UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
ÁREA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

PROGRAMA COMPUTACIONAL PARA UM SIMULADOR DE VÔO

Carlos Eduardo Beluzo

DEDALUS - Acervo - EESC
31100104295

Dissertação apresentada à Escola de Engenharia de São Carlos, da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Mestre em Engenharia Mecânica.

ORIENTADOR: Prof. Titular Eduardo Morgado Belo

EESC-USP
Biblioteca
CAMPUS DE S. CARLOS

São Carlos

2006

Serviço de Pós-Graduação EESC/USP
EXEMPLAR REVISADO
Data de entrada no Serviço 04/08/2006
Ass.:

FOLHA DE JULGAMENTO

Candidato: Bacharel **CARLOS EDUARDO BELUZO**

Dissertação defendida e julgada em 27/04/2006 perante a Comissão Julgadora:

Eduardo Morgado Belo — Aprovado

Prof. Titular **EDUARDO MORGADO BELO (Orientador)**
(Escola de Engenharia de São Carlos/USP)

Carlos De Marqui Junior — Aprovado

Dr. **CARLOS DE MARQUI JUNIOR**
(Autônomo)

Roseli Ap. Francelin Romero — APROVADA

Profa. Associada **ROSELI APARECIDA FRANCELIN ROMERO**
(Instituto de Ciências Matemáticas e de Computação/USP)

Prof. Associado **JONAS DE CARVALHO**
Coordenador do Programa de Pós-Graduação
em Engenharia Mecânica

Profa. Titular **MARIA DO CARMO CALIJURI**
Presidente da Comissão de Pós-Graduação

AGRADECIMENTOS

A Deus.

Ao Prof. Titular Dr. Eduardo Morgado Belo pela orientação e discussões, pela dedicação, paciência e incentivo neste trabalho.

Ao Prof. Dr. Flávio Donizete Marques pelo apoio e pelas discussões.

Aos meus Pais José Paulo e Regina Célia pelo apoio e dedicação e por todo amor que eles me dão.

A minha noiva Tatiana por todo amor, compreensão, paciência, apoio e incentivo.

Aos meus irmãos Paula e José Rodolfo pelo apoio e incentivo.

A minha companheira de pesquisa Doutoranda Luciane de Fátima por toda ajuda e apoio.

A todos os demais amigos do Laboratório de Aerelasticidade, Dinâmica Vôo e Controle Elizangela, Caixeta, De Marqui, Daniela, Andréa, Edson, e todos que passaram por lá enquanto estive desenvolvendo este trabalho.

Aos meus irmãos do Grupo de Oração Universitário Mario, fiel companheiro, André, Lucas, Vanessa, e todos os "Luquinhas" do Projeto Universidades Renovadas.

Aos meus irmãos do Grupo de Oração *Kairós* Beta, Talita, Rafa, Vagner, Dinei, Du, Lica, e todos os demais.

A todos os meus amigos e a todos que me incentivaram de alguma maneira e acreditaram no meu trabalho muito obrigado.

RESUMO

BELUZO, C. E. (2006). *Programa Computacional para um Simulador de Vôo.* São Carlos, 2006. 85p. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo.

Os simuladores de vôo têm sido uma importante ferramenta para treinamento de pilotos e análise de vôo sem ter que se desembolsar grandes quantias monetárias, economizando combustível e evitando acidentes. Conseqüentemente, a demanda por Simuladores de Vôo tem aumentado tanto na indústria quanto na pesquisa. Com o intuito de futuramente construir um simulador de vôo, foi desenvolvido um projeto para elaboração de um software capaz de simular uma aeronave em vôo, do ponto de vista de dinâmica do vôo.

O software SIMAERO foi desenvolvido na linguagem de programação C++ e simula a dinâmica de vôo de uma aeronave. Esta simulação consiste em resolver as equações de movimento da aeronave, utilizando o modelo matemático de equações diferenciais ordinárias proposto por ETKIN & REID, *et al* (1996). O modelo matemático é solucionado através do método de integração numérica Runge-Kutta de 4ª ordem conforme apresentado em CONTE (1977).

Como parâmetros de entrada são informadas as seguintes características da aeronave: dados geométricos, dados aerodinâmicos e derivadas de estabilidade. Os resultados das simulações são apresentados em gráficos cartesianos e gravados em arquivos. Os gráficos são úteis para que possa ser feita uma posterior análise do comportamento da aeronave. Os arquivos gravados com os resultado das simulações podem ser utilizados em alguma aplicação futura, como sinas de entrada para uma plataforma de simulação, por exemplo.

Neste trabalho será descrito como o SIMAERO foi desenvolvido e ao final serão apresentados alguns resultados obtidos.

Palavras Chave: simulador de vôo; dinâmica de vôo; simulação; Runge-Kutta.

ABSTRACT

BELUZO, C. E. (2006). *A Computer Program for a Flight Simulator*. São Carlos, 2006. 85p. Dissertation (Master Degree) – Escola de Engenharia de São Carlos, Universidade de São Paulo.

Flight simulators have been an important tool for pilots training and for flight analyses, without having to spend a high quantity of money, saving gas and prevent accidents. Because of this, the demand for Flight Simulators has increased both in industry and in research centers. With the objective of in future build a flight simulator, a project to develop a software that is able to simulate the dynamics of flight of a flying aircraft was developed.

The SIMAERO Software was developed using C++ and its principal functionality is to simulate the dynamics of fligth of an aircraft. This simulation basically is the solution of the system of motion equations of the aircraft, using the mathematical model described by ETKIN & REID, et al (1996). The mathematical model is solved using the 4th order Runge-Kutta numeric integration method, as presented in CONTE (1977).

For the simulation, the geometric data, the aerodynamic data, and the Dimensional Derivates are passed to the software as input arguments. The results of the simulations are displayed as Cartesians Graphics and recorded as data files. The graphics are useful for visual analyses of the aircraft behavior, and the file, with the results of the simulation, can be used as input data for ground based simulator, for example.

In this work, the development of the software SIMAERO will be presented, and then some results of the simulation of one aircraft will be shown.

Key Words: flight simulator; dynamics of flight; simulation; Runge-Kutta.

SUMÁRIO

Sumário i

Lista de Figuras iii

Lista de Tabelas v

Lista de Símbolos vi

CAPÍTULO 1

INTRODUÇÃO **1**

1.1 – Objetivos do Trabalho 4

1.2 – Ambiente de Desenvolvimento 5

1.3 – Organização da Dissertação 6

CAPÍTULO 2

REVISÃO BIBLIOGRÁFICA **8**

2.1 – Modelos Matemáticos para Simulação de Vôo 9

2.2 – Elementos do Modelo 12

2.3 – Definição do Modelo Matemático 13

CAPÍTULO 3

MODELO MATEMÁTICO DE DINÂMICA DE VÔO DE UMA AERONAVE **15**

3.1 – Definição do Sistema de Coordenadas e Eixos 16

3.2 – Segunda Lei de Newton e Sistemas de Referência 19

3.2.1 – Suposição 1 – A Aeronave é um Corpo Rígido 20

3.2.2 – Suposição 2 – A Terra está Fixa no Espaço 20

3.3 – Momentos e Produtos de Inércias 26

3.4 – Ângulos de Orientação para uma Aeronave 28

3.4.1 – Orientação e Posição de uma Aeronave 28

3.5 – Matrizes de Transformação 29

CAPÍTULO 4

EQUAÇÕES DE MOVIMENTO 31

4.1 – Contextualização no Espaço 32

4.2 – Equações de Movimento para um Corpo Rígido 33

4.3 – Desenvolvimento das Equações de Movimento de uma Aeronave 33

4.4 – Desenvolvimento das Equações para Obtenção da Posição 39

4.5 – Equações de Forças e Momentos Aerodinâmicos 41

CAPÍTULO 5

ANÁLISE DE REQUISITOS PARA CONSTRUÇÃO DO SOFTWARE SIMAERO 44

5.1 – Introdução 44

5.2 – Descrição da Informação 46

5.2.1 – Reconhecimento do Problema 46

5.3 – Descrição Funcional 47

5.3.1 – Avaliação e Síntese da Solução 47

5.4 – Definição das Variáveis e Constantes 50

5.5 – Algoritmo Descritivo do Programa SIMAERO 57

5.5.1 – Seqüência de eventos da execução do software 57

5.5.2 – Algoritmo Descritivo 58

CAPÍTULO 6

RESULTADOS 64

CAPÍTULO 7

CONCLUSÕES E TRABALHOS FUTUROS 77

REFERÊNCIAS BIBLIOGRÁFICAS 81

LISTA DE FIGURAS

CAPÍTULO 2

Figura 2.1 – Modelo Conceitual, Modelo Computacional e Realidade (ROLFE & STAPLES, 1986). 11

Figura 2.2 – Elementos do Modelo Conceitual (ROLFE & STAPLES, 1986). 13

CAPÍTULO 3

Figura 3.1 – Sistema de Referência Fixo na Terra - $F_E(O_E, x_E, y_E, z_E)$ (ETKIN & REID,1996). 17

Figura 3.2 - Sistema de Referência Fixo no Corpo (ETKIN & REID,1996). 18

Figura 3.3 - Planos de Simetria da Aeronave. 27

Figura 3.4 - Orientação da aeronave (ETKIN & REID,1996). 29

CAPÍTULO 5

Figura 5.1 – Diagrama de Fluxo de Dados do SIMAERO. 46

Figura 5.2 – Estrutura do Software SIMAERO. 49

Figura 5.3 – Fluxograma do Software SIMAERO. 56

Figura 5.4 – Implementação computacional do método de integração numérica Runge-Kutta de 4ª Ordem. 62

Figura 5.5 – Interface gráfica do software SIMAERO. 63

CAPÍTULO 6

Figura 6.1 – Componentes das velocidades lineares (ETKIN & REID, 1996). 67

Figura 6.2 – Componentes das velocidades angulares (ETKIN & REID, 1996). 68

Figura 6.3 – Ângulos de Orientação (ETKIN & REID, 1996). 68

Figura 6.4 – Componentes das velocidades lineares – Resultado do SIMAERO. 69
Figura 6.5 – Componentes das velocidades angulares – resultado do SIMAERO. 70
Figura 6.6 – Ângulos de Orientação – resultado do SIMAERO. 70
Figura 6.7 – Velocidade Linear u (m/s) *vs* Tempo (s) – resultado do SIMAERO. 73
Figura 6.8 – Velocidade Linear w (m/s) *vs* Tempo (s) – resultado do SIMAERO. 73
Figura 6.9 – Velocidade Angular q (rad/s) *vs* Tempo (s) – resultado do SIMAERO. 74
Figura 6.10 – Ângulo de Orientação θ (graus) *vs* Tempo (s) – resultado do SIMAERO. 74
Figura 6.11 – Deslocamento em x (m) *vs* Tempo (s) – resultado do SIMAERO. 75
Figura 6.12 – Deslocamento em z (m) *vs* Tempo (s) – resultado do SIMAERO. 75

LISTA DE TABELAS

CAPÍTULO 6

Tabela 6.1 – Dados do Boeing 747-101 (ETKIN & REID, 1996). 65

Tabela 6.2 – Derivadas de Estabilidade Dimensionais do Boeing 747-101. 65

Tabela 6.3 – Dados do Boeing 747-101 (ETKIN & REID, 1996). 71

Tabela 6.4 – Derivadas de Estabilidade Dimensionais do Boeing 747-101. 72

LISTA DE SÍMBOLOS

$\ddot{x}$	Aceleração
F	Força
m	Massa
CG	Centro Gravitacional
O	Ponto de Origem Localizado no CG
Ox	Eixo da aeronave na direção horizontal
Oy	Eixo da aeronave na direção lateral
Oz	Eixo da aeronave na direção vertical
F_E	Sistema de Referência Fixo na Terra
O_E	Centro do Sistema de Referência Fixo na Terra
x_E	Vetor no Sistema F_E saindo de O_E na direção Ox
y_E	Vetor no Sistema F_E saindo de O_E na direção Oy
z_E	Vetor no Sistema F_E saindo de O_E na direção Oz
F_B	Sistema de Referência Fixo no Corpo
C	Centro do Sistema de Referência Fixo no Corpo
Cxz	Plano de Simetria da Aeronave
X_B	Eixo X no Sistema F_B
Y_B	Eixo Y no Sistema F_B
Z_B	Eixo Z no Sistema F_B
u	Componente da Velocidade de Translação na Direção CX_B
v	Componente da Velocidade de Translação na Direção CY_B
w	Componente da Velocidade de Translação na Direção CZ_B
X	Componente da Força na Direção CX_B
Y	Componente da Força na Direção CY_B
Z	Componente da Força na Direção CZ_B

p	Componente da Velocidade de Rotação na Direção CX_B
q	Componente da Velocidade de Rotação na Direção CY_B
r	Componente da Velocidade de Rotação na Direção CZ_B
L	Momento de Rolagem na Direção CX_B
M	Momento de Rolagem na Direção CY_B
N	Momento de Rolagem na Direção CZ_B
$\mathbf{p}$	Vetor de Quantidade Movimento Linear
$\mathbf{H}$	Vetor de Quantidade de Movimento Angular
t	Tempo
V	Velocidade do Total da Aeronave
v_e	Velocidade Aplicada
T_e	Força de Tração
F_x	Força Total Aplicada na Direção CX_B
F_y	Força Total Aplicada na Direção CY_B
F_z	Força Total Aplicada na Direção CY_B
H	Vetor Quantidade de Movimento Angular
$\underline{\underline{I}}$	Momento de Inércia
Ω	Vetor Velocidade Angular
ξ	Eixo Principal
$I_{\xi\xi}$	Momento de Inércia Referente ao Eixo Principal
I_x	Momento de Inércia Relativo à Rolagem
I_y	Momento de Inércia Relativo à Arfagem
I_z	Momento de Inércia Relativo à Guinada
I_{xz}	Produtos de Inércia Relativo ao Plano de Simetria Cxy
I_{zy}	Produtos de Inércia Relativo ao Plano de Simetria Czy
I_{xz}	Produtos de Inércia Relativo ao Plano de Simetria Cxz
ψ	Ângulo de azimute

θ	Ângulo de arfagem
ϕ	Ângulo de rolagem
g	Aceleração da Gravidade
L	Sustentação
D	Arrasto
ρ	Densidade
S	Área de Referência
C_L	Coeficiente de Sustentação
C_D	Coeficiente de Arrasto
x_E	Posição da Aeronave com Relação ao Eixo $O_E X$
y_E	Posição da Aeronave com Relação ao Eixo $O_E Y$
z_E	Posição da Aeronave com Relação ao Eixo $O_E Z$
δ_a	Ângulo de deflexão dos ailerons
δ_e	Ângulo de deflexão do profundor
δ_r	Ângulo de deflexão do leme
X_u	Derivada da Força em X em Relação à Velocidade u
X_w	Derivada da Força em X em Relação à Velocidade w
X_q	Derivada da Força em X em Relação à Velocidade q
$X_{\dot{w}}$	Derivada da Força em X em Relação à Velocidade $\dot{w}$
X_{δ_e}	Derivada da Força em X em Relação à δ_e
Z_u	Derivada da Força em Z em Relação à Velocidade u
Z_w	Derivada da Força em Z em Relação à Velocidade w
Z_q	Derivada da Força em Z em Relação à Velocidade q
$Z_{\dot{w}}$	Derivada da Força em Z em Relação à Velocidade $\dot{w}$
Z_{δ_e}	Derivada da Força em Z em Relação à δ_e
M_u	Derivada do Momento Aerodinâmico em M em Relação à Velocidade u
M_w	Derivada do Momento Aerodinâmico em M em Relação à Velocidade w

M_q	Derivada do Momento Aerodinâmico em M em Relação à Velocidade q
$M_{\dot{w}}$	Derivada do Momento Aerodinâmico em M em Relação à Velocidade $\dot{w}$
M_{δ_e}	Derivada do Momento Aerodinâmico em M em Relação à Velocidade δ_e
Y_v	Derivada da Força em Y em Relação à Velocidade v
Y_p	Derivada da Força em Y em Relação à Velocidade p
Y_r	Derivada da Força em Y em Relação à Velocidade r
Y_{δ_r}	Derivada da Força em Y em Relação à δ_r
Y_{δ_a}	Derivada da Força em Y em Relação à δ_a
L_v	Derivada do Momento L em Relação à Velocidade v
L_p	Derivada do Momento L em Relação à Velocidade p
L_r	Derivada do Momento L em Relação à Velocidade r
L_{δ_r}	Derivada do Momento L em Relação à δ_r
L_{δ_a}	Derivada do Momento L em Relação à δ_a
N_v	Derivada do Momento N em Relação à Velocidade v
N_p	Derivada do Momento N em Relação à Velocidade p
N_r	Derivada do Momento N em Relação à Velocidade r
N_{δ_r}	Derivada do Momento N em Relação à δ_r
N_{δ_a}	Derivada do Momento N em Relação à δ_a

CAPÍTULO 1

INTRODUÇÃO

A prática de simular computacionalmente o modelo matemático da dinâmica de vôo de aeronaves, de uma grande importância na aeronáutica, tem sido muito empregada ao logo de sua história. Atualmente os simuladores de vôo têm sido uma importante ferramenta para treinamento de pilotos e análise de vôo sem ter que desembolsar grandes quantias monetárias, economizando combustível e até evitando acidentes. Devido a isto, tem-se observado que a demanda por simuladores de vôo tem aumentado tanto na indústria quanto na pesquisa (ROLFE & STAPLES, 1986). O baixo custo para treinamento de pilotos; testes sobre o comportamento dinâmico do ser humano enquanto piloto de uma aeronave; e também para simulação do comportamento dinâmico de vôo de diferentes modelos de aeronaves têm sido o principal atrativo dos simuladores de vôo.

Os modelos matemáticos utilizados em simulação, baseados em derivadas de estabilidade e com algumas simplificações, tornam-se modelos relativamente simples, considerando os resultados muito próximos do real que estas simulações apresentam. As simulações e testes de vôo têm sido de grande importância no progresso e desenvolvimento da aeronáutica (ROLFE & STAPLES, 1986).

Com o avanço tecnológico, os computadores pessoais têm se tornado cada vez mais potentes. Isto tem estimulado a construção de softwares capazes de simular fenômenos reais. A simulação de aeronaves tem sido uma das áreas mais

complicadas e estimulantes da engenharia. Com o auxílio de computadores pessoais, as pesquisas e os trabalhos de simulação de movimento e do comportamento de aeronaves foram muito favorecidos.

Com o intuito de futuramente construir uma plataforma de simulação, foi desenvolvido um projeto para elaboração de software capaz de simular matematicamente uma aeronave em vôo, do ponto de vista da dinâmica do vôo da aeronave. A primeira parte deste trabalho consiste em apresentar um modelo de simulação matemática da dinâmica de vôo de uma aeronave, para em seguida desenvolver um software capaz de simular este modelo. Isto significa resolver o sistema de equações de movimento da aeronave.

Com isto, pretende-se também em trabalhos futuros, utilizar dados obtidos como resposta deste software como dados de entrada para uma plataforma de simulação. O software SIMAERO foi desenvolvido no LADinC (Laboratório de Aeroelasticidade, Dinâmica de Vôo e Controle) do NPA (Núcleo de Pesquisas em Aeronáutica) da EESC (Escola de Engenharia de São Carlos) da Universidade de São Paulo. Este software realiza a simulação da dinâmica de vôo de aviões a partir dos dados geométricos, dados aerodinâmicos e derivadas de estabilidade de modo a encontrar as velocidades lineares e angulares, as acelerações lineares e angulares, a posição e a orientação da aeronave simulada.

Outros trabalhos já foram desenvolvidos dentro no LADinC, pensando na plataforma de simulação com relação à modelagem matemática, computacional e mecânica. Nestes trabalhos foram realizados estudos em cima da plataforma de Stewart. A plataforma de Stewart é um manipulador paralelo com alta relação força-peso e acuracidade de posicionamento comparada a manipuladores seriais convencionais. Uma aplicação bastante popular da plataforma de Stewart é o simulador de vôo onde a plataforma executa movimento com acelerações similares àquelas de uma aeronave. A plataforma de Stewart tem recebido recentemente considerável interesse de pesquisadores devido ao sucesso de suas

aplicações e potencial vantagens sobre os manipuladores convencionais (CAPORALI, 2003).

Em um destes trabalhos realizados no LADinC podemos citar um estudo realizado sobre técnicas de projeto de controle neural e controle difuso para uma Plataforma de Stewart acionada hidraulicamente (CAPORALI, 2003).

Ainda nesta linha temos um trabalho onde foi realizado um estudo apresentando o controle de posição e orientação de um modelo não linear da Plataforma de Stewart com seis graus de liberdade (MONTEZUMA, 2003).

Também com o intuito de construir futuramente uma plataforma deste tipo foi realizado um trabalho apresentando a modelagem, simulação e análise de um servomecanismo eletro-hidráulico de controle de posição, desenvolvendo também neste trabalho modelos matemáticos que descrevem o funcionamento dos elementos que compõem o sistema eletro-hidráulico (CAPORALI, 1999).

Visando obter dados para realizar simulações, temos um trabalho em andamento propondo um método para simulação e identificação de envelopes de vôo utilizando redes neurais (SOUZA, 2005).

Podemos citar por fim um trabalho similar ao presente propondo um programa computacional, o qual fornece como saídas respostas com relação à dinâmica de vôo de uma aeronave (TANARA, 1995).

Motivado por estes trabalhos e especialmente por esta última citação, foi desenvolvido o presente trabalho visando obter-se um programa similar, de modo fornecer um software inserido no contexto tecnológico atual, e com algumas diferenças quanto à modelagem matemática e à implementação.

1.1 – OBJETIVOS DO TRABALHO

Como brevemente mencionado acima, o objetivo deste trabalho é construir um software para auxiliar na análise comportamental de uma aeronave em vôo. Foi escolhida para implementação a linguagem de programação C++, devido ao fato de esta ser bastante difundida tanto no meio acadêmico, como industrial. Isso favorece futuras alterações no software, ou ainda, que este mesmo software possa ser estendido para outras aplicações, como já citado anteriormente.

O modelo matemático escolhido também é muito difundido no meio acadêmico, especificamente nas diversas sub-áreas de pesquisa em aeronáutica. Além disso, é o modelo que os trabalhos e as pesquisas atuais, deste mesmo gênero, têm utilizado como base para equacionamento da dinâmica de vôo de uma aeronave, com resultados coerentes.

O método de integração numérica Runge-Kutta também é bastante utilizado nas aplicações de integração numérica em Engenharia e tem apresentado resultados aceitáveis em diversos trabalhos de diversas áreas para um grande número de aplicações.

Com a construção deste software, espera-se fornecer uma ferramenta de auxílio para outras áreas de pesquisa em aeronaves, que utilizam respostas de simulação de aeronaves em vôo como condições iniciais ou parâmetros de entrada, como por exemplo, identificação com redes neurais. Com a utilização deste software, seriam diminuídos o tempo e o custo de obtenção destes dados. Além disso, espera-se fornecer uma base para pesquisa e desenvolvimento de simuladores de vôo, e trabalhos nesta área.

1.2 – AMBIENTE DE DESENVOLVIMENTO

O software foi desenvolvido em um PC com plataforma Windows 2000 e utiliza a linguagem de programação C++. A princípio os dados de entrada para a simulação são numéricos e passados diretamente para o programa através da interface desenvolvida com o auxílio do aplicativo Borland C++ Builder©. Os resultados das simulações são apresentados em gráficos cartesianos e gravados em arquivos. Pretende-se também em trabalhos futuros, adaptar o programa para que, através de interfaces de hardware, receba os dados de sinais provenientes de uma plataforma de simulação. Da mesma maneira os sinais de saída serão enviados para a plataforma de simulação.

O modelo aerodinâmico completo é relativamente complexo, logo, foram feitas algumas simplificações de modo a não comprometer os resultados, para que estes possam ser o mais próximo dos valores reais quanto possíveis. Estas simplificações também estão de acordo com ETKIN & REID *et al* (1996).

Baseado neste modelo, o SIMAERO foi desenvolvido e realiza a simulação da dinâmica de vôo de uma aeronave, ou seja, implementa e resolve estas equações, obtendo então os valores numéricos que descrevem a dinâmica de vôo da aeronave. Esta simulação é realizada a partir dos seguintes dados da aeronave: dados geométricos, dados aerodinâmicos e derivadas de estabilidade, os quais são passados como parâmetros de entrada para o software. Estes dados, em uma boa parte são obtidos em ensaios em vôo e para este trabalho foram retirados de ETKIN & REID *et al* (1996) e de MCRUER, ASHKENAS & GRAHAM, (1973). Como parâmetros de saída, a simulação retorna os valores das seguintes variáveis: velocidades lineares, velocidades angulares, acelerações lineares, acelerações angulares e orientação e deslocamento da aeronave. Estes resultados são apresentados em valores numéricos e desenhados em gráficos cartesianos.

Com os computadores pessoais atuais, com velocidade de processamento nominal em torno de três Ghz é possível implementar estas equações de movimento em um software que gere respostas com rapidez equivalente ou muito próxima daquela com que o avião responderia. Isto é possível, pois o modelo dinâmico da aeronave implementado no SIMAERO foi adaptado fazendo a linearização de alguns termos, de modo a deixar o sistema de equações resultantes não muito complexo (ETKIN & REID *et al*, 1996).

A escolha dos modelos utilizados na implementação, modelo de orientação da aeronave, complexidade do modelo aerodinâmico, algoritmo de integração e demais, foram baseados no que atualmente temos encontrado nos trabalhos publicados, e que têm sido utilizados para simulações de vôo, obtendo resultados próximo do real.

1.3 – ORGANIZAÇÃO DA DISSERTAÇÃO

Este primeiro capítulo apresenta uma introdução do trabalho desenvolvido. A princípio é feita uma breve explicação sobre simuladores de vôo e sua utilização. Em seguida descreve-se os modelos matemáticos, e o emprego destes com as tecnologias atuais. Por fim, descreve-se os objetivos deste trabalho.

No capítulo 2 é feita uma revisão bibliográfica. Inicialmente é feito um detalhamento dos objetivos do trabalho, mostrando o ambiente computacional utilizado no desenvolvimento do software. Em seguida é apresentado o modelo de simulação e os elementos contidos neste modelo. Por fim, descreve-se brevemente o modelo matemático implementado.

No capítulo 3 é feita a apresentação e o detalhamento do modelo dinâmico de uma aeronave a partir da 2ª Lei de Newton.

No capítulo 4 são apresentadas as deduções das equações de movimento da aeronave, assim como as equações de forças e momentos aerodinâmicos, equações estas que serão implementadas no Software SIMAERO.

No capítulo 5 apresenta-se a construção do software, definindo seus requisitos, apresentado a estrutura, o fluxo de dados e o algoritmo de integração numérica implementado para resolver as equações.

No capítulo 6 são apresentados os gráficos resultantes da resolução das equações de movimento, provenientes de simulações realizadas com o software SIAMERO. Estes gráficos são apresentados com a intenção de validar e testar a funcionalidade do software.

No último capítulo são feitas as conclusões deste trabalho juntamente com sugestões para trabalhos futuros, e alterações e atualização que poderiam ser feitas no software.

CAPÍTULO 2

REVISÃO BIBLIOGRÁFICA

A simulação tem sido uma ferramenta importante para estudo do comportamento físico de objetos de pesquisa da engenharia, especialmente pelo baixo custo se comparado ao de construção do objeto real. O desempenho, disponibilidade e o custo relativamente baixo de computadores e hardwares em geral, têm favorecido a construção de simuladores de modelos de dinâmica de vôo e sistemas de controle de vôo, possibilitando visualizar características e repostas de uma aeronave em tempo de execução próximo do da resposta do avião real (simulação em tempo real) (HORN, BRIDGES, LOPES, BRENTNER, 2005).

Desenvolver o modelo de uma nova aeronave, testar e avaliar suas características dinâmicas são algumas das maiores contribuições que um simulador de vôo oferece. Com isso, o uso de simuladores de vôo tem crescido cada vez mais resultando numa evolução nas técnicas utilizadas e modelos de simuladores de dinâmica e controle de vôo (MOODY, 2004).

Com a simulação de vôo é possível que o piloto seja colocado em um ambiente "real" de vôo permitindo realizar o treinamento de pilotos, assim como de engenheiros (WEINGARTEN, 2005).

Dentro de uma vasta literatura em termos de simulação de vôo e modelagem matemática de aeronaves, envolvendo estudos de hardware e

software, é notável a necessidade de uma plataforma de simulação como ferramenta para estudos devido ao alto nível de interação existente entre a plataforma e o usuário (PRALIO, GUGLIERI & QUAGLIOTTI, 2004).

Na engenharia aeronáutica a simulação tem sido uma ferramenta bastante importante também na prevenção e reconstrução de acidentes. Para isto métodos de modelagem aerodinâmica para simulação que descrevem detalhadamente características da dinâmica de vôo têm sido desenvolvidos para aplicações como estas (FOSTER, CUNNINGHAM, FREMAUX, SHAH, STEWART, RIVERS, WILBORN & GATO, 2005).

A modelagem matemática para representar o comportamento dinâmico de uma aeronave pode ser definida de acordo com cenários e condições pré-determinados, definindo os subsistemas envolvidos na situação em que se deseja simular. Alguns softwares simuladores de vôo são desenvolvidos para resolver problemas específicos relacionados à dinâmica de vôo, como por exemplo o estudo de técnicas de identificação de um modelo aerodinâmico para aeronaves sob efeito natural de acumulo de gelo sobre a superfície da aeronave (HUI, WOLDE, BROWN, ISAAC, COBER, 2005).

1.1 – MODELOS MATEMÁTICOS PARA SIMULAÇÃO DE VÔO

A simulação de sistemas de controle é útil quando contribui para melhorar a modelagem da dinâmica de uma aeronave. Isto requer que os dados de forças e momentos aerodinâmicos sejam definidos corretamente para calcular acelerações, além de um sistema de equações de movimento juntamente com um método de integração numérica computacional, para obter velocidades e posições a partir das acelerações. Várias técnicas têm sido desenvolvidas para adaptar estes conceitos matemáticos aos computadores atuais (MOODY, 2004).

A base para análise, computação ou simulação do movimento de uma aeronave é o modelo matemático da aeronave e de seus subsistemas. A representação de uma aeronave em vôo resulta em sistema dinâmico muito complicado. Devido a isso, iremos simplificar nosso sistema considerando em primeiro lugar a aeronave como sendo um corpo rígido com seis graus de liberdade. Este corpo movimenta-se livremente na atmosfera sujeita a ação da aceleração da gravidade, das forças aerodinâmicas (ROLFE & STAPLES, 1986).

A principal tarefa para modelar o comportamento dinâmico de um veículo em vôo é uma extensão dos princípios básicos de mecânica do vôo. A natureza do problema consiste no movimento de um veículo em vôo que é governado pelas leis de Newton que resultam nas equações de movimento na forma genérica:

$$\ddot{x} = F/m \tag{2.1}$$

onde $\ddot{x}$ é a aceleração do veículo, F é a força aplicada e m a massa (ROLFE & STAPLES, 1986).

Para uma aeronave voando, este modelo matemático é primeiramente um relacionamento entre reação do ar e movimento relativo entre aeronave e ar ao seu redor. Isto pode ser chamado de modelo aerodinâmico. Outras forças externas e momentos aparecem devido ao mecanismo de propulsão e superfícies de controle. A definição de todas estas forças e momentos é a chave para uma descrição realística das características de uma aeronave em vôo. O desempenho e o comportamento dinâmico de uma aeronave podem ser calculados para todas as condições de vôo em que as forças possam ser definidas (ROLFE & STAPLES, 1986).

A simulação é fundamentalmente a geração ou identificação destas forças e a solução das equações de movimento. A simulação é descrita por 3 elementos

básicos: Modelo Conceitual, Modelo Computacional e Realidade (ROLFE & STAPLES, 1986).

Começando pela realidade, um modelo conceitual é descrito pelas equações que são obtidas por análise da realidade. A implementação por um programa de computador resulta no modelo computacional, o qual por meio de simulação pode ser comparado com a realidade. A credibilidade do modelo conceitual é avaliada por procedimentos que testam a adequação do modelo para fornecer um aceitável nível de comparação com a realidade, enquanto o modelo computacional utilizando-se de um programa, é admitido como uma representação adequada do modelo conceitual, por procedimentos de verificação. Finalmente a validação do modelo demonstra que o modelo computacional possui, ou não, uma taxa de precisão satisfatória comparado com a realidade e consistente para a aplicação desejada (ROLFE & STAPLES, 1986).

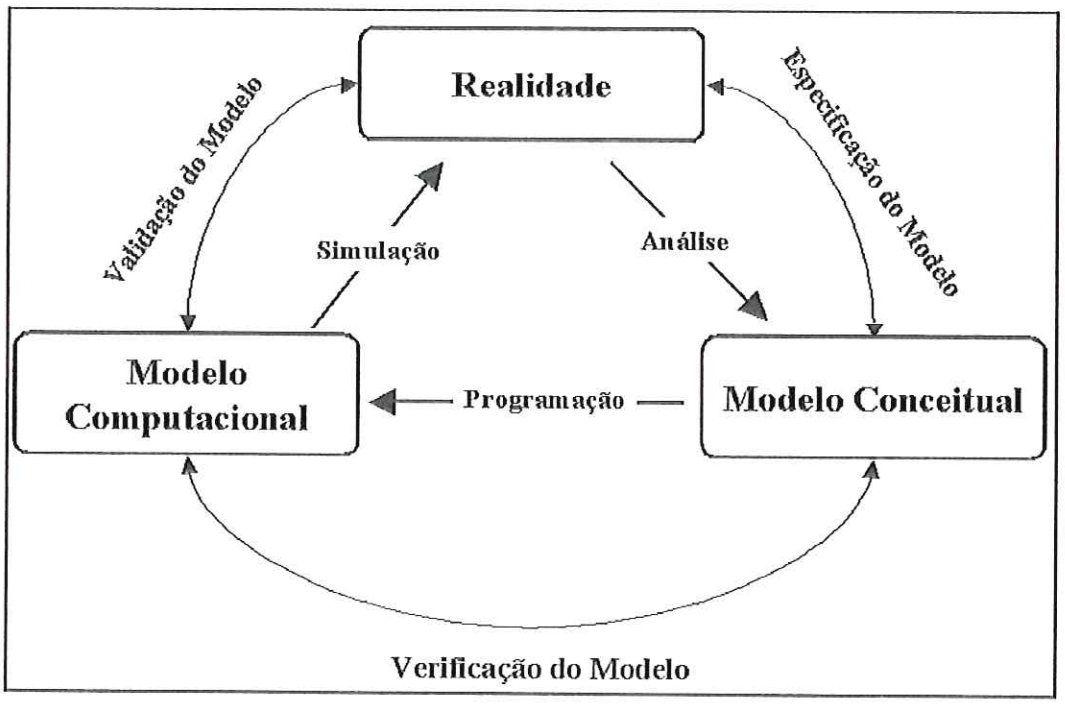

Figura 2.1 – Modelo Conceitual, Modelo Computacional e Realidade (ROLFE & STAPLES, 1986).

Na criação ou desenvolvimento de modelos matemáticos, é importante que a pessoa que está desenvolvendo o modelo, tenha em mente o objetivo do modelo a ser desenvolvido, isso porque o propósito do modelo influência na forma e na qualidade, podendo ser complexo demais para uma certa aplicação ou simplesmente não ser útil ou pouco confiável para outra determinada aplicação (ROLFE & STAPLES, 1986).

A derivação de modelos matemáticos para simulação possui algumas similaridades com as técnicas básicas da teoria de estabilidade e controle, mas existem diferenças. As similaridades relacionam-se com notação matemática, sistemas de coordenadas e nomenclatura básica. As diferenças são significantes e incluem: altas variações de velocidade, configuração da aeronave e manobra, solução em tempo real, entre outras (ROLFE & STAPLES, 1986). Neste estudo o simulador deve ser o mais próximo possível do real, mas deve ser capaz de fornecer respostas em tempo de execução, portanto será simplificado.

1.2 – ELEMENTOS DO MODELO

O modelo aerodinâmico deve reproduzir as características dominantes de forças e momentos em uma aeronave: sustentação e arrasto, os principais contribuintes para o desempenho da aeronave, e momentos sobre todos os eixos, oriundos de controles aplicados pelo piloto. Modelos matemáticos normalmente consideram componentes físicos do veículo como, asa, fuselagem, cauda, e constroem equações matemáticas para as contribuições destes elementos (ROLFE & STAPLES, 1986).

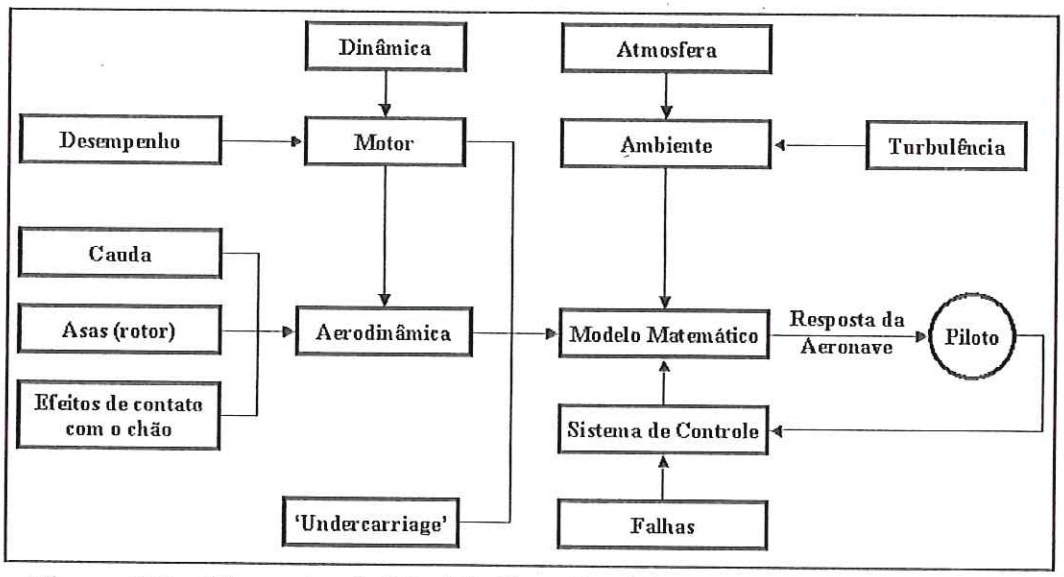

Figura 2.2 – Elementos do Modelo Conceitual (ROLFE & STAPLES, 1986).

Modelos aerodinâmicos normalmente são estacionários, supondo-se que o escoamento de ar pode restabelecer as condições estáveis em uma escala de tempo pequena por uma manobra realizada pela própria aeronave. Este tempo não aparece explicitamente no modelo matemático, mas o comportamento transitório de uma aeronave é uma função do tempo (ROLFE & STAPLES, 1986).

1.3 – DEFINIÇÃO DO MODELO MATEMÁTICO

Para a simulação iremos utilizar as equações de movimento de uma aeronave que são descritas em diversos textos como ETKIN & REID, *et al* (1996), MCRUER, ASHKENAS & GRAHAM, *et al* (1973), e BABISTER, *et al* (1986), entre outros. Porém existem algumas diferenças na maneira em que cada autor define as unidades dos coeficientes aerodinâmicos, e também na definição dos termos das equações, assim como símbolos usados para cada termo nas equações. Sendo assim, para este trabalho iremos adotar como referência base o

modelo de equacionamento de acordo com o proposto por ETKIN & REID, *et al* (1996).

Para a dedução destas equações, com respeito a simplificações do modelo considera-se ainda que a aeronave possui um plano longitudinal de simetria, a atmosfera esta repouso; os efeitos de rotação da Terra são desprezados, assim como as forças de propulsão.

CAPÍTULO 3

MODELO MATEMÁTICO DE DINÂMICA DE VÔO DE UMA AERONAVE

O desenvolvimento de um modelo matemático para representar uma aeronave em vôo envolve a descrição física externa e as forças que atuam sobre o corpo (aeronave). Para este trabalho foi usado o modelo baseado nas equações de movimento apresentadas por Newton.

Este modelo de dinâmica de vôo de aeronave descreve a posição, as velocidades lineares e angulares, integradas com as acelerações, e calculadas a partir das forças e momentos aerodinâmicos. A dedução das equações consiste basicamente da descrição geométrica e de massa da aeronave, juntamente com as equações de movimento e influências externas que atuam sobre ela. Este modelo consiste de um conjunto de equações não lineares.

Dentre alguns modelos que descrevem a dinâmica de vôo, foi adotado o modelo matemático não linear apresentado por ETKIN & REID *et al* (1996). A representação deste sistema é feita através de um conjunto de 12 equações diferenciais ordinárias.

3.1 – DEFINIÇÃO DO SISTEMA DE COORDENADAS E EIXOS

Geralmente em simulação de aeronaves são usados dois sistemas de referências conhecidos como *Sistema de Referência Fixo no Corpo* e *Sistema de Referência Fixo na Terra*. Para este trabalho também serão usados estes dois sistemas de coordenadas e que estão descritos a seguir.

Para formular as equações de movimento de uma aeronave, é necessário primeiramente definir um sistema de referência. Este sistema será denotado por $F_E(O_E, x_E, y_E, z_E)$, que é o *Sistema de Eixos de Referência Fixo na Terra*. Sua origem possui localização definida como o ponto, na Terra, em que a aeronave se encontra no momento inicial do vôo. O eixo $O_E z_E$ aponta verticalmente na direção da aceleração da gravidade (**Figura 3.3** adiante). O eixo $O_E x_E$ é horizontal e aponta para a mesma direção do eixo x da aeronave no instante inicial.

Também neste trabalho a Terra é considerada plana e estacionária no espaço inercial. Com isto, todo e qualquer sistema de referência fixo na Terra pode ser considerado um sistema inercial no qual são válidas as Leis de Newton (ETKIN & REID *et al*, 1996). A força gravitacional também é assumida uniforme, o centro de massa e o centro gravitacional (CG) na aeronave são assumidos como sendo o mesmo ponto. A localização do CG é dada de acordo com as Coordenadas Cartesianas no sistema F_E.

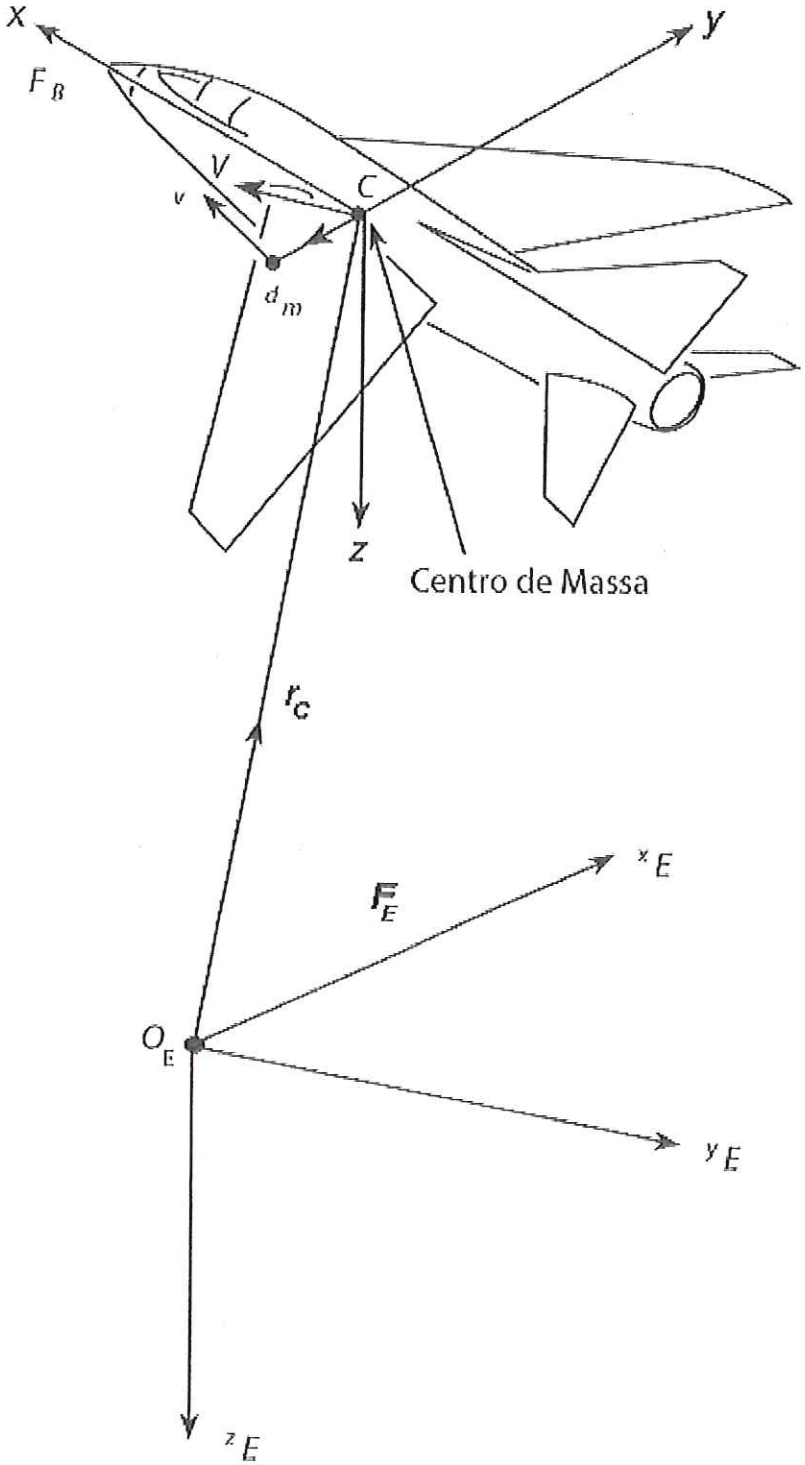

Figura 3.1 – Sistema de Referência Fixo na Terra - $F_E(O_E, x_E, y_E, z_E)$ (ETKIN & REID, 1996).

Além deste sistema, é necessário ainda definir um segundo sistema de referência para formular as equações de movimento. Este segundo sistema é o *Sistema de Referência Fixo no Corpo* (**Figura 3.1**) e movimenta-se com ele. Sua origem C coincide com o CG. Este sistema será denotado por F_B e é conhecido como sistema de referência no corpo. Cxz é o plano de simetria da aeronave. Existem ainda outros sistemas que poderiam ser utilizados para formular as equações do corpo rígido, como por exemplo, *Sistema de Referência Fixo no Espaço*, porém resultaria em sistemas de equação mais complicados do que utilizando o sistema F_B, por isto escolheu-se o sistema F_B (MCRUER, ASHKENAS & GRAHAM, 1973). Assim, na formulação do sistema de equações, utilizaremos os seguintes sistemas de referência:

- F_E = *Sistema de Referência Fixo na Terra*;
- F_B = *Sistema de Referência Fixo no Corpo*.

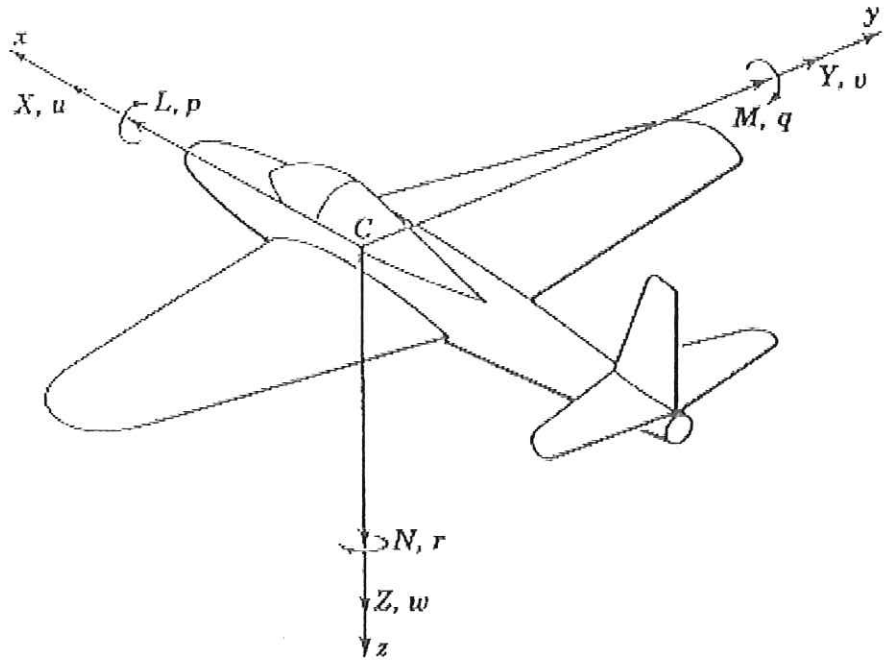

Figura 3.2 - Sistema de Referência Fixo no Corpo (ETKIN & REID,1996).

Pode-se observar na **Figura 3.2** que para F_B têm-se os seguintes termos:

- C = origem do Sistema F_B;
- u = componente da velocidade de translação do centro de massa na direção do eixo x;
- v = componente da velocidade de translação do centro de massa na direção do eixo y;
- w = componente da velocidade de translação do centro de massa na direção do eixo z;
- X = componente da força na direção do eixo x;
- Y = componente da força na direção do eixo y;
- Z = componente da força na direção do eixo z;
- p = velocidade de rotação da aeronave na direção eixo x;
- q = velocidade de rotação da aeronave na direção eixo y;
- r = velocidade de rotação da aeronave na direção eixo z;
- L = Momento de rolagem, em torno do eixo x;
- M = Momento de arfagem, em torno do eixo y;
- N = Momento de guinada, em torno do eixo z.

Geralmente, calculam-se todas as forças aerodinâmicas, velocidades e acelerações no sistema F_B, em seguida convertem-se para o sistema F_E para determinar a posição e orientação da aeronave.

3.2 – SEGUNDA LEI DE NEWTON E SISTEMAS DE REFERÊNCIA

Conforme citado no capítulo anterior, as duas suposições que seguem são as necessárias para contextualizar a apresentação do modelo matemático que é deduzido a partir da 2ª Lei de Newton.

3.2.1 – Suposição 1 – A aeronave é um corpo rígido

Em um corpo rígido, a distância entre quaisquer pontos específicos no corpo é fixa e devido a isto, a força resultante individual sobre cada elemento de massa do corpo pode ser desconsiderada. Conseqüentemente, o movimento do corpo pode ser completamente descrito através da translação do centro de massa do corpo e da rotação sobre este ponto (MCRUER, ASHKENAS & GRAHAM, 1973).

As representações dos veículos atuais partem da suposição de corpo rígido de duas maneiras. Na primeira suposição, tem-se a aeronave composta de alguns elementos principais, os quais são necessários para o movimento relativo do corpo, tais como motor, rotor, ou superfícies de controle, incluindo as deformações elásticas que ocorrem em sua estrutura. Na segunda, além destes elementos, são consideradas também outras mudanças envolvendo a dinâmica da estrutura, as quais podem aumentar os graus de liberdade a serem considerados nas equações de movimento, e estão além do escopo do propósito deste trabalho. Como todo movimento é relativo, devemos sempre escolher um sistema de referência não muito complexo e apropriado para descrever o movimento do corpo (MCRUER, ASHKENAS & GRAHAM, 1973).

3.2.2 – Suposição 2 – A Terra está fixa no espaço

O sistema de referência inercial definido para este propósito, isto é, um sistema de referência fixo ou em movimento com velocidade constante com relação à Terra, permite que seja feita uma descrição precisa do movimento do veículo para um período de tempo relativamente pequeno, com o intuito de analisar a orientação e o controle do veículo. Existem limitações práticas quando se deseja analisar navegação para períodos de tempo relativamente longo ou condições atmosféricas anormais (MCRUER, ASHKENAS & GRAHAM, 1973).

Com as duas suposições assumidas, tem-se um sistema de referência no qual as leis de Newton são válidas e um corpo rígido no qual podem ser aplicadas estas leis. Considerando **p** o vetor de quantidade movimento linear de uma aeronave e **H** o vetor de quantidade de movimento angular, representados no sistema de coordenadas inerciais, pela 2ª Lei de Newton tem-se que a taxa de variação do vetor de quantidade de movimento linear é igual à soma de todas as forças aplicadas (MCRUER, ASHKENAS & GRAHAM, 1973):

$$dp / dt = F \tag{3.1}$$

Sabe-se também que a taxa de variação do vetor de quantidade de movimento angular é igual à soma de todos os torques aplicados:

$$dH / dt = M \tag{3.2}$$

Estas equações diferenciais são o ponto inicial para uma descrição completa do movimento de um corpo rígido de um veículo.

Em quase todos os veículos aeronáuticos, algumas partes das forças de impulsão são produzidas devido às forças geradas com o consumo de massa do veículo (combustível); a variação de massa deve ser considerada quando se deseja encontrar a taxa de variação do vetor quantidade de movimento linear. Para o instante t pode-se definir o vetor quantidade de movimento linear como (MCRUER, ASHKENAS & GRAHAM, 1973):

$$p_1 = m \cdot V \tag{3.3}$$

onde, m é a massa e **V** a velocidade do veículo. Em seguida, para o instante $t + \Delta t$, sendo $-\Delta m$ a quantidade de massa que o veiculo perdeu. Para uma velocidade aplicada v_e, o vetor quantidade de movimento linear será:

$$p_2 = (m + \Delta m)\cdot(V + \Delta V) + (-\Delta m)\cdot(V + v_e) \qquad \textbf{(3.4)}$$

Nesta equação $(V + \Delta V)$ e $(m + \Delta m)$ representam respectivamente a velocidade e a massa do veículo no instante $(t + \Delta t)$; e $(V + v_e)$ é o incremento de velocidade com relação ao espaço inercial, correspondente a um incremento no consumo da massa. Este incremento de velocidade depende da área externa, das pressões de sustentação e da velocidade com que o veículo perde sua massa. Sendo assim, a variação do vetor quantidade de movimento do instante t para o instante $(t + \Delta t)$ é (MCRUER, ASHKENAS & GRAHAM, 1973):

$$\begin{aligned} \Delta p &= (p_2 - p_1) \\ &= (m + \Delta m)\cdot(V + \Delta V) - \Delta m\cdot(V + v_e) - m\cdot V \\ &= m\cdot\Delta V - v_e\cdot\Delta m + \Delta m\cdot\Delta V \end{aligned} \qquad \textbf{(3.5)}$$

Dividindo por Δt, e fazendo o limite de $\Delta t \to 0$, é encontrada a variação no vetor quantidade de movimento linear no tempo. O resultado é:

$$dp/dt = \left(m\cdot\frac{dv}{dt}\right) - \left(v_e\cdot\frac{dm}{dt}\right) = F$$

ou

$$m\cdot\frac{dV}{dt} = F + \left(v_e\cdot\frac{dm}{dt}\right) = F + T_e \qquad \textbf{(3.6)}$$

Na equação **(3.6)** a força de tração T_e, é definida por $v_e\cdot(dm/dt)$ e representa apenas a componente de tração resultante diretamente da perda de massa do veículo. Assim, a taxa de variação da quantidade de movimento linear pode ser computada como se a massa fosse constante e o produto da variação de massa por unidade de tempo pela velocidade relativa entre a massa expelida pela exaustão dos propulsores e o veículo fosse uma força externa.

A equação **(3.6)** pode ser deduzida diretamente da equação **(3.1)** quando a tração é resultante de um outro momento que não envolve diretamente a massa do veículo, como o propulsor, por exemplo, (porém neste caso a tração continua sendo uma força externa). Entretanto se a força de tração incluir forças resultantes da perca de massa, então a equação **(3.1)** está correta para veículos trafegando em velocidade relativamente baixa, com relação à velocidade da luz. Conseqüentemente, neste ponto deve-se considerar que a componente de força de tração T_e, está embutida nas forças gerais aplicadas, F (MCRUER, ASHKENAS & GRAHAM, 1973).

Se o sistema de referência inercial for um sistema do tipo da mão-direita de eixos cartesianos, os vetores velocidade V, e força total aplicada F, são resolvidos através de suas componentes u, v e w, e X, Y e Z, ao longo dos eixos x, y e z respectivamente e então os vetores da equação **(3.6)** podem ser escritos pelas seguintes equações escalares (MCRUER, ASHKENAS & GRAHAM, 1973):

$$m\frac{du}{dt} = X$$

$$m\frac{dv}{dt} = Y \qquad \textbf{(3.7)}$$

$$m\frac{dw}{dt} = Z$$

Estas equações descrevem o movimento do centro de massa de uma aeronave, observado no sistema F_B conforme **Figura 3.1**. Até agora, a definição do sistema de referência x, y e z, continua sendo arbitrária – pode ter qualquer orientação e pode mover-se com velocidade constante com relação à terra. Adiante será definido um sistema mais específico.

Para a rotação o raciocínio é equivale ao usado na equação do vetor de quantidade de movimento linear, sendo que no caso o vetor quantidade de movimento a ser considerado é o angular, $\boldsymbol{H}$. A quantidade equivalente à massa m no movimento angular é o momento de inércia, que é representado por $\underline{\underline{I}}$. O vetor quantidade de movimento angular é igual ao produto escalar do momento de inércia $\underline{\underline{I}}$ pelo vetor velocidade angular Ω, ou seja (MCRUER, ASHKENAS & GRAHAM, 1973):

$$\boldsymbol{H} = \underline{\underline{I}} \cdot \Omega \tag{3.8}$$

No caso mais simples, considerando o vetor quantidade de movimento angular na direção de um dos eixos x, y ou z, onde ξ representa genericamente um destes eixos, a expressão se reduz à equação escalar:

$$\boldsymbol{H} = I_{\xi\xi} \cdot \Omega \tag{3.9}$$

onde $I_{\xi\xi}$ é o momento de inércia referente ao eixo escolhido (MCRUER, ASHKENAS & GRAHAM, 1973).

No caso geral, existem nove componentes do vetor quantidade de movimento angular, os quais resultam da diferenciação da expressão, dependendo do sistema de referência escolhido para as equações de movimento. Considere a derivada da equação **(3.8)**,

$$d\boldsymbol{H} / dt = \underline{\underline{I}} \cdot \frac{d\Omega}{dt} + \frac{d\underline{\underline{I}}}{dt} \cdot \Omega = M \tag{3.10}$$

Se os momentos de inércia e as velocidades angulares são calculados em um sistema de coordenadas x, y e z, fixo no espaço, o momento de inércia em cada um dos eixos, geralmente varia conforme o veiculo movimenta-se, com respeito à

origem do eixo. No entanto, se o movimento da aeronave for restringido considerando velocidades constantes, os eixos x, y e z podem ser fixados no centro de massa da aeronave e assim as inércias irão mudar de acordo com a rotação da aeronave sobre cada eixo. Cada alteração na inércia, contribuirá para a taxa de variação do momento angular pelo termo $\left(d\underline{\underline{I}}/dt\right)\cdot\Omega$ na equação **(3.10)**. Devido a isto, as equações resultantes tornam-se muito complicadas devido à variação dos parâmetros no tempo (MCRUER, ASHKENAS & GRAHAM, 1973).

Uma alternativa é escolher um sistema de coordenadas fixo no corpo. Neste tipo de sistema, as propriedades de rotação inercial medidas são constantes, além disso a massa também pode ser considerada constante; outra simplificação é que os termos contendo a variável $\left(dI_{\xi\xi}/dt\right)$ são sempre nulos. O *Sistema de Coordenadas Fixo no Corpo* possui outra vantagem: é o sistema natural de referência para a maioria dos veículos que transitam em fluidos (ar ou água), quanto à representação e descrição de seus movimentos. Por exemplo, um piloto percebe os movimentos de rotação do veículo em torno do centro de gravidade, não de um ponto fixo no espaço. Ele percebe, também, forças de aceleração de acordo com o seu alinhamento com relação ao sistema fixo no corpo da aeronave, e não de acordo com um sistema fixo no espaço. Da mesma maneira, alguns instrumentos de vôo e sensores, são similarmente construídos, ou seja, em geral eles medem movimentos com relação ao sistema fixo no corpo (MCRUER, ASHKENAS & GRAHAM, 1973).

Infelizmente, todas estas vantagens não são obtidas sem algumas perdas. Por exemplo, a forma simples da força inercial na equação **(3.7)** é substituída por uma mais complexa quando a velocidade linear é medida no sistema de coordenadas fixo no corpo. Além disto, algumas forças aplicadas são expressas de maneira mais simples e natural no sistema de referência inercial. A instrumentação designada para navegação e orientação durante o vôo é necessariamente alinhada para medir velocidades lineares e distâncias no espaço

ou no sistema de coordenadas fixo na Terra; o piloto orienta-se pelo sistema fixo na terra tão bem quanto no sistema fixo no veículo. Estes fatores mostram a necessidade de ter-se uma mistura de sistemas de coordenadas, ou seja, um sistema de referência fixo na Terra para as forças e acelerações lineares, e um sistema de referência fixo no corpo para os momentos e movimentos angulares. Assim os dois sistemas são de fato necessários para um tratamento completo do movimento do veículo. Entretanto para aplicações de controle de vôo, as expansões dos lados esquerdos das equações **(3.1)** e **(3.2)** são usualmente realizadas de maneira apropriada para um *sistema de referência fixo no corpo com sua origem no centro de massa do veículo*. Este sistema de eixos é apresentado na **Figura 3.2**, juntamente com suas notações e convenções de sinais, utilizados para identificar velocidades lineares e angulares, e forças e momentos aplicados. O sistema de coordenadas fixo no corpo está orientado na aeronave com x para a frente (proa), y saindo da asa direita e z para baixo, conforme **Figura 3.2**.

3.3 MOMENTOS E PRODUTOS DE INÉRCIAS

Considerando o plano de simetria Cxz da aeronave temos que:

- I_x = Momento de Inércia com relação ao eixo Cx relativo à rolagem;
- I_y = Momento de Inércia com relação ao eixo Cy relativo à arfagem;
- I_z = Momento de Inércia com relação ao eixo Cz relativo à guinada.

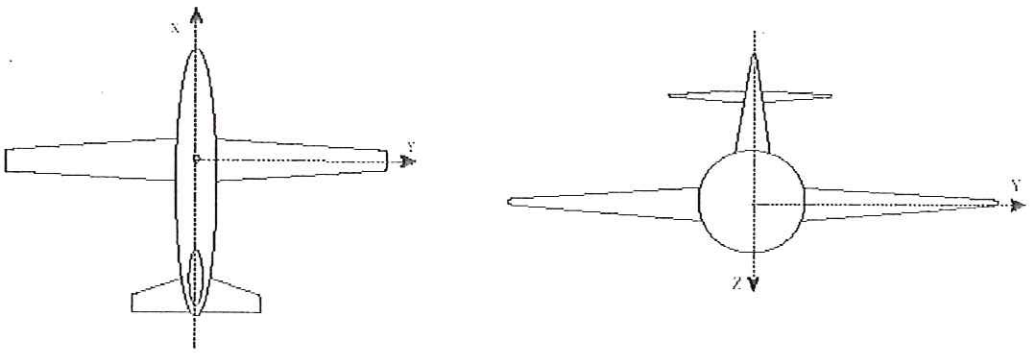

Figura 3.3 – Planos de Simetria da Aeronave.

As expressões dos momentos de inércia são:

$$
\begin{aligned}
I_X &= \int \left(y^2 + z^2\right) dm \\
I_Y &= \int \left(z^2 + x^2\right) dm \\
I_Z &= \int \left(x^2 + y^2\right) dm
\end{aligned}
\qquad \textbf{(3.11)}
$$

e devido ao plano de simetria Cxz, temos que e os Produtos de Inércia são:

$$
\begin{aligned}
I_{XY} &= \int xy \cdot dm = 0 \\
I_{YZ} &= \int yz \cdot dm = 0 \\
I_{XZ} &= I_{ZX} = \int zx \cdot dm
\end{aligned}
\qquad \textbf{(3.12)}
$$

3.4 – ÂNGULOS DE ORIENTAÇÃO PARA UMA AERONAVE

3.4.1 – Orientação e posição de uma aeronave

A posição e a orientação de uma aeronave no espaço são dadas com relação ao sistema de referência fixo na Terra (F_E), . A posição do CG é definida por um vetor r_c no sistema F_E (**Figura 3.1**). A orientação da aeronave é definida por uma série de três rotações consecutivas, que dão origem aos ângulos conhecidos como *ângulos de Euler*, na qual a ordem de rotação é importante(**Figura 3.4**). São usados os *ângulos de Euler*, por ilustrarem melhor a atitude da aeronave do que outros métodos (ROLFE & STAPLES, 1986). São definidos por:

- $\psi =$ ângulo de azimute – rotação em torno do eixo oz_1, mudando os eixos para $Cx_2y_2z_2$;
- $\theta =$ ângulo de arfagem – rotação em torno do eixo oy_2, mudando os eixos para $Cx_3y_3z_3$;
- $\phi =$ ângulo de rolagem – rotação em torno do eixo ox_3, mudando os eixos para posição final $Cxyz$;

A seqüência de rotação é importante e é de acordo com a ordem acima. Para evitar ambigüidades na orientação, são definidos os seguintes limites para os ângulos:

- $-\pi \leq \psi < \pi \quad ou \quad 0 \leq \psi < 2\pi$
- $-\dfrac{\pi}{2} \leq \theta < \dfrac{\pi}{2}$ **(3.13)**
- $-\pi \leq \phi < \pi \quad ou \quad 0 \leq \phi < 2\pi$

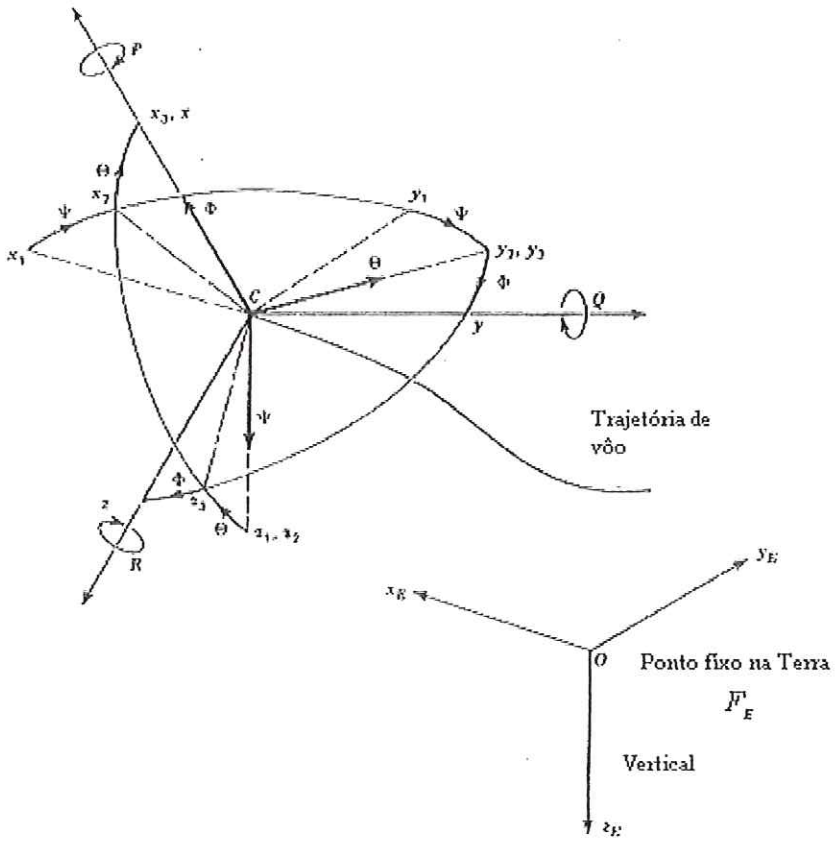

Figura 3.4 – Orientação da aeronave (ETKIN & REID,1996).

3.5 – MATRIZES DE TRANSFORMAÇÃO

Para o caso dos ângulos de *Euler*, a matriz de transformação do sistema F_E para F_B, obedece a seguinte ordem na rotação dos ângulos: ψ, θ e ϕ, resultando. na matriz de transformação L_{EB} descrita da seguinte maneira (ETKIN & REID *et al*, 1996):

$$L_{EB} = \begin{bmatrix} \cos\theta \cdot \cos\psi & \operatorname{sen}\phi \operatorname{sen}\theta \cos\psi - \cos\phi \operatorname{sen}\psi & \cos\phi \operatorname{sen}\theta \cos\psi + \operatorname{sen}\phi \operatorname{sen}\psi \\ \cos\theta \cdot \operatorname{sen}\psi & \operatorname{sen}\phi \operatorname{sen}\theta \operatorname{sen}\psi + \cos\phi \cos\psi & \cos\phi \operatorname{sen}\theta \operatorname{sen}\psi - \operatorname{sen}\phi \cos\psi \\ -\operatorname{sen}\theta & \operatorname{sen}\phi \cos\theta & \cos\phi \cos\theta \end{bmatrix}$$

(3.14)

Assim, a matriz L_{EB} multiplicada pelo vetor velocidade (u, v, w) dado no sistema F_B, resulta nos valores dos termos $\dot{x}_B, \dot{y}_B, \dot{z}_B$, respectivamente. Entretanto, para este trabalho o objetivo é encontrar os valores de $\dot{x}_E, \dot{y}_E, \dot{z}_E$, pois basta integrar os valores de $\dot{x}_E, \dot{y}_E, \dot{z}_E$ e obter a posição ou calcular trajetória da aeronave. Logo as equações para $\dot{x}_E, \dot{y}_E, \dot{z}_E$ são encontradas multiplicando a matriz L_{EB} pelo vetor velocidade V_B.

$$\begin{bmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{bmatrix} = L_{EB} \cdot V_B^E \qquad \textbf{(3.15)}$$

CAPITULO 4

EQUAÇÕES DE MOVIMENTO

Para calcular, simular ou analisar o movimento de uma aeronave, é necessário primeiramente definir um modelo matemático para a aeronave, assim como os subsistemas desta. Conforme já comentado, o modelo matemático de uma aeronave em vôo é bastante complexo, visto que envolve diversos movimentos relativos a corpos rígidos e corpos elásticos. Assim, pode-se encontrar na literatura diversos modelos desenvolvidos bastante precisos e realistas, para possibilitar uma simulação que apresente resultados próximos do comportamento real (ETKIN & REID *et al*, 1996).

O modelo de dinâmica de vôo de uma aeronave consiste basicamente na sua descrição geométrica e de massa, juntamente com as equações de movimento e das influências externas que atuam sobre ela.

Neste capítulo, será apresentado o conjunto de equações de movimento de uma aeronave, ou seja, o modelo matemático que descreve o comportamento dinâmico de uma aeronave. O modelo escolhido para a simulação é o modelo não linear, isto devido ao fato deste modelo ser atualmente bastante usado (ZYSKOWSKI, 2003).

4.1 – CONTEXTUALIZAÇÃO NO ESPAÇO.

O ambiente definido para o desenvolvimento das equações de movimento da aeronave foi apresentado no capítulo anterior. Resumidamente definiu-se que:

- Seja um sistema de coordenadas $Oxyz$, onde O é o centro de gravidade da aeronave. Os eixos são fixados na aeronave e movimentam-se com ela (sistema de referência F_B). Ox e Oz estão no plano de simetria da aeronave, com Oz para baixo; Oy na direção de estibordo;

- u, v e w são as componentes da velocidade linear do centro de gravidade na direção dos eixos Ox, Oy e Oz respectivamente; e p, q e r as componentes da velocidade angular do sistema $Oxyz$ na direção dos eixos Ox, Oy e Oz respectivamente. O sentido positivo das velocidades angulares é no sentido horário sobre os respectivos eixos;

- m é a massa da aeronave, e I_x, I_y e I_z os momentos de inércia da aeronave sobre os eixos Ox, Oy e Oz, e I_{yz}, I_{zx} e I_{xy} os produtos de inércia com relação aos planos O_{yz}, O_{zx} e O_{xy} respectivamente;

- As forças externas sobre a aeronave são definidas por X, Y e Z sobre os eixos Ox, Oy e Oz respectivamente; e os momentos externos sobre a aeronave são L, M e N sobre os eixos Ox, Oy e Oz respectivamente.

O modelo matemático que representa a aeronave em vôo, e que conseqüentemente será usado para simular a situação proposta pelo presente trabalho, é composto por 12 equações diferenciais ordinárias não lineares que serão apresentadas a seguir. Neste modelo, serão representadas na aeronave as componentes das forças, dos momentos e das velocidades angulares e lineares.

4.2 – EQUAÇÕES DE MOVIMENTO PARA UM CORPO RÍGIDO

Para dedução das equações, é necessário destacar algumas considerações que abrangem o modelo matemático que será utilizado. Assim serão utilizadas as seguintes hipóteses para o desenvolvimento das equações, algumas já foram citadas, mas julga-se importante relembrar neste ponto:

- A aeronave é considerada um corpo rígido, com distribuição de massa constante;
- A Terra é considerada uma superfície plana e fixa no espaço;
- A atmosfera encontra-se em repouso em relação à Terra;
- O escoamento de ar sobre a aeronave é considerado estável;
- O vetor da gravidade (g) é um vetor radial;
- A aeronave possui um plano de simetria (Plano Ozx), conforme já visto;
- As perturbações da condição de vôo estável são assumidas relativamente pequenas.

As deduções das equações são realizadas a partir da 2ª lei de Newton, conforme descrito no Capítulo 3.

4.3 – DESENVOLVIMENTO DAS EQUAÇÕES DE MOVIMENTO DE UMA AERONAVE

Para o desenvolvimento das equações de movimento de uma aeronave, devemos definir os seguintes termos:

- L = sustentação;
- D = arrasto;

- V = velocidade;
- ρ = densidade (no caso, do ar);
- S = área de referência;
- C_L = coeficiente de sustentação;
- C_D = coeficiente de arrasto;
- b = envergadura de referência da asa;
- c = corda média aerodinâmica.

Basicamente três forças atuam sobre uma aeronave quando ela se desloca no espaço (CALNCY, 1975):

- Tração;
- Peso;
- Resultante aerodinâmica.

A Resultante aerodinâmica normalmente é decomposta em sustentação (L) e arrasto (D), em suas componentes nos eixos z e x respectivamente.

Considerando o ar como um fluido incompressível e não viscoso, sabe-se que a força aerodinâmica é proporcional à densidade (ρ), ao quadrado da velocidade (V^2) e à área de referência (S). Portanto as forças aerodinâmicas L e D são definidas pelas seguintes equações:

$$L = C_L \frac{1}{2} \rho V^2 S \qquad \text{e} \qquad D = C_D \frac{1}{2} \rho V^2 S \tag{4.1}$$

Assim pode-se considerar que são quatro as forças que agem sobre uma aeronave em vôo: Sustentação, Arrasto, Tração e Peso. Para facilitar nosso trabalho, essas forças são decompostas nos três eixos do sistema de referência, e

escritas em função de seus respectivos coeficientes aerodinâmicos. Assim, decompondo todas as forças e somando-as tem-se que:

$$X = C_x \frac{1}{2} \rho V^2 S$$

$$Y = C_y \frac{1}{2} \rho V^2 S \qquad \textbf{(4.2)}$$

$$Z = C_z \frac{1}{2} \rho V^2 S$$

Além disso, as forças geram momentos, pois o ponto de aplicação das forças não é o mesmo. Logo, estes momentos também podem ser decompostos e somados, resultando em:

$$L = C_l \frac{1}{2} \rho V^2 S b$$

$$M = C_m \frac{1}{2} \rho V^2 S \bar{c} \qquad \textbf{(4.3)}$$

$$N = C_n \frac{1}{2} \rho V^2 S b$$

Reagrupando essas forças e momentos, tem-se um conjunto de seis equações, uma para cada grau de liberdade da aeronave, e que são válidas para um movimento com aceleração retilínea nula:

$$\begin{aligned} &\sum X = 0 \\ &\sum Z = 0 \\ &\sum M = 0 \\ &\\ &\sum L = 0 \\ &\sum Y = 0 \\ &\sum N = 0 \end{aligned} \qquad \textbf{(4.4)}$$

As equações em X, L e Z são conhecidas como grupo de equações simétricas, pois pertencem ao plano de simetria da aeronave que se refere ao movimento longitudinal e representam respectivamente:

- A soma algébrica de todas as componentes ao longo do eixo x é zero, ou seja, não há aceleração nesta direção;

- A soma algébrica de todas as componentes ao longo do *eixo y* é zero, ou seja, não há aceleração nesta direção;

- A soma de todos os momentos de arfagem em relação ao CG é zero.

O segundo grupo de equações refere-se aos movimentos laterais, ou seja, movimentos assimétricos. Similarmente representam:

- Força Lateral Y;

- Momento de Rolagem L; e

- Momento de Guinada N.

Para um vôo acelerado, as equações (4.4) são válidas, entretanto é acrescida a constante *ma* (massa x aceleração – 2ª lei de Newton) no lado direito da igualdade. Desenvolvendo os somatórios das equações e as acelerações nos três eixos, e já considerando que devido ao plano de simetria C_{xz}, os produtos de inércia $I_{xy} = I_{yz} = 0$, resultam em (ETKIN & REID *et al*, 1996):

$$X - mgsen\theta = m(\dot{u} + qw - rv) \tag{4.5}$$

$$Y + mg\cos\theta sen\phi = m(\dot{v} + ru - pw) \tag{4.6}$$

$$Z + mg\cos\theta\cos\phi = m(\dot{w} + pv - qu) \tag{4.7}$$

$$L = I_x\dot{p} - I_{zx}(\dot{r} + pq) - (I_y - I_z)qr \tag{4.8}$$

$$M = I_y\dot{q} - I_{zx}(r^2 - p^2) - (I_z - I_x)rp \tag{4.9}$$

$$N = I_z\dot{r} - I_{zx}(\dot{p} - qr) - (I_x - I_y)pq \tag{4.10}$$

Resolvendo estas equações para $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}$ e $\dot{r}$:

$$\dot{u} = \frac{X}{m} - g \cdot sen\theta - q \cdot w + r \cdot v \tag{4.11}$$

$$\dot{v} = \frac{Y}{m} + g \cdot \cos\theta \cdot sen\phi - r \cdot u + p \cdot w \tag{4.12}$$

$$\dot{w} = \frac{Z}{m} + g \cdot \cos\theta \cdot \cos\phi - p \cdot v + q \cdot u \tag{4.13}$$

$$\dot{p} = \frac{L + I_{zx}(\dot{r} + p \cdot q) + (I_y - I_z)q \cdot r}{I_x} \tag{4.14}$$

$$\dot{q} = \frac{M + I_{zx}(r^2 - p^2) + (I_z - I_x)r \cdot p}{I_y} \tag{4.15}$$

$$\dot{r} = \frac{N + I_{zx}(\dot{p} - qr) + (I_x - I_y)p \cdot q}{I_z} \tag{4.16}$$

Para transformar as componentes de velocidade angular da aeronave do sistema F_E para o sistema F_B, projetam-se as componentes das velocidades angulares $\dot{\phi}$, $\dot{\theta}$ e $\dot{\psi}$, ao longo dos eixos *x*, *y* e *z*, resultando em (ETKIN & REID *et al*, 1996):

$$p = \dot{\phi} - \dot{\psi}\,\mathrm{sen}\,\theta \tag{4.17}$$

$$q = \dot{\theta}\cos\phi + \dot{\psi}\cos\theta\,\mathrm{sen}\,\phi \tag{4.18}$$

$$r = -\dot{\theta}\,\mathrm{sen}\,\phi + \dot{\psi}\cos\theta\cos\phi \tag{4.19}$$

Resolvendo estas equações para $\dot{\phi}$, $\dot{\theta}$ e $\dot{\psi}$ tem-se que:

$$\dot{\theta} = q\cos\phi - r\,\mathrm{sen}\,\phi \tag{4.20}$$

$$\dot{\psi} = q\,\mathrm{sen}\,\phi\sec\theta + r\cos\phi\sec\theta \tag{4.21}$$

$$\dot{\phi} = p + q\,\mathrm{sen}\,\phi\tan\theta + r\cos\phi\tan\theta \tag{4.22}$$

4.4 – DESENVOLVIMENTO DAS EQUAÇÕES PARA OBTENÇÃO DA POSIÇÃO

Define-se a posição da aeronave com relação à Terra (sistema F_E) pelas coordenadas x_E, y_E e z_E, representando respectivamente distância horizontal, distância lateral e altitude de vôo. Portanto, determinando essas variáveis, temos a posição ou a trajetória da aeronave no decorrer do tempo (ETKIN & REID *et al*, 1996). O equacionamento necessário para determinar essas coordenadas é:

$$\begin{bmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{bmatrix} = L_{EB} \cdot \begin{bmatrix} u \\ v \\ w \end{bmatrix} \text{ ou seja}$$

$$\begin{bmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{bmatrix} = \begin{bmatrix} \cos\theta \cdot \cos\psi & sen\phi sen\theta \cos\psi - \cos\phi sen\psi & \cos\phi sen\theta \cos\psi + sen\phi sen\psi \\ \cos\theta \cdot sen\psi & sen\phi sen\theta sen\psi + \cos\phi \cos\psi & \cos\phi sen\theta sen\psi - sen\phi \cos\psi \\ -sen\theta & sen\phi \cos\theta & \cos\phi \cos\theta \end{bmatrix} \cdot \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

(4.23)

Logo, as três equações que definem a posição da aeronave são:

$$\begin{aligned} \dot{x}_E &= u \cdot \cos\theta \cdot \cos\psi + v \cdot (sen\phi \cdot sen\theta \cdot \cos\psi - \cos\phi \cdot sen\psi) \\ &\quad + w \cdot (\cos\phi \cdot sen\theta \cdot \cos\psi + sen\phi \cdot sen\psi) \end{aligned} \qquad \textbf{(4.24)}$$

$$\begin{aligned} \dot{y}_E &= u \cdot \cos\theta \cdot \sin\psi + v \cdot (\sin\phi \cdot \sin\theta \cdot \sin\psi + \cos\phi \cdot \cos\psi) \\ &\quad + w(\cos\phi \sin\theta \sin\psi - \sin\theta \cos\psi) \end{aligned} \qquad \textbf{(4.25)}$$

$$\dot{z}_E = -u \cdot \sin\theta + v \cdot \sin\phi \cdot \cos\theta + w \cdot \cos\phi \cdot \cos\theta \qquad \textbf{(4.26)}$$

Assim, tem-se o conjunto de equações que descrevem o movimento completo de uma aeronave (ETKIN & REID *et al*, 1996), onde:

- u, v e w representam as velocidades lineares horizontal, lateral e vertical da aeronave, respectivamente;

- p, q e r representam as velocidades angulares de rolagem, arfagem e guinada respectivamente;

- X, Y e Z representam as forças aerodinâmicas nas direções horizontal, lateral e vertical da aeronave, respectivamente;

- L, M e N representam os momentos aerodinâmicos de rolagem, arfagem e guinada respectivamente;

- ϕ, θ e ψ representam os ângulos de rolagem, arfagem e guinada respectivamente (ângulos de Euller);

- x_E, y_E e z_E representam a posição da aeronave com relação a F_E;

- m representa a massa total da aeronave;

- I_x, I_y e I_z representam os momentos de inércia em torno de x, y e z, respectivamente;

- I_{xy}, I_{yz} e I_{zx} representam os produtos de inércia com relação aos planos Oxy, Oyz e Ozx.

4.5 – EQUAÇÕES DE FORÇAS E MOMENTOS AERODINÂMICOS

O método e o desenvolvimento das equações de forças e momentos aerodinâmicos que atuam sobre uma aeronave em vôo escolhido, é o mesmo citado por ETKIN & REID, *et al* (1996), onde pode ser encontrado de forma mais detalhada. Para este trabalho, interessa apenas definir a forma final das equações, as quais serão posteriormente implementadas no software.

Este modelo considera que as forças e momentos aerodinâmicos são funções de valores instantâneos das perturbações na velocidade, nos ângulos das superfícies de controle e das suas derivadas.

Considerando A uma reação aerodinâmica qualquer, temos que:

$$\Delta A = A_u u + A_{\dot{u}} \dot{u} + A_{\ddot{u}} \ddot{u} + \ldots + A_{\delta_a} \delta_a + A_{\dot{\delta}_a} \dot{\delta}_a \qquad \textbf{(4.27)}$$

onde: $A_u = \dfrac{\delta A}{\delta u}$, etc.

As derivadas do tipo A_u são chamadas de derivadas de estabilidade dimensionais da aeronave. Assim as forças e momentos aerodinâmicos que atuam na aeronave são funções das componentes de velocidade, e podem ser representadas pelas seguintes equações (ETKIN & REID *et al*, 1996):

$$\Delta X = X_u \cdot \Delta u + X_w \cdot \Delta w + X_{\dot{w}} \cdot \Delta \dot{w} + X_q \cdot \Delta q + X_{\delta_e} \cdot \Delta \delta_e \qquad \textbf{(4.28)}$$

$$\Delta Y = Y_v \cdot \Delta v + Y_p \cdot \Delta p + Y_r \cdot \Delta r + Y_{\delta_a} \cdot \Delta \delta_a + Y_{\delta_r} \cdot \Delta \delta_r \qquad \textbf{(4.29)}$$

$$\Delta Z = Z_u \cdot \Delta u + Z_w \cdot \Delta w + Z_{\dot{w}} \cdot \Delta \dot{w} + Z_q \cdot \Delta q + Z_{\delta_e} \cdot \Delta \delta_e \qquad \textbf{(4.30)}$$

$$\Delta L = L_v \cdot \Delta v + L_p \cdot \Delta p + L_r \cdot \Delta r + L_{\delta_a} \cdot \Delta\delta_a + L_{\delta_r} \cdot \Delta\delta_r \quad \textbf{(4.31)}$$

$$\Delta M = M_u \cdot \Delta u + M_w \cdot \Delta w + M_{\dot{w}} \cdot \Delta\dot{w} + M_q \cdot \Delta q + M_{\delta_e} \cdot \Delta\delta_e \quad \textbf{(4.32)}$$

$$\Delta N = N_v \cdot \Delta v + N_p \cdot \Delta p + N_r \cdot \Delta r + N_{\delta_a} \cdot \Delta\delta_a + N_{\delta_r} \cdot \Delta\delta_r \quad \textbf{(4.33)}$$

Onde:

- X_u, X_w, $X_{\dot{w}}$, X_q, X_{δ_e}, Y_v, Y_p, Y_r, Y_{δ_a}, Y_{δ_r}, Z_u, Z_w, $Z_{\dot{w}}$, Z_q, Z_{δ_e}, L_v, L_p, L_r, L_{δ_a}, L_{δ_r}, M_u, M_w, $M_{\dot{w}}$, M_q, M_{δ_e} N_v, N_p, N_r, N_{δ_a} e N_{δ_r} representam as derivadas aerodinâmicas de estabilidade;

- Δu, Δv e Δw representam os valores instantâneos das perturbações nas velocidades lineares;

- Δp, Δq e Δr representam os valores instantâneos das perturbações nas velocidades angulares;

- $\Delta\delta_e$, $\Delta\delta_a$ e $\Delta\delta_r$ variações nos controles;

Considerando que a aeronave se encontra em equilíbrio, é necessário acrescentar os valores iniciais das forças aerodinâmicas nas equações. Estas forças são representadas pelas seguintes equações:

$$X_0 = m \cdot g \sin\theta_0 \quad \textbf{(4.34)}$$

$$Y_0 = 0 \quad \textbf{(4.35)}$$

$$Z_0 = -m \cdot g \cos\theta_0 \tag{4.36}$$

Desta forma, as forças aerodinâmicas X, Y e Z são substituídas nas equações de **(4.11)** a **(4.13)** e são definidas por:

$$X = \Delta X - X_0 \tag{4.37}$$

$$Y = \Delta Y - Y_0 \tag{4.38}$$

$$Z = \Delta Z - Z_0 \tag{4.39}$$

CAPITULO 5

ANÁLISE DE REQUISITOS PARA CONSTRUÇÃO DO SOFTWARE SIMAERO

Para construção de um software, é necessário antes definir quais os requisitos, características, parâmetros de entrada e saídas deste software. Neste capítulo será feita uma análise e descrição dos requisitos básicos e necessários para o desenvolvimento do software SIMAERO.

Em linhas gerais o software SIMAERO será capaz de simular matematicamente uma aeronave em vôo, recebendo como entradas valores numéricos, retornando dados numéricos e gráficos cartesianos. Isto será feito implementando o modelo matemático apresentado no capítulo anterior. Além disso, o software será implementado de modo que possa facilitar uma futura integração com uma plataforma de simulador de vôo.

5.1 – INTRODUÇÃO

O programa SIMAERO é um software desenvolvido utilizando a linguagem de programação orientada a objetos C++. O software é um simulador de vôo no qual o usuário entra com os valores iniciais para algumas variáveis que dizem respeito à dinâmica e ao controle do vôo, e obtém como saída a resposta do comportamento dinâmico da aeronave. Em outras palavras, o software tem como

objetivo principal implementar e resolver as equações de dinâmica e controle de vôo, de modo a simular o comportamento dinâmico de uma aeronave em vôo, assim como retornar as respostas do comportamento da aeronave após acionamento de seus controles (leme, profundor e aileron).

A princípio a idéia é apenas implementar e obter as respostas numericamente e em gráficos cartesianos, de modo que as saídas possam ser transmitidas em um trabalho futuro, através das portas de saída do PC, para uma plataforma de simulação. Sendo assim, a interface gráfica necessária neste momento deve permitir ao usuário entrar com os valores das variáveis e apresentar os resultados numéricos em gráficos cartesianos para análise dos resultados.

A **Figura 5.1** representa um Diagrama de Fluxo de Dados do SIMAERO. Neste diagrama o arquivo texto com os resultados representa o 'Banco de Simulações Realizadas'. O Bloco simular Aeronave representa o software SIMAERO.

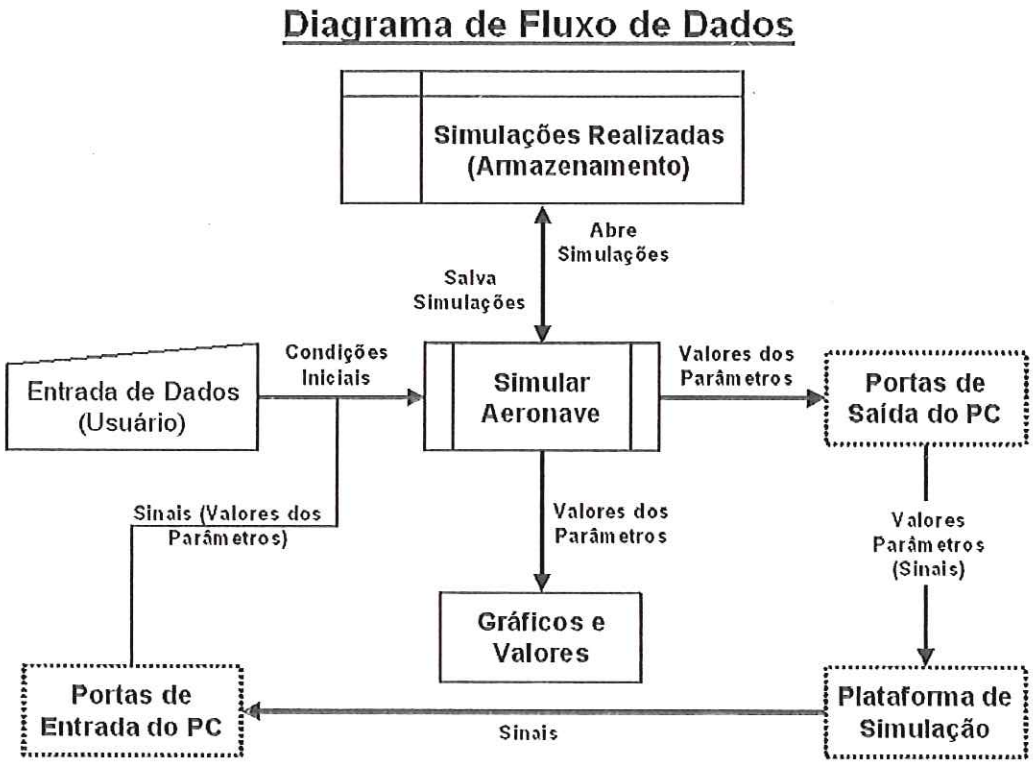

Figura 5.1 – Diagrama de Fluxo de Dados do SIMAERO.

5.2 – DESCRIÇÃO DA INFORMAÇÃO

5.2.1 – Reconhecimento do Problema

A partir do modelo matemático apresentado no capítulo anterior, foi desenvolvido um software que realiza a simulação da dinâmica de vôo de uma aeronave. O software consiste na implementação e na resolução deste sistema de equações, e é desenvolvido na linguagem de programação C++. O programa resolve este sistema através do método interativo de integração numérica *Runge-Kutta* de 4ª ordem. A simulação é realizada a partir dos seguintes dados da aeronave: dados geométricos, dados aerodinâmicos e derivadas de estabilidade, os quais são passados como parâmetros de entrada para o software. Como parâmetros de saída, o programa retorna os valores das seguintes variáveis:

velocidade linear, velocidade angular, aceleração linear, aceleração angular e orientação da aeronave. Estes resultados são apresentados em valores numéricos e armazenados em arquivos, e desenhados em gráficos cartesianos. Os resultados obtidos da simulação descrevem o comportamento da aeronave. A análise do comportamento é realizada comparando simulações realizadas para diferentes condições de vôo e variando as entradas nos controles da aeronave (aileron, profundor e leme).

5.3 – DESCRIÇÃO FUNCIONAL

5.3.1 – Avaliação e síntese da solução

O objetivo deste trabalho consiste em implementar e resolver as equações de dinâmica e controle de vôo em um algoritmo computacional. Para isto, utilizamos a linguagem de programação C++, como já foi dito. A estrutura de arquivos do software está implementada de acordo com a **Figura 5.2** adiante.

O programa principal está no arquivo *simaero.cpp*, o qual contém todas as funções de controle da interface gráfica e dos eventos internos do programa, ou seja, eventos ou funções que não são controlados pelo usuário. Além disso, também faz parte do arquivo a rotina principal que controla toda a simulação, a qual é disparada por interação do usuário ao pressionar o botão *Calcular*. Este evento resulta na chamada à função *Button_calcularClick*, a qual denominaremos aqui simplesmente por *Calcular*. Neste arquivo também são implementadas as funções *rk4*, *gera_f*, *carrega_controles*, *carrega_matrizes* e que serão detalhadas mais adiante.

Além disto, são feitas também chamadas a outras funções que estão em outros arquivos denominados bibliotecas. São eles: *definiçao.h e edo.h*. Estes são

referenciados em uma outra biblioteca chamada *simaero.h*, que é a biblioteca principal do software e é referenciada pelo programa principal *simaero.cpp*. Nesta biblioteca, além das bibliotecas externas (ou seja, desenvolvidas especificamente para este programa) também estão referenciadas as bibliotecas internas (aquelas que acompanham o compilador para eventos não controlados pelo usuário).

O arquivo *definicao.h*, contém a declaração das constantes, variáveis, vetores e matrizes. Neste arquivo, estes termos são criados e recebem como valor inicial zero. Apenas as constantes recebem valores específicos e diferentes de zero. Além disso, também estão implementadas neste arquivo as funções *calcula_densidade* e *calcula_vel_total*.

O arquivo *edo.h*, é composto das seguintes funções:

- *calcula_forca_momento*, a qual calcula as forças e momentos aerodinâmicos da aeronave;

- *calcula_derivadas*, a qual contém a implementação das equações diferenciais ordinárias não-lineares que descrevem a dinâmica da aeronave, e que devem ser resolvidas para simular o vôo.

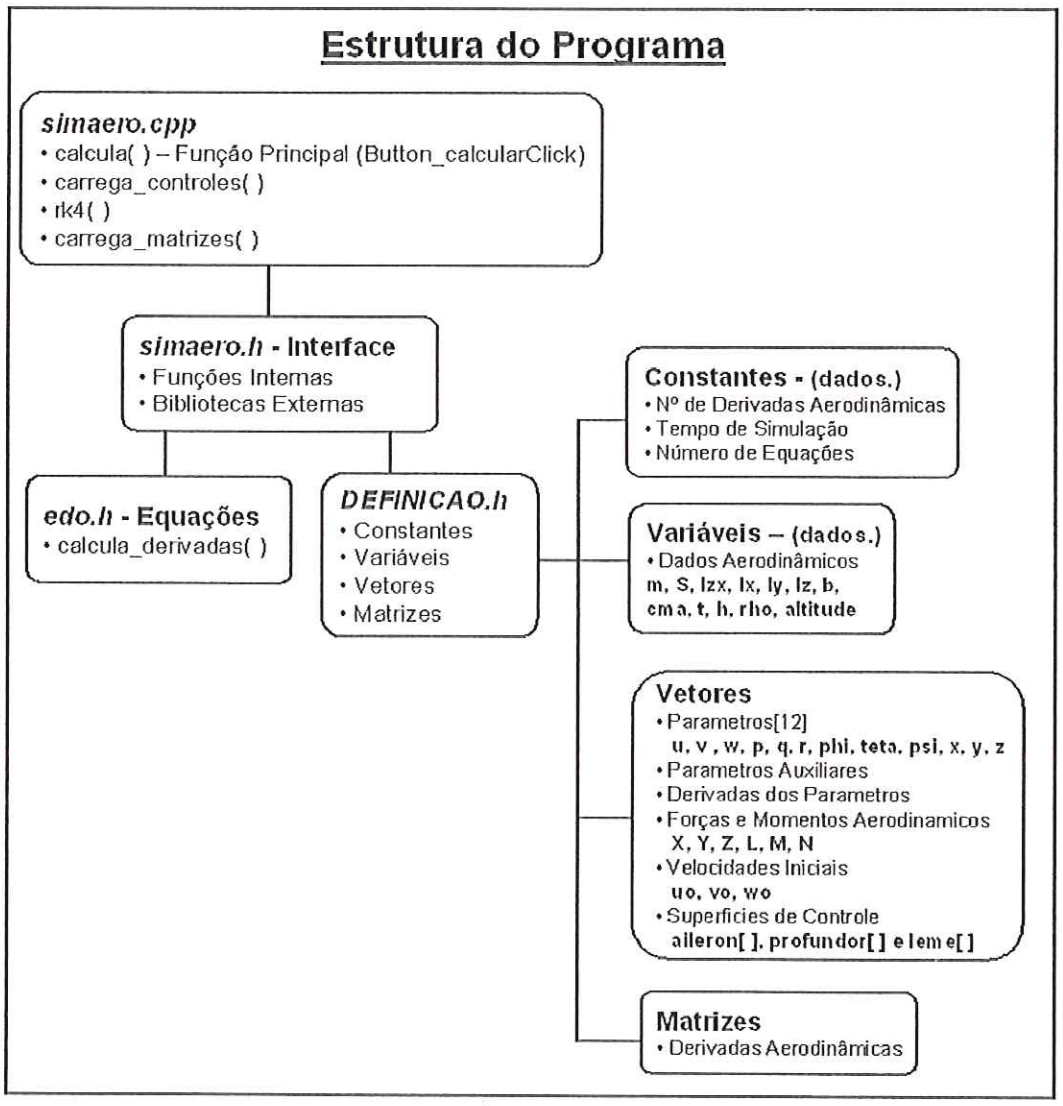

Figura 5.2 – Estrutura do Software SIMAERO.

5.4 – DEFINIÇÃO DAS VARIÁVEIS E CONSTANTES

No programa, serão usadas as seguintes constantes:

- **g** – representa a aceleração da gravidade que será considerada constante com valor de $9{,}80665\, m/seg^2$;
- **t** – representa o tempo total em segundos;
- **i** – instante atual da simulação;
- **h** – representa o valor do passo de integração em segundos (valor assumido como padrão no SIMAERO é **h** = 0,01s);

Também serão usadas as seguintes variáveis durante a simulação:

- **m** – representa a massa da aeronave em $[Kg]$;
- **S** – representa a área de referência da aeronave em $[m^2]$;
- **IZX** – representa o produto de inércia com relação ao plano de coordenadas cartesianas OZX em $[Kg \cdot m^2]$;
- **IZY e IXY** – representam os produtos de inércia com relação aos planos OZY e OXY respectivamente, porém não estão presentes no software, pois possuem valor zero devido à simetria da aeronave (conforme considerado no desenvolvimento das equações);
- **IX** – representa o momento de inércia com relação ao eixo de coordenadas cartesianas OX e será medido em $[Kg \cdot m^2]$;
- **IY** – representa o momento de inércia com relação ao eixo de coordenadas cartesianas OY e será medido em $[Kg \cdot m^2]$;
- **IZ** – representa o momento de inércia com relação ao eixo de coordenadas cartesianas OZ e será medido em $[Kg \cdot m^2]$;
- **b** – representa a envergadura de referência da asa em $[m]$;
- **cma** – representa a corda média aerodinâmica de referência em $[m]$;

EESC-USP Biblioteca CAMPUS DE S. CARLOS

- **altitude** – representa a altitude em que a aeronave se encontra em $[m]$;
- **alt_ref** – altitude de referência (no caso ao nível do mar) em $[m]$;
- **rho** – densidade relativa do ar em $[Kg/m^3]$;
- **vel_total** – representa a velocidade total da aeronave em $[m/s]$.

Todos estas constantes e variáveis fazem parte de um registro com o nome *dados*, logo cada uma das variáveis e constantes é acessada através deste registro (*dados.g*, por exemplo, acessa a constante que contém o valor da aceleração da gravidade).

Além das constantes e variáveis que são usadas nas funções e procedimentos no decorrer do programa, será feito uso também de outras variáveis na forma de vetores e matrizes que serviram de auxílio aos cálculos.

Os vetores usados são:

- **parametro[12]** – este vetor armazena os parâmetros: $u, v, w, p, q, r, \phi, \theta, \psi, x, y, z$, nesta ordem, os quais são calculados nas equações de movimento. As unidades para cada um dos parâmetros são definidas de acordo com o Sistema Internacional de Unidades, sendo:

  - $u = [m/s]$;
  - $v = [m/s]$;
  - $w = [m/s]$;
  - $p = [rad/s]$;
  - $q = [rad/s]$;
  - $r = [rad/s]$;
  - $\phi = [rad]$;

- $\theta = [rad]$;
- $\psi = [rad]$;
- $x = [m]$;
- $y = [m]$; e
- $z = [m]$.

Neste vetor são armazenadas as soluções para cada um destes parâmetros, e isto é realizado na função *rk4*, onde acontece a integração numérica;

- **parametro_aux[12]** – representa também um vetor para os parâmetros $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z, mas agora utilizados como variáveis auxiliares no armazenamento da solução no começo do "iésimo" passo de integração;

- **parametro_der[12]** – este vetor armazena o valor das derivadas dos parâmetros em função do tempo $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$, nesta ordem e que são calculados durante a integração;

- **vel_ini[3]** – armazena os valores iniciais das componentes da velocidade;

- **forca_momen[6]** – este vetor representa as forças *X*, *Y* e *Z*, e os momentos *L*, *M* e *N* que atuam na aeronave, e seus valores são calculados e armazenados nesta ordem;

- **controles[3][Tempo]** – armazena os valores a cada instante da simulação para o acionamento dos controles da aeronave: profundor (controles[0][tempo]), aileron (controles[1][tempo]) e leme (controles[2][tempo]);

As matrizes utilizadas no software são as seguintes:

- **derivadas** – esta matriz armazena os valores de todas as derivadas aerodinâmicas na seguinte ordem X_u, X_w, $X_{\dot{w}}$, X_q, X_{δ_e}, Y_v, Y_p, Y_r, Y_{δ_a}, Y_{δ_r}, Z_u, Z_w, $Z_{\dot{w}}$, Z_q, Z_{δ_e}, L_v, L_p, L_r, L_{δ_a}, L_{δ_r}, M_u, M_w, $M_{\dot{w}}$, M_q, M_{δ_e} N_v, N_p, N_r, N_{δ_a} e N_{δ_r}, que serão usadas para resolução das equações, uma para cada linha da matriz. As colunas representam diferentes condições de vôo;

Julga-se necessário descrever também, antes de citar as funções e rotinas, alguns eventos importantes realizados pelo programa e que estão implementados de maneira implícita:

- *Inicialização das variáveis, matrizes e vetores* – todas as variáveis, vetores e matrizes recebem como valor inicial 0 (zero), evitando que o programa comece sua execução com as variáveis definidas com valores indesejados. Isto é realizado nas rotinas de criação das variáveis no arquivo *definicao.h*;

- *Definir condições iniciais dos parâmetros* – para este trabalho foi definido que para as condições iniciais dos parâmetros a serem calculados, será atribuído o valor inicial de 0 (zero), com exceção das velocidades u_0, v_0 e w_0 que podem ser definidas pelo usuários com valores diferentes de zero. . Como isto já é feito no momento em que as variáveis são criadas, não se faz necessário mais nenhum procedimento. Assim os valores para as condições iniciais do problema para os parâmetros $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z,, já estão definidos antes de iniciar a simulação. Isto poderá futuramente ser alterado, permitindo que o usuário defina as

condições iniciais com entradas pelo teclado ou através de um arquivo;

- *Definição das velocidades inicias* – as velocidades iniciais u_0, v_0 e w_0 também recebem o valor inicial 0 (zero) quando são criadas, porém podem ser definidas pelo usuário na tela principal antes de iniciar a simulação;

- *Carregar dados geométricos da aeronave* – os dados geométricos são passados ao programa através de um arquivo texto no início da execução do programa. Isto é feito no momento em que são carregadas as matrizes das derivadas aerodinâmicas.

- *Salvar simulação* – os valores obtidos a cada passo da simulação são gravados em um arquivo texto no momento em que são calculados. São armazenados respectivamente o tempo atual (t) e os parâmetros com um espaço de tabulação entre eles na seguinte ordem: $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z.

Será feita agora uma descrição das funções e rotinas utilizadas no programa. São elas:

- *calcula* – esta função chamada no programa pelo nome *Button_calcularClick,* e é a função principal que é executada por interação do usuário. Nesta função define-se a condição de vôo que será simulada; criam-se as variáveis, vetores e matrizes; define-se o tempo de simulação, passo de integração, atitude para a condição de vôo escolhida; carrega-se a matriz de coeficientes aerodinâmicos e as entrados para os controles. Por fim, faz-se também a chamada à função *rk4* e ao final da simulação é feita a limpeza das variáveis;

- *carrega_matrizes* – carrega os dados aerodinâmicos e geométricos da aeronave, e os valores das derivadas aerodinâmicas lidos de um arquivo texto;

- *carrega_controles* – atribui valores para os ângulos de deflexão do profundor, aileron e leme, os quais são armazenados no vetor controles[3][tempo], onde as linhas 0, 1 e 2 representam respectivamente profundor, aileron e leme. Os valores dos comandos são passados pelo usuário em graus e o programa converte para radianos para realizar os cálculos; e o tempo representa cada instante (em segundos) da simulação. A princípio, a entrada nos controles é uma função do tipo "degrau". Isto significa que, define-se um único valor (em radianos) para uma faixa de tempo limitada. Futuramente poderá ser alterada para outros tipos de entrada nos controles;

- *rk4* – esta função contém a implementação do método Runge-Kutta de 4ª ordem, que é o método de integração numérica utilizado no cálculo dos valores dos parâmetros que estão sendo avaliados. Além disto, são realizados todos os cálculos da simulação através de chamadas às demais funções, onde estão implementados estes cálculos;

- *gera_f* – esta função realiza chamadas as funções *calcula_derivadas* e *calcula_forca_momento* que são usadas durante a integração;

- *calcula_forca_momento* – calcula as forças e momentos aerodinâmicos *X, Y, Z, L, M e N,* utilizando as derivadas aerodinâmicas. Os valores calculados nesta função serão usados

para o cálculo das derivadas dos parâmetros $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z;

- *calcula_derivadas* – esta função utiliza os dados e valores calculados anteriormente para calcular as derivadas de $u, v, w, x, y, z, p, q, r, \phi, \theta, \psi$, no tempo atual. Estes são posteriormente utilizados para realizar a integração numérica, e encontrar o valor dos parâmetros $u, v, w, x, y, z, p, q, r, \phi, \theta, \psi$, que descrevem o comportamento da aeronave em vôo em cada instante.

O fluxograma da **Figura 5.3** mostra como o SIMAERO se comporta do início ao fim de sua execução, ou seja, as tarefas que o software realiza durante o processo de simulação. O laço de repetição relativo à simulação e à solução do sistema de equações ocorre dentro do bloco *Integração Numérica*.

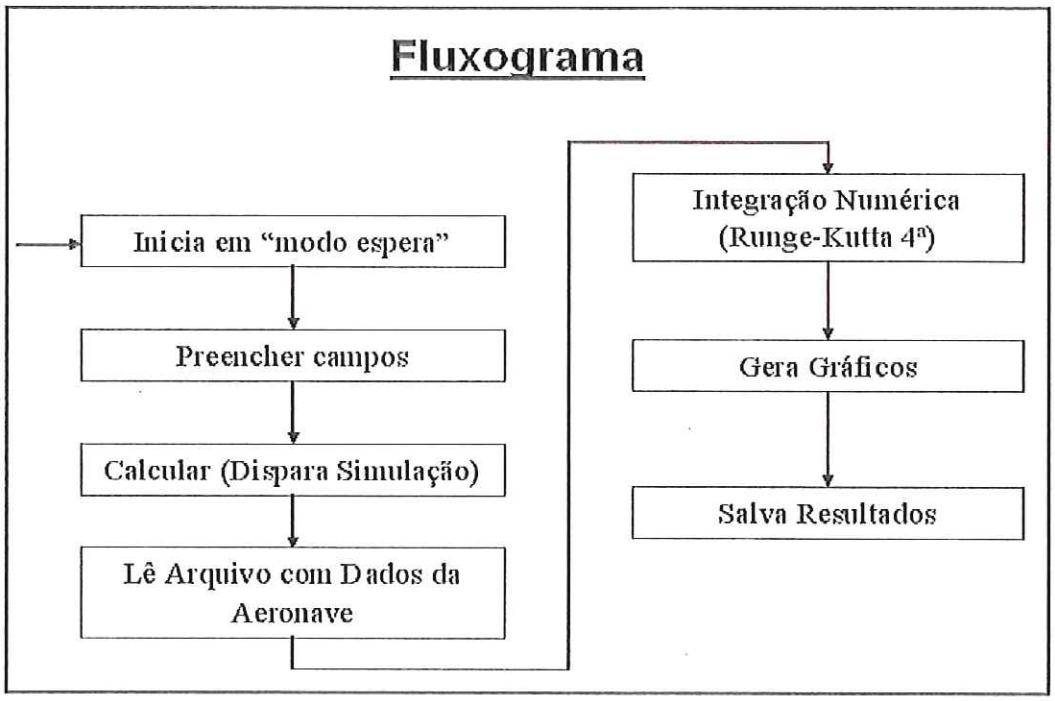

Figura 5.3 – Fluxograma do Software SIMAERO.

5.5 – ALGORITMO DESCRITIVO DO PROGRAMA SIMAERO

Será apresentada agora a seqüência de eventos que ocorre durante a simulação do programa. Primeiramente em forma de texto descritivo em seguida para melhor compreensão será apresentado um pseudo-algoritmo descritivo.

5.5.1 – Seqüência de eventos da execução do software

A princípio, após ser executado, o programa fica em "modo de espera", ou seja, aguardando por um evento do usuário. Primeiramente o usuário define a condição de vôo e entra com os respectivos valores iniciais para a condição escolhida. É necessário definir também o passo de integração e o tempo de simulação. Os valores de entrada para os controles (aileron, profundor e leme) também são definidos pelo usuário, porém deverão ser do tipo "degrau", ou seja, é estipulado um valor fixo para um determinado intervalo de tempo. Todos estes valores são passados para o programa através da interface gráfica.

Os valores para as derivadas aerodinâmicas e para os dados geométricos são lidos pelo programa de um arquivo texto previamente definido. Em seguida inicia-se a simulação pressionando o botão *Calcula* (também disponível na interface).

Para a simulação da dinâmica de vôo, ou seja, para a solução do sistema de equações, a altitude, a densidade relativa do ar, e a velocidade total da aeronave, devem ser previamente definidas.

Após serem carregadas as derivadas aerodinâmicas, são calculados os valores das Forças e Momentos Aerodinâmicos, usando os valores iniciais dos parâmetros u, v, w, p, q, e r, (ou valores encontrados anteriormente durante as

iterações da integração numérica). Em seguida são calculados os valores para as equações de $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$, ou seja, para as derivadas dos parâmetros.

Feito isto, inicia-se a integração numérica usando o método *Runge-Kutta*, repetindo-se os dois últimos passos para cada iteração do método, para encontrar os valores dos parâmetros $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z, no instante atual. Isto é realizado para cada instante do intervalo de tempo definido. A cada passo da iteração, os valores encontrados para os parâmetros, são utilizados na iteração seguinte.

A cada passo da iteração, são gravados em arquivos textos os resultados para cada um dos parâmetros e desenhados em gráficos cartesianos, também disponíveis na interface.

5.5.2 – Algoritmo Descritivo

O programa é executado e fica em modo de espera até que seja disparado algum evento pelo usuário. Após o usuário ter preenchido todos os campos da tela, deve-se pressionar o botão *Calcular*, dando assim início à seguinte seqüência de eventos:

1° Passo - Cria e carrega as variáveis, vetores e matrizes globais:

dados – Estrutura de dados (registro) com todos os dados da aeronave (*m, S, IZX, IX, IY, IZ, b, t, h, altitude, rho* e *c*; atribuindo-se o valor 0 para todos os campos);

vet – Estrutura de dados (registro) com os vetores *parâmetros*[], *parametros_der*[], *parametros_aux*[], *forca_momen*[] e *controles*[][];

mat – Matriz para armazenar as Derivadas Aerodinâmicas;

2º Passo - Cria os vetores locais:

aux – do tipo inteiro, com valor inicial 0, utilizado para definir a posições nos vetores;

x – do tipo ponto flutuante (número real), com valor inicial 0, representa o instante atual da simulação;

variacao – do tipo ponto flutuante, com valor inicial *dados.t/dados.h* (onde *t* equivale ao tempo total de simulação em segundos e *h* ao passo de integração);

f1[n], f2[n], f3[n], f4[n] – do tipo ponto flutuante representando os parâmetros $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$, usados como variáveis auxiliares durante o processo integração numérica pelo método Runge-Kutta. (n é uma constante definida na biblioteca DEFINICAO.h com valor 12, que é o numero de equações do modelo matemático).

3º Passo – Define valores iniciais para *parametro*[], *parametro_aux*[], *parametro_der*[]:

Passo 3.1 – *parametro*[] recebe os valores dos parâmetros $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z, para a condição inicial de vôo;

Passo 3.2 – *parametro_aux*[] recebe *parametro*[];

Passo 3.3 – *parametro_der*[] recebe os valores de *parametro_der*[] dos parâmetros $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z, para a condição inicial de vôo;

4º Passo – Executa os sub-passos abaixo dentro de um laço de repetição, afim de realizar a integração numérica de acordo com o método de Runge-Kutta de 4ª Ordem, variando de 0 até *variação* (= *dados.t/dados.h*. Este valor para *variação* é definido pelo método Runge-Kutta, assim como o algoritmo abaixo – Passos 4.1 – 4.16).

Passo 4.1 – Resolve as equações de *X, Y, Z, L, M e N,* utilizando os valores de *paramtero*[] e armazena em *forca_momen*[]. **(Equações 4.30-32, 4.36-38).**

Passo 4.2 – Resolve as equações de $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$, utilizando *forca_momen*[]. **(Equações 4.11-16, 4.20-25).** Armazena resultado em *parametro_der*[].

Passo 4.3 – *f1*[] recebe o valor do vetor *parametro_der*[];

Passo 4.4 – *parâmetro*[] recebe o valor de:
parametro_aux[] + dados.h/2 * *f1*[];

Passo 4.5 – Resolve as equações de *X, Y, Z, L, M e N,* utilizando os valores de *paramtero*[] e armazena em *forca_momen*[]. **(Equações 4.30-32, 4.36-38).**

Passo 4.6 – Resolve as equações de $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$, utilizando *forca_momen*[]. **(Equações 4.11-16, 4.20-25).** Armazena resultado em *parametro_der*[].

Passo 4.7 – *f2*[] recebe o valor do vetor *parametro_der*[];

Passo 4.8 – *parâmetro*[] recebe o valor de:
parametro_aux[] + dados.h/2 * *f2*[];

Passo 4.9 – Resolve as equações de *X, Y, Z, L, M e N,* utilizando os valores de *paramtero*[] e armazena em *forca_momen*[]. **(Equações 4.30-32, 4.36-38).**

Passo 4.10 – Resolve as equações de $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$, utilizando *forca_momen*[]. **(Equações 4.11-16, 4.20-25).** Armazena resultado em *parametro_der*[].

Passo 4.11 – *f3*[] recebe o valor do vetor *parametro_der*[];

Passo 4.12 – *parâmetro*[] recebe o valor de:
parametro_aux[] + dados.h/2 * *f3*[];

Passo 4.13 – Resolve as equações de *X, Y, Z, L, M e N,* utilizando os valores de *paramtero*[] e armazena em *forca_momen*[]. **(Equações 4.30-32, 4.36-38).**

Passo 4.14 – Resolve as equações de $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$, utilizando *forca_momen*[]. **(Equações 4.11-16, 4.20-25).** Armazena resultado em *parametro_der*[].

Passo 4.15 – *f4*[] recebe o valor do vetor *parametro_der*[];

Passo 4.16 – *parâmetro*[] recebe o valor de:
parametro_aux[] + ((dados.h/6) * (*f1*[] + (2**f2*[]) + (2**f3*[]) + (*f4*[])));

Passo 4.17 – Incrementa instante atual x com o valor de h;

Passo 4.18 – Desenha o ponto valores calculado para os parâmetros $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z, para o instante atual x, e que estão armazenados no vetor *parâmetro*[].

Passo 4.19 – *parametro_aux*[] = *parametro*[];

Passo 4.20 – Salva o valor de x e de *parametro*[] no arquivo;

Passo 4.21 – Se valor x igual ao valor de t finaliza sai do laço de repetição, senão retornar Passo 4.1;

5° Passo – Limpa as variáveis, vetores e matrizes, e encerra o processo de simulação.

O fluxograma da **Figura 5.4** representa a implementação computacional do método de integração numérica Runge-Kutta de 4ª Ordem, para integração do sistema de equações, ou seja dos parâmetros $\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}$, e $\dot{z}$. A **Figura 5.4** representa o 3° e 4° passos do algoritmo descrito acima.

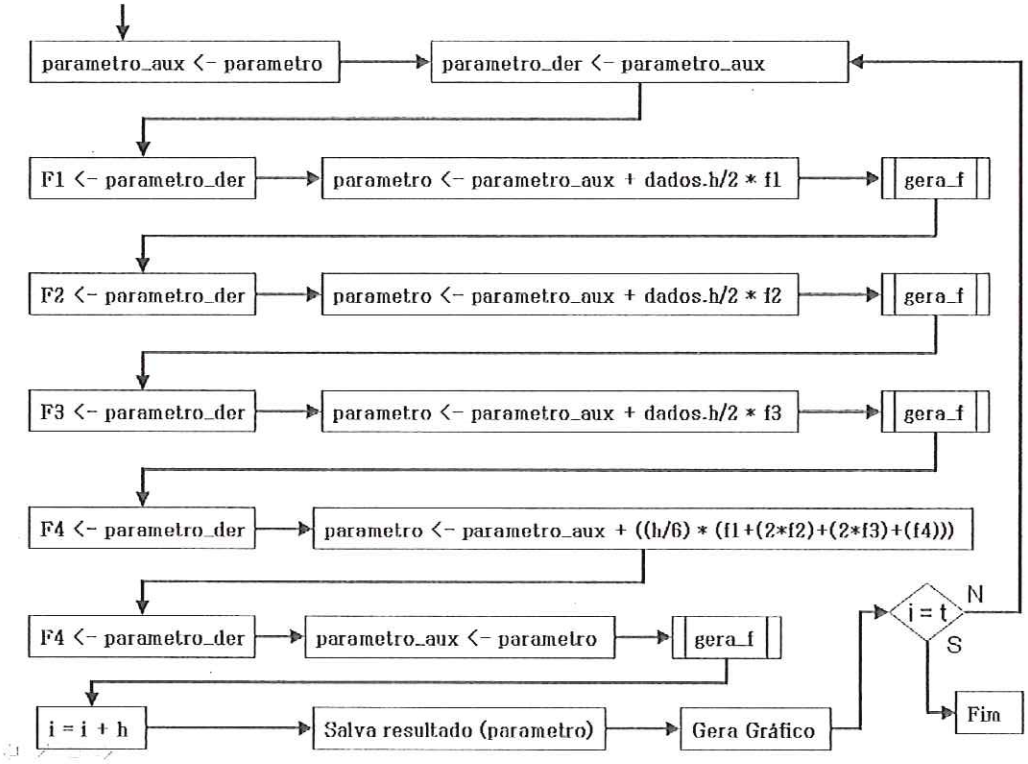

Figura 5.4 – Implementação computacional do método de integração numérica Runge-Kutta de 4ª Ordem.

Como já citado foi criada uma interface gráfica para que o usuário possa interagir com o software, ou seja, informar ao software o nome do arquivo contendo as informações relativas à geometria da aeronave a ser simulada, assim como os valores para as derivadas aerodinâmicas. Através desta interface também são definidos o tempo de simulação, passo de integração e entradas nos controles da aeronave.

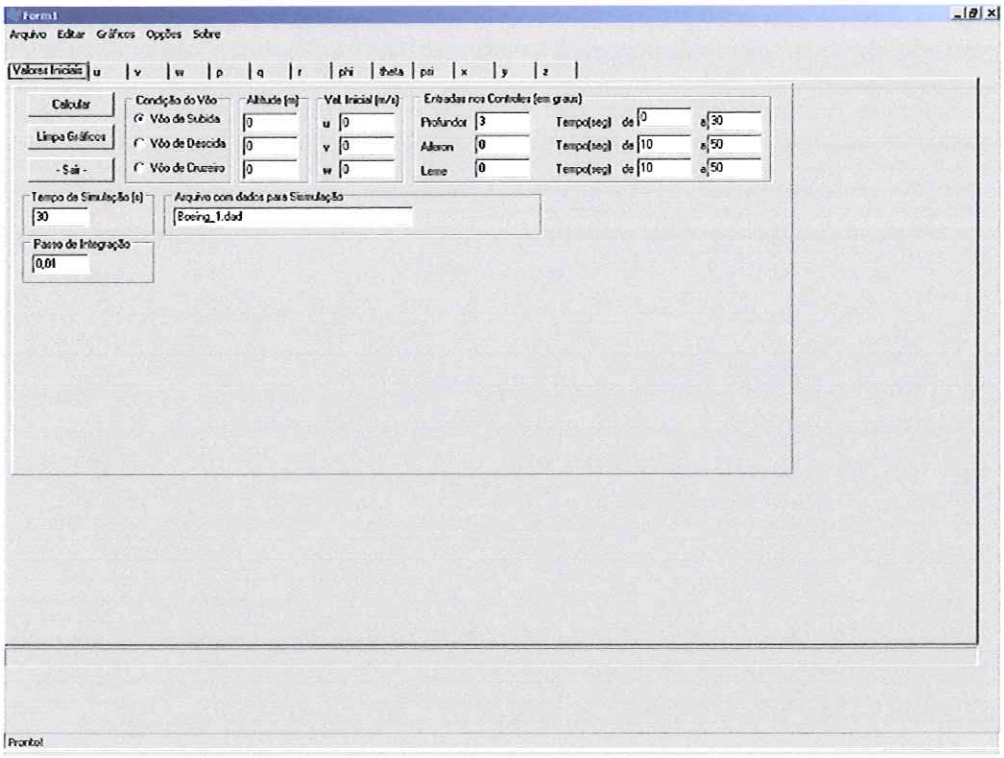

Figura 5.5 – Interface gráfica do software SIMAERO

Após realizada a simulação de uma aeronave, os gráficos de $u, v, w, p, q, r, \phi, \theta, \psi, x, y$, e z estarão desenhados em cada uma das abas na parte superior, bastando selecionar cada uma delas para visualizar o respectivo gráfico. O arquivo contendo o resultado da simulação estará gravado na mesma pasta em que se encontra o executável do software com o nome 'result.dad'.

CAPITULO 6

RESULTADOS

Como já foi descrito, o software SIMAERO realiza a simulação da dinâmica de vôo de uma aeronave, a partir dos dados geométricos e das derivadas de estabilidade dimensionais de uma aeronave. Como resultando desta simulação obtêm-se os valores das velocidades lineares e angulares, dos ângulos de Euller e da posição da aeronave com relação à posição inicial. Com o intuito de testar o software SIMAERO, foi realizada uma simulação para um caso da aeronave Boeing 747-101, em condição estável de vôo, para um vôo de cruzeiro a 12.103 m de altitude, a uma velocidade de 235.9 m/s. Para a simulação, esta condição é considerada como sendo a condição de equilíbrio da aeronave para o instante inicial. Os dados desta aeronave foram retirados de (ETKIN & REID, 1996) e convertidos para o Sistema Internacional de Unidades. A simulação foi realizada utilizando os dados da **Tabelas 6.1.**

Como se pode ver, a **Tabela 6.1** a seguir apresenta os dados para os valores de altitude, número de Mach (**M**), velocidade inicial (**V**), peso (**W**), massa (**m**), inércia com relação os eixos X (**Ix**), Y (**Iy**), e Z (**Iz**) respectivamente, e produto de inércia com relação ao plano ZX (**Izx**).

Altitude (m)	12103
M	0,8
V (m/s)	235,9
W (Kg * m/s^2)	2,8317E+06
m (Kg)	2,8374E+05
Ix (Kg/m^2)	2,477E+07
Iy (Kg/m^2)	4,490E+07
Iy (Kg/m^2)	6,730E+07
Izx (Kg/m^2)	-2,120E+06

Tabela 6.1 - Dados do Boeing 747-101 (ETKIN & REID, 1996).

A **Tabela 6.2** abaixo apresenta os valores das derivadas de estabilidade dimensional do Boeing 747-101, separadas por derivadas longitudinais e derivadas laterais.

Derivadas Longitudinais

	X (N)	Z (N)	M (m * N)
u (m/s)	-1,982+E03	-2,595+E04	1,593+E04
w (m/s)	4,025+E03	-9,030+E04	-1,563+E04
q (rad/s)	0	-4.524+E05	-1,521+E07
w' (m/s^2)	0	1,909+E03	-1,702+E04
Delta e (rad)	-1,653E+00	-1,579E+06	-5,204E+07

Derivadas Laterais

	Y (N)	L (m * N)	N (m * N)
v (m/s)	-1,610+E04	-3,062+E0	2,131+E0
p (rad/s)	0	-1,076+E0	-1,330+E0
r (rad/s)	0	9,925+E0	-8,934+E0
Delta a (rad)	0	-1,368E-02	-1,973E-04
Delta r (rad)	1,146E-01	6,976E-03	-1,257E-01

Tabela 6.2 – Derivadas de Estabilidade Dimensionais do Boeing 747-101.

Estes dados são passados para o Software SIMAERO através de um arquivo texto, de onde o software lê e realiza as simulações.

A simulação que deu origem aos gráficos a seguir foi realizada com os dados da **Tabela 6.1** e **Tabela 6.2** fazendo as devidas conversões para o Sistema Americano de Unidades. Os valores de entradas para os controles da aeronave e o tempo de simulação estão de acordo com o exemplo encontrado em (ETKIN & REID, 1996), facilitando a comparação entre os gráficos dos resultados obtidos da simulação pelo SIMAERO com os resultados apresentados em (ETKIN & REID, 1996).

Na interface original do software as respostas são exibidas separadas cada uma em um gráfico, porém com a finalidade de facilitar a comparação com as respostas da literatura, os gráficos das respostas geradas pelo SIMAERO foram apresentados da mesma maneira que os da literatura. Além disto, como o software apresenta os resultados no Sistema Internacional de Unidades, também para facilitar a comparação, os resultados deste caso foram convertidos para o Sistema Americano de Unidades. Logo, na **Figura 6.1** e na **Figura 6.4,** as velocidades lineares estão em pés/s. Para os demais resultados as unidades permanecem inalteradas.

Assim, após executar uma simulação, obtêm-se como resultados os gráficos dos seguintes parâmetros:

- Velocidade Linear u (m/s);
- Velocidade Linear v (m/s);
- Velocidade Linear w (m/s);
- Velocidade Angular p (rad/s);
- Velocidade Angular q (rad/s);
- Velocidade Angular r (rad/s);
- Deslocamento Angular ϕ (graus);

- Deslocamento Angular θ (graus);
- Deslocamento Angular ψ (graus);
- Deslocamento em x (metros);
- Deslocamento em y (metros);
- Deslocamento em z (metros).

Assim, a simulação foi realizada para um tempo total de 30 segundos com uma entrada do tipo degrau de $-15°$ de deflexão no aileron do instante zero ao instante final de simulação. Os demais controles permanecem fixos com valor zero.

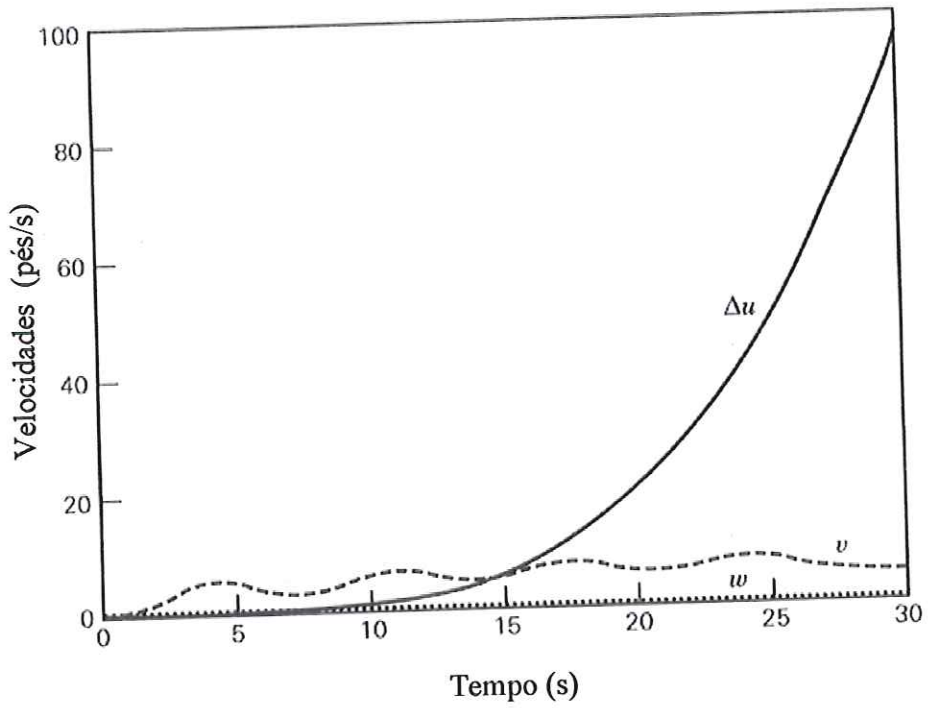

Figura 6.1 – Componentes das velocidades lineares (ETKIN & REID, 1996).

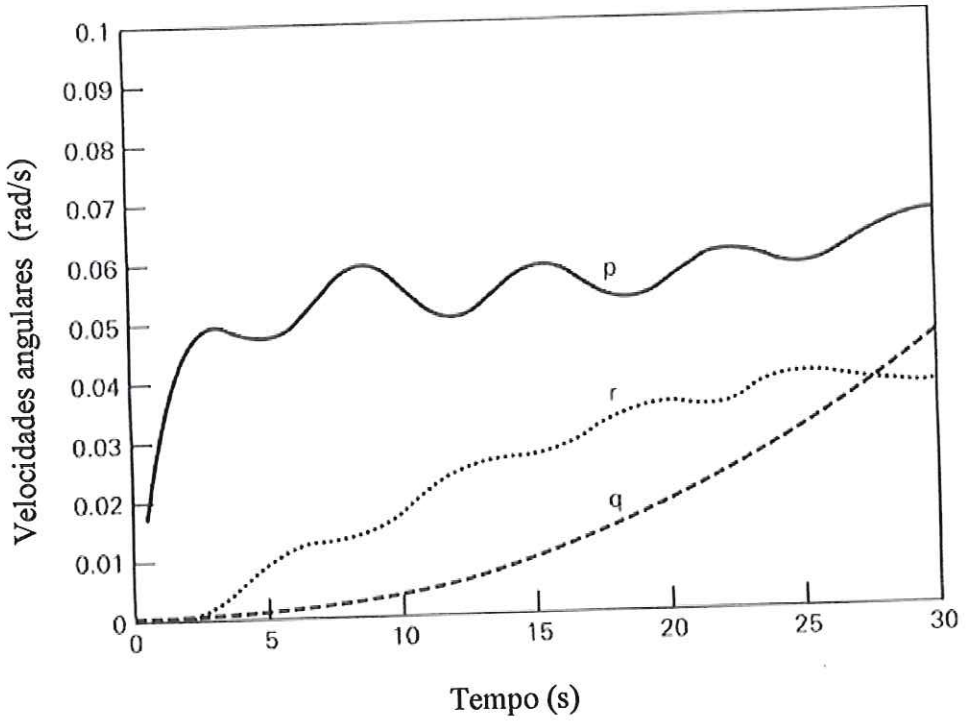

Figura 6.2 – Componentes das velocidades angulares (ETKIN & REID, 1996).

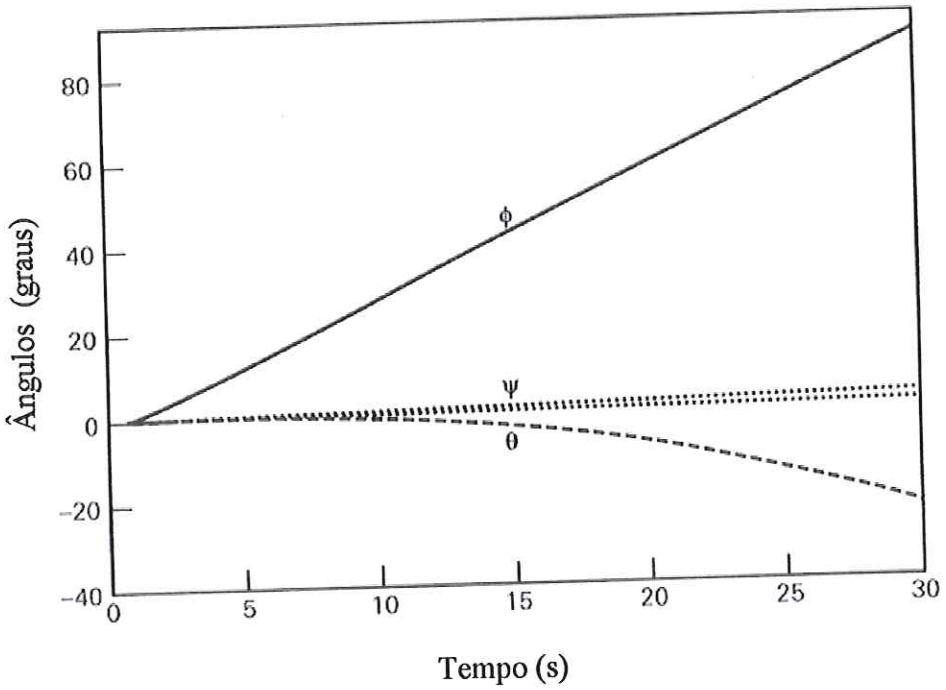

Figura 6.3 – Ângulos de Orientação (ETKIN & REID, 1996).

Analisando os gráficos, a principal característica que pode-se perceber é um rápido incremento na componente de velocidade angular p conforme **Figura 6.2**, juntamente com aumento do ângulo de rolagem ϕ, que chega quase a 90° em 30 segundos (**Figura 6.3**). Por outro lado, a componente de velocidade angular q, assim como o ângulo de guinada ψ permanecem com variações pequenas ao longo do tempo. À medida que a aeronave realiza o movimento de rolagem, com a sustentação praticamente igual ao peso, a componente vertical de força aerodinâmica diminui rapidamente, como conseqüência levando θ a um valor negativo, e aumentando a velocidade. Após 30 segundos pode-se perceber um aumento de quase 10% com relação à velocidade inicial (**Figura 6.1**).

Importante lembrar que na **Figura 6.1**, temos o gráfico de Δu que representa a componente de velocidade u menos a velocidade inicial u_0 (velocidade com relação à condição inicial, ou seja, antes do início simulação) que neste caso não é igual a zero. Logo o valor não representa a velocidade final u da aeronave, e sim o incremento nesta componente com relação à condição inicial.

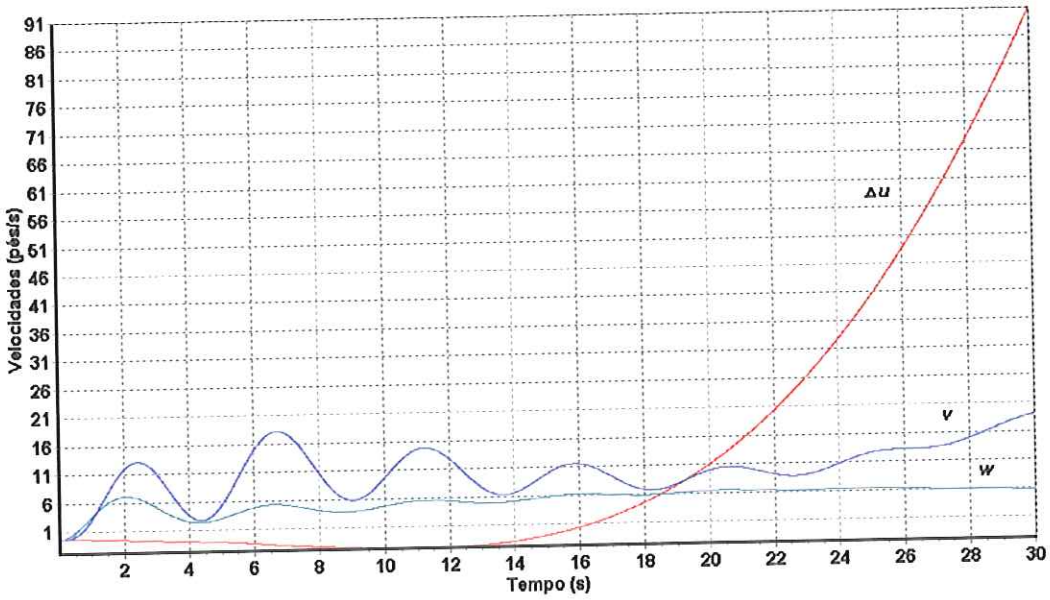

Figura 6.4 – Componentes das velocidades lineares – Resultado do SIMAERO.

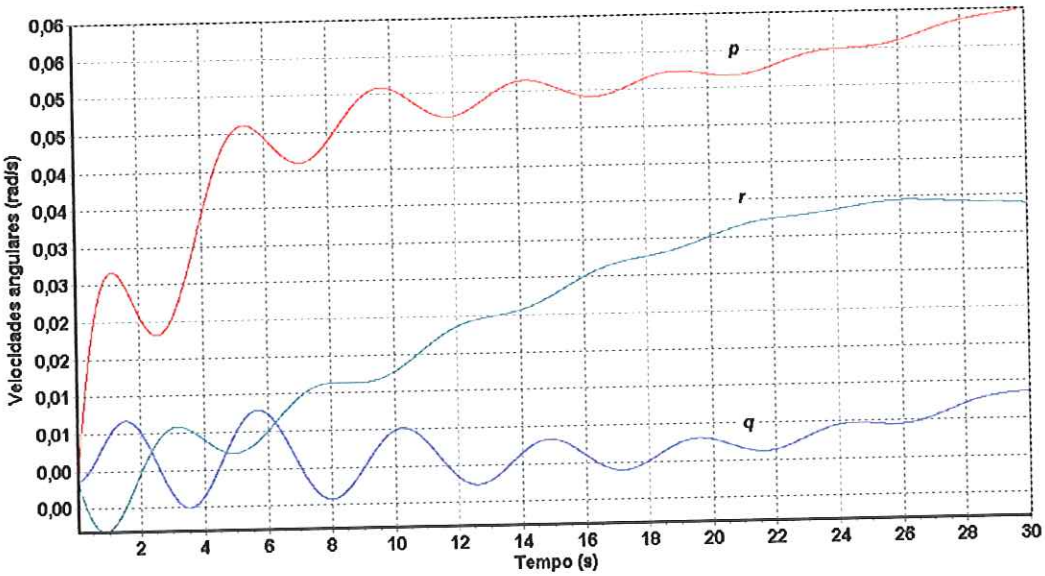

Figura 6.5 – Componentes das velocidades angulares – resultado do SIMAERO.

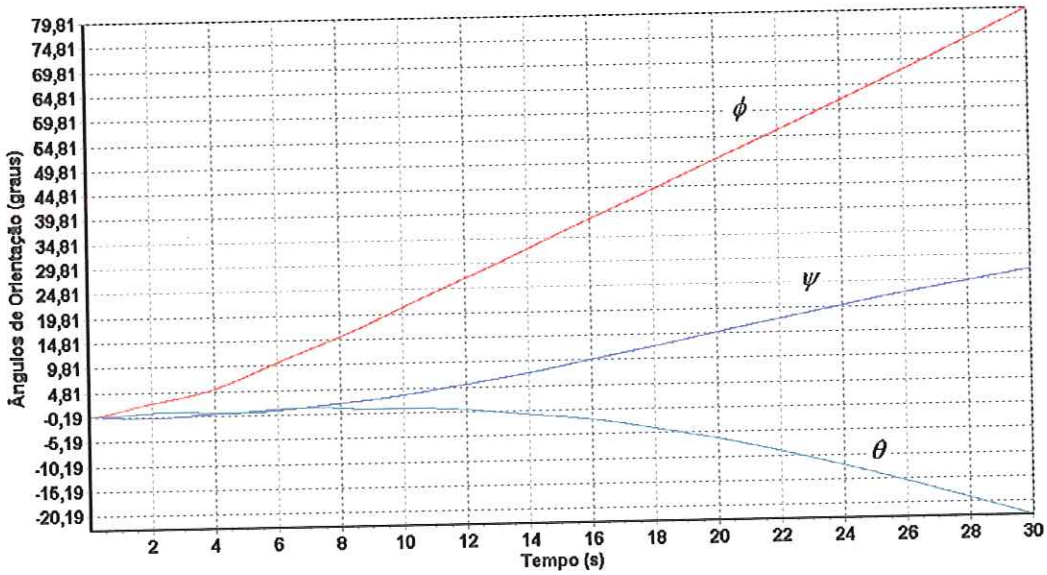

Figura 6.6 – Ângulos de Orientação – resultado do SIMAERO.

Os gráficos das **Figuras 6.4, 6.5** e **6.6,** correspondem respectivamente aos gráficos das **Figura 6.1, 6.2** e **6.3**. Porém estes gráficos foram gerados com os resultados oriundos da simulação pelo SIMAERO, na condição de vôo descrita

anteriormente. Como os resultados entre os gráficos são bastante próximos, pode-se considerar que os resultados obtidos da simulação com o SIMAERO estão de acordo com o esperado.

A simulação acima mostra o comportamento da aeronave com respeito ao movimento lateral da mesma, pois neste exemplo o controle da aeronave acionado foi o aileron, com deflexão de –15°. A fim de ilustrar o comportamento longitudinal da aeronave e avaliar as respostas do SIMAERO, considerando que seus resultados já foram validados no exemplo anterior, foi realizado um segundo caso para a aeronave Boeing 747-101, também em condição estável de vôo, para um vôo de cruzeiro a 12.103 m de altitude, agora a uma velocidade de 235.9 m/s.

Neste exemplo os gráficos estão exatamente como implementado na versão final (considerada para este trabalho) do SIMAERO, portanto os resultados foram apresentados no Sistema Internacional de Unidades. Os dados de entrada para a simulação estão na **Tabela 6.3** e **Tabela 6.4** adiante. Estes dados correspondem ao Caso III apresentado em (ETKIN & REID, 1996), com as devidas conversões para o Sistema Intenacional de Unidades.

Altitude (m)	12192
M	0.9
V (m/s)	265
W (Kg * m/s^2)	2,832E+06
m (Kg)	2,887E+05
Ix (Kg/m^2)	2,636E+07
Iy (Kg/m^2)	4,794E+07
Iy (Kg/m^2)	7,199E+07
Izx (Kg/m^2)	1,405E+06

Tabela 6.3 - Dados do Boeing 747-101 (ETKIN & REID, 1996).

Derivadas Longitudinais

	X (N)	Z (N)	M (m * N)
u (m/s)	-2,70E+01	-5,74E+01	-4,64E+02
w (m/s)	2,15E+01	-5,43E+02	-1,41E+04
q (rad/s)	0	-9,09E+03	-2,98E+06
w' (m/s^2)	0	8,32E+00	-1,19E+03
Delta e (rad)	1,06E+03	-2,52E+04	-9,08E+06

Derivadas Laterais

	Y (N)	L (m * N)	N (m * N)
v (m/s)	-8,21E+01	-6,44E+03	1,28E+04
p (rad/s)	0	-1,88E+06	-1,32E+05
r (rad/s)	0	1,18E+06	-1,64E+06
Delta a (rad)	0	-7,62E+05	1,09E+05
Delta r (rad)	5,47E+03	5,06E+05	-4,96E+06

Tabela 6.4 – Derivadas de Estabilidade Dimensionais do Boeing 747-101.

Para esta simulação foi considerado um tempo total de simulação de 1.000 segundos, aplicando um ângulo de deflexão no profundor de 3° no instante 10s, o qual fica acionado até o instante 50s, retornando então seu ângulo de deflexão para zero graus. Os demais controles permanecem inalterados. Os gráficos de v, p, r, ψ, ϕ e y foram omitidos pois como o movimento é longitudinal estes parâmetros permanecem inalterados.

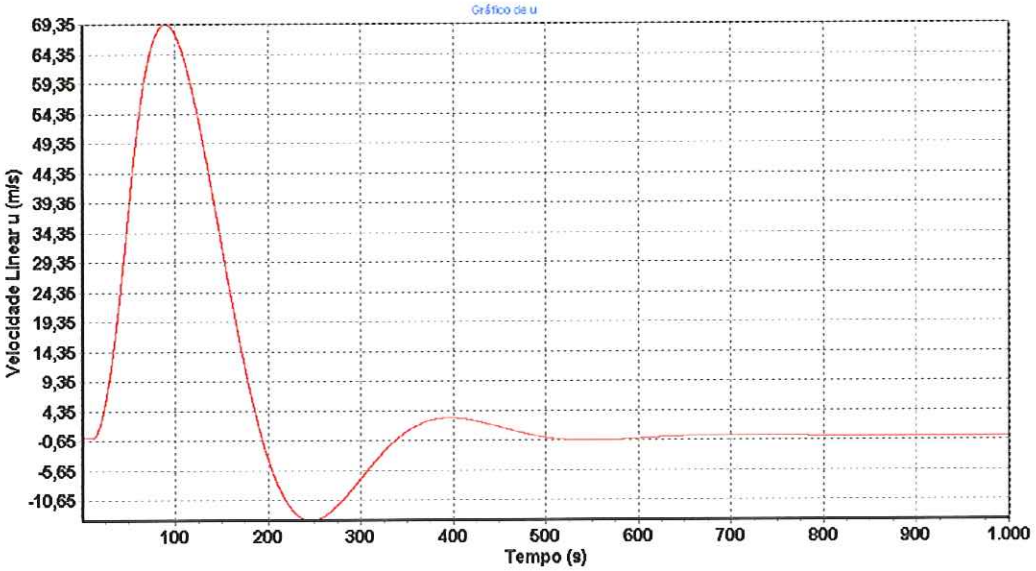

Figura 6.7 – Velocidade Linear *u* (m/s) *vs* Tempo (s) – resultado do SIMAERO.

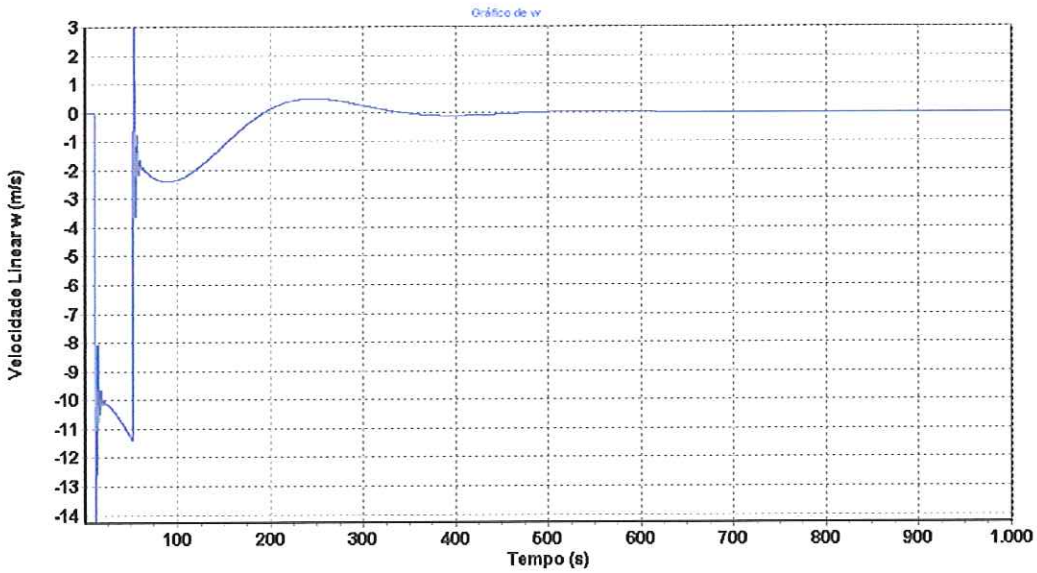

Figura 6.8 – Velocidade Linear *w* (m/s) *vs* Tempo (s) – resultado do SIMAERO.

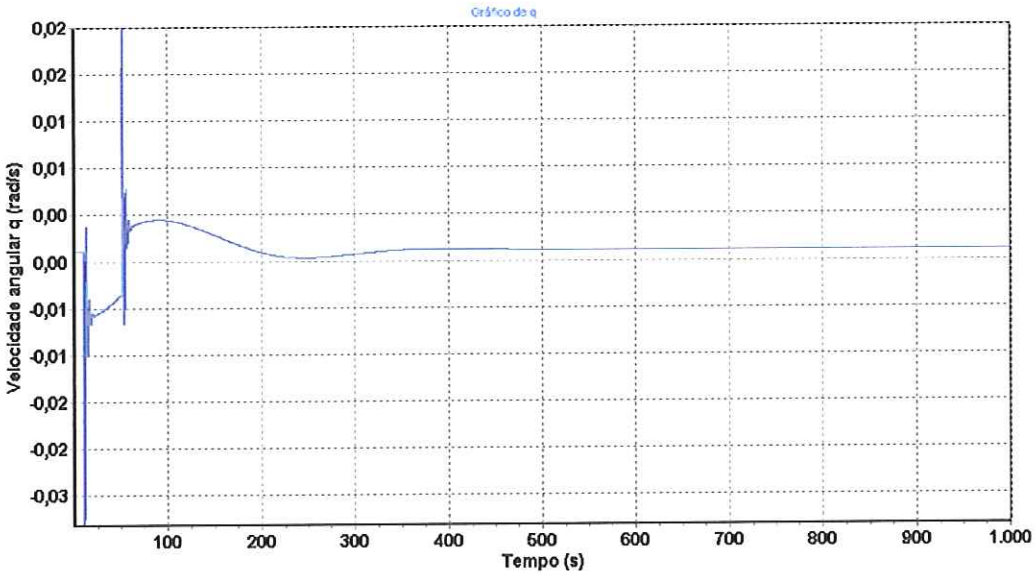

Figura 6.9 – Velocidade Angular q (rad/s) *vs* Tempo (s) – resultado do SIMAERO.

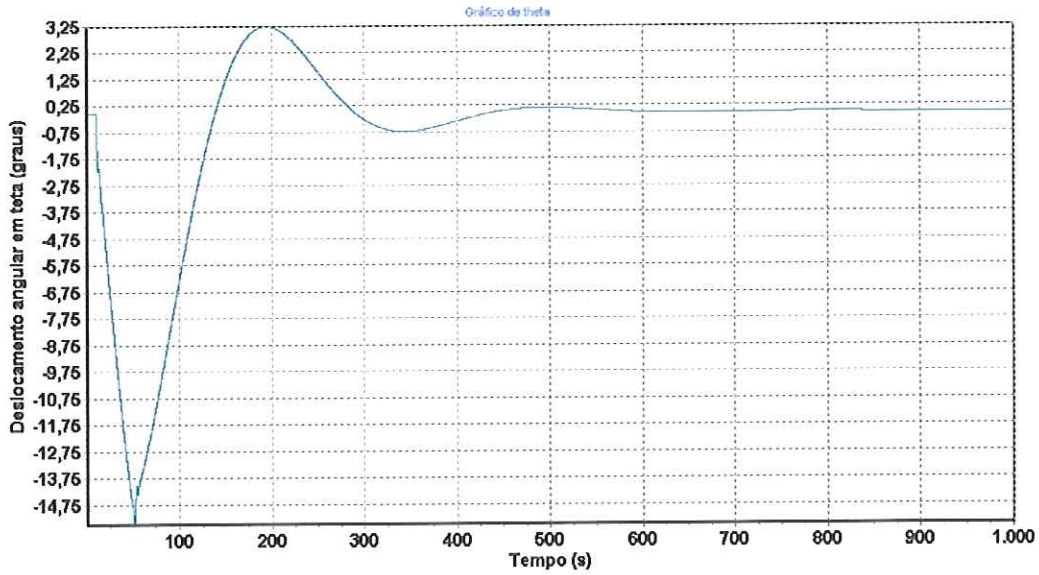

Figura 6.10 – Ângulo de Orientação θ (graus) *vs* Tempo (s) – resultado do SIMAERO.

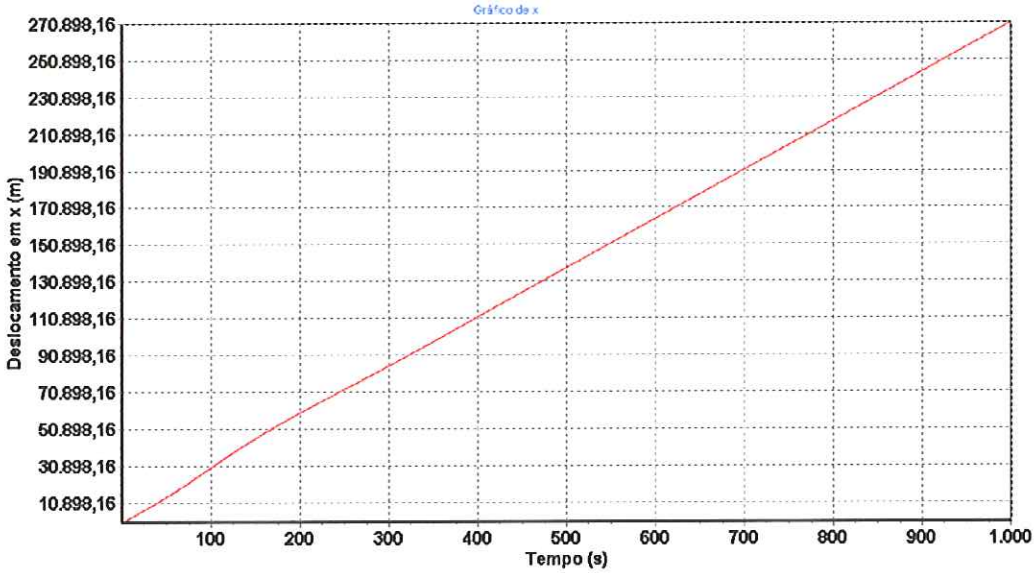

Figura 6.11 – Deslocamento em *x* (m) *vs* Tempo (s) – resultado do SIMAERO.

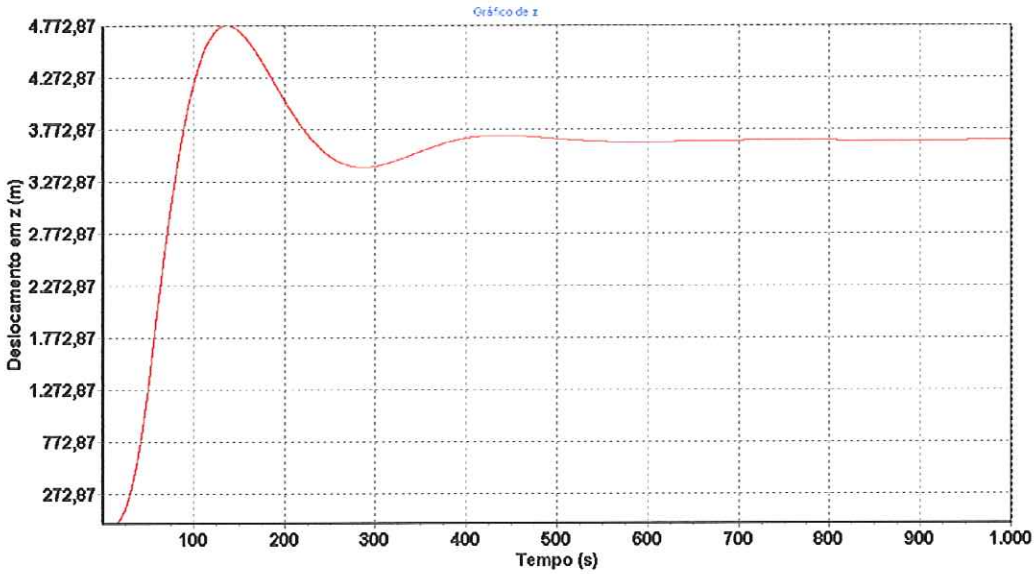

Figura 6.12 – Deslocamento em *z* (m) *vs* Tempo (s) – resultado do SIMAERO.

Analisando as respostas nesta simulação fica nítido o momento em que é acionado o profundor no instante 10s, pois os gráficos de u, w, q, θ e z permanecem retilíneos até este ponto, onde inicia-se então uma perturbação nos gráficos (**Figura 6.7, 6.8, 6.9, 6.10** e **6.12** respectivamente). Da mesma maneira é notável nos gráficos de w, q, θ e z o momento em que o ângulo de deflexão retorna a seu valor inicial zero (instante 50s), pois neste instante os gráficos de u, w, q, θ e z começam a convergir. Este comportamento já era esperado, pois demonstra a situação em que a aeronave tende a retornar a um vôo equilibrado e estável no qual se encontrava no instante inicial. Assim as velocidades u, w, q tendem a seus respectivos valores iniciais. Por outro lado, a altitude se estabelece em torno de 3.700m acima da posição em que a aeronave se encontrava no instante inicial (**Figura 6.12**). Este resultado no gráfico de z já era esperado pelo fato de ter-se aplicado um ângulo de deflexão no profundor. Por fim, podemos perceber o deslocamento da aeronave com relação à sua posição em x, que ao final da simulação tem um valor em torno de 270.900m (**Figura 6.11**), o que também já era esperado considerando que para este caso a velocidade inicial da aeronave é de 265m/s e portanto após 1.000s espera-se que a aeronave tenha percorrido uma distância próxima à obtida como resultado no gráfico.

Da mesma maneira que na simulação anterior, o gráfico de u representa a variação da velocidade linear u com relação à velocidade inicial, assim como o gráfico de z representa a variação da posição em z com relação à altitude inicial. Os demais gráficos também representam esta variação com relação a seus valores no instante inicial, porém nos casos aqui simulados estes possuem valor inicial zero.

CAPITULO 7

CONCULSÃO E TRABALHOS FUTUROS

O Software SIMAERO foi desenvolvido com o intuito de que no futuro ele seja modificado, para controlar uma plataforma de simulação. Desta forma as respostas resultantes da simulação não seriam apresentadas somente em gráficos, como é feito atualmente, mas seriam direcionadas para uma placa controladora, que por sua vez responsável por realizar a interface entre o software e a plataforma de simulação, converteria os valores obtidos na simulação em movimentos na plataforma.

Da forma como está implementado, o Software responde a entradas nos controles da aeronave (profundor, aileron, e leme) somente do tipo degrau. Para que seja capaz de receber outros tipos de entradas, basta alterar a rotina 'carrega_controles()', passando os valores para os controles em cada instante da simulação através de um vetor.

Com os resultados apresentados nos gráficos do Capítulo 6, podemos concluir que as simulações apresentaram respostas compatíveis com as respostas de uma aeronave real, a encontrada na referência (ETKIN & REID, 1996). Um fator que faz com que os resultados não sejam mais próximos do real é o fato de

algumas variáveis serem desconsideradas, a fim de simplificar o modelo, porém sem grandes percas qualitativas. Em trabalhos futuros, caso deseje-se apurar a qualidade das respostas, pode-se modificar as equações no arquivo 'edo.h', na função 'Calcula_derivadas()'.

Da maneira como está implementado, o Software permite que uma aeronave seja simulada em no máximo 2.000 segundos. Caso seja necessário realizar uma simulação em um tempo maior, basta alterar a constante que define o tempo máximo de simulação no arquivo 'DEFINICAO.h'.

Os casos simulados foram retirados da literatura (ETKIN & REID, 1996), como já foi descrito. Ao simular outras aeronaves, deve-se ter uma atenção especial com as unidades dos dados e a maneira em que estão definidas as Derivadas de Estabilidade, de modo que estejam compatíveis com o sistema aqui apresentado, ou seja, de acordo com o modelo de equações proposto por ETKIN & REID at al (1996).

Este Software, apesar de não ter sido implementado de forma a prover imediata comunicação com uma plataforma de simulação, pode ser usado para realizar uma análise inicial dos dados de uma aeronave, considerando as características da aeronave, isto é, os parâmetros de entrada para o Software. Este tipo de simulação permite que seja realizada comparação entre resultados obtidos e esperados, determinando o comportamento da aeronave, já que a simulação apresenta um comportamento para a aeronave, próximo do comportamento real.

O Software apresenta ainda uma função que gera um arquivo texto com os resultados obtidos na simulação. Estes resultados são os mesmo resultados apresentados nos gráficos, porém em formato texto. Este arquivo é gerado em

tempo de execução, ou seja, paralelamente à geração dos gráficos. O arquivo é gerado no mesmo diretório do programa executável com o nome 'result.dad'.

As características da aeronave passadas como parâmetros de entrada para o software estão dispostas em arquivos textos. Nestes arquivos é importante a ordem em que são inseridos os dados, pois o programa lê linha a linha do arquivo, de modo que em cada linha do arquivo tem-se uma variável específica. Assim a ordem é essencial, e segue a seguinte seqüência de variáveis: g, $u0$, I_x, I_y, I_z, I_{zx}, rho, X_u, X_w, $X_{\dot{w}}$, X_q, X_{δ_e}, Z_u, Z_w, $Z_{\dot{w}}$, Z_q, Z_{δ_e}, M_u, M_w, $M_{\dot{w}}$, M_q, M_{δ_e}, Y_v, Y_p, Y_r, Y_{δ_a}, Y_{δ_r}, L_v, L_p, L_r, L_{δ_a}, L_{δ_r}, N_v, N_p, N_r, N_{δ_a} e N_{δ_r}.

O Software foi desenvolvido em Plataforma Windows 2000, utilizando o Borland C++ Builder©, de modo que seja necessário ter este mesmo ambiente para executar o programa. Com o intuito de tornar o software multi-plataforma, sugere-se que em uma próxima versão, o Software seja convertido para a linguagem de programação JAVA. Desta forma, é possível que o mesmo software seja executado em diversos ambientes e Sistemas Operacionais. Com isso, é possível também que seja desenvolvido um ambiente específico para a plataforma, de modo que o software seja executado diretamente em uma interface hardware construída ou programada especificamente para controlar a plataforma de simulação.

Atualmente o número de equações simuladas pelo software SIMAERO é fixo de 12 equações e cada uma destas equações estão relacionadas a uma posição em um vetor. Em um trabalho futuro seria interessante fazer com este armazenamento ou definição do número de equações seja dinâmico. No caso da linguagem C seria utilizando ponteiros, no caso C++ ou outra linguagem orientada a objetos, o software seria adaptado de modo que cada nova equação seja uma nova instância de um objeto, permitindo implementar um número de equações

variáveis. Desta forma pode-se implementar modelos diferentes, mais complexos envolvendo outros subsistemas da aeronave.

É sugerido também, que em um próximo trabalho, sejam utilizados arquivos do tipo XML (*Extended Markup Language*), tanto para os arquivos de entradas do software, como para o arquivo com os resultados, pois uma vez que os dados são armazenados em um arquivo XML, estes dados recebem marcações que identificam cada dado, permitindo que um mesmo arquivo seja facilmente interpretado por vários ambientes (JACKSON & HILDRETH, 2002).

Com a utilização destas duas tecnologias (JAVA e XML), é possível também que uma futura versão do Software seja baseada na WEB, ficando assim independente do ambiente.

REFERÊNCIAS BIBLIOGRÁFICAS

BAARSPUL, M. (1990). *Review of Flight Simulation Techiniques*. Progress in Aerospace Sciences, 27 (1): 1-120.

BABISTER, A. W. (1986). *Aircraft Dynamic Stability and Response*. New York, Pergamon Press, 700p

BLAKELOCK, John H.(1965). *Automatic control of aircraft and missiles*. New York, John Wiley & Sons, Inc., 348p.

CALEFFI, T. (1995). *Desenvolvimento de um Programa para Simuladores de Vôo*. Dissertação de Mestrado. Universidade de São Paulo. Escola de Engenharia de São Carlos. São Carlos - SP.

CARVALHO, J. A. D. (1995). *Arquitetura de Hardware e Software para um Simulador de Vôo*. Dissertação de Mestrado. Instituto Tecnológico de Aeronáutica. Engenharia Eletrônica e Computação. São José dos Campos – SP.

CONTE, S. D. (1977). *Elementos de Análise Numérica*. Enciclopédia Técnica Universal Globo, Porto Alegre, Editora Globo, 331p.

CAPORALI, A. S. (1999). *Modelagem, Simulação e Análise de Servomecanismo Eletro-Hidráulico.* Dissertação de Mestrado. Universidade de São Paulo. Escola de Engenharia de São Carlos. São Carlos - SP.

CAPORALI, A. S. (2003). *Estratégias de Controle Não-Convencional para uma Plataforma de Stewart Acionada Hidraulicamente.* Tese de Doutorado. Universidade de São Paulo. Escola de Engenharia de São Carlos. São Carlos - SP.

DEITEL, H. M (2001). *C++: Como Programar.* Tradução Carlos Arthur Lang Lisboa. 3ª Edição. Porto Alegre: Bookman, 1098 p.

ETKIN, B.; REID L. D. (1996). *Dynamics of Flight: Stability and Control.* Third Edition. John Wiley & Sons, Inc..

FOSTER, J. V.; CUNNINGHAM, K.; FREMAUX, C. M.; SHAH, G.H; STEWART, E. C.; RIVERS, R. A.; WILBORN, J. E.; GATO, W. (2005). *Dynamics Modeling and Simulation of Large Transport Airplanes in Upset Conditions.* American Institute of Aeronautics and Astronautics. Guidance, Navegation, and Control Conference and Exhibit, San Francisco, California. AIAA 2005-5933.

GOODWIN, D.; JOHNSON, S (2000). *Teamwork Training – An Innovative Use of Flight Simulators.* Industrial and Commercial Training. v 32, n 4, p. 132-134. MCB University Press.

HORN, J. F.; BRIDGES, D. O.; LOPES, L. V.; BRENTNER, K. S (2005). *Devlopment of a Low-Cost, Multi-Disciplinary Rotorcraft Simulation Facility.* Journal of Aerospace Computing, Information, and Communication, v 2. –

Department of Aerospace Engineering, The Pennsylvania State University, University Park.

JACKSON, E. B.; HILDRETH, B. L. (2002). *Flight Dynamic Model Exchange Using XML.* Modeling and Simulation Technologies Conference and Exhibit, Monterey, California. AIAA 2002-4482.

LACERDA, J. (1997). *Estudo de uma estrutura com três graus de liberdade de orientação visando seu uso como simulador de vôo.* Dissertação de Mestrado. Universidade Federal de Santa Catarina. Escola de Engenharia Mecânica. Florianópolis – SC.

MANZANO, J. A. N. G.; OLIVEIRA, J. F. (1996). *Algoritmos: Lógica Para Desenvolvimento de Programação.* 2ª Edição. São Paulo: Érica, 265 p.

MCRUER, D.; ASHKENAS, I.; GRAHAM, D. (1973). *Aircraft Dynamics and Automatic Control.* Princeton University Press, Princeton, New Jersey, 784 p.

MIZRAHI, V. V. (1995). *Treinamento em Linguagem C++ modulo 2.* São Paulo: Makron Books, 318 p.

MONTEZUMA, M. A. F. (2003). *Modelagem e Controle de Posição e Orientação de uma Plataforma de Stewart.* Dissertação de Mestrado. Universidade de São Paulo. Escola de Engenharia de São Carlos. São Carlos - SP.

MOODY, L. A. (2004). *The History and Future of Flight Control Simulation.* American Institute of Aeronautics and Astronautics. Modeling and Simulation Technologies Conference and Exhibit, Providence, Rhode Island. AIAA 2004-5149

NELSON, R. C. (1942). *Flight Stability and Automatic Control.* New York: McGraw-Hill, 284 p.

PALLETT, E. H. J.; COYLE, S. (1993). *Automatic Flight Control.* 4th Edition. Cambridge, MA: Blackwell Science, 323 p.

PRALIO, B.; GUGLIERI, G.; QUAGLIOTTI, F. (2004). *Design of a Flight Simulation Software Tool for Education Applications.* Modeling and Simulation Technologies Conference and Exhibit, Providence, Rhode Island. AIAA 2004-5160.

PRATT, R. W. (2000). *Flight control systems.* Progress in Astronautics and Aeronautics: v. 184. Reston, Va.; Stevenage, Herts, UK: American Institute of Aeronautics and Astronautics: Institution of Electrical Engineers, 382 p.

ROLFE, J. M.; STAPLES, K. J. (1986). *Flight simulation.* Cambrige, Press Syndicate of the University of Cambrige, 282p.

ROSKAM, J. (1979). *Airplane Flight Dynamics and Automatic Flight Controls.* Ottawa, Kan.: Roskam Aviation and Engineering Corp., 1388 p.

SCHMIDT, L. V. (1998). *Introduction to Aircraft Flight Dynamics.* Reston, Va.: American Institute of Aeronautics and Astronautics 397 p.

SOUZA, L. F. R.; BELUZO, C. E.; BELO, E. M.; MARQUES, F. D. (2005). *Simulation and Identification of a Flight Envelope Using Neural Networks.* 18Th Congress os Mechanical Engineering, Ouro Preto, MG.

STINTON, D. (1996). *Flying Qualities and Flight Testing of the Aero plane*. Oxford, OX; Cambridge, Mass., USA: Blackwell Science: American Institute of Aeronautics and Astronautics, 667 p.

WEINGARTEN, N. C. (2005). *History of In-Flight Simulation at General Dynamics*. General Dynamics – Advanced Information Systems, Buffalo, New York. – Journal of Aircraft, v. 42, n. 2.

WETZEL, R. A. S. (1997). *Metodologia de Projeto e Análise de Simulador de Atitude*. Dissertação de Mestrado. Instituto Tecnológico de Aeronáutica. Engenharia Eletrônica e Computação. São José dos Campos – SP.

YURT, S. N.; OZKOL, I.; HACIYEZ, C. (2004). *Optimization of the PD. Error Analysis and Motion Determination of a Flight Simulator*. Aircraft Engineering and Aerospace Technology. v 76, n 2, p. 185–192.

YURT, S. N.; OZKOL, I.; KAYA, M. O.; HACIYEZ, C. (2002). *Optimization of the PD Coefficient in a Flight Simulator Control Via Genetic Algorithms*. Aircraft Engineering and Aerospace Technology. v 74, n 2, p. 147–153.

ZYSKOWSKI, M. K. (2003). *Aircraft Simulation Techniques Used in Low-Cost, Commercial Software*. Modeling and Simulation Technologies Conference and Exhibit, Austin, Texas.