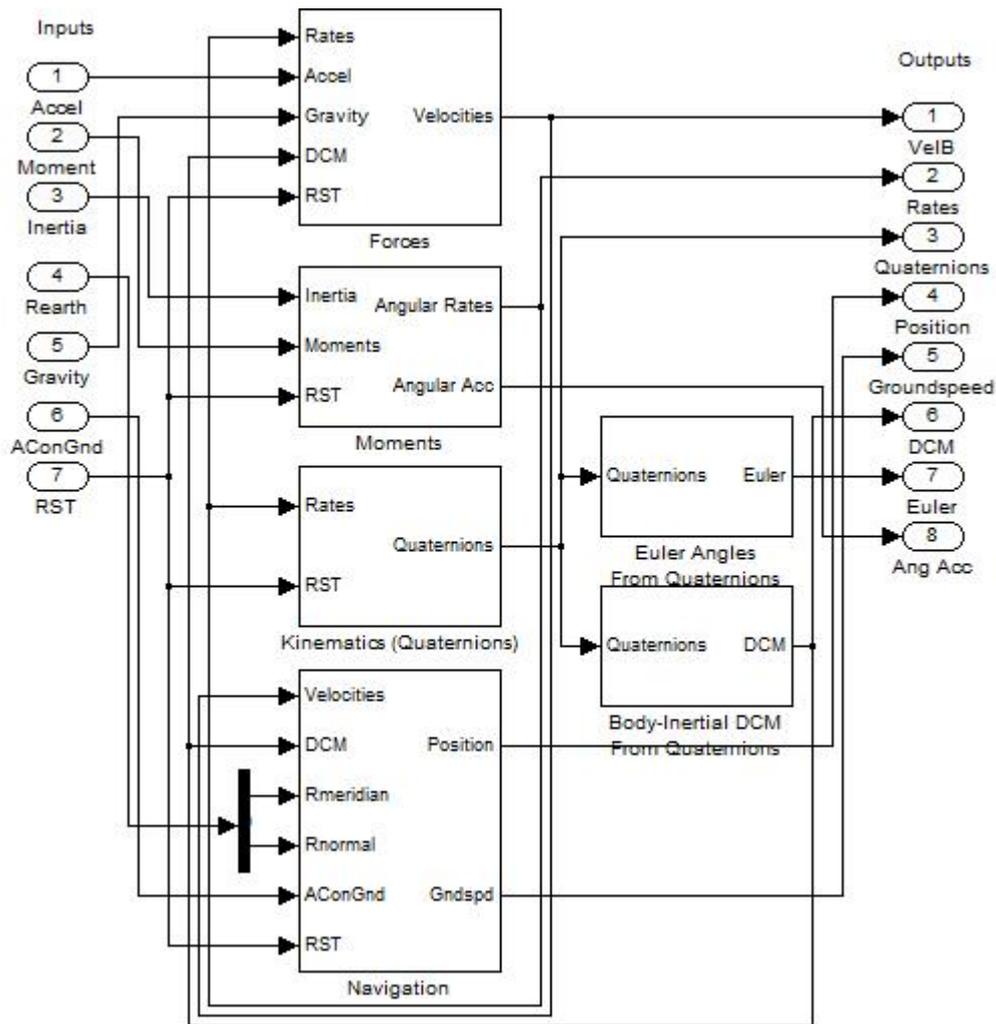


Figura 3.17 – Bloco das Equações do Movimento interior.

- **Forces** – Trata-se do bloco que implementa as equações do corpo rígido com 6 *Degree of Freedom*, **DOF**, para descrever a variação, no tempo, das velocidades da aeronave, no referencial desta. As equações deste modelo são [35]:

$$\dot{u} = rv - qw + g_x + a_x \quad (3.41)$$

$$\dot{v} = -ru + pw + g_y + a_y \quad (3.42)$$

$$\dot{w} = qu - pv + g_z + a_z \quad (3.43)$$

- **Moments** – Bloco semelhante ao anterior, no entanto este implementa as equações do corpo rígido com 6 **DOF** para obter as velocidades angulares instantâneas da aeronave, no referencial local. As equações implementadas são [35]:

$$\dot{p} = (c_1 r + c_2 p)q + c_3 L + c_4 N \quad (3.44)$$

$$\dot{q} = c_5 p r - c_6 (p^2 - r^2) + c_7 M \quad (3.45)$$

$$\dot{r} = (c_8 p - c_2 r)q + c_4 L + c_9 N \quad (3.46)$$

Onde os coeficientes c_1 a c_9 são calculados pelo bloco **Inertia Coefficients**. Os coeficientes de inércia são calculados através dos momentos de inércia e podem ser obtidos através das equações do Anexo A.

- **Kinematics (Quaternions)** – Bloco que integra as velocidades angulares da aeronave para obter os parâmetros de atitude como representação com quatérniões. O objectivo deste sistema de representação de parâmetros é eliminar as singularidades, existentes na representação da rotação em termos dos ângulos de Euler, associadas às funções trigonométricas seno e co-seno. As equações principais deste bloco estão escritas seguidamente e podem ser vistas com mais detalhe em [36].

$$\begin{bmatrix} \dot{e}_0 \\ \dot{e}_x \\ \dot{e}_y \\ \dot{e}_z \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \cdot \begin{bmatrix} e_0 \\ e_x \\ e_y \\ e_z \end{bmatrix} \quad (3.47)$$

- **Euler Angles From Quaternions** – Bloco que transforma os quatérniões em ângulos de Euler. As equações deste bloco são [36]:

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \text{atan2}[2(e_0 e_x + e_y e_z), (e_0^2 + e_z^2 - e_x^2 - e_y^2)] \\ \text{asin}[2(e_0 e_y - e_x e_z)] \\ \text{atan2}[2(e_0 e_z + e_x e_y), (e_0^2 + e_x^2 - e_y^2 - e_z^2)] \end{bmatrix} \quad (3.48)$$

Os ângulos de Euler são extremamente importantes no presente trabalho, pois é através deles que definimos a orientação relativa da aeronave no espaço tridimensional Euclidiano (NED).

- **Body-Inertial DCM From Quaternions** – Bloco usado para o cálculo da matriz dos co-senos directores a partir dos quatérniões. As equações que regem este bloco são [36]:

$$DCM = \begin{bmatrix} e_x^2 + e_0^2 - e_y^2 - e_z^2 & 2(e_x e_y + e_z e_0) & 2(e_x e_z - e_y e_0) \\ 2(e_x e_y - e_z e_0) & e_y^2 + e_0^2 - e_x^2 - e_z^2 & 2(e_y e_z + e_x e_0) \\ 2(e_x e_z + e_y e_0) & 2(e_y e_z - e_x e_0) & e_z^2 + e_0^2 - e_x^2 - e_y^2 \end{bmatrix} \quad (3.49)$$

- **Navigation** – Bloco que integra as equações de navegação para calcular a posição da aeronave assim como a velocidade relativa ao chão, *Ground Speed*, no referencial **NED**. As equações da navegação estão escritas de seguida [36]:

$$\begin{bmatrix} V_{North} \\ V_{East} \\ V_{Down} \end{bmatrix} = DCM^T \cdot \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (3.50)$$

Sendo a posição geográfica obtida através desta velocidade segundo [32]:

$$\dot{Lat} = \frac{V_{North}}{R_{meridian} + Alt} \quad (3.51)$$

$$\dot{Lon} = \frac{V_{East}}{(R_{normal} + Alt) \cos(Lat)} \quad (3.52)$$

$$Alt = \begin{cases} -V_{Down}, & ACondGnd = 0 \\ 0, & ACondGnd = 1 \end{cases} \quad (3.53)$$

A variação de altitude, na última equação, depende do estado da *flag* para a condição da aeronave no chão, tendo o valor um ou zero consoante a aeronave se encontre ou não no solo.

3.3.10. Sensores e Actuadores

Em seguida está descrito o bloco dos actuadores, figura 3.18. Os actuadores são usados em sistemas de controlo de voo sendo úteis para testar a estabilidade e a desempenho de sistemas de piloto automático [38], uma vez que são eles que fazem a ponte entre o sistema de controlo e as superfícies aerodinâmicas que controlam a atitude da aeronave.

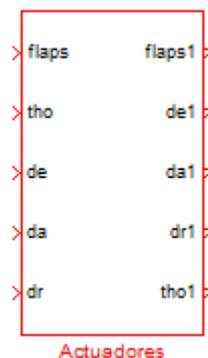


Figura 3.18 – Bloco dos Actuadores.

Os actuadores foram desenvolvidos para cinco variáveis de controlo da aeronave, ou seja, flaps, *throttle*, leme de profundidade, ailerons e leme de direcção, e os diagramas de blocos podem ser vistos na figura 3.19.

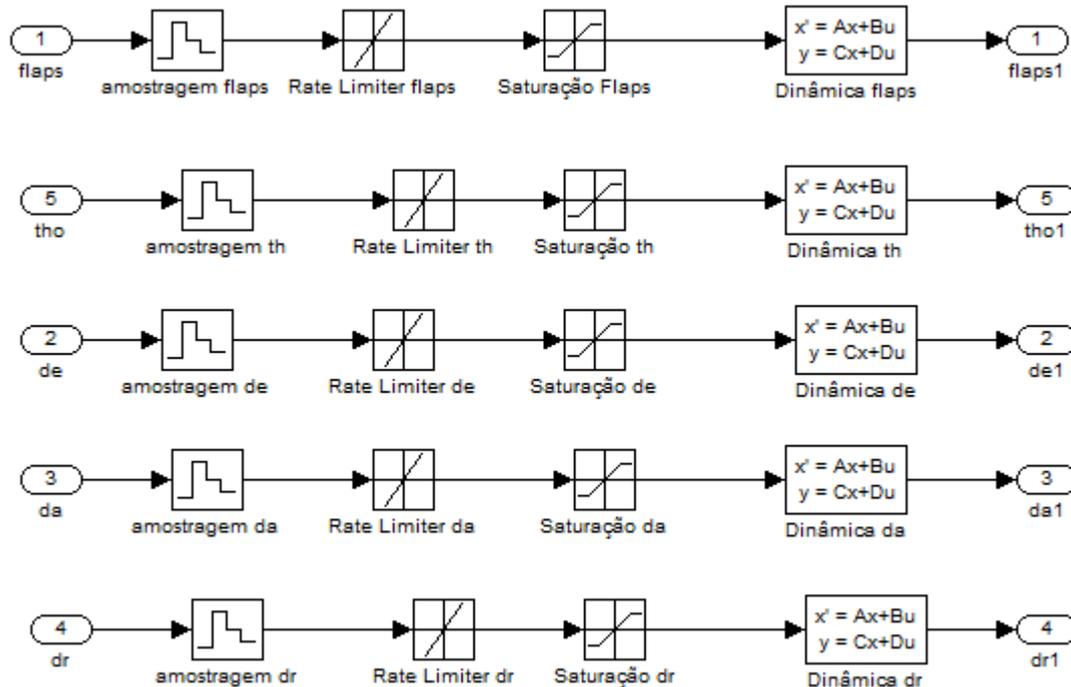


Figura 3.19 – Actuadores para as superfícies de controlo.

Cada actuador é composto por quatro blocos. O primeiro bloco definido como **amostragem** define a frequência do actuador. O bloco **Rate Limiter** define as velocidades máxima e mínima permitidas para o movimento do actuador. O terceiro bloco, denominado de **Saturação**, limita o valor máximo e mínimo que a posição do actuador pode ter. Por último o bloco **Dinâmica** define o modelo linear do movimento do actuador, onde é possível também introduzir a posição inicial do actuador. Estes parâmetros são definidos no *Script* inicial. No Anexo C encontra-se o ficheiro *Modelo.m* onde estão definidos estes valores.

De momento, ao invés dos actuadores, não foram implementados modelos realistas dos sensores, estando estes projectados como ideais. No entanto, o módulo de interface com o piloto já desenvolvido [10], possui um modelo do receptor ILS.

3.4. Controlo de Voo

Em controlo, pode definir-se a aeronave como um sistema em que a entrada é a acção de controlo e a saída é o movimento ou a trajectória [37]:

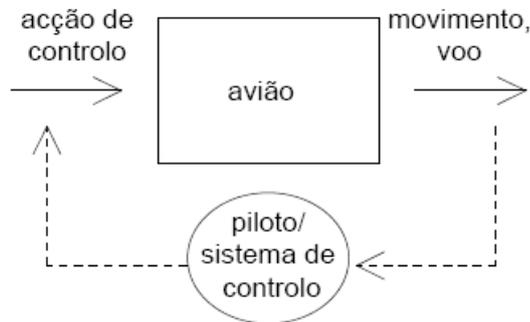


Figura 3.20 – Aeronave como sistema.

O controlo de uma aeronave é conseguido através da presença de:

- Actuadores, meios de controlo para aplicar forças de forma a gerar e corrigir o movimento da aeronave.
- Sensores para fornecerem a informação sobre o voo, ou seja, o estado da aeronave.
- Um piloto ou um sistema de controlo que feche o anel, de maneira a que com base na informação recebida através dos sensores, actue e controle a aeronave de forma a cumprir os objectivos específicos do voo.

Para analisar e desenvolver o controlo de uma aeronave é usual definir as equações do movimento da aeronave, linearizadas em torno de um ponto de equilíbrio em voo rectilíneo nivelado, sob a forma de espaço de estados, ou seja:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (3.54)$$

Onde x é o vector de estado, u é o vector de controlo (entradas) e A, B, C, D são matrizes constantes que contêm as derivadas de estabilidade da aeronave, invariantes no tempo.

O controlador escolhido, segundo a teoria de controlo moderno, foi o *Linear Quadratic Regulator*, LQR. É um controlador caracterizado pela sua fácil implementação. O LQR resume-se a uma função de custo que pode ter variadíssimas formas, correspondendo a situações e objectivos diferentes. Este método baseia-se na optimização e minimização da função de custo do sistema, J , feita em conjunto com a equação da dinâmica, \dot{x} , de onde se obtém o valor que minimiza u^0 e fornece a solução linear óptima para o ganho K da realimentação pretendida.

A função de custo do sistema enunciada atrás pode ser definida por:

$$J = \frac{1}{2} \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (3.55)$$

Onde as matrizes Q e R são quadradas e simétricas. Q é semi-definida positiva e R é definida positiva, as quais podem ser funções de tempo ou não e permitem ponderar estados e entradas de forma diferente.

A resolução da função de custo conduz a uma solução óptima linear dada pela seguinte equação:

$$u^0 = -Kx \quad (3.56)$$

As equações deste modelo podem ser analisadas com maior detalhe em [37].

A solução do LQR é conseguida através do programa Matlab com as matrizes A, B, Q e R, gerando uma solução da matriz K conhecida como matriz de ganho. Esta ganho é colocado de formar a realimentar o estado para fechar o anel e ao mesmo tempo estabilizar o sistema.

As matrizes Q e R são designadas por matrizes de ponderação dos estados e entradas. Estas matrizes diagonais foram aqui obtidas através do método de Bryson. Segundo este método a matriz Q é determinada através do inverso do quadrado de um valor máximo e considerado aceitável para os estados. A matriz R é calculada de forma idêntica à matriz Q, ou seja, através do inverso do quadrado de um valor máximo e considerado aceitável para as entradas.

Pode-se assim regular a velocidade da resposta das variáveis do sistema através do aumento ou diminuição destes valores ponderados.

Para calcular as matrizes A e B, é utilizada uma função fornecida pelo Aerosim denominada por **TRIM**. O cálculo do modelo linear é então conseguido sabendo as condições de TRIM assim como os estados das variáveis. O modelo não linear da aeronave é linearizado iterativamente usando uma função existente no ficheiro *Modelo.m*, no Anexo C. Como esperado, o resultado do modelo linear é desacoplado em dois movimentos, o longitudinal e lateral.

O movimento longitudinal apresenta o vector de estados e entradas da seguinte forma:

$$x = [u \ w \ q \ \theta \ h \ \omega]^T \quad (3.57)$$

$$u = [\delta_e \ \delta_T]^T \quad (3.58)$$

Onde u é a velocidade longitudinal, w é a velocidade vertical, q é a razão de picada, θ é o ângulo de picada, h é a altitude, ω é a velocidade angular do motor, δ_e é deflexão do leme de profundidade e δ_T é o *Throttle*.

Como o motor é muito rápido e estabiliza facilmente, foi eliminada a última entrada do vector de estados, ou seja, a velocidade angular do motor (ω). Por conseguinte as matrizes A e B foram modificadas ficando apenas com a dimensão para os cinco estados, mas com o efeito do ω nas outras entradas. O vector de entradas manteve-se o mesmo, no entanto o vector de estados passou a ser representado por:

$$x = [u \ w \ q \ \theta \ h]^T \quad (3.59)$$

Estas manipulações matemáticas encontram-se no ficheiro *Modelo.m*, no Anexo C.

O movimento lateral apresenta os seguintes vectores de estados e entradas:

$$x = [v \ p \ r \ \phi \ \psi]^T \quad (3.60)$$

$$u = [\delta_a \ \delta_r]^T \quad (3.61)$$

Onde v é a velocidade lateral, p é a razão de rolamento, r é a razão de guinada, ϕ é o ângulo de rolamento e ψ é ângulo de guinada, δ_a é deflexão dos ailerons e δ_r é a deflexão do leme de direcção.

Este modo não foi alterado, dado que a matriz de estados já apresenta os cinco estados pretendidos.

Após a linearização do sistema assim como o cálculo das matrizes A e B, calculou-se o ganho através do controlador óptimo **LQR**. Com este ganho foi possível fechar o anel do sistema e projectar os pilotos automáticos.

Na figura 3.21 observa-se um esquema geral do método utilizado para a realização dos pilotos automáticos. Este esquema é implementado tanto para o controlo longitudinal como para o controlo lateral e permite que o modelo siga uma entrada de referência, entrada esta que depende do tipo de piloto automático desejado.

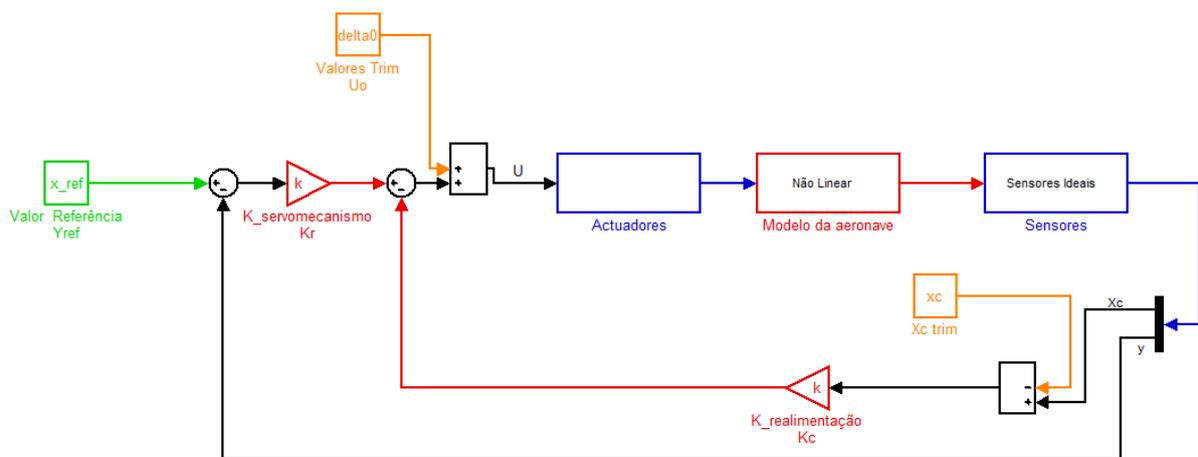


Figura 3.21 – Sistema de controlo genérico para os pilotos automáticos.

O sistema de controlo para os pilotos automáticos é um sistema de realimentação em servomecanismo. O sistema de servomecanismo, anel exterior, actua sobre o erro entre o valor efectivo da saída e o valor de referência desejado. No sistema de controlo por realimentação, anel interior, as saídas do sistema são controladas pela acção da matriz K de realimentação e são subtraídas à entrada de referência de modo a que permita controlar a saída do sistema e aumente a estabilidade. A equação deste modelo é definida da seguinte forma:

$$U = U_0 - K_r(y - y_{ref}) - K_c(x_c - x_c^{trim}) \quad (3.62)$$

A identificação das variáveis está de acordo com a figura 3.21.

É de salientar, que neste controlo são subtraídas às saídas os parâmetros iniciais calculadas pelo Trim, de forma a que o sistema de controlo actue apenas no valor do estado. No final da realimentação são somados os valores dos parâmetros iniciais calculados pelo Trim, obtendo o vector de entradas U para realimentar o sistema, como se pode analisar através da figura 3.21.

O sistema de controlo em anel fechado garante um funcionamento estável com um bom seguimento entre a entrada de referência e a saída, de modo a que se consiga uma dinâmica rápida e correcta na resposta aos comandos do piloto ou no seguimento das referências.

3.5. Pilotos Automáticos

O controlo da aeronave é obtido por um sistema de controlo de realimentação em servomecanismo através dos ganhos K , obtidos pelo LQR, como referido anteriormente. Este controlo é dividido em dois controlos diferentes, sendo um correspondente ao controlo longitudinal e outro ao controlo lateral da aeronave.

Como os controladores são implementados de forma independente, em seguida realiza-se uma análise detalhada de cada controlador.

3.5.1. Controlo Longitudinal

No controlo longitudinal os estados a controlar em servomecanismo foram a velocidade e a altitude. Para uma determinada velocidade ou altitude de referência a aeronave segue esses valores através do controlo do leme de profundidade e do *Throttle*, dando as entradas necessárias para o sistema convergir para essas velocidades e altitudes de referência. Os outros estados do movimento longitudinal são controlados por uma realimentação no anel interior do sistema através do ganho correspondente para esses estados.

Na figura 3.22 observa-se o controlador desenvolvido para o controlo longitudinal da aeronave. Pode visualizar-se o controlo de realimentação assim como o controlo em servomecanismo para as entradas de referência pretendidas. Neste tipo de controlo é feito o controlo através do erro, calculado entre a diferença do valor da saída do sistema não linear e o valor de referência que se pretende. O resultado final deste controlador são as saídas para o sistema não linear da aeronave da deflexão do leme de profundidade e do valor do *Throttle*.

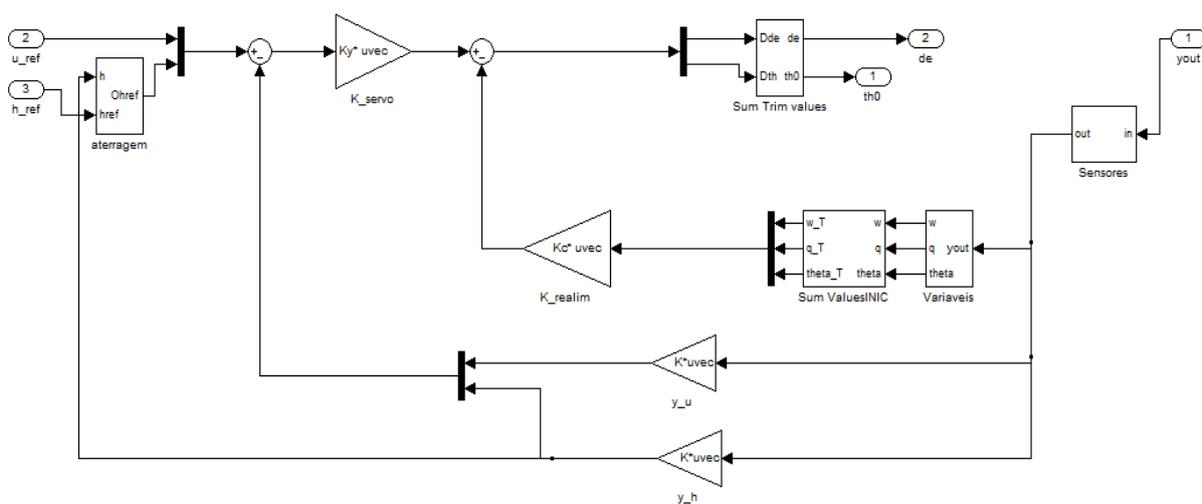


Figura 3.22 – Piloto automático de controlo longitudinal.

Como indicado anteriormente, o bloco dos sensores apenas está implementado de forma representativa, ou seja, são sensores ideais, sendo um dos trabalhos a desenvolver em futuros projectos do SVI.

O modelo matemático que representa este controlador é dado por:

$$\begin{bmatrix} \delta_E \\ \delta_T \end{bmatrix} = \begin{bmatrix} \delta_E^\circ \\ \delta_T^\circ \end{bmatrix} - K_y \begin{bmatrix} u - u_{ref} \\ h - h_{ref} \end{bmatrix} - K_c \begin{bmatrix} w - w_0 \\ q - q_0 \\ \theta - \theta_0 \end{bmatrix} \quad (3.63)$$

3.5.2. Controlo Lateral

O controlo lateral é efectuado da mesma forma que o controlo longitudinal, no entanto neste caso o estado a controlar é o rumo da aeronave. Para um determinado rumo de referência imposto ela segue esse rumo, em volta coordenada, controlando a deflexão dos ailerons e do leme de direcção. Os outros estados, tal como no sistema longitudinal, são realimentados pelo anel interior.

Na figura 3.23 esquematiza-se o controlo lateral da aeronave:

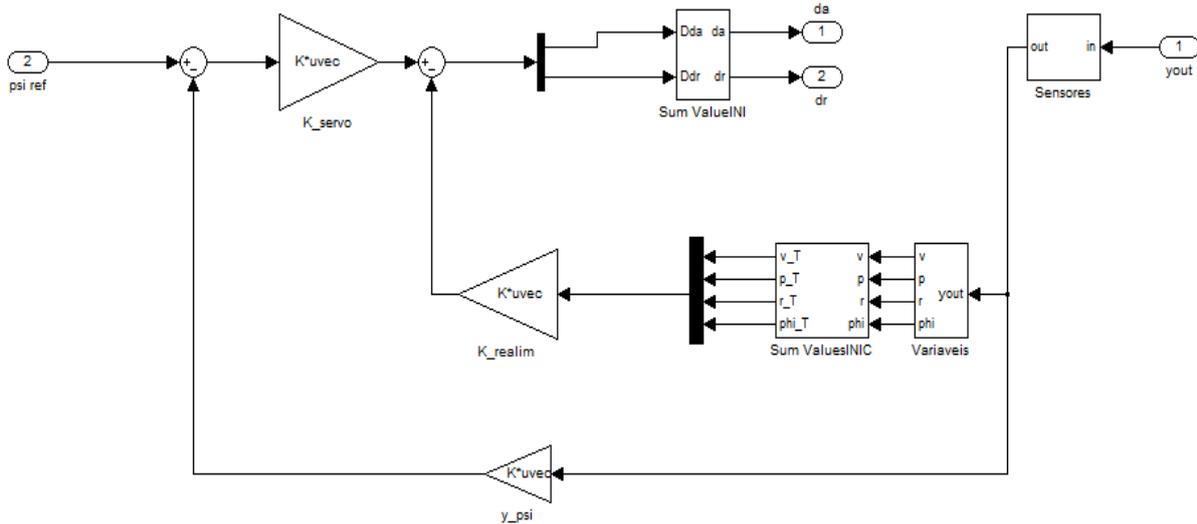


Figura 3.23 – Piloto automático de controlo lateral.

O valor do ângulo de rumo, λ , é dado pela soma do ângulo de guinada, ψ , com o ângulo de derrapagem, β . Caso se pretenda volta coordenada, o ângulo de derrapagem, β , tem que ser reduzido a zero, ou seja, o valor de referência a ser introduzido nessa entrada é uma constante igual a zero. Sendo assim, o ângulo de guinada (ψ) fornece uma referência para o rumo a ser seguido pela aeronave.

O modelo matemático que representa este controlador é dado por:

$$\begin{bmatrix} \delta_A \\ \delta_R \end{bmatrix} = \begin{bmatrix} \delta_A^\circ \\ \delta_R^\circ \end{bmatrix} - K_y (\psi - \psi_{ref}) - K_c \begin{bmatrix} v - v_0 \\ p - p_0 \\ r - r_0 \\ \phi - \phi_0 \end{bmatrix} \quad (3.64)$$

3.5.3. Aterragem Automática

O controlo longitudinal tem um controlo de aterragem automática para que a aeronave consiga aterrar no chão e assim testar a operacionalidade do modelo da dinâmica, mais concretamente o bloco do trem de aterragem.

O processo de aproximação e aterragem é constituído por três fases. Numa primeira fase o avião encontra-se em voo estável horizontal até atingir o ponto onde principia a descida. Nesta altura inicia-se a segunda fase, que se baseia no seguimento da ladeira com um ângulo de descida de aproximadamente 3° . A última fase do processo resume-se ao arredondamento, onde a aeronave realiza o ajuste necessário ao seu ângulo de descida para que o trem de aterragem traseiro toque primeiro na pista e posteriormente o trem dianteiro. Normalmente, nesta última fase é necessário que se faça tender o ângulo de descida para zero.

Para atingir os objectivos acima descritos é feito um controlo de forma a controlar a altitude da aeronave, de modo a que a sua razão de descida seja a pretendida e o ângulo de descida o desejado. O controlo é feito em servomecanismo, à entrada de referência da altitude, de modo a que a aeronave efectue a descida até à pista de aterragem. O controlo da velocidade é também realizado neste controlo, para que a velocidade diminua e que a aeronave consiga parar na pista de aterragem.

O controlo lateral é feito de modo a que a aeronave se alinhe com o rumo da pista e posteriormente é efectuada a descida com esse rumo de referência regulado.

4. Integração da Dinâmica no SVI

A integração do módulo da dinâmica da aeronave foi realizada por fases. Todo o software foi desenvolvido com recurso ao programa MATLAB 7.1, recorrendo a uma ferramenta muito importante chamada Simulink. Para a integração da dinâmica no SVI foi necessário, numa primeira fase, implementar a comunicação entre todos os módulos. Após esta fase, foi então desenvolvido o módulo de imagem para a simulação do ambiente exterior, de forma a tornar o simulador mais realista, e o modo do *joystick* para receber os comandos do piloto e controlar a aeronave.

De seguida são descritos todos os passos desenvolvidos ao longo da integração do módulo da dinâmica da aeronave no SVI.

4.1. Modo de Comunicação

Para a realizar o módulo final do SVI é necessário proceder à interligação dos computadores através de um *Hub* e fazer a comunicação entre eles usando uma ligação *100 Base-Tx STP Ethernet*, entre cada um dos computadores. Esta ligação *Ethernet* faz uso do *User Datagram Protocol* (UDP), uma vez que o UDP é um mecanismo simples de comunicação processo a processo, baseado em pacotes de dados. Os pacotes são enviados preferencialmente a uma frequência de 100Hz e para garantir a transmissão de dados assim como o funcionamento do SVI, esta transmissão deverá ocorrer de forma contínua, caso contrário ocorrerá erro de transmissão.

A comunicação processa-se em modo *master-slave* (mestre-escravo). O envio e a recepção de dados ocorrem assincronamente. A figura 4.1 mostra como os computadores do SVI estão interligados.

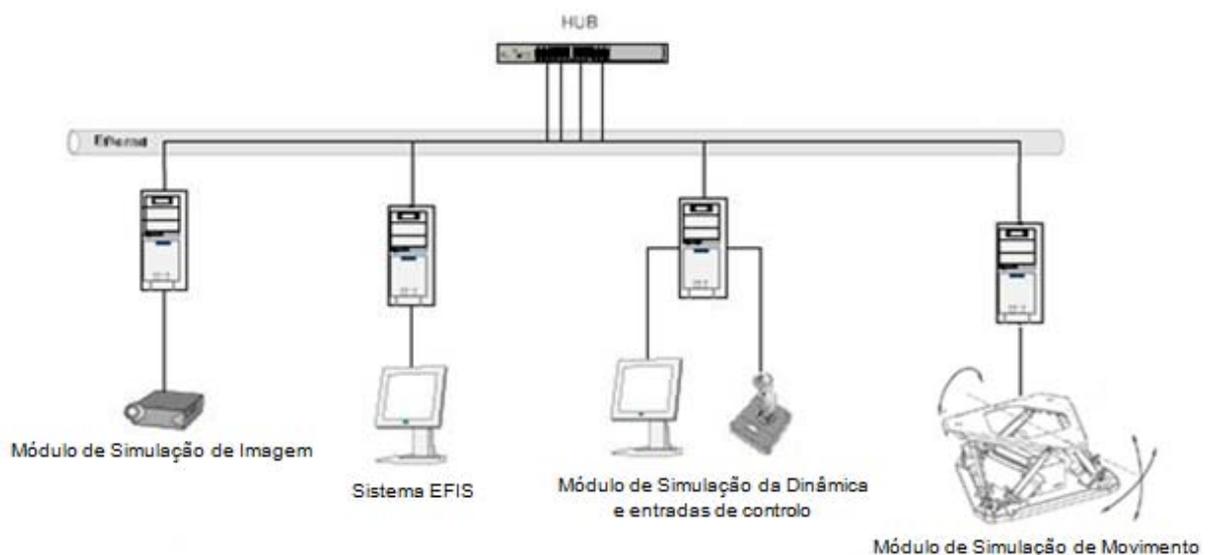


Figura 4.1 – Esquema do sistema de interligação entre os computadores do SVI.

O computador central escolhido foi o computador onde está o módulo da dinâmica. Este computador é responsável por enviar os dados da dinâmica da aeronave para cada módulo, para que depois estes dados sejam tratados devidamente por cada módulo.

Na figura 4.2 está representado o módulo central implementado para o SVI. O computador central é responsável pelo cálculo de toda a dinâmica da aeronave, através do bloco **Módulo da Dinâmica**. O bloco **Pilotos Automáticos e Joystick** é responsável pelos pilotos automáticos assim como pelas entradas dos comandos do *Joystick*.

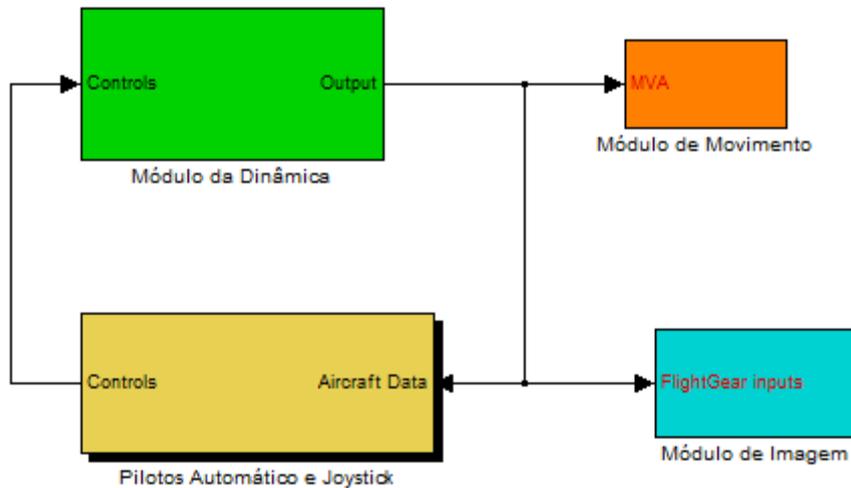


Figura 4.2 – Módulo Central do SVI.

O módulo central do SVI, é responsável por enviar os parâmetros da dinâmica da aeronave para os outros três módulos, ou seja, para o módulo de movimento do simulador através do bloco **Módulo de Movimento**, para o módulo de simulação visual do simulador pelo bloco **Módulo de Imagem** e para o sistema EFIS através de um bloco inserido no bloco **Piloto Automático e Joystick**.

O fluxo de informação é proveniente essencialmente do módulo da dinâmica da aeronave para os restantes módulos, como se pode observar na figura anterior, pois este módulo processa toda a informação relativamente ao estado da aeronave garantido assim o funcionamento dos restantes módulos. Seguidamente são explicados os tipos de dados enviados para cada módulo.

4.1.1. Módulo do movimento

Este módulo envia os dados provenientes da dinâmica da aeronave para o computador onde está desenvolvido o módulo de simulação do movimento. Os dados enviados para este módulo são as velocidades angulares, as acelerações angulares, as velocidades lineares e os ângulos de Euler.

O módulo de simulação de movimento recebe estes dados e trata-os através do algoritmo de simulação do movimento [9] para que a dinâmica da aeronave seja traduzida em movimentos da plataforma utilizada pelo SVI.

Na figura 4.3 pode observar-se o esquema utilizado para o envio de dados.

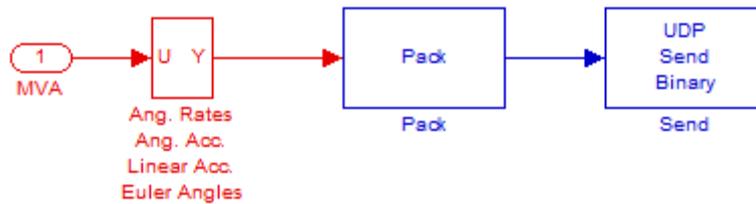


Figura 4.3 – Módulo de envio de dados para o módulo do Movimento.

Os dados são inicialmente selecionados por um bloco, de forma a que só sejam enviados os dados necessários para este módulo e de seguida passam por um bloco denominado *Pack* para os compilar. Os dados são compilados em formato *Double* para que o dado a enviar seja tratado como um número real, ou seja, com parte inteira e fracionária. O bloco *Send* é o bloco que estabelece o protocolo UDP e que envia os dados. É neste bloco que se define o IP de destino, a porta de destino e a frequência de envio. No Anexo G encontram-se os parâmetros utilizados no bloco *Send*.

Realiza-se também uma configuração de IP nos computadores do SVI para que o IP de cada computador esteja predefinido. Por conseguinte, impõe-se uma base, sendo só alterado o último dígito que corresponde ao número do computador, ou seja, a base é 192.168.1.x sendo x o número do computador em questão.

O algoritmo de recepção de dados, no módulo de movimento, é de forma semelhante ao de envio mas neste caso é pela ordem inversa, ou seja, os dados são recebidos através de um bloco *Receive* semelhante ao utilizado no envio de dados, apresentado no Anexo G. Neste bloco de recepção de dados define-se o IP de origem, a porta de origem, o número de *bytes* de cada quadro a receber e a frequência de recepção que deve ser igual à de envio. O bloco *Unpack*, que é o bloco seguinte, será o responsável por descompilar os dados, ficando os dados após este bloco iguais aos dados enviados, no que diz respeito ao formato.

Na figura 4.4 pode observar-se o modelo de recepção implementado no módulo de simulação de movimento:

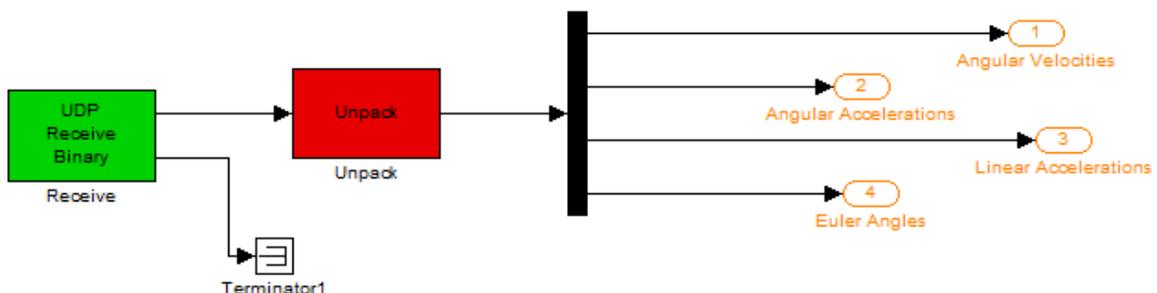


Figura 4.4 – Módulo de recepção de dados para o módulo do Movimento.

4.1.2. Módulo de simulação de imagem

O módulo de simulação visual envia a informação relativa à posição e orientação da aeronave para o FlightGear [40] para simular o ambiente externo da aeronave. O envio de dados é semelhante ao apresentado anteriormente. Este módulo pertence ao módulo imagem por conseguinte irá ser tratado na secção 4.2.

4.1.3. Módulo de interface com o piloto

Este último módulo é o único que tem comunicação bidireccional com o módulo da dinâmica. Recebe os dados necessários para serem apresentados nos indicadores e envia os comandos para controlar a simulação através das entradas do módulo da dinâmica, que correspondem às deflexões das superfícies de controlo da aeronave e de controlo do motor. Este bloco também envia os dados para o desenvolvimento dos pilotos automáticos da aeronave.

Os dados da dinâmica são enviados para este módulo para serem tratados no bloco de sistemas de navegação (*Navigation Systems*). Posteriormente são enviados os dados referentes às ajudas de navegação, dos estados dos pilotos automáticos e dos parâmetros da dinâmica da aeronave para o bloco *EFIS Data*, bloco este que faz o tratamento dos dados e os envia para o sistema de navegação EFIS [10] para se poder criar os instrumentos utilizados pelo piloto.

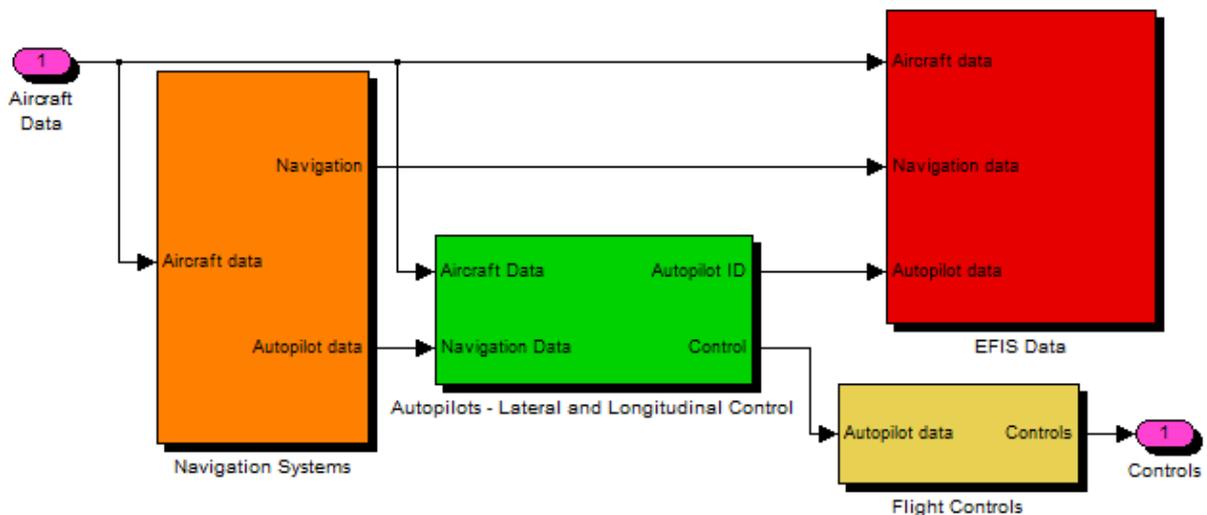


Figura 4.5 – Módulo de interface com o piloto.

Os dados são enviados recorrendo ao bloco *To Instrument da Instrument Control Toolbox* que permite o envio de dados através de uma conexão UDP, enviando datagramas para um computador com um determinado IP e porta específica. Neste bloco define-se também a frequência de envio dos dados e o tamanho do buffer, usando-se respectivamente 100Hz e 1024bytes para garantir a rapidez e total envio dos dados. No Anexo G pode consultar-se os parâmetro definidos para este bloco.

4.2. Módulo de Simulação de Imagem

O módulo de simulação de imagem permite a criação de imagem para o simulador, por conseguinte este módulo revela alguma importância visto ser o responsável pelo ambiente exterior produzido, para dar a sensação de voo ao piloto. Para a criação deste ambiente utiliza-se o FlightGear [40] visto consistir num projecto aberto que permite modificações para o melhoramento do cenário a simular. Um dos desejos principais deste projecto, é persuadir a comunidade de utilizadores de forma a aumentar e melhorar as capacidades do simulador, o FlightGear oferece uma vasta gama de áreas de trabalho.

Para se gerar a imagem, o módulo central do SVI envia os dados para o **Módulo de Imagem** e este por sua vez envia os dados para o FlightGear para a criação da mesma. A interface com o simulador FlightGear, incluída na biblioteca *Aerospace Blockset* do Simulink, é unidireccional para a transmissão do Simulink para o FlightGear usando o protocolo de troca de dados em binário *net_fdm* inserido no FlightGear. Os dados são transmitidos via UDP para o IP do computador responsável da geração de imagem, visto que, para melhores performances é recomendada a separação dos computadores entre o Matlab e as aplicações do FlightGear, uma vez que qualquer uma destas aplicações coloca uma carga significativa no CPU.

Na figura 4.6 mostra-se o esquema utilizado para o envio dos parâmetros necessários à geração de imagem. Como se pode observar os dados enviados são de posição e orientação da aeronave, ou seja, longitude, latitude, altitude e ângulos de Euler.

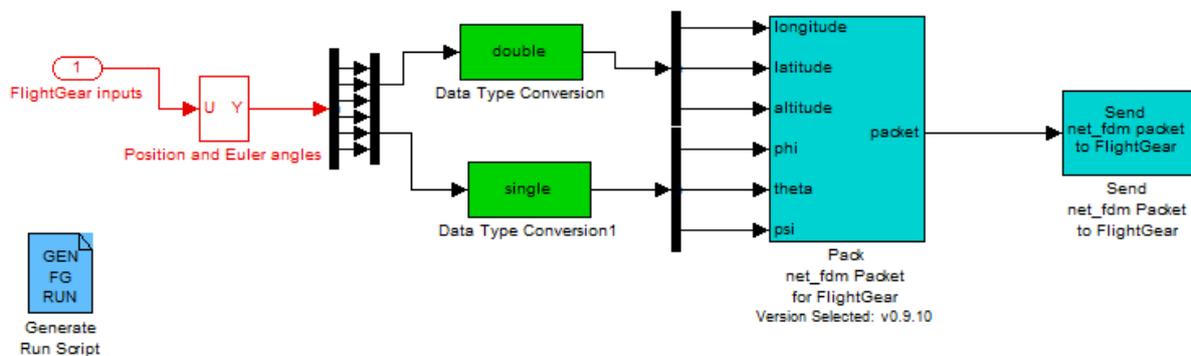


Figura 4.6 – Esquema responsável pelo envio de dados para o FlightGear.

O bloco *Generate Run Script* permite seleccionar as características do ambiente a simular, tais como aeroporto, número da pista e parâmetros de orientação da aeronave para simular de forma exacta a imagem correcta.

A imagem de inicialização do modo de simulação de imagem está modificada de modo a personalizar o SVI.

4.3. Comandos

Na implementação dos comandos tem-se especial cuidado em adotar comandos genéricos para que seja possível simular diversas aeronaves. Por isso, além de controlarem as diversas superfícies de controlo e parâmetros do motor têm que permitir também a sua parametrização, em ambiente Simulink, dependendo do tipo de aeronave que se esteja a simular.

Os comandos de uma aeronave podem ser divididos em dois grupos, os primários e os secundários [11]. Os comandos primários permitem controlar as principais superfícies de controlo (ailerons, leme de profundidade e leme de direcção) e também controlar a força propulsiva dos motores da aeronave. Os comandos secundários servem preferencialmente para otimizar o desempenho da aeronave (flaps, slots e slats) e para libertar o piloto de excessivas forças de controlo durante o voo. Estes tipos de superfícies de controlo podem ser observadas na figura 4.7 e melhor estudadas em [12].



Figura 4.7 – Superfícies de controlo de uma aeronave.

O comando é implementado através do bloco *S-Function*, proveniente da biblioteca *Aerosim* [38]. Este bloco identifica *Joystick* e devolve as várias saídas de um *joystick* genérico USB compatível com o Windows. As saídas do bloco representam os eixos do *joystick* e os botões, que podem assumir o valor de um ou zero conforme estão a ser pressionados ou não, respectivamente.

Os eixos do *joystick* estão parametrizados no intervalo $[-1,1]$, por conseguinte é necessário multiplicar estas saídas por ganhos de maneira a que as deflexões das superfícies de controlo operem no intervalo adequado para a aeronave a simular, assim como o valor do *Throttle* varie no intervalo $[0,1]$. Esta parametrização do *joystick* está representada na figura 4.8.

No Anexo E encontra-se o *joystick* utilizado neste modelo, assim como as funções dos botões implementados.

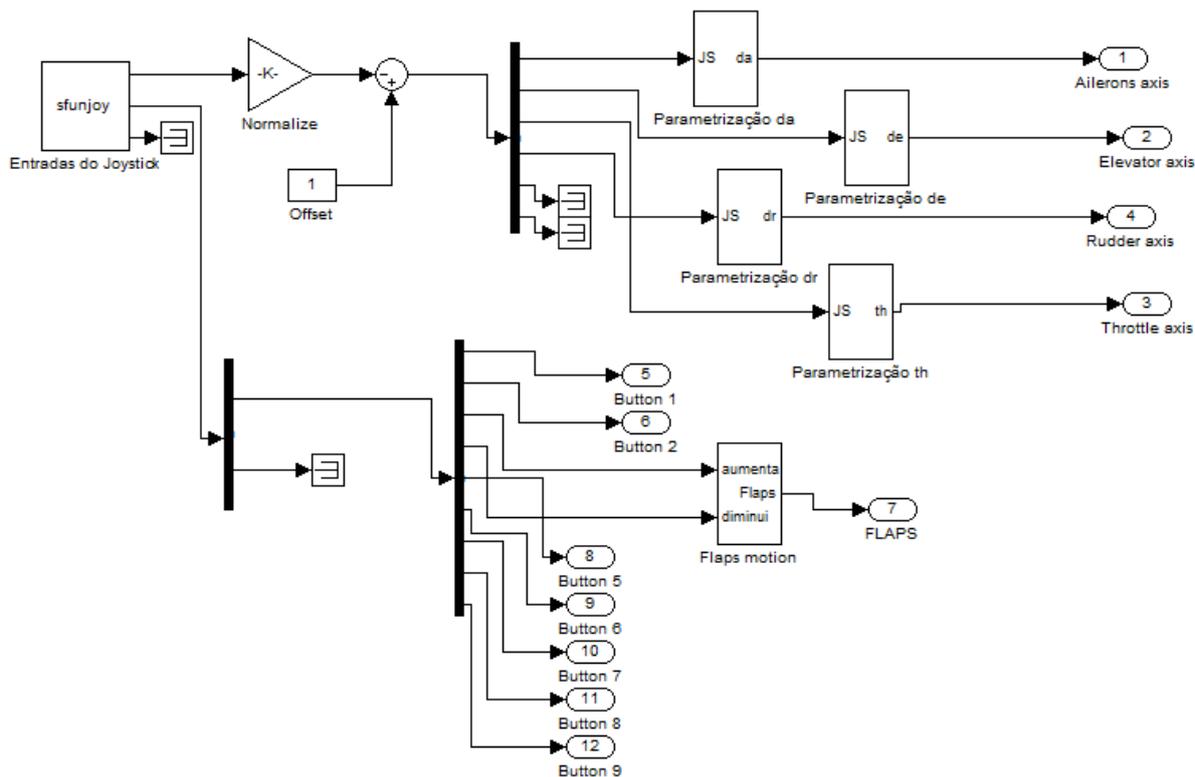


Figura 4.8 – Blocos de parametrização do *Joystick*.

A aeronave simulada é o *Navion* e são à partida definidos os valores das deflexões máximas das superfícies de controlo. Estes valores estão escritos na tabela 4.1.

A frequência de amostragem dos valores dos eixos do *joystick* assim como dos botões do mesmo é definida a 100Hz, de modo a garantir fluidez e segurança no envio de dados.

Superfícies de Controlo	Deflexão Mínima	Deflexão Máxima
Leme de profundidade	-30°	+20°
Leme de direcção	-30°	+30°
Ailerons	-30°	+30°
Flaps	0°	+50°
Throttle	0	1

Tabela 4.1 – Intervalo de variação das superfícies de controlo e *Throttle*.

O *joystick* apresenta sistema de quatro eixos variáveis continuamente, por conseguinte atribui-se a estes eixos os comandos principais da aeronave, ou seja, o leme de profundidade, o leme de direcção, os ailerons e o *Throttle*.

Como se pode verificar através da figura 4.8 os flaps são modelados de maneira a poderem variar continuamente no intervalo pretendido, apresentado na tabela 4.1. Este controlo dos flaps é conseguido através de botões do *joystick* uma vez que já não é possível implementá-los num eixo do mesmo. No Anexo E pode verificar-se o funcionamento desta superfície de controlo.

5. Avaliação Operacional do módulo

Neste capítulo são testados e avaliados todos os modelos implementados, ou seja, os pilotos automáticos, os modos de comunicação entre os outros módulos do SVI e o módulo de imagem.

Numa primeira fase são testados os pilotos automáticos do controlo da aeronave implementados na secção 3.5, efectuando o estudo dos diversos parâmetros relevantes da aeronave de forma a avaliar a operacionalidade dos mesmos. De seguida, são abordados os métodos para estabelecer e testar os modos de comunicação, de forma a validar todo o modelo desenvolvido. Por último é dada uma perspectiva final do SVI, indicando todas as potencialidades deste simulador.

5.1. Testes dos Pilotos Automáticos

Com os pilotos automáticos implementados é possível controlar o movimento longitudinal e lateral da aeronave. Através do movimento longitudinal controla-se a velocidade e a altitude, independentemente ou em simultâneo. O movimento lateral é utilizado apenas para controlar o rumo da aeronave.

Para o cálculo dos controladores K é necessário definir uma condição de voo da aeronave. Por conseguinte estabelece-se os parâmetros de voo iniciais ou de referência (Trim), apresentados na tabela 5.1.

Velocidade	Altitude	Phi	Massa de combustível	Flaps
50 (m/s)	200 (m)	0 (°)	200 (kg)	0 (°)

Tabela 5.1. – Parâmetros iniciais de voo da aeronave.

Para o movimento longitudinal a matriz K tem o seguinte valor:

$$K_{Long} = \begin{bmatrix} -0.0007 & 0.0301 & -0.3197 & -2.5323 & -0.0429 \\ 0.2099 & -0.0037 & 0.0655 & 0.7697 & 0.0453 \end{bmatrix} \quad (5.1)$$

As entradas desta matriz correspondem ao vector de estados, em que as colunas da matriz correspondem aos estados $[u \ w \ q \ \theta \ h]^T$ a realimentar, e por sua vez as duas linhas equivalem ao vector de entradas $[\delta_e \ \delta_r]^T$.

Para o movimento lateral a matriz K tem o valor:

$$K_{Lat} = \begin{bmatrix} 0.1120 & -2.1044 & -0.9721 & -0.5003 & -0.3308 \\ 0.3168 & 0.3324 & -2.4254 & 0.5425 & 0.0163 \end{bmatrix} \quad (5.2)$$

Para este movimento as colunas da matriz correspondem aos estados $[v \ p \ r \ \phi \ \psi]^T$, e as linhas correspondem respectivamente ao vector de entradas $[\delta_a \ \delta_r]^T$

Os pilotos automáticos implementados são conseguidos através de uma realimentação em servomecanismo, tal como é abordado no capítulo 3. Estes sistemas de controlo seguem uma dada referência imposta pelo piloto.

Para que o sistema estabilize são necessários dez segundos após o início da simulação, por conseguinte, só são impostos ao sistema entradas de referência após este tempo de estabilização. Nas figuras 5.1 e 5.2 pode observar-se a estabilização da altitude e da velocidade. Observa-se que o sistema estabiliza mais rapidamente para a altitude do que para a velocidade. Este tempo de estabilização maior da velocidade deve-se a um erro estático que pode ser anulado com um integrador, no entanto, visto ser um erro pouco significativo e como a estabilização ocorre num curto intervalo de tempo não há necessidade de aprofundar este assunto, podendo no entanto vir a ser aperfeiçoado num futuro trabalho do SVI.

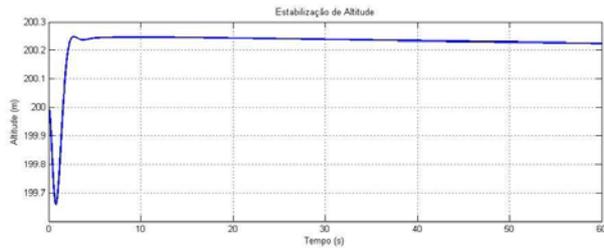


Figura 5.1 – Estabilização de altitude.

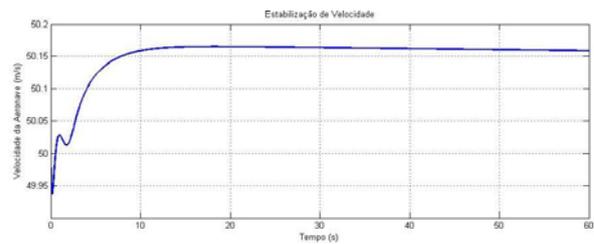


Figura 5.2 – Estabilização de velocidade.

Posteriormente são apresentados os testes efectuados para cada caso.

5.1.1 Seguimento de Altitude

Para o teste de seguimento de altitude simula-se um voo a diferentes altitudes, capaz de demonstrar que a aeronave consegue manter altitude e efectuar subidas e descidas para valores predefinidos. Este teste é realizado para uma velocidade constante de 50 m/s, embora nas subidas e descidas haja uma pequena variação do valor da velocidade. Na figura 5.3 observa-se um voo a diferentes altitudes predefinidas pelo piloto.

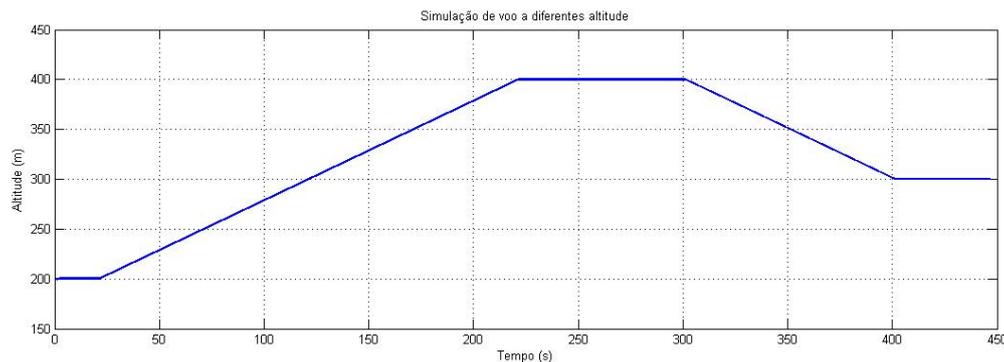


Figura 5.3 – Simulação de voo a diferentes altitudes.

Para este teste é imposto que a aeronave suba dos 200 m de altitude para os 400 m, de seguida mantenha esta altitude e posteriormente desça até aos 300 m, onde deve manter este valor até ao final da simulação. É imposta uma razão de subida para esta aeronave de maneira a que simulação seja o mais real possível. Por conseguinte a razão de subida imposta é de 120 metros por minuto, um valor comum para este tipo de aeronaves.

De seguida, pode observar-se na figura 5.4 a variação da velocidade ao longo do circuito vertical de mudança de altitude. A velocidade da aeronave é imposta, de forma a que se mantenha constante, e de valor igual a 50 m/s, no entanto, este valor sofre uns pequenos desvios. A velocidade decresce quando a aeronave inicia a subida, volta a aumentar para o valor estabilizado quando o voo fixa a uma dada altitude, subindo posteriormente quando a aeronave inicia a descida. É de salientar que a escala do gráfico em relação à velocidade da aeronave está num intervalo de valores muito pequeno de forma a permitir visualizar a mudança de velocidade. No entanto, é visível que a variação de velocidade, ao longo da simulação do voo a diferentes altitudes, é muito pouco significativa.

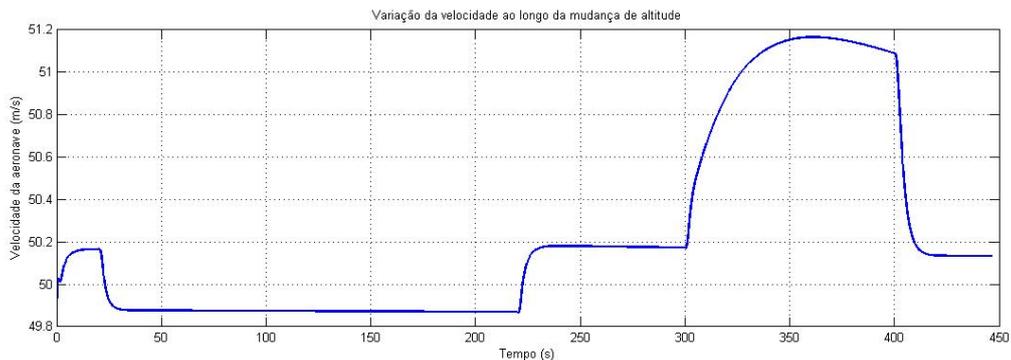


Figura 5.4 – Variação da velocidade ao longo do voo a diferentes altitudes.

As entradas para o controlo de altitude são o leme de profundidade e o *Throttle*, por conseguinte para a simulação deste voo a diferentes altitudes estas entradas variam de maneira a que a aeronave consiga efectuar o voo pretendido.

Na figura 5.5 pode observar-se a deflexão das superfícies de controlo para efectuar este tipo de manobra. A deflexão mais acentuada diz respeito ao leme de profundidade, como seria de esperar, já que a aeronave mantém rumo constante. A deflexão do leme de profundidade é efectuada no momento em que a aeronave muda de altitude, estabilizando depois aproximadamente no valor inicial. Sempre que a aeronave pretende atingir uma referência para a altitude, o valor da deflexão do leme de profundidade é alterado voltando depois a estabilizar.

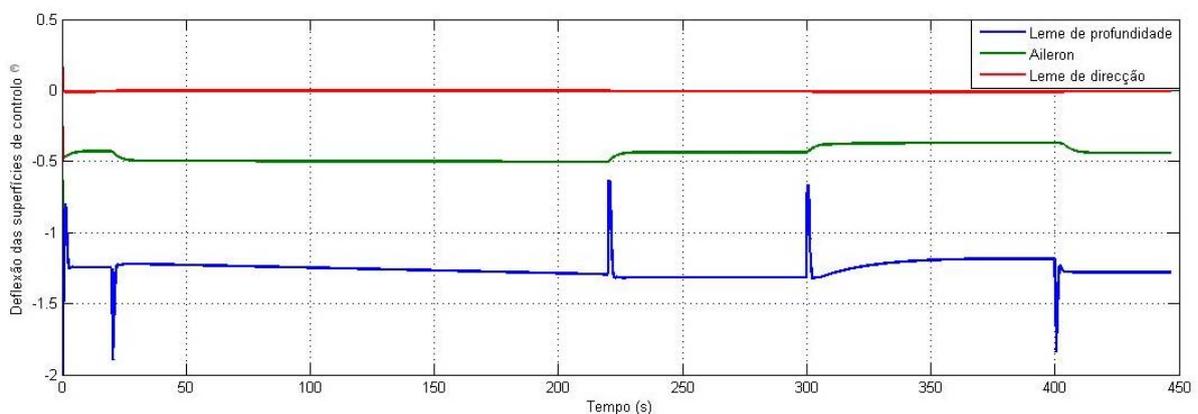


Figura 5.5 – Deflexão das superfícies de controlo ao longo do voo a diferentes altitudes.

Na figura 5.6 observa-se a variação de *Throttle* ao longo do circuito vertical. É fácil perceber a condição de voo para cada patamar do gráfico. É de salientar, que para efectuar a subida é necessária rotação do motor mais acentuada, o que corresponde a *Throttle* mais elevado e na fase da descida a aeronave tem *Throttle* na posição zero, o que equivale a motor em *Idle*. Quando a aeronave se fixa novamente a uma determinada altitude o valor do *Throttle* estabiliza num valor de maneira a que seja possível manter a aeronave a altitude e velocidade constante.

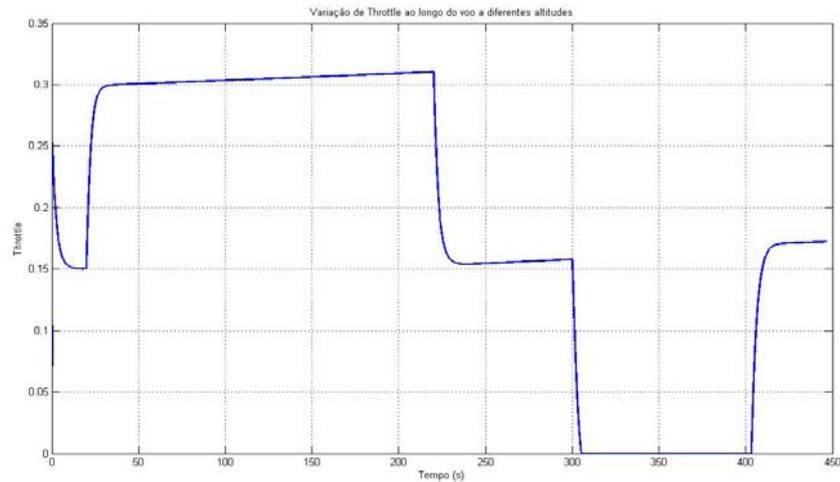


Figura 5.6 – Variação do *Throttle* ao longo do voo a diferentes altitudes.

Outros parâmetros podem ser estudados quando se efectua os testes de pilotos automáticos. No entanto, nesta secção só são referenciados os mais importantes para o estudo em questão, embora seja sempre possível efectuar outras análises, de outros parâmetros, consoante o tema ou o estudo em questão.

5.1.2 Seguimento de Velocidade

O seguimento de velocidade é testado de forma a que a aeronave siga um sinal de referência de velocidade. Este teste permite verificar que a aeronave é capaz de seguir uma determinada velocidade, assim como aumentar ou diminuir esse valor. Na figura 5.7 é possível visualizar a variação da velocidade da aeronave. Observa-se que a velocidade aumenta, até aproximadamente 56 m/s, mantendo-se depois constante nesse valor durante um determinado período de tempo. De seguida, a velocidade decresce até o valor de 46 m/s, aproximadamente, aumentando depois gradualmente até atingir os 50 m/s. Os valores de velocidades atingidos não são exactamente os pretendidos, ou seja, inicialmente o valor para o qual a velocidade deveria aumentar seria de 60 m/s, diminuindo depois até aos 45 m/s e de seguida aumentasse linearmente até aos 50m/s. É possível verificar que existe um pequeno erro inerente ao controlador. Este erro deve-se ao facto de não se ter efectuado um escalamento de ganhos para diferentes velocidades, ou seja, deveria ter-se calculado uma tabela de ganhos consoante as condições de voo da aeronave. Dado que o objectivo primordial desta dissertação é a integração e funcionamento do SVI, não se melhorou o escalamento de ganhos mas o mesmo deverá ser feito em estudos posteriores.

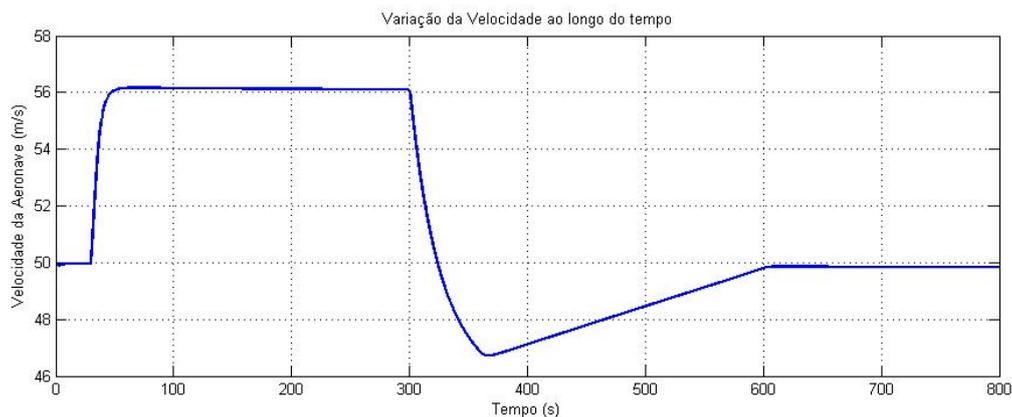


Figura 5.7 – Simulação de voo a diferentes velocidades.

Um aspecto importante neste teste que se deve ter em atenção é o facto de o motor saturar, ou seja, se os pedidos de seguimento de velocidade forem muito grandes a aeronave pode não atingir essas velocidades visto não possuir características suficientes. Por conseguinte, deve-se ter cuidado em escolher a velocidade, que a aeronave deve seguir, de modo a que a aeronave seja capaz de atingir esse valor de referência.

O seguimento da velocidade de referência, neste teste de piloto automático, é efectuado para altitude constante. No entanto, como se pode observar na figura 5.8 o valor de altitude não se mantém constante. Quando a aeronave varia de velocidade, a altitude tende a variar, no entanto pode visualizar-se na figura 5.8 que depois de estabelecida a velocidade de referência, a aeronave tende também estabilizar a sua altitude, para a referência pretendida. É de salientar que esta alteração de altitude é pouco significativa, como se pode observar na figura 5.8 em que a escala da altitude está definida utilizando um intervalo de valores muito pequeno, para tornar possível a observação da pequena alteração sofrida pela altitude.

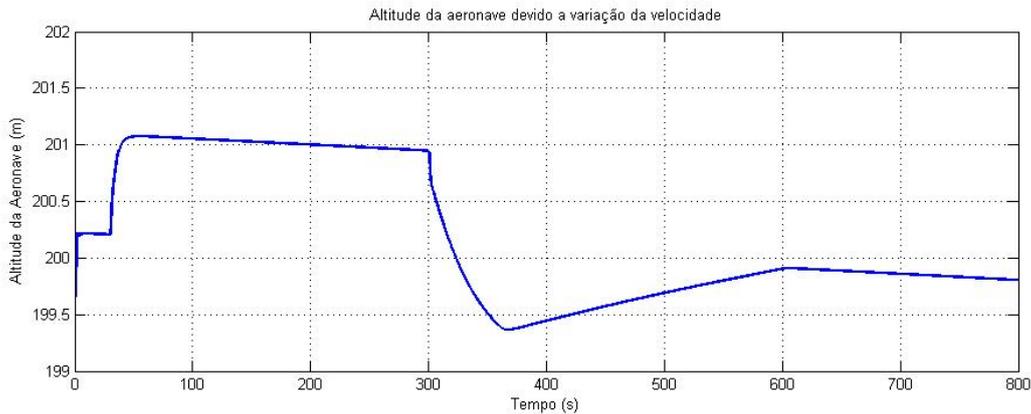


Figura 5.8 – Variação da altitude ao longo do voo a diferentes velocidades.

As deflexões das superfícies de controlo são visualizadas na figura 5.9. No movimento longitudinal do controlo de velocidade as entradas do sistema são o leme de profundidade e o *Throttle*, por conseguinte as outras superfícies de controlo sofrem pequenas deflexões, para manter e corrigir o rumo da aeronave.

Na figura 5.9 pode observar-se a deflexão das superfícies de controlo. Para este controlo interessa salientar a deflexão do leme de profundidade. Para o seguimento de velocidade a aeronave muda de altitude, por conseguinte para que se consiga o seguimento de altitude é necessária uma correcção no leme de profundidade.

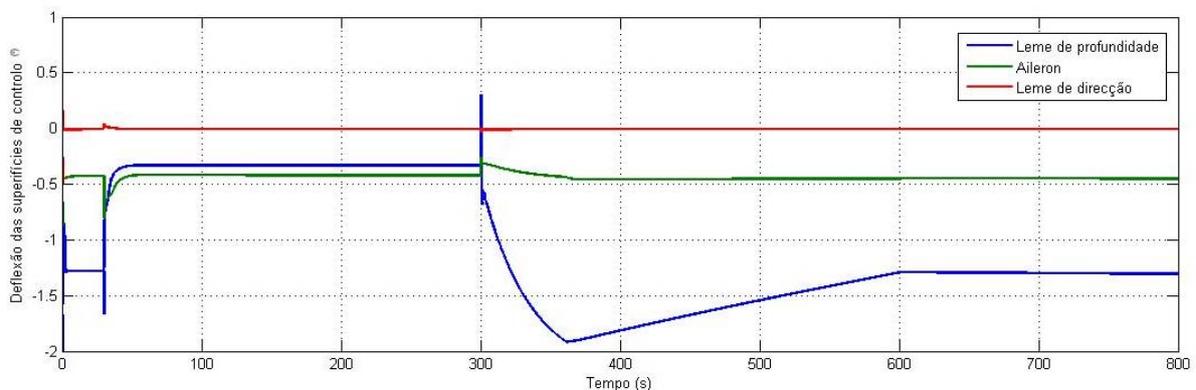


Figura 5.9 – Deflexão das superfícies de controlo ao longo do voo a diferentes velocidades.

É por causa da saturação, entre os 300s e os 360s, que o controlador não consegue regular a velocidade (figura 5.7), mas pode-se verificar que quando a saturação acaba o controlo fica de novo efectivo.

A variação do *Throttle* ao longo da simulação do seguimento de velocidade pode ser observada na figura 5.10. Como se pode notar o aumento de *Throttle* até ao limite máximo no início da simulação deve-se ao facto de se exigir um aumento de velocidade e por conseguinte um aumento da potência do motor. Após se atingir a velocidade pretendida o valor do *Throttle* decresce visto não ser necessário um valor de potência tão elevado, para manter a velocidade constante. O decréscimo de *Throttle* até ao valor mínimo é explicado pela diminuição de velocidade exigida na simulação.

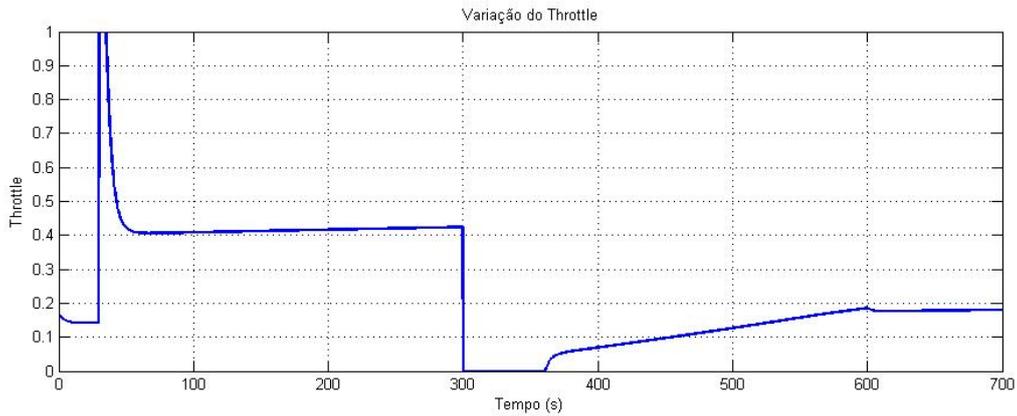


Figura 5.10 – Variação do *Throttle* ao longo do voo a diferentes velocidades.

5.1.3 Seguimento de Velocidade e de Altitude

É testado também o seguimento de altitude e de velocidade em simultâneo. Como é de esperar, os valores de velocidade e de altitude são atingidos com sucesso como se pode constatar nas figuras 5.11 e 5.12. No entanto, a estabilização da velocidade demora mais tempo, como é referenciado anteriormente.

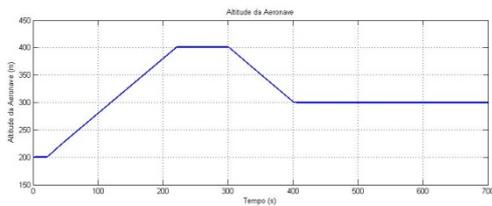


Figura 5.11 – Simulação de voo a diferentes altitudes.

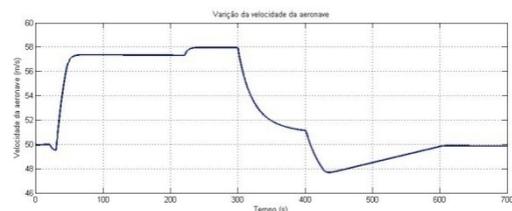


Figura 5.12 – Simulação de voo a diferentes velocidades.

Nas figuras 5.13 e 5.14 observa-se a deflexão das superfícies de controlo e a variação do *Throttle*, respectivamente. A variação do leme de profundidade é mais evidenciada para este controlo, sendo a variação de *Throttle* muito semelhante ao demonstrado na simulação de seguimento de velocidade.

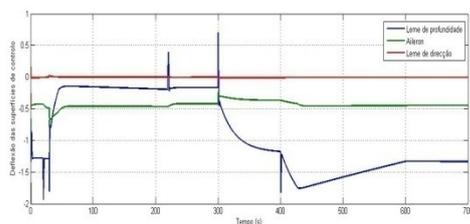


Figura 5.13 – Deflexão das superfícies de controlo.

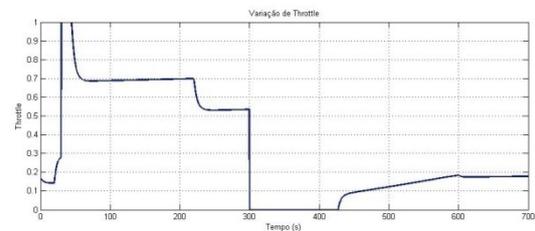


Figura 5.14 – Variação do *Throttle*.

5.1.4 Controlo de Rumo

Para o teste de controlo de rumo simula-se uma trajectória com diversos segmentos e mudanças de rumo. Esta simulação pode ser observada na figura 5.15.

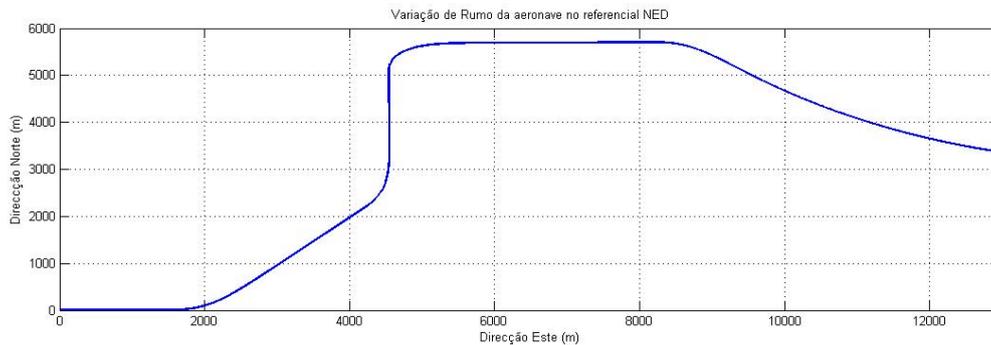


Figura 5.15 – Simulação de voo a diferentes rumos.

Na figura 5.15 está representada a trajectória da aeronave no referencial NED. Inicialmente o rumo da aeronave é de 90° , direcção Este, passando depois para Nordeste e para direcção Norte, ou seja 180° , e de seguida volta para a direcção Este. No final da simulação é feita uma mudança de rumo de forma contínua. Como se pode observar, o piloto automático implementado para o controlo de rumo é eficaz, fazendo com que a aeronave adquira o rumo pretendido para a simulação.

O controlo de rumo está implementado para o movimento lateral da aeronave, por conseguinte, as entradas que são realimentadas pelo sistema de controlo são o leme de direcção e os ailerons. Na figura 5.16 pode visualizar-se a deflexão destas superfícies de controlo durante a simulação do voo a diferentes rumos.

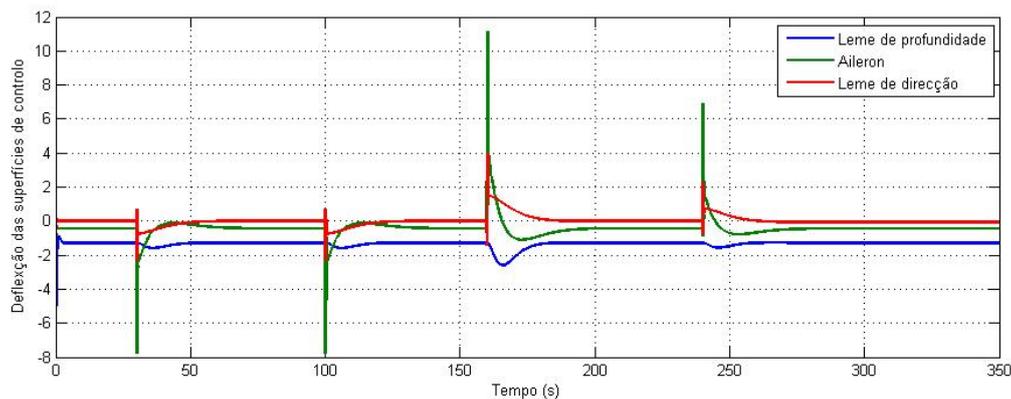


Figura 5.16 – Deflexão das superfícies de controlo ao longo do voo a diferentes rumos.

Como se pode observar através da figura 5.16, o leme de direcção e os ailerons sofrem grandes variações quando é necessário alterar o rumo, estabilizando posteriormente no valor de referência e seguindo o rumo pretendido.

O leme de profundidade como não é uma entrada no sistema para o controlo lateral, não sofre grandes variações como seria de esperar. Esta superfície de controlo apenas faz ajustes para seguir a velocidade e altitude de referência.

Poderiam ser testados mais pilotos automáticos, interligando os movimentos longitudinais e laterais da aeronave, no entanto, visto estes controlos automáticos estarem a corresponder ao pedido não se achou necessário fazer um estudo mais exaustivo para estes controlos automáticos de voo.

5.1.5 Aterragem Automática

Em seguida é testado o controlo de aterragem automática. Nesta aterragem efectua-se um teste de aproximação desde os 200 m de altitude até ao chão. É possível verificar que o trem toca na pista e de seguida a aeronave diminui a velocidade. A aproximação à pista é feita com um ângulo de descida, de aproximadamente 3° , seguindo-se uma fase de arredondamento, onde a aeronave reduz o ângulo de descida de modo que este tenda para zero para depois as rodas tocarem o chão.

Na figura 5.17 é possível observar a aterragem automática para testar este controlo da aeronave.

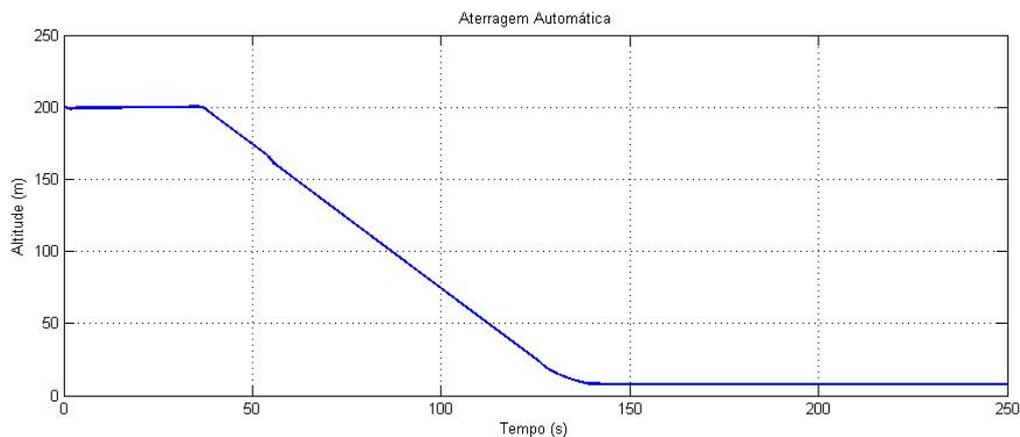


Figura 5.17 – Aterragem automática.

O controlo de aterragem automática está implementado para o movimento longitudinal. Esta aeronave faz uma aproximação à pista a 200 m de altitude com velocidade de 40 m/s, aumentando este valor de velocidade durante a descida, no entanto, a partir da fase de arredondamento a velocidade diminui até ser zero, como se pode observar na figura 5.18.

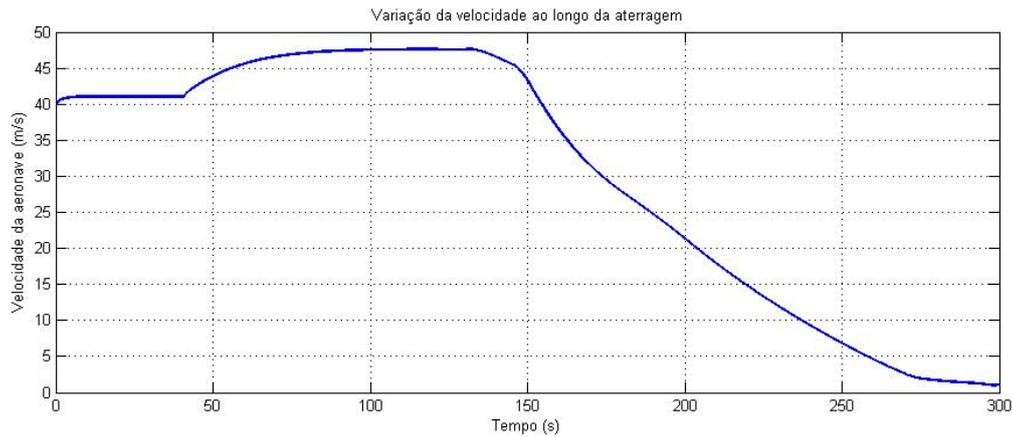


Figura 5.18 – Variação da velocidade ao longo da aterragem automática.

Outro parâmetro muito importante no estudo da aterragem automática é a deflexão dos amortecedores quando a aeronave toca no chão. Na figura 5.19 é possível visualizar essa deflexão.

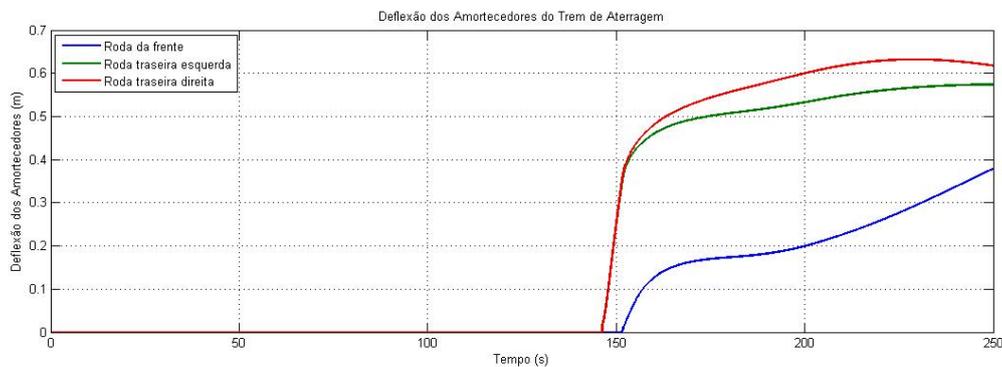


Figura 5.19 – Variação da deflexão dos amortecedores do trem.

Como se pode observar na figura 5.19, o trem traseiro da aeronave toca primeiro no chão e só passado algum tempo é que toca o trem do nariz da aeronave, como seria de esperar numa aterragem normal. Verifica-se também que a deflexão dos amortecedores é maior no trem traseiro em relação ao trem do nariz da aeronave devido às características da suspensão.

Quanto à variação de *Throttle* foi fácil verificar que se mantém no valor zero desde o início da descida até ao final da aterragem. Este modelo implementado para o motor não tem reversão do mesmo, o que poderia facilitar a travagem na pista de aterragem.

O rumo da aeronave é mantido constante desde o início da aterragem. Ou seja, durante a fase inicial a aeronave alinha-se com a pista e depois a partir do momento em que se inicia a descida a aeronave mantém constante o seu rumo.

Para além dos parâmetros estudados, podem ser ainda consultados outros parâmetros da aeronave. Apesar de neste trabalho não haver um estudo pormenorizado de todos os parâmetros, de forma a não tornar extremamente longa a presente dissertação, estes podem ser melhor analisados e avaliados a qualquer momento da simulação ou posteriormente.

5.2. Testes dos modos de Comunicação

A comunicação entre os módulos é conseguida através de um *Hub*, estabelecendo-se assim um protocolo UDP capaz de transmitir dados. Tendo em conta que o módulo da dinâmica da aeronave é o módulo central de todo o simulador, ele é também responsável pelo envio de todos os seus parâmetros para os vários módulos. Por conseguinte, é necessário testar a comunicação implementada.

Na figura 5.20 pode visualizar-se o *Hub* utilizado no SVI, com as respectivas ligações aos computadores.



Figura 5.20 – *Hub* utilizado no SVI.

Cada módulo está instalado num computador diferente para tornar possível a execução de todos os módulos em tempo próximo do real. Tendo em conta este aspecto, cada cabo de rede da figura 5.21 estabelece ligação a um computador diferente. É de salientar que a plataforma possui um computador interno, que comunica com o módulo do movimento, estando por isso ligado também através de um cabo de rede ao *Hub*. Esta comunicação pode ser vista com mais detalhe em [9].

A transmissão de dados entre os módulos de cada computador foi testada de modo a assegurar que todos os módulos recebam e transmitam todos os parâmetros desejados. Não são notados atrasos na recepção de dados nem a sua corrupção. A transmissão de dados é feita a 100Hz, como se refere anteriormente, velocidade suficiente para que os dados sejam transmitidos rapidamente e sem se verificarem atrasos.

O envio de dados foi efectuado com sucesso visto que todos os módulos funcionam correctamente e com todas as potencialidades operacionais. O módulo da simulação de movimento recebe os dados necessários para o algoritmo de movimento transmitindo depois ao computador da plataforma. A plataforma manteve sempre um movimento contínuo, pois a transmissão de dados era rápida e eficiente. Se o envio de dados pára, por qualquer razão, a plataforma pára e envia uma mensagem de erro ao seu computador. Ao longo dos testes do simulador só por razões previstas e previamente estabelecidas é que a plataforma parou.

O módulo de interface com o piloto recebe os dados transmitidos de forma correcta e eficiente, de modo a que o seu funcionamento seja íntegro. Na figura 5.21 visualiza-se o sistema EFIS do SVI implementado por outro módulo do simulador [10]. Os testes de comunicações deste módulo tiveram

bons resultados. Em várias fases da simulação, o processo é interrompido, propositadamente, e verifica-se que o último valor enviado pelo módulo da dinâmica é o valor exacto que o sistema EFIS apresenta nos ecrãs.



Figura 5.21 – Sistema EFIS do SVI.

A comunicação feita com o computador para a geração de imagem revela também bons resultados, ou seja, a imagem gera-se tal como se pretende e durante a simulação dos testes verifica-se que a imagem corresponde às condições de voo definidas.

A visualização da imagem gerada pode ser vista na figura 5.22.

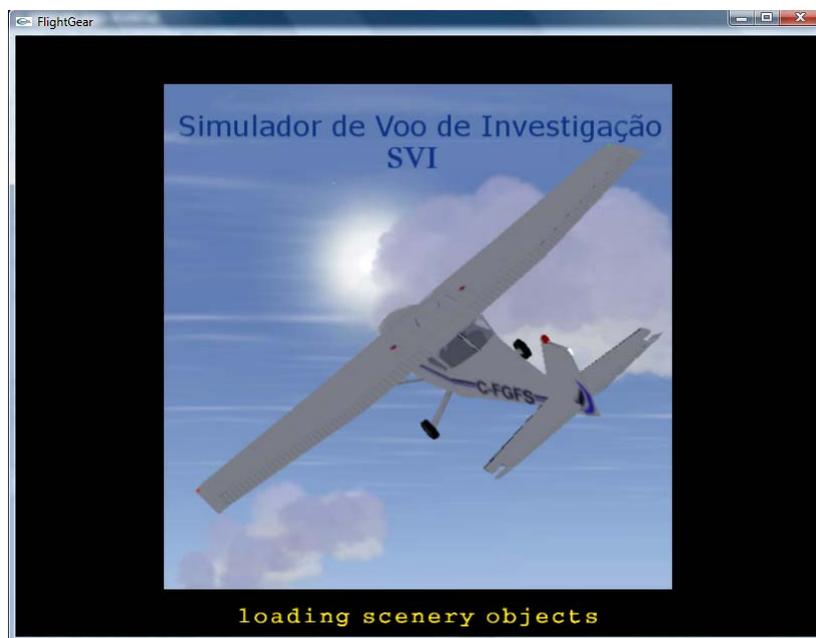


Figura 5.22 – Modo simulação de imagem.

5.3. Perspectiva Final do Simulador

Esta secção apresenta o estado actual do simulador, oferecendo imagens que permitem criar uma concepção do aspecto físico do SVI.

Para inicializar o simulador é necessário uma série de procedimentos que podem ser consultados no Anexo F. O utilizador pode utilizar uma interface gráfica de modo a alterar as condições de voo dos pilotos automáticos, sem ter de estar no ambiente Simulink. Esta interface gráfica permite controlar as constantes de navegação assim como os pilotos automáticos. Além das potencialidades de navegação que esta interface gráfica possui [10], ela também permite a escolha dos três pilotos automáticos concebidos, ligar ou desligar os comandos e efectuar um reset à simulação, voltando esta à posição inicialmente definida. Na figura 5.23 visualiza-se esta interface gráfica, relativamente aos pilotos automáticos e ao comando de controlo da aeronave, visto os outros parâmetros da interface gráfica já terem sido estudo de outra tese [10].

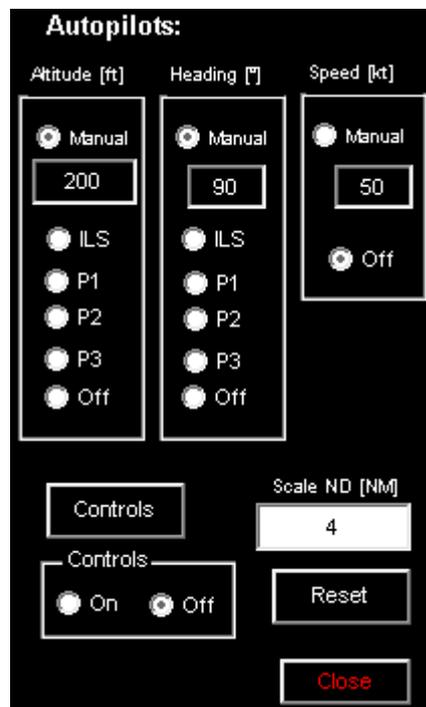


Figura 5.23 – Interface gráfica para a escolha dos pilotos automáticos e comando.

Neste momento o módulo da dinâmica da aeronave encontra-se interligado com os vários módulos, o que torna possível realizar testes de operacionalidade do SVI, usando todas as potencialidades desenvolvidas até este momento.

O simulador permite simular condições de voo específicas, definidas pelo piloto. Simultaneamente o piloto pode, a qualquer momento, interagir no simulador controlando a aeronave com as suas instruções.

As figuras 5.24 e 5.25 mostram o aspecto final do SVI na fase de testes e de simulações para os pilotos.

A figura 5.24 mostra a perspectiva do piloto durante a simulação do voo. É possível verificar que o piloto tem a imagem do cenário exterior gerada num LCD e consegue ao mesmo tempo visualizar os painéis de instrumentos do sistema EFIS. O utilizador do simulador consegue ainda controlar o módulo do movimento.

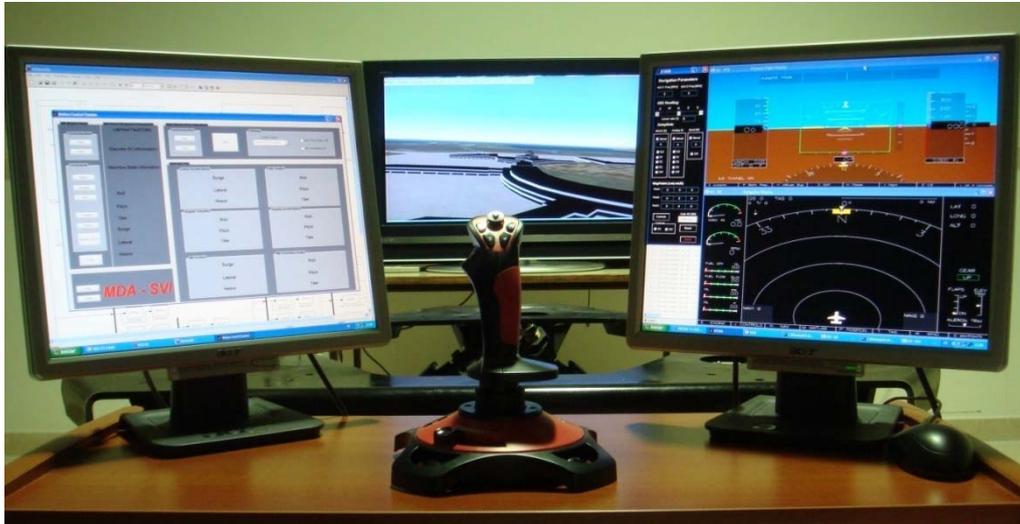


Figura 5.24 – Perspectiva do piloto no SVI.

Na figura 5.25 pode-se observar o SVI em simulação, onde se podem visualizar todos os módulos em funcionamento.



Figura 5.25 – Módulo final do SVI em simulação.

6. Conclusões

6.1. Conclusões

Esta dissertação desenvolve o módulo de controlo e de simulação da dinâmica de uma aeronave para a sua integração num simulador de voo. O objectivo proposto inicialmente é conseguido e permite que o SVI se encontre actualmente em estado operacional.

De modo a finalizar todo o projecto proposto inicialmente, este trabalho abrangeu áreas para além das estabelecidas no princípio do trabalho. Por conseguinte o módulo de controlo e da dinâmica da aeronave neste simulador fica como base para futuros trabalhos, dado que a sua abordagem é feita de modo geral e não específico.

Os ensaios experimentais realizados no âmbito da avaliação operacional de todo o módulo revelam-se muito satisfatórios. Comprova-se o adequado funcionamento dos pilotos automáticos implementados, assim como a interacção entre o GUI, de interface gráfica, e os respectivos comandos que o piloto transmite para o módulo da dinâmica.

A comunicação entre os módulos está estabelecida com êxito, estando neste momento todos os módulos do SVI interligados e operacionais.

O módulo da imagem foi gerado com sucesso, no entanto, dado que inicialmente este não era tema proposto para esta dissertação, este pode ser melhorado, tal como é explicado na secção 6.2.

Apesar do cockpit do SVI ainda não estar instalado na plataforma de movimento, este facto não invalidou a realização de testes. Para a realização dos testes finais contou-se com a ajuda de um piloto, segundo o qual a dinâmica se encontra adequada às características da aeronave, o que permite a simulação de um voo em condições muito próximas das reais. Este facto reforça positivamente as conclusões dos testes realizados, no que respeita à sua operacionalidade e funcionalidade do simulador.

Sendo o SVI um projecto desenvolvido para o IST, este poderá ser utilizado como base de trabalho e investigação. O trabalho desenvolvido até ao momento permite a operacionalidade do sistema, no entanto, há ainda muito trabalho até que o SVI tenha todas as suas potencialidades em funcionamento. Na secção 6.2 são apresentadas algumas sugestões para futuros trabalhos

Muito trabalho há ainda a ser desenvolvido no SVI a todos os níveis. No entanto o primeiro passo está dado e o sistema encontra-se operacional. Na secção seguinte são referidos alguns temas relevantes, directamente relacionados com o trabalho apresentado, com interesse para futuros desenvolvimentos.

6.2. Sugestões para trabalho futuro

Dado que o SVI tem como objectivo a criação de um instrumento de investigação e aplicação de conhecimentos na área de aeronáutica é importante dar continuidade ao projecto. Ao longo desta secção são mencionados assuntos que devem ser alvo de futuros trabalhos.

Relativamente ao módulo de imagem seria importante um maior desenvolvimento deste, de forma a tornar possível a projecção da imagem em vários ecrãs, conseguindo assim uma visão no interior do cockpit de 180°, dando ao piloto um cenário de voo mais próximo da realidade.

Apesar dos bons resultados obtidos nos testes dos pilotos automáticos, será sempre útil proceder ao seu desenvolvimento, nomeadamente realizar um escalamento de ganhos para várias condições de voo.

Um tema interessante para futuros trabalhos será o desenvolvimento de um módulo de turbulência capaz de gerar e simular condições ambientais extremas. Deste modo o piloto poderá testar as potencialidades da aeronave, assim como a sua agilidade para pilotar nestas condições.

Neste momento apenas está implementada a dinâmica de uma aeronave, o Navion. Seria de importância extrema a criação de uma base de dados com várias dinâmicas, de diferentes aeronaves, de modo a permitir ao piloto a escolha da aeronave pretendida.

Um ponto importante para o funcionamento do SVI é a simulação em tempo real do sistema. Neste momento, tal é conseguido através de um bloco *Set Pace*. Sabe-se, porém, que este bloco apenas o faz indirectamente, controlando o relógio de simulação do Simulink e sincronizando-o com o relógio do computador. Este facto induz erros e o sistema não funciona, exactamente, em tempo real, ou seja, não apresenta a precisão adequada. Este é um aspecto fundamental para a integridade do sistema. O estudo de meios de implementação que conduzam a um funcionamento em tempo real exacto é de muita importância.

Seria relevante o controlo de todos os módulos através de um computador central. Este computador controlaria todos os outros de módulos, de forma a iniciar a simulação sem recorrer aos computadores dos módulos do SVI. Tendo em conta este facto seria importante a criação de um Output neste computador, para que o utilizador possa seleccionar a aeronave a simular assim como todos os parâmetros importantes para a simulação do voo.

Todas estas funcionalidades seriam usadas com a máxima potencialidade após a integração do Cockpit. Este permitiria avaliar de forma mais realista as características do simulador assim como criar a sensação de voo real.

7. Bibliografia

- [1] *Control System Design* , Graham C. Godwin, Stefan F. Graebe, Mario E. Salgado, 2000, Prentice Hall; Bk&CD-Rom edition, ISBN: 0139586539.
- [2] *Dynamics of Flight - Stability and Control* , B. Etkin L. D. Reid, 1996, John Wiley & Sons.
- [3] *Flight Stability and Automatic Control*, 2nd ed., Robert C. Nelson, 1998, McGraw-Hill.
- [4] Hosman, R., Advani, S. and Haeck, N., *Integrated Design of Flight Simulator Motion Cueing Systems. Royal Aeronautical Society Conference of Flight Simulation*. 2002, London.
- [5] *Aircraft Control and Simulation, 2nd Edition* , Brian Stevens & Frank Lewis, 2003, Wiley-Interscience.
- [6] HJ.Farrell and M.Barth, *The Global Positioning System & Inertial Navigation*. McGraw Hill 1999.
- [7] Branco, P., *Introdução aos actuadores de superfícies de comando de voo*. Folhas da cadeira de Sistemas de Controlo de Superfícies de Voo, Instituto Superior Técnico, 2004.
- [8] MATLAB User's Guide. The Math Works Inc., 2007.
- [9] Vaz, L.M., *Desenvolvimento e Integração de um Módulo de Simulação de Movimento*. Master Thesis, Instituto Superior Técnico, 2007.
- [10] Alves, J.P., *Desenvolvimento e Integração do Módulo de Interface com o Piloto*. Master Thesis, Instituto Superior Técnico, 2008.
- [11] Costa, B., Fonseca A. Folhas da cadeira de Sistemas Aviónicos Integrados. Instituto Superior Técnico, 2007.
- [12] Brederode, V., *Fundamentos de Aerodinâmica Incompressível*. Edição do Autor, Instituto Superior Técnico, 1997.
- [13] Cook, E., *Flight simulation boundaries - barriers to progress or guidelines to success*. In *Proc. Flight Simulation - Expanding the boundaries*, pages 4.1–4.10. The Royal Aeronautical Society, 1997.

- [14] Kim, S.H., Shin, H.P. In, W.S. and Kim, J., *Design of a 6 D.O.F. Flight Simulator Based on a Full Spinning Parallel Mechanism Platform*. AIAA Modeling and Simulation Technologies Conference and Exhibit. 2005, AIAA-6105, San Francisco, California.
- [15] Stroosma, O., Paassen, R. and Mulder, M., *Using the SIMONA research simulator for human-machine interaction research*. AIAA Modeling and Simulation Technologies Conference and Exhibit. 2003, AIAA-5525, Austin, California.
- [16] Skans, N.S. and Barnes, A.G., *Fifty years of success and failure in flight simulation*. In *Proc. 50 Years of Flight Simulation*, pages 1–24. The Royal Aeronautical Society, 1979.
- [17] Telban, J.R. and Cardullo, F.M., *Motion Cueing Algorithm Development: Human-Centered Linear and Nonlinear Approaches*. 2005, NASA CR-213747, NASA Langley Research Center, Hampton, VA.
- [18] Koekebakker, S.H., *Model based control of a flight simulator motion system*. PhD thesis, Delft University of Technology, 2001.
- [19] NASA VMS, <http://www.frc.arc.nasa.gov/vms/vms.html>.
- [20] TU Delft SIMONA, <http://www.simona.tudelft.nl>.
- [21] Giovannetti, D.P., *NASE AMES Research Center Simulation Laboratories*. NASA Ames Research Center, 2003.
- [22] Conijn, C., *Developments in high quality motion systems for flight and driving simulators used in r&d and training*. In *CEAS Symposium on Simulation Technologies - Making it real*, pages MSy03/1–12, 1995.
- [23] Martin, E.A., *Motion and force simulation systems i, whole-body motion simulators*. *Flight Simulation Update*, 1995.
- [24] Advani, S.K., *The kinematic design of flight simulator motion-bases*. PhD thesis, Delft University of Technology, 1998.
- [25] Rodchenko, V.V., Zaichik, L.E., and Yashin, Y.P., *Motion cueing on research flight simulators*. In *CEAS Symposium on Simulation Technologies - Making it real*, pages MC02/1–12, 1995.
- [26] R.P.A.M. Teunissen. *Ab-initio zft: Creating pilots without wings*. In *Proc. Flight Simulation - Expanding the boundaries*, pages 13.1–13.8. The Royal Aeronautical Society, 1997.

- [27] Joint Aviation Requirements. JAR-STD 1A: Aeroplane Flight Simulators. 2001.
- [28] Hall, J.R., *Motion versus visual cues in piloted flight simulation*. In *Proc. AGARD Conf. on Flight Simulation*, pages 17.1–17.13, 1978.
- [29] Reid, L.D., *Technical evaluation report*. In *AGARD Conference on Flight Simulation – Where are the challenges*, pages T1–T9, 1996.
- [30] Anderson, S.B., *Historical review of piloted simulation at NASA AMES*. In *AGARD Conference on Flight Simulation - Where are the challenges*, pages 1.1–1.13, 1995.
- [31] Simulink User's Guide. The Math Works Inc., 2007.
- [32] Rogers R. M. *Applied Mathematics in Integrated Navigation Systems*. American Institute of Aeronautics and Astronautics, Inc., 2000. ISBN 1-56347-397-6.
- [33] <http://cddisa.gsfc.nasa.gov/926/egm96/egm96.html>.
- [34] <http://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml>.
- [35] Stevens B. L. and Lewis F. L., *Aircraft Control and Simulation*, John Wiley & Sons, Inc., 992. ISBN 0-471-61397-5.
- [36] Phillips W. F. Hailey C. E. and Gebert G. A. Review of Attitude Representations Used for Aircraft Kinematics. *Journal of Aircraft*, 38:718–737, 2001.
- [37] Azinheira, J. R., *Folhas da Cadeira de Controlo de Voo*, Lisboa, 2008.
- [38] AeroSim Blockset User's Guide, Unmanned Dynamics, Version 1.2, 2002.
<http://www.u-dynamics.com>.
- [39] Lau, F. J., *Folhas da Cadeira de Estabilidade de Voo*, Lisboa, 2006.
- [40] <http://www.flightgear.org>.

Anexos

Anexo A

A.1. Coeficientes de Inércia

Os coeficientes do tensor de inércias são dados por:

$$I_{xx} = \int_v (y^2 + z^2) \rho_A dV \quad (\text{A.1})$$

$$I_{yy} = \int_v (x^2 + z^2) \rho_A dV \quad (\text{A.2})$$

$$I_{zz} = \int_v (x^2 + y^2) \rho_A dV \quad (\text{A.3})$$

$$I_{xy} = \int_v xy \rho_A dV = I_{yx} \quad (\text{A.4})$$

$$I_{xz} = \int_v xz \rho_A dV = I_{zx} \quad (\text{A.5})$$

$$I_{yz} = \int_v yz \rho_A dV = I_{zy} \quad (\text{A.6})$$

Em aeronaves simétricas em relação ao plano XZ considera-se $I_{xy} = I_{yz} = 0$.

Os coeficientes de inércia calculados pelo bloco da dinâmica podem ser obtidos através das seguintes equações [35]:

$$c_1 = \frac{(I_{yy} - I_{zz})I_{zz} - I_{xz}^2}{\Gamma} \quad (\text{A.7})$$

$$c_2 = \frac{(I_{xx} - I_{yy} + I_{zz})I_{xz}}{\Gamma} \quad (\text{A.8})$$

$$c_3 = \frac{I_{zz}}{\Gamma} \quad (\text{A.9})$$

$$c_4 = \frac{I_{xz}}{\Gamma} \quad (\text{A.10})$$

$$c_5 = \frac{I_{zz} - I_{xx}}{I_{yy}} \quad (\text{A.11})$$

$$c_6 = \frac{I_{xz}}{I_{yy}} \quad (\text{A.12})$$

$$c_7 = \frac{1}{I_{yy}} \quad (\text{A.13})$$

$$c_8 = \frac{I_{xx}(I_{xx} - I_{yy}) + I_{xz}^2}{\Gamma} \quad (\text{A.14})$$

$$c_9 = \frac{I_{xx}}{\Gamma} \quad (\text{A.15})$$

Onde:

$$\Gamma = I_{xx}I_{zz} - I_{xz}^2.$$

A.2. Produto Externo

Seja o vector \vec{a} e \vec{b} definidos em componentes Cartesianas:

$$\vec{a} = a_x \vec{e}_x + a_y \vec{e}_y + a_z \vec{e}_z \quad (\text{A.16})$$

$$\vec{b} = b_x \vec{e}_x + b_y \vec{e}_y + b_z \vec{e}_z \quad (\text{A.17})$$

O produto externo é:

$$\vec{c} = \vec{a} \times \vec{b} = \begin{vmatrix} \vec{e}_x & \vec{e}_y & \vec{e}_z \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix} = \begin{vmatrix} a_y & a_z \\ b_y & b_z \end{vmatrix} \vec{e}_x + \begin{vmatrix} a_z & a_x \\ b_z & b_x \end{vmatrix} \vec{e}_y + \begin{vmatrix} a_x & a_y \\ b_x & b_y \end{vmatrix} \vec{e}_z \quad (\text{A.18})$$

$$\vec{c} = \vec{a} \times \vec{b} = (a_y b_z - a_z b_y) \vec{e}_x + (a_z b_x - a_x b_z) \vec{e}_y + (a_x b_y - a_y b_x) \vec{e}_z$$

O produto externo é dado pelo vector \vec{c} .

Anexo B

B.1. Eixos de Referência e Coordenadas

Ao longo do trabalho são utilizados maioritariamente dois referenciais. Neste anexo são sucintamente abordados, em conjunto como o sistema WGS84, recorrendo a [6].

B.1.1. Sistema Roll Pitch Yaw

O primeiro diz respeito ao referencial da aeronave RPY, da figura B.1. Neste referencial o eixo X aponta na direcção da aeronave, o Y encontra-se na asa direita e o Z aponta para baixo.

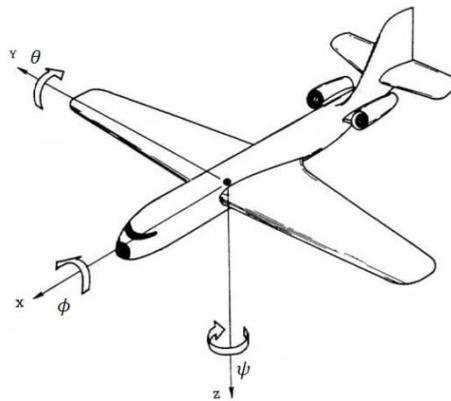


Figura B.1. – Referencial da aeronave RPY.

São definidos os ângulos de Euler em torno destes eixos. O ângulo de picada é dado por θ , o de rolamento por ϕ e o de guinada por ψ . Os sentidos dos ângulos são os concordantes com os da figura B.1. Dado que a trajectória da aeronave pode não coincidir com o seu eixo longitudinal definem-se o ângulo de ataque α e o ângulo de derrapagem β . Estes ângulos assim como os seus sentidos estão definidos na figura B.2.

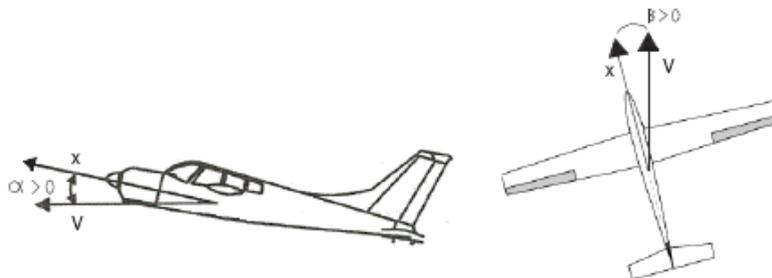


Figura B.2. – Ângulos de ataque e derrapagem.

B.1.2. Sistema North East Down

O outro referencial usado neste trabalho é o referencial NED. O eixo X aponta para Norte, o Y para Este e o Z para o centro da Terra, como é visível na figura B.3.

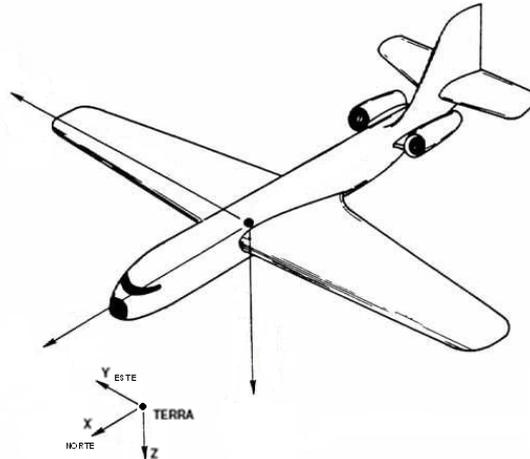


Figura B.3. – Referencial NED.

A passagem deste referencial para o RPY realiza-se através da sequência de rotações ψ , θ e ϕ em torno dos eixos respectivos. Assumindo que o eixo Z nos dois referenciais assume a mesma direcção, basta realizar uma translação.

B.1.3. Sistema WGS84

O modelo WGS84 define um elipsóide com o eixo maior a e um menor b , como na tabela B.1. As coordenadas de um ponto P à superfície dadas pela latitude λ , longitude ϕ , e altitude h . Os parâmetros da elipse e a sua excentricidade e são dados pelas expressões B.1 e B.2.

Êixo maior (a)	6378137.0 m
Êixo menor (b)	6356752.314245 m

Tabela B.1. – Parâmetros WGS84.

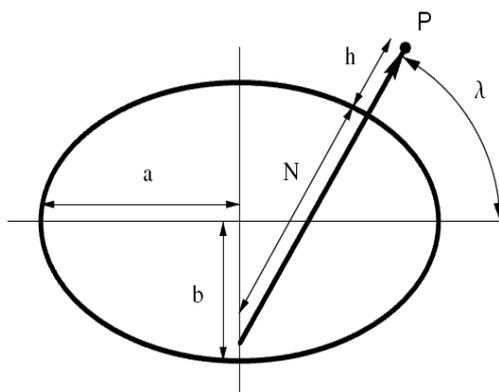


Figura B.4. – Modelo elipsóide WGS84.

$$f = \frac{a-b}{a} \quad (\text{B.1})$$

$$e = \sqrt{f(2-f)} \quad (\text{B.2})$$

B.1.4. Conversão de referenciais ECEF para NED

Recorrendo à excentricidade da elipse e à latitude do ponto consegue-se obter a distância normal à superfície da terra ao eixo z dada por $N(\lambda)$ em B.3.

$$N(\lambda) = \frac{a}{\sqrt{1-e^2 \sin^2(\lambda)}} \quad (\text{B.3})$$

Torna-se então possível saber o valor das coordenadas do ponto P no referencial cartesiano.

$$x = (h + N) \cos(\lambda) \cos(\phi) \quad (\text{B.4})$$

$$y = (h + N) \cos(\lambda) \sin(\phi) \quad (\text{B.5})$$

$$z = (h + (1 - e^2)N) \sin(\lambda) \quad (\text{B.6})$$

Após a transformação para coordenadas cartesianas, transforma-se estas para o que será o centro do referencial NED dado pelos pontos (x_0, y_0, z_0) em B.7.

$$x' = \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (\text{B.7})$$

Com a latitude λ e longitude ϕ basta realizar as respectivas rotações sobre os eixos a que estas correspondem e obtém-se A.9.

$$x'' = \begin{bmatrix} e \\ n \\ d \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\sin(\lambda) & \cos(\lambda) \\ 0 & -\cos(\lambda) & -\sin(\lambda) \end{bmatrix} \begin{bmatrix} -\sin(\phi) & \cos(\phi) & 0 \\ \cos(\phi) & \sin(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} x' \Leftrightarrow$$

$$x'' = \begin{bmatrix} -\sin(\phi) & \cos(\phi) & 0 \\ -\cos(\phi)\sin(\lambda) & -\sin(\lambda)\sin(\phi) & \cos(\lambda) \\ -\cos(\lambda)\cos(\phi) & -\cos(\lambda)\sin(\phi) & -\sin(\lambda) \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \quad (\text{B.8})$$

$$\begin{bmatrix} n \\ e \\ d \end{bmatrix} = \begin{bmatrix} -\cos(\phi)\sin(\lambda) & -\sin(\lambda)\sin(\phi) & \cos(\lambda) \\ -\sin(\phi) & \cos(\phi) & 0 \\ -\cos(\lambda)\cos(\phi) & -\cos(\lambda)\sin(\phi) & -\sin(\lambda) \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \quad (\text{B.9})$$

Anexo C

C.1. Ficheiros utilizados

Modelo.m

```
% dnavion_OL.m
% example open loop script to load NAVION data and launch simulink model
%
%original from:
% Navion GA airplane - sample model from AeroSim Library
% Copyright 2002 Unmanned Dynamics, LLC
% Revision: 1.0 Date: 05/13/2002
%
%adapted by:
% Jose Raul Azinheira, 01/09/2008
%Modified by:
% João Pedro Santos Jorge, 03/22/2008

global deg
deg=pi/180; % 1 deg in rad

%load model data
matfile='navioncfg.mat';
eval(['load ' matfile])

%initial conditions
%(these are mere examples to be adapted for specific case)
Velini=[40 0 0]';
pqrini=[0 0 0]';
quatini=[1 0 0 0]';
phiini=0;
thetaini=0;
psiini=117.9*deg;
quatini=euler2quat([phiini thetaini psiini]); quatini=quatini(:);
lat0=37.62*deg;lon0=-122.4*deg;alt0=0; %ref position to convert from LLA to
NED
altini=200;
LLAini=[lat0 lon0 altini]';
Fuelini=200;
omegaini=2400*pi/30;
altG=alt0-20; % ground altitude

% landing gear introduced to cope with landing and takeoff
% location is referred to Aerodynamic Center [xw2n+1/4*MAC 0 zw2n]
LGNlocation=[3 0 1/2]; LGLlocation=[-.2 -1.5 1]; LGRlocation=[-.2 1.5 1];
LGNstiffness=500*2*3;LGNdamping=1.42;
LGLstiffness=1500*2*3;LGLdamping=1.42;
LGRstiffness=1500*2*3;LGRdamping=1.42;
LGXfriction=.05*2.3; LGYfriction=.1/5;
LGNmaxForce=3000; LGLmaxForce=10000; LGRmaxForce=10000;

% added wind input (with turbulence) JRA 071228
windNED=[0 0 0]'; %adapt to specific case
simdate=clock;
```

```

%-----OUTPUTS
% vectors for simulink outputs ( "to workspace" blocks):
Toutput=.1; % output sampling rate
%
% tout: simulation time
% uout: flight control inputs
% yout: sensors outputs
%
% example to visualize after simulation: plot AOA and pitch in deg
% figure(1),plot(tout,yout*[y_alpha' y_theta']/deg)

ny=23;yv=1:ny;% yout=[uvw pqr NED Euler Vba LLA omega ground]
y_u=1==yv;y_v=2==yv;y_w=3==yv;
y_p=4==yv;y_q=5==yv;y_r=6==yv;
y_N=7==yv;y_E=8==yv;y_D=9==yv;
y_phi=10==yv;y_theta=11==yv;y_psi=12==yv;
y_Vt=13==yv;y_beta=14==yv;y_alpha=15==yv;
y_lat=16==yv;y_lon=17==yv;y_alt=18==yv;
y_om=19==yv;y_mass=20==yv;
y_LGN=21==yv;y_LGL=22==yv;y_LGR=23==yv;
y_h=-y_D;

%-----
%simulation duration (to adapt according to needs)
tsim=12000;
Ts=.01; %control sampling rate

%-----LAUNCH
%launch simulink model
%navion_OL
SVI
EFISND

%_____TRIM_____

global TrimParam

Vt00=40;h00=200;ff0=0;mF0=100;flap0=20*deg;
%Vt00=30;h00=100;ff0=0;mF0=100;flap0=50*deg;

[Alon,Blon,Clon,Alat,Blat,Clat,...

de0,da0,dr0,th0,u0,v0,w0,p0,q0,r0,phi0,theta0,psi0,alt0,fuel0,om0,as0,beta0
,alpha0]...
    =ftrim_navion_1(Vt00,h00,ff0,mF0,flap0)

%__ Eliminar o estado omega do motor_____
ablon=[Alon Blon];
omega=-ablon(6,[1:5 7:8])/ablon(6,6);
alon=Alon(1:5,1:5)+Alon(1:5,6)*omega(1:5);
blon=Blon(1:5,1:2)+Alon(1:5,6)*omega(6:7);

Q=diag([10 1 10 1 10]);
R=diag([1/(5*deg)^2;1/(0.5)^2]);

%_____LQR_____
R=40*R;

```

```

K=lqr(alon,blon,Q,R);

damp(alon)
damp(alon-blon*K)

C=eye(5);
Cy=[C(1,:);C(5,:)];
Cc=[C(2,:);C(3,:);C(4,:)];
Kc=[K(:,2), K(:,3), K(:,4)];
Ky=[K(:,1), K(:,5)];
D=zeros(5,2);

damp(Alat)
Klat=lqr(Alat,Blat,diag(1./[.1 2*deg 5*deg 15*deg
15*deg].^2),diag(1./[5*deg 2*deg].^2));
damp(Alat-Blat*Klat)

CyLat=[C(5,:)];
CcLat=[C(1,:);C(2,:);C(3,:);C(4,:)];
KcLat=[K(:,1), K(:,2), K(:,3), K(:,4)];
KyLat=[K(:,5)];

alt0=0;

#####ACTUADORES#####
%_____ variaveis dos blocos dos actuadores_____
Ts_act=0.01;
Vde=1.5;
Vda=1.5;
Vtho=1.5;
Vdr=1.5;
Vflaps=0.5;

identidade=eye(4);
Tflaps=0.5;
Tcontrols=0.02;
Kflaps=1;
Kcontrols=1;

Flapmax=50*deg;
Flapmin=0*deg;
thomax=1;
thomin=0;
demax=20*deg;
demin=-30*deg;
damax=30*deg;
damin=-30*deg;
drmax=30*deg;
drmin=-30*deg;

```

LLA2NED.m

```
function [n,e,d]=lla2ned(refLat, refLon, refH, lat,lon,h)

% function [n,e,d]=lla2ned(refLat, refLon, refH, lat,lon,h)
% or
% function ned=lla2ned([refLat, refLon, refH, lat,lon,h])

if nargin==1;
    u=refLat;refLat=u(1);refLon=u(2);refH=u(3);lat=u(4);lon=u(5);h=u(6);
end

[X,Y,Z] = lla2ecef(lat,lon, h);
[Xr,Yr,Zr] = lla2ecef(refLat,refLon, refH); % location of reference point
[e,n,u] = ecef2enu(Xr, Yr, Zr, X, Y, Z);
d=-u;
if nargin==1;n=[n e d]';end
return

function [X,Y,Z] = lla2ecef(lat,long, h)

% function [X,Y,Z] = lla2ecef(lat,long, h)
%
% Convert lat, long, height in WGS84 to ECEF X,Y,Z
%
% from Geodetic system, in Wikipedia, the free encyclopedia
%
http://en.wikipedia.org/wiki/Geodetic\_system#From\_geodetic\_coordinates\_to\_l
ocal_ENU_coordinates
%
% JRA070908

a = 6378137.0; % earth semimajor axis in meters
f = 1/298.257223563; % reciprocal flattening
e2 = 2*f -f^2; % eccentricity squared

chi = sqrt(1-e2*(sin(lat)).^2);
X = (a./chi +h).*cos(lat).*cos(long);
Y = (a./chi +h).*cos(lat).*sin(long);
Z = (a*(1-e2)./chi + h).*sin(lat);

return

function [e,n,u] = ecef2enu(Xr, Yr, Zr, X, Y, Z)

% function [e,n,u] = ecef2enu(Xr, Yr, Zr, X, Y, Z)
%
% convert ECEF coordinates to local east, north, up
%
% from Geodetic system, in Wikipedia, the free encyclopedia
%
http://en.wikipedia.org/wiki/Geodetic\_system#From\_geodetic\_coordinates\_to\_l
ocal_ENU_coordinates
%
% JRA070908

phiP = atan2(Zr,sqrt(Xr^2 + Yr^2));
lambda = atan2(Yr,Xr);
```

```
e = -sin(lambda).*(X-Xr) + cos(lambda).*(Y-Yr);
n = -sin(phiP).*cos(lambda).*(X-Xr) - sin(phiP).*sin(lambda).*(Y-Yr) +
cos(phiP).*(Z-Zr);
u = cos(phiP).*cos(lambda).*(X-Xr) + cos(phiP).*sin(lambda).*(Y-Yr) +
sin(phiP).*(Z-Zr);

return
```

Anexo D

D.1. Aeronave Simulada

Os parâmetros utilizados para configurar a dinâmica da aeronave simulada estão presentes no ficheiro *navioncfg.mat*. Este ficheiro contém a informação do *Script* definido inicialmente. De seguida apresenta-se este ficheiro onde se pode verificar todos os parâmetros da aeronave simulada.

```
%%% AIRCRAFT CONFIGURATION SCRIPT %%%
%%% Navion GA airplane - sample model from AeroSim Library %%%

% Copyright 2002 Unmanned Dynamics, LLC
% Revision: 1.0 Date: 05/13/2002

% corrected VaBnd for landing 071228
% added flaps JRA 071229

% Clear workspace
clear all;

% Name of the MAT-file that will be generated
cfgmatfile = 'navioncfg';

%%% AERODYNAMICS %%%
% Aerodynamic force application point (usually the aerodynamic center)[x y
z]
rAC = [0 0 0]; % m

%%% Aerodynamic parameter bounds %%%
% Airspeed bounds
%VaBnd = [30 100]; % m/s
VaBnd = [0 100]; % m/s 071228 for ground case
% Sideslip angle bounds
BetaBnd = [-0.5 0.5]; % rad
% Angle of attack bounds
AlphaBnd = [-0.1 0.5]; % rad

%%% Aerodynamic reference parameters %%%
% Mean aerodynamic chord
MAC = 1.73736; % m
% Wind span
b = 10.18032; % m
% Wing area
S = 184*0.3048^2; % m^2

% ALL aerodynamics derivatives are per radian:
%%% Lift coefficient %%%
% Zero-alpha lift
CL0 = 0.3;
% alpha derivative
CLa = 4.44;
% Lift control (flap) derivative
CLdf = 0;
CLdf = 0.7; % add flaps JRA 071229/080108
% Pitch control (elevator) derivative
```

```

CLde = 0.355;
% alpha-dot derivative
CLalphadot = 0;
% Pitch rate derivative
CLq = 3.8;
% Mach number derivative
CLM = 0;

%%% Drag coefficient %%%
% Lift at minimum drag
CLmind = 0.3;
% Minimum drag
CDmin = 0.04;
% Lift control (flap) derivative
CDdf = 0;
CDdf = 0.1; % add flaps JRA 071229
% Pitch control (elevator) derivative
CDde = 0;
% Roll control (aileron) derivative
CDda = 0;
% Yaw control (rudder) derivative
CDdr = 0;
% Mach number derivative
CDM = 0;
% Oswald's coefficient
osw = 0.7;

%%% Side force coefficient %%%
% Sideslip derivative
CYbeta = -0.564;
% Roll control derivative
CYda = 0;
% Yaw control derivative
CYdr = 0.157;
% Roll rate derivative
CYp = 0;
% Yaw rate derivative
CYr = 0;

%%% Pitch moment coefficient %%%
% Zero-alpha pitch
Cm0 = 0;
% alpha derivative
Cma = -0.683;
% Lift control derivative
Cmdf = 0;
% Pitch control derivative
Cmde = -0.923;
% alpha_dot derivative
Cmalphadot = -4.36;
% Pitch rate derivative
Cmq = -9.96;
% Mach number derivative
CmM = 0;

%%% Roll moment coefficient %%%
% Sideslip derivative
Clbeta = -0.074;
% Roll control derivative
Cllda = -0.134;
% Yaw control derivative

```

```

Cldr = 0.107;
% Roll rate derivative
Clp = -0.41;
% Yaw rate derivative
Clr = 0.107;

%%% Yaw moment coefficient %%%
% Sideslip derivative
Cnbeta = 0.071;
% Roll control derivative
Cnda = -0.0035;
% Yaw control derivative
Cndr = -0.072;
% Roll rate derivative
Cnp = -0.0575;
% Yaw rate derivative
Cnr = -0.125;

%%% PROPELLER %%%
% Propulsion force application point (usually propeller hub) [x y z]
rHub = [1.25*1.75 0 0]; % m
% Advance ratio vector
J = [-1.0 0.1 0.2 0.3 0.35 0.4 0.45 0.5 0.6 0.7 0.8 0.9 1 1.2 2];
% Coefficient of thrust look-up table CT = CT(J)
CT = [0.0492 0.0286 0.0266 0.0232 0.0343 0.034 0.0372 0.0314 0.0254 0.0117
-0.005 -0.0156 -0.0203 -0.0295 -0.04 -0.1115];
% Coefficient of power look-up table CP = CP(J)
CP = [0.0199 0.0207 0.0191 0.0169 0.0217 0.0223 0.0254 0.0235 0.0212 0.0146
0.0038 -0.005 -0.0097 -0.018 -0.0273 -0.0737];
% Propeller radius
Rprop = 1.88/2; % m
% Propeller moment of inertia
Jprop = 25.4*0.0254*0.4536; % kg*m^2

%%% ENGINE %%%
% Engine rpm vector
RPM = [2000 2200 2400 2600 2700]; % rot/min
% Manifold pressure vector
MAP = [57.6 64.35 71.12 77.89 84.66 91.44 98.21]; % kPa

% Sea-level fuel flow look-up table fflow = fflow(RPM, MAP)
% RPM -> rows, MAP -> columns
FuelFlow = [
    6 6.7 7.5 8.5 9.6 12 14.3
    6.5 7.3 8.2 9.4 10.6 12 14.3
    7.2 8 9 10 11.4 12.8 14.3
    7.8 8.6 9.6 10.9 12.3 14 15.6
    8.3 9.2 10.3 11.6 13.2 15 17
]*0.0037854*800*1000;
% Sea-level power look-up table P = P(RPM, MAP)
% RPM -> rows, MAP -> columns
Power = [
    70 83 96 110 124 137 151
    77 90 105 120 134 150 164
    84 98 114 128 144 160 174
    88 104 118 134 150 166 180
    91 107 122 138 154 170 184
]*745.7; % W

```

```

% Sea-level pressure and temperature at which the data above is given
pSL = 102300; % Pa
TSL = 291.15; % deg K
% Engine shaft moment of inertia
Jeng = 0; % Neglected

%%% INERTIA %%%
% Empty aircraft mass (zero-fuel)
mempty = 1000; % kg
% Gross aircraft mass (full fuel tank)
mgross = 1247.392; % kg
% Empty CG location [x y z]
CGempty = [-0.04*MAC 0 -0.25]; % m
% Gross CG location [x y z]
CGgross = [-0.045*MAC 0 -0.2]; % m
rAC = CGgross; % m
% Empty moments of inertia [Jx Jy Jz Jxz]
Jempty = [1040 2900 3500 100]*14.594*0.3048^2; % kg*m^2
% Gross moments of inertia [Jx Jy Jz Jxz]
Jgross = [1048 3000 3530 110]*14.594*0.3048^2; % kg*m^2

%%% OTHER SIMULATION PARAMETERS %%%
% WMM-2000 date [day month year]
dmy = [13 05 2002];

% Save workspace variables to MAT file
save(cfgmatfile);

% Output a message to the screen
fprintf(strcat('\n Aircraft configuration saved as:\t',
strcat(cfgmatfile), '.mat'));
fprintf('\n');

```

Anexo E

E.1. Comando utilizado e parametrização

O comando utilizado, figura E.1., foi parametrizado de maneira a ser possível controlar a aeronave. Este comando é composto por quatro eixos, que variam continuamente, e por onze botões. Na tabela seguinte estão definidas as funcionalidades de cada botão.

Botão pressionado	Função
1	Reset da Simulação
2	-
3	Aumenta a deflexão dos flaps
4	Diminui a deflexão dos flaps
5	Aumenta a mistura do combustível
6	Diminui a mistura do combustível
7	Trem Down
8	Trem Up
9, 10 e 11	Botões auxiliares

Tabela E.1. – Funcionalidades dos botões do *Joystick*.

Os eixos do comando servem para controlar o *Throttle*, a deflexão dos ailerons, do leme de direcção e do leme de profundidade.



Figura E.1. – *Joystick* utilizado.

Relativamente à calibração do comando, dado que é reconhecido como um periférico genérico USB, este pode ser calibrado através do próprio sistema operativo nas suas configurações.

Anexo F

F.1. Manual de utilização do SVI

Todo o modelo é implementado através do software Matlab.

1. Ligar todos os computadores, inclusive o computador da plataforma
Em cada computador deve inicializar-se o respectivo módulo, através do programa Matlab. Os módulos encontram-se nas pastas denominada SVI_Nome_do_Módulo.
2. Inicializar o sistema EFIS
No computador do módulo interface com o piloto deve inicializar-se o sistema EFIS, segundo [10].
3. Inicializar o FlightGear através do MATLAB
No bloco *Generate Run Script* que está dentro do esquema do módulo de simulação de imagem, deverá introduzir-se os dados do aeroporto a simular, ou da posição da aeronave. De seguida, carrega-se no botão *Generate Script* e pode abrir-se o FlightGear através do Matlab.
4. Inicializar o Módulo do movimento
Este módulo inicia-se com a execução do ficheiro **MDA.m**. Após esta inicialização aparecerá uma janela de interface gráfica para o utilizador. Deve carregar-se no botão **START**, em seguida **Engage** e por fim **Ready**. Depois desta inicialização a plataforma está preparada para começar a receber valores do módulo da dinâmica.
5. Inicializar o Módulo da Dinâmica introduzindo as condições de voo desejadas
Abrir o ficheiro **Modelo.m**, onde devem ser introduzidas as condições iniciais do voo assim como as condições da função Trim, caso se deseje pilotos automáticos. Após a configuração das características de voo deve executar-se este ficheiro. O modelo Simulink assim como a interface gráfica para o controlo do comando ou dos pilotos automáticos aparecerão na janela do computador.
6. Inicializar a simulação
Para inicializar a simulação após o SETUP de todos os parâmetros e condições de voo desejadas é necessário inicializar os módulos através da opção **START**. Por conseguinte deve inicializar-se o módulo da imagem, em seguida o módulo de interface com o piloto e por fim o módulo da dinâmica, já que o módulo de movimento tinha sido iniciado anteriormente.

Depois destes procedimentos o SVI está pronto a ser utilizado.

Anexo G

G.1. Parâmetros utilizados pelo bloco *Send*

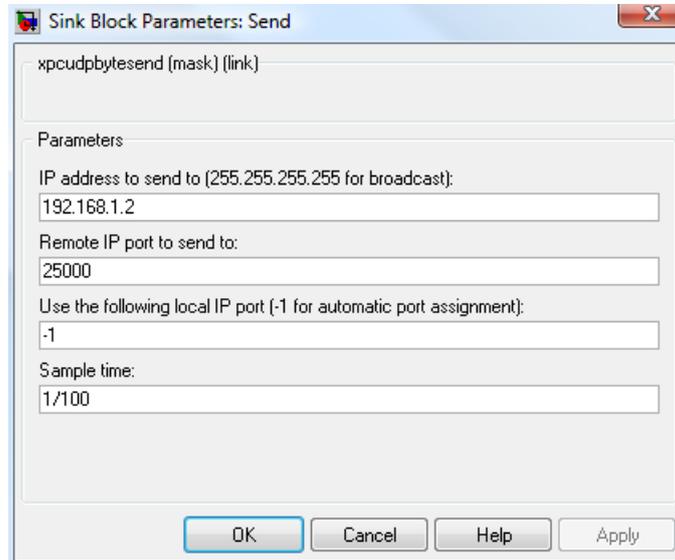


Figura G.1 – Parâmetros do bloco *Send*.

G.2. Parâmetros utilizados pelo bloco *Receive*

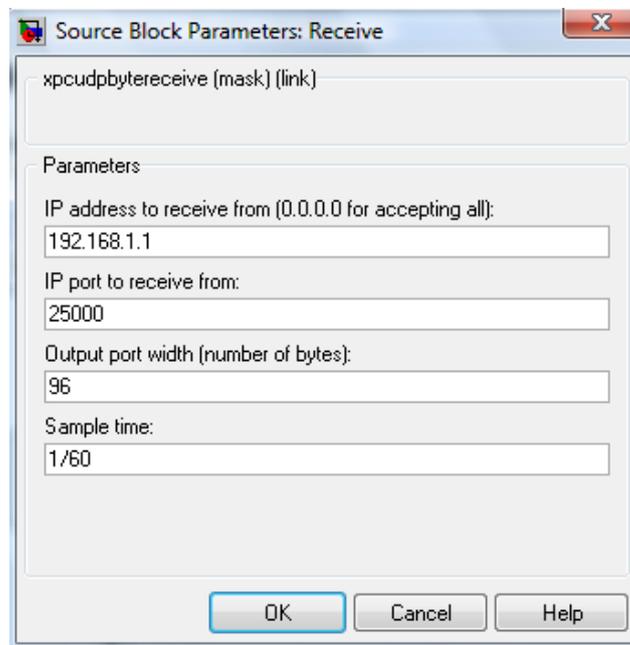


Figura G.2 – Parâmetros do bloco *Receive*.

G.3. Parâmetros utilizados pelo bloco *To Instrument*

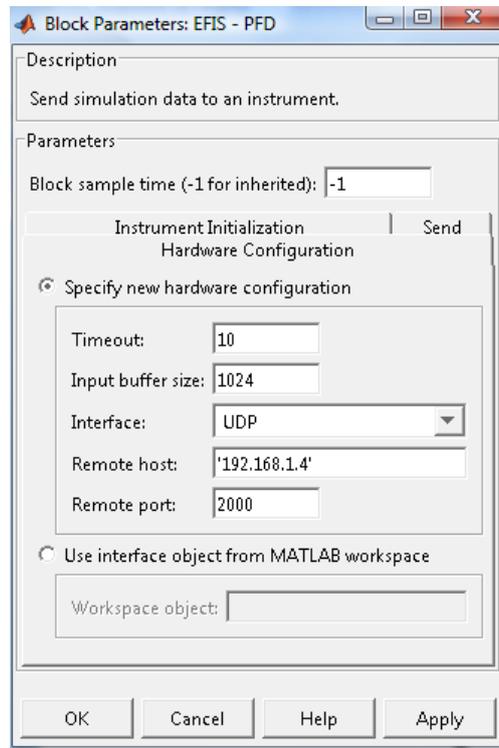


Figura G.3 – Parâmetros do bloco *To Instrument*.