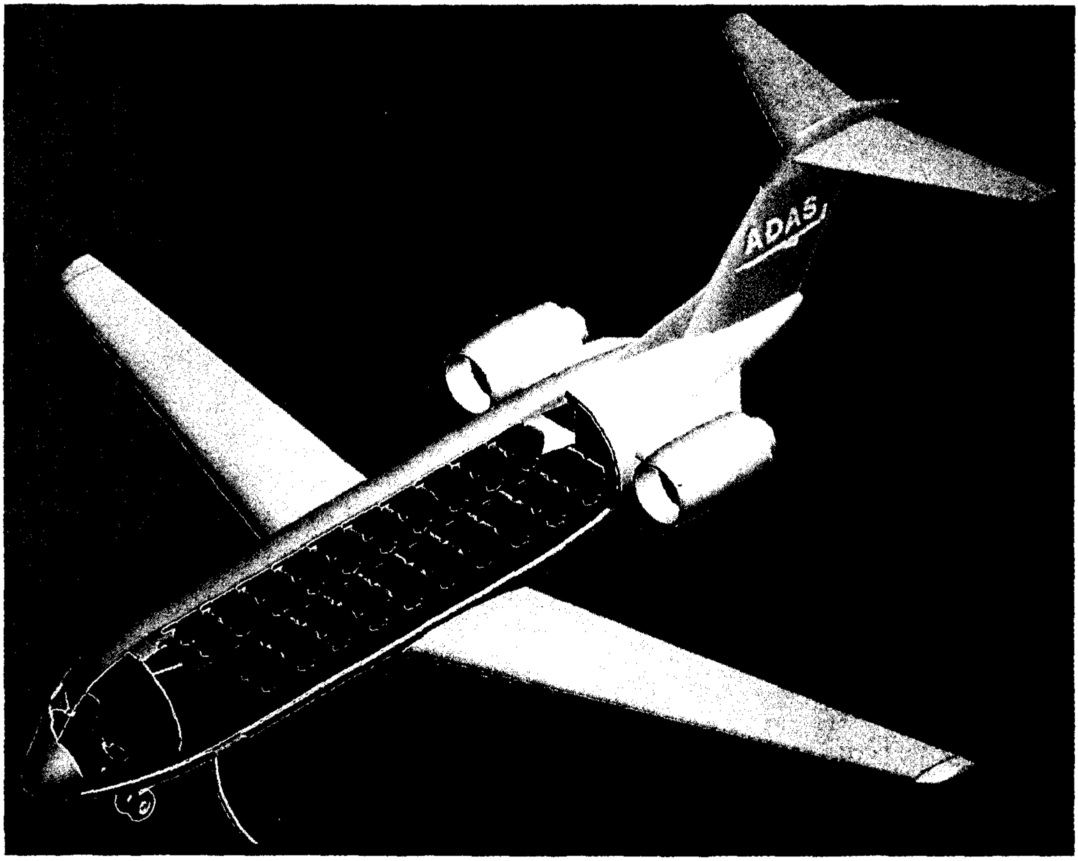
Development and application of a computer-based system for conceptual aircraft design



TR diss
1681

C. Bil

Development and application of a computer-based system for conceptual aircraft design

466661
717 89 02
TR diss 1621

Development and application of a computer-based system for conceptual aircraft design

PROEFSCHRIFT

Ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus,
prof.drs. P.A. Schenck,
in het openbaar te verdedigen
ten overstaan van een commissie
aangewezen door het College van Dekanen
op *8 november 1988* te *16.00* uur

door

Cornelis Bil

Geboren te Oud-Beijerland
Vliegtuigbouwkundig ingenieur

BIBLIOTHEEK TECHNISCHE UNIVERSITEIT
Prometheusplein 1
2628 ZC
DELFT

Delft University Press / 1988

TR diss
1681

Dit proefschrift is goedgekeurd door de promotor

Prof.ir. E. Torenbeek

STELLINGEN

1. De toepassing van een computergesteund voorontwerpsysteem bij het ontwerpen van vliegtuigen biedt de mogelijkheid het ontwerpproces te versnellen en te komen tot een beter koncept-ontwerp (zie dit proefschrift).

2. Bij het gebruik van statistische analyse methoden bij multivariabel optimaliseren dient uiterste zorgvuldigheid te worden betracht.

3. Een ontwerpsysteem dient te worden ontwikkeld uitgaande van het traditionele ontwerpproces: een dergelijk systeem dient hier mogelijkheden aan toe te voegen.

4. De toepassing van statistische analyse methoden vanwege de onvolledigheid aan ontwerpgegevens, betekent slechts dat deze gegevens binnen zekere grenzen impliciet worden vastgelegd.

5. De beschikbaarheid van steeds snellere computers stimuleert de tendens fundamentele rekenmethoden steeds vroeger in het ontwerpproces te gebruiken.

6. De introduktie van geïntegreerde ontwerpsystemen vereist een aanpassing van de beslissingsstruktuur.

7. Het gebruik van een ontwerpsysteem ontslaat de gebruiker niet van het hebben van fundamentele kennis en inzicht op het gebied van ontwerpen van vliegtuigen.

8. Bij het ontwikkelen van computer programmatuur vormt het schrijven van een deugdelijke handleiding ten onrechte een sluitpost.

9. Gezien de bezuinigingen op de universitaire begrotingen, dient de samenwerking met de Nederlandse vliegtuigindustrie op commerciële basis plaats te vinden.

10. Een sterke koppeling tussen het studierendement en de beschikbare financiële middelen bedreigt de kwaliteit van het onderwijs.

Delft, 8 november 1988 C. Bil

Aan mijn ouders

Published and distributed by
Delft University Press

Copyright © 1988 by author.
All rights reserved.
No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without written permission from the publisher: Delft University Press.

List of Contents

page

List of symbols 6

PART 1: INTRODUCTION TO COMPUTER-AIDED AIRCRAFT DESIGN 12

1. COMPUTER-AIDED ENGINEERING IN AIRCRAFT DESIGN 13
   - 1.1 A historical overview 13
   - 1.2 Potential benefits of computer-aided engineering 17
   - 1.3 Computer-aided engineering at the Delft University of Technology 18

2. THE AIRCRAFT DESIGN PROCESS 21
   - 2.1 The design synthesis 21
   - 2.2 The search for the optimum design 22
   - 2.3 Procedures for design optimization 25

PART 2: THE AIRCRAFT DESIGN AND ANALYSIS SYSTEM 27

3. THE AIRCRAFT DESIGN AND ANALYSIS SYSTEM 28
   - 3.1 Principal requirements for ADAS 28
   - 3.2 The ADAS system architecture 30
   - 3.3 Project organization and data protection 32

4. DESIGN DEFINITION 34
   - 4.1 The user-system interface 34
   - 4.2 Configuration geometry definition 36
     - 4.2.1 The MEDUSA drafting and modelling system 37
       - 4.2.1.1 2D drafting 37
       - 4.2.1.2 3D geometric modelling 38
     - 4.2.2 The MEDUSA → ADAS interface 40

4.3 Design data storage and retrieval 42
  4.3.1 Database management systems 42
  4.3.2 The ADAS design database 42
  4.3.3 The database data structure 44
4.4 Definition of design analysis methods 49
  4.4.1 The analysis program 49
  4.4.2 The program library 50

5. DESIGN ANALYSIS 52
5.1 The project file 52
5.2 ADAP compile, load and run step 53
5.3 The ADAP executive program 54
  5.3.1 ADAP data storage and retrieval 55
  5.3.2 ADAP analysis modes 57
    5.3.2.1 Parametric survey mode 59
    5.3.2.2 Optimization mode 60
    5.3.2.3 Design point analysis mode 65

6. DESIGN EVALUATION 66
6.1 Data postprocessing with ADAS 66
6.2 Graphical representation of parametric data 67
  6.2.1 Types of engineering diagrams 69
    6.2.1.1 Carpet plots 69
    6.2.1.2 Surface plots 71
    6.2.1.3 Contour plots 71
  6.2.2 Auto-scaling and annotation 73
  6.2.3 Curve fitting and interpolation techniques 74
    6.2.3.1 The osculatory method 74
    6.2.3.2 Butland's method 76
6.3 The ADAS → MEDUSA interface 76

PART 3: ANALYSIS METHODS FOR CONCEPTUAL AIRCRAFT DESIGN 81

7. WEIGHT ANALYSIS 82
7.1 Fuselage weight 83
7.2 Wing weight 86

7.3 Empennage weight 87
7.4 Undercarriage weight 89
7.5 Control systems weight 89
7.6 Propulsion group weight 90
7.7 Engine nacelles and pylons weight 91
7.8 Airframe systems and instruments weight 91
7.9 Furnishing and equipment weight 93
7.10 Operational items weight 94
7.11 Payload weight 95
7.12 Fuel weight 95

8. AERODYNAMIC ANALYSIS 97
8.1 Lifting surfaces 96
8.1.1 The wetted area of lifting surfaces 101
8.1.2 The mean aerodynamic chord 102
8.1.3 The flat-plate friction coefficient 103
8.2 Fuselage 105
8.3 Engine nacelles 109
8.4 Interference drag 110
8.5 Trim drag 110
8.6 Aerodynamic curves generation 110

9. ATMOSPHERIC PROPERTIES 116

10. RANGE PERFORMANCE 115
10.1 The equivalent range concept 115
10.2 Payload versus range 117

11. FIELD PERFORMANCE 124
11.1 Takeoff performance 124
11.1.1 The takeoff run 126
11.1.2 The rotation phase 127
11.1.3 The airborne phase 128
11.1.4 The stop distance 130
11.1.5 The Balanced Field Length 131
11.2 Landing performance 132

12. OPERATIONAL CLIMB PERFORMANCE 135
  12.1 Angle of attack and climb angle for quasi-stationary climb 135
  12.2 Operational climb procedure 137
  12.3 Maximum speed in horizontal flight 139

13. LOAD AND BALANCE ANALYSIS 140
  13.1 Center of gravity at Operational Empty Weight 140
  13.2 Center of gravity travel 141
    13.2.1 Passengers 141
    13.2.2 Cargo 142
    13.2.3 Fuel 143

14. DIRECT OPERATING COST 145

PART 4: APPLICATION EXAMPLE OF ADAS 152

15. DESIGN EXAMPLE OF A SHORT-HAUL PASSENGER AIRLINER 153
  15.1 The design specification 153
  15.2 The baseline design 154
  15.3 Optimum aspect ratio and wing loading for minimum MTOW 158
    15.3.1 Engines fixed 160
    15.3.2 Engines sized for cruise 161
    15.3.3 Engines sized for takeoff 163
    15.3.4 Design with multivariate optimization 165
  15.4 Optimum design evaluation 169
  15.5 Effect of design range 170
  15.6 Effect of alternative merit functions 173

16. CONCLUSION 174

17. REFERENCES 179

Appendices

A: ADAS COMMAND SUMMARY 197

B: The MEDUSA - ADAS interface protocol 201
- B-1 Lifting surfaces planforms 203
- B-2 Airfoil selection and positioning 204
- B-3 High-lift devices and control surfaces 205
- B-4 Wing fuel tanks geometry 206
- B-5 Fuselage geometry 207
- B-6 Doors and windows 211
- B-7 Fuselage internal arrangement 214
- B-8 Engine selection and positioning 217
- B-9 Extended air intake ducts 219
- B-10 Undercarriage geometry and disposition 220
- B-11 Center of gravity locations 222
- B-12 ADAS significant layer numbers 223

C: THE ENGINE LIBRARY 228
- C-1 Engine nacelle geometry 228
- C-2 Engine performance data 229
- C-3 Engine scaling techniques 231

D: THE AIRFOIL LIBRARY 243
- D-1 Airfoil geometry 243
- D-2 Airfoil aerodynamic data 244

E: ADAS PROGRAM MODULES SUMMARY 247

Summary 252
Samenvatting 255
Curriculum Vitae 258

List of Symbols

a	-	acceleration, speed of sound
A	-	aspect ratio
b	-	span
B	-	bounds of free variables and constraints
c	-	chord length, coefficient
C	-	rate of climb, cost, coefficient, circumference
CO	-	constraint functions
D	-	drag, diameter
e	-	Oswald factor
exp	-	exponent
E	-	induced-drag factor for low-speed parabolic drag polar
f	-	function, factor
fmt	-	print format, i.c the number of significant decimal digits
F	-	form factor
FV	-	free variables
g	-	gravitational acceleration
h	-	altitude, height
H	-	Hessian matrix
i	-	index, counter
j	-	index, counter
k	-	equivalent sand grain size
l	-	length
L	-	length, lift
m	-	mass
M	-	pitching moment, Mach number
MAC	-	mean aerodynamic chord
n	-	load factor
N	-	number of ...
O	-	origin
OBJ	-	objective function
p	-	ambient atmospheric pressure

q - dynamic pressure $q = \frac{1}{2} \rho V^2$, curve slope

r - recovery factor

R - range

R_e - Reynolds number

S - area (no index: reference area)

SF - survey function

SV - survey variable

t - airfoil thickness, time

T - thrust, ambient temperature

U - aircraft utilization

V - speed, volume, bounds of survey variables

W - weight, curve-fit weight factor

α - angle of attack

β - Prandtl-Glauert compressibility factor, scale factor for finite-difference gradient approximations

γ - climb angle, ratio of specific heats

Γ - engine rating

δ - flap/slat setting

ε - geometric twist angle

Δ - difference

η - non-dimensional spanwise station

θ - pitch angle

λ - taper ratio, fuselage slenderness ratio

Λ - sweep angle

μ - runway friction coefficient

ν - ambient kinematic viscosity

ρ - ambient atmospheric density

σ - engine scale factors

τ - lifting surface thickness ratio

φ - user-specified factors in statistical equations

subscripts

ac - aircraft, aerodynamic center
acs - airconditioning system
af - airframe
afsp - airframe spare parts
ai - additional items
ais - anti-icing system
A - afterbody, airborne phase
APU - Auxilary Power Unit
ATC - Air Traffic Control
av - avionics
b - span, basic
bh - bulkheads
car - cargo
cc - cabin crew
ccp - cabin crew pay
ccpr - cabin crew provisions
ccs - cabin crew seats
co - compartment
comm - communications system
cowl - cowling
cr - cruise
D - diameter, dive
depr - depreciation
dw - doors and windows
e - engines, empty
els - electrical system
ep - escape provisions
es - engines support
esp - engines spare parts
f - fuselage, friction
fc - flight crew
fco - floor covering
fcp - flight crew pay
fcpr - flight crew provisions

fps - flight controls system
fpcs - flap control system
fd - flight deck
fda - flight deck accommodations
fdc - flight deck controls
fdes - fire detection and extinguishing system
feq - furnishing and equipment
fl - flight, floor
flp - flaps
fms - flight management system
fr - standard frames
fs - fuel system
ft - fuel tank
fts - fuel tank support
G - ground, gross
gal - galley
h - horizontal tailplane
hs - hydraulics system
i - induced
init - initial
ins - insurance
insp - inspection
instr - instruments
l - lower
lam - laminar boundary layer
land - landing
lpws - lavatory provisions and water system
ls - lighting system
mcs - manoeuvering control system
nac - nacelles
nav - navigation system
no - non-optimum
oe - operational empty
oi - other instruments
os - oxygen system
p - lifting surface panel

pass - passengers
pay - payload
pc - passenger cabin
pcf - passenger cabin floor
pcs - passenger cabin supplies
pi - propulsion instruments
prop - propeller
ps - pneumatic system
pwtc - potable water and toilet chemicals
pyl - pylon
r - root
ref - reference
rev - thrust reversal
rfo - residual fuel and oil
R - rotation phase
s - structural, steps
sbcs - speed brakes control system
scs - slat control system
se - safety equipment
serv - aircraft servicing
sk - skin
slt - slats
spb - speed brakes
spcs - spoilers control system
spi - sound proofing and insulation
stab - horizontal stabilizer
str - stringers and longerons
struc - structures
t - tail, tip
to - takeoff
ts - tail support
turb - turbulent boundary layer
T - boundary layer transition
u - upper
uc - undercarriage
ult - ultimate

v - vertical tailplane

vs - variable stabilizer

vscs - variable stabilizer control system

w - wing

wb - wheelbays

wind - windows

ws - windshield

wss - wing/fuselage support

wws - water/waste system

PART 1: INTRODUCTION TO COMPUTER-AIDED AIRCRAFT DESIGN

This part gives a brief and general introduction into the field of computer-aided design and engineering. A historical overview is given, leading into an assessment of the present situation with respect to computer applications in aircraft design in general and conceptual and preliminary design in particular.

1. COMPUTER-AIDED ENGINEERING IN AIRCRAFT DESIGN

Computer applications have become commonplace in many areas of design and engineering. Advances in computer technology have resulted in a continuous improvement in computational performance and memory capacity. With the introduction of single-user workstations, either stand-alone or included in a network interlinked with other processors and peripheral devices, considerable dedicated computing resources have become available to the engineer at a relatively low cost [Ref. 82]. Besides the traditional function of solving large-scale numerical problems ('number crunching'), additional computer capabilities have emerged. For example, the so-called 4D graphics workstations allow real-time object visualization for animation and simulation. Special-purpose machines for symbolic manipulation have been developed for the implementation of knowledge-based systems and other applications in the field of artificial intelligence [Ref. 72]. Since the beginning of the computer era, the aerospace industry in particular is playing a leading role in the application of these new technologies to improve aircraft development and manufacturing.

1.1 A historical overview

Before 1960, the computer was hardly integrated into the design process. It was mainly used for running self-contained analysis programs, usually in a batch-mode environment. Each department or design team availed of their own specific analysis codes, generally developed and operated by specialists. However, the interchange of design information between these 'isolated islands' was still a manual task and therefore time-demanding and error prone. In 1950, the technical feasibility to display computer-generated pictures on a CRT was demonstrated at MIT [Ref. 99]. In 1962, after appreciable advances in interactive computer technology, this new technique was implemented in the first experimental drafting system (SKETCHPAD). In 1965, the Lockheed Aircraft Company developed the Computer-Aided Drafting And Manufacturing system (CADAM) [Refs. 18 and 117], one of the first commercial drafting systems which is at present in use with many aircraft manufacturers. Around 1980, the CADAM capabilities were augmented with the introduction of a specific 3-dimensional system referred to as CATIA (Computer-graphics Aided Three-dimensional Interactive Application system), also developed by an aircraft manufacturer: the Avions Marcel Dassault - Breguet Aviation company [Ref. 22]. At present, a wide range of CAD/CAM-systems are commercially available on various types of computer hardware. Drafting and modelling systems are usually associated with Computer-Aided Design (CAD).

The introduction of the interactive computer graphics capability prompted a change in the scope of computer applications. One became aware of the potential benefits of computer application in the overall design process. This initiated a trend towards integrating self-contained engineering programs, e.g. mesh-generators for structural analysis, geometric modelling and NC-tooling, into design systems. Usually, engineering programs are interfaced with, and configured around, central database systems. Gradually, the role of the computer evolved into a powerful design tool, practically indispensible for the design and manufacturing of today's complex and efficient aerospace vehicles. The infrastructure of computer systems for the overall support of all design and engineering activities, upto manufacturing, is generally referred to as Computer-Aided Engineering (CAE).

A classification of the currently available CAE-systems according to their dependency on type of application, design phase and discipline is given in Figure 1.1 [Ref. 51]. With the traditional CAD/CAM systems practically any object can be geometrically defined and visualized with a high degree of detail and accuracy. Therefore, these systems are catagorized as extremely application independent. Because of their fundamental solution to the problem, computer-assisted engineering systems, e.g. for structural and flow analysis (CFD), are also generally applicable. Inherent to the use of these systems is that the object needs to be known in quite some detail, hence a large amount of data is involved and computing times are relatively long. Therefore, these systems can only be efficiently used in the detailed design phase when major changes in the configuration are no longer expected.

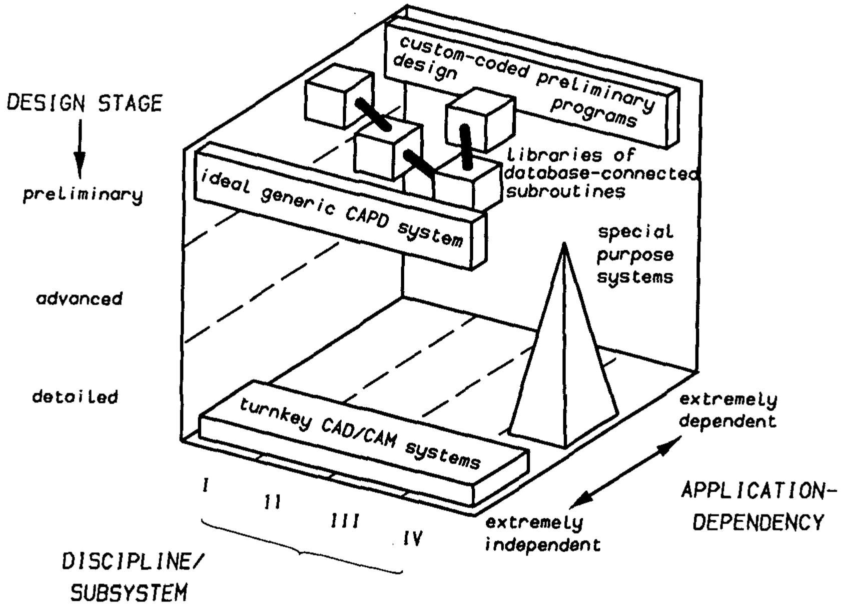


Figure 1.1: Classification of system types in the current CAE-spectrum [Ref. 51].

As yet, CAE has not established itself as an accepted design technique in the initial, i.c. conceptual and preliminary, phases of the design process. There are some general factors that have hindered the penetration of CAE into the configuration development phase:

- Conceptual design is usually not a predetermined and strictly rational process. The heuristic and intuitive nature is less suited for the formalized structure required by a computer.

- Computer programs tend to appear less transparent to the designer. Hidden or obscured design decisions may confuse the decisionmaking process and results are not or only cautiously accepted. Especially in those cases where many design parameters are varied, e.g. in automated design, the analysis may become to complex to understand the physical relationships between the different disciplines.

- Close interaction between the computer and designer is essential and requires special attention to a user-friendly communication interface and ease of operation.

Computer applications in the pre-design phase are generally restricted to (large) custom-coded programs developed and used on an ad hoc basis. Since 1965, research has been directed toward integrating design methods into synthesis programs. A classical example is SYNAC (Synthesis of Aircraft) for configuration development of military aircraft at General Dynamics [Ref. 91]. A more recent example is the ACSYNT-program (Aircraft Synthesis) developed by NASA [Ref. 15]. This program also forms a part of Northrop's Conceptual Design System (CDS) [Refs. 105 and 106]. Some of these synthesis programs even provide the option to automatically perform sensitivity studies and aircraft sizing. However, the incorporated design methods and their respective input and output variables are pre-selected and can not be easily modified by the designer. Therefore, these programs are likely to require reprogramming or additional code for each new application, not accommodated by the original program. Thus, innovative design is resisted. In addition, the time required to adapt the program may render it already obsolete before it becomes productive. Because of the built-in design procedures and processes, the designer tends to adjust the way of thinking to the capabilities and the mode of program operation.

Essential to the development of a generic preliminary design system, is that the analysis methods are to be considered as input to the system, similar to data, rather than integrating, i.c. fixing, them into the system itself. Thus, these modules, embodying the design knowledge, should reside outside the system, where they can be easily modified and reorganized by the designer and be tailored to the specific design problem structure and aircraft category. The Boeing's Computer-aided Preliminary Design System (CPDS) [Ref. 132] is a typical example of this concept.

1.2 Potential benefits of computer-aided engineering

Although the objectives to introduce CAE may differ per industry and type of application, some general areas of practical and economic potential can be identified with respect to aircraft design [Ref. 70]:

- Reduced design time
  CAE can reduce the development time needed to introduce a new design. Early introduction and availability, ahead of competative designs, can be an advantage.

- A better product through improved design
  Within a given time frame, more alternative design solutions can be considered, which potentially leads to an improved design quality.

- A greater design capacity
  For a given design staff, several design projects can be accommodated simultaneously. A faster response to customer queries and requests is possible.

- Solving outsized design problems
  CAE introduces opportunities and capabilities, which are practically impossible to do in a traditional design environment. Sophisticated analysis codes require computer-assisted pre- and postprocessing.

- Reduced design cost
  A reduction in design and development cost can be expected, in particular in the areas of drafting and manufacturing [Ref. 54].

In principle, these advantages also apply to conceptual design. However, although at least 80% of the development and production cost of a project is related to decisions made in the early stages of design, the actual investments during the conceptual design are relatively small. Therefore, emphasis is put on improving the design quality rather than cost reduction, in general:

- An improved efficiency of the design process.
  Automation of routine and repetative activities relieves the designer of standard tasks. Formalization of design data improves data exchange between disciplinary teams and down-stream design levels.

- An improved design quality.
  Within a given time/cost frame, more design alternatives can be evaluated. Sensitivity studies, trade-off studies and multivariate optimization can be more easily applied in an integrated design environment.

Although the procurement of CAD/CAE hard- and software is costly and the implementation will undoubtedly have an impact on the company's organization, infrastructure and working procedures, these initial problems do not outweigh the long-term benefits and the implementation of CAE in the aerospace industries and research laboratories is well under way: A better concept is essential for a better aircraft design.

1.3 Computer-aided engineering at the Delft University of Technology

Considering these rapid developments in the aerospace industry with respect to CAE, it is to be expected that aerospace engineers will become increasingly more involved with computer applications in aircraft design. Therefore, the aeronautical faculties must adjust their curricula to this trend and prepare future design engineers for this new environment.

Around 1980, the Delft University of Technology furnished funds to stimulate general research in CAE. In September 1983, a general-purpose CAD-system, referred to as the Interfaculty CAD-Installation (ICI), became available for university-wide use. The ICI is a "turnkey" CAD-system based on a PRIME 750 computer on which the MEDUSA drafting and modelling package is implemented. The ICI hardware configuration is schematically illustrated in Figure 1.2. It shows resemblance with the system in use at the NASA Langly Research Center for Computer-Aided Research (CAR) [Ref. 118]. The central processing unit (PRIME 750) and its peripheral devices are situated at the DUT's Computing Centre. The host can accommodate several remote MEDUSA-workstations and other terminals in a multi-user operation mode. A MEDUSA-workstation comprises a graphics and an alphanumeric display with local hardcopy and printing facilities. A small data tablet is used for menu-driven command input and digitizing. A direct communication link exists with the Central Digital Installation (CDI), an IBM 3083-JX1 mainframe, for Remote Job Entry (RJE).

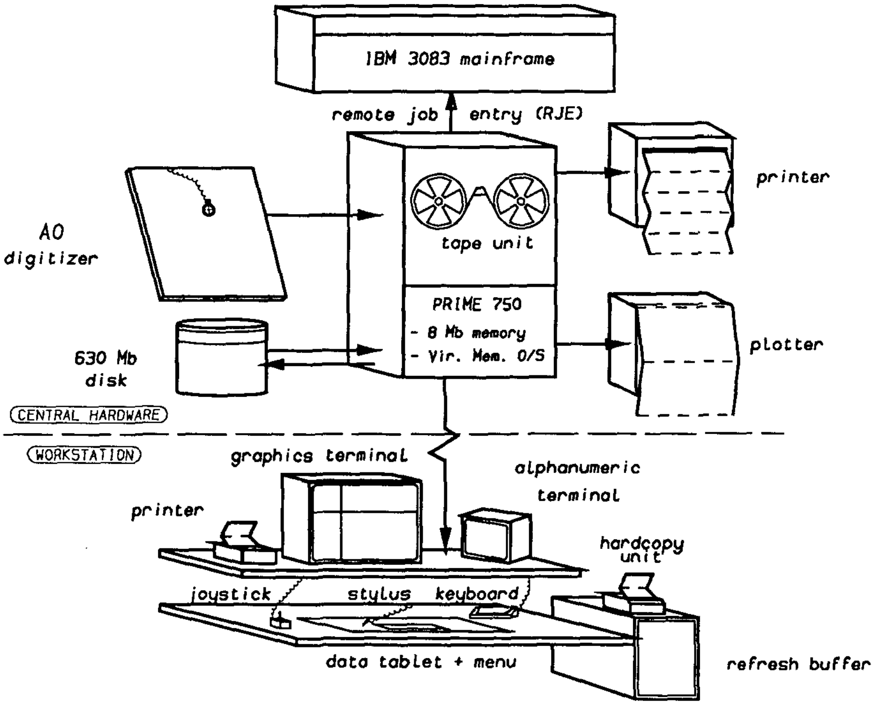


Figure 1.2: The DUT's central CAD hardware configuration (ICI).

The ICI is currently in use with 4 engineering faculties, i.e. Industrial Design, Naval Engineering, Mechanical Engineering and Aerospace Engineering. The DUT's Computing Center is responsible for system supervision, maintenance and support. In 1985, these faculties drew up and submitted a proposal for a joined project, extending for a 10 years period, covering several related topics in the field of CAD/CAM/CAE for which each faculty would put in a specific research effort.

A contribution to this project from the Faculty of Aerospace Engineering is represented by the development of a computer-aided system for conceptual aircraft design. The purpose of this project, initiated in 1983, was to assess the practical capabilities and pitfalls of computer-aided design as applied to conceptual aircraft design and to obtain hands-on experience in this field. The system should be general and flexible enough to be applicable in a wide spectrum of design problems, both for research, e.g. alternative design concepts, multivariate optimization, as well as aeronautical teaching, e.g. design synthesis, parameter studies. This design system is referred to as the Aircraft Design and Analysis System (ADAS).

This dissertation concludes a 4-years period of development work in which the design process was analyzed, functional requirements were drawn up, the system architecture was conceived and the individual programs were coded. Although, development work on ADAS will be an on-going process as new requirements arise and additional enhancements are implemented, the current pilot-version of ADAS meets the basic requirements set for an aircraft design system geared toward a university environment. This dissertation gives a general overview of ADAS as it is currently implemented on the ICI. Individual modules will be highlighted and discussed in detail. In conclusion, Chapter 15 describes a typical design problem and illustrates how ADAS can be employed to obtain an optimum design solution.

2. THE AIRCRAFT DESIGN PROCESS

The design and development of a new aircraft type involves a great financial investment. Therefore, the decision to initiate a new aircraft project is preceded by intensive market surveys to assess the commercial prospects and to inventory the requirements of potential customers for a future air transport. However, conceptual and preliminary design studies are also carried out just to assess e.g. the impact of emerging technologies or to study the feasibility of alternative concepts, but without the implicit intention to ultimately build the aircraft: a new aircraft type appears only about every 20 years!

2.1 The design synthesis

Aircraft design is generally not a continuous and straightforward process: it involves many repetative procedures and feedbacks. However, it is common practice to divide the overall aircraft design process into 3 logical phases, as shown in Figure 2.1 [Ref. 124]. The objective in the conceptual design phase is to conceive a global definition of a number of design configurations that best comply with the design requirements. Typical for conceptual design is the design synthesis: the designer attempts to combine all technical disciplines, e.g. weight and balance, aerodynamics, stability and control, performance, costs and noise, into a well-balanced design solution. As only little design information is available at this stage, relatively simple analysis methods have to be used. These prediction methods are referred to as class I methods and are generally derived from (semi-) empirical and statistical analysis on existing aircraft designs.

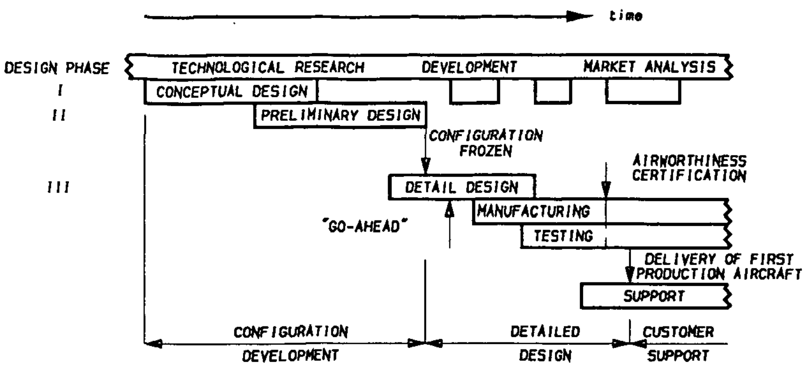


Figure 2.1: Typical phases in the aircraft design process [Ref. 124].

A selection of the most promising conceptual designs are subsequently analysed with more sophisticated methods, e.g. finite-element modelling and computational fluid dynamics. Usually, at this stage, the design staff is divided into disciplinary teams each with their specific specialism. A final selection is made and the design configuration is frozen. In the detail design phase, the aircraft is designed at a component level resulting in many engineering drawings. Windtunnel experiments and structural tests are carried out. At this time, the management must decide whether to "go-ahead" and build the aircraft as development cost will increase progressively beyond this point.

Although the general concept of the ADAS system does not restrict its application to a particular design stage, it is in principle intended for conceptual design.

2.2 The search for the optimum design

In order to develop a suitable design system, it is essential to identify the basic activities and procedures that take place in the initial stages of design. Design is generally an iterative process which starts with an initial, tentative design configuration, referred to as the baseline design. Given the design requirements and objectives, the configuration is repeatedly modified in subsequent design cycles until a satisfactory design solution is found. A design cycle can in turn be divided into 3 basic steps:

1. <u>Design definition</u>
   A new or modified design configuration is geometrically defined in sufficient detail.

2. <u>Design analysis</u>
   Suitable analysis methods are utilized to compute selected design characteristics.

3. <u>Design evaluation</u>
   The design is evaluated by comparing the analysis results with the given design requirements and objectives. If deemed necessary, the design configuration is changed and the process repeats.

This procedure can be represented in a schematic flow chart, adopted for implementation on a computer (Figure 2.2):

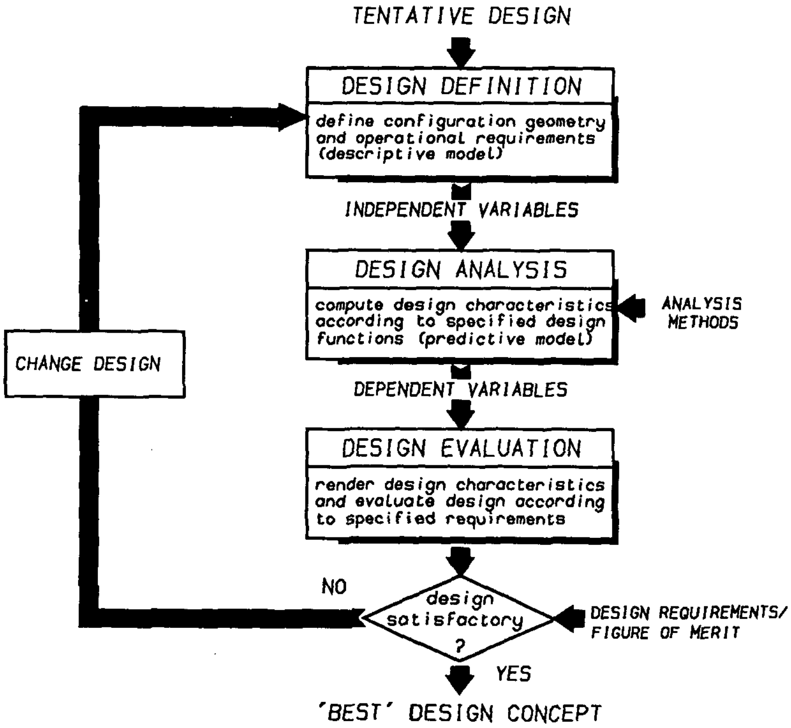

TENTATIVE DESIGN
DESIGN DEFINITION
define configuration geometry and operational requirements (descriptive model)
INDEPENDENT VARIABLES
DESIGN ANALYSIS
compute design characteristics according to specified design functions (predictive model)
ANALYSIS METHODS
CHANGE DESIGN
DEPENDENT VARIABLES
DESIGN EVALUATION
render design characteristics and evaluate design according to specified requirements
NO
design satisfactory ?
DESIGN REQUIREMENTS/ FIGURE OF MERIT
YES
'BEST' DESIGN CONCEPT

Figure 2.2: Basic steps in an aircraft design cycle.

Figure 2.2 shows that conceptual design is basically a search process: design data and analysis methods are manipulated until the process is converged to an acceptable design solution. In mathematical terms there are 4 ingredients involved:

1. Independent variables or design parameters which can have arbitrary values within certain limits. Design parameters can be assigned a value either directly by the designer or indirectly by an executive program. Design data that is not subject to change during a design study are referred to as design constants. Examples of typical design parameters are wing aspect ratio, wing loading, thrust/weight ratio and tailplane dimensions.

2. Dependent variables whose values depend on the independent variables. Typical dependent variables are design characteristics such as performance criteria, cost, noise, etc.

3. Analysis methods which define the physical relationships between the independent and dependent variables. Analysis methods may be relatively simple or extremely complex, dependent on the accuracy required and the amount of input information available.

4. Design requirements and figure of merit to assess the feasibility and efficiency of the design configuration. The design requirements impose technical constraints which may originate from operational performance and mission requirements, airworthiness requirements, technological aspects and practical considerations.

The selection of the primary wing and tailplane parameters such that the aircraft meets the defined mission requirements ("sizing"), is common practice in conceptual design. However, it is generally found that several different design concepts represent a feasible design. The total collection of feasible designs is referred to as the design space. An additional requirement, a figure of merit, may then be considered according to which the best possible (feasible) design can be selected. This process is referred to as design optimization.

2.3 Procedures for design optimization

There are 3 basically different procedures which the designer can employ to search for the best conceivable ("optimum") design [Refs. 113 and 126]:

- In the traditional intuitive (optimum) design approach, the designer relies mainly on intuition and experience to select and change design parameters. The principal advantages (+) and disadvantages (-) of this approach are:

  + The designer can make full use of experience, augmented by proven and simple design methods;
  + Simple or no programming is required;
  + Number of designs to be analyzed is limited;
  + Maximum use is made of calculated results;
  + No a priori choice of one merit function;
  + Arbitrary, though limited number of variations and design modifications;
  - No guarantee that a real optimum is obtained;
  - No useful result outside the designer's experience;
  - It is time consuming: the designer will therefore tend to resist desirable changes in the design specifications or other previous decisions.

- With explicit or parametric optimization, a multitude of designs, each with different parameter values, are generated and analyzed. All designs are subsequently evaluated and the "best" design is selected. The advantages (+) and disadvantages (-) are:

  + It is rooted in the industrial approach;
  + Requires relatively simple programming;
  + The designer has complete control over decisions;
  + No a priori choice of a single merit function;
  + Sensitivity of off-optimum design conditions remains visible;
  - No guarantee that a global optimum is obtained: it may be outside the selected design space;
  - Only practical for a limited number of independent variables, 3 or 4;

- Many designs are evaluated, only a few are actually used;
- The designer is not encouraged to extend the number of variables: the amount of data to be analyzed increases exponentially;
- Changes in the design specification make previously generated results obsolete and are resisted.

- <u>Implicit or multivariate optimization</u> requires the design process to be fully automated. The figure of merit and design requirements are quantitatively formulated as an objective function and constraints respectively. An optimization algorithm (optimizer) changes the specified design parameters (free variables) based on mathematical information acquired during the optimization process. The advantages (+) and disadvantages (-) are:

  + It potentially leads to an improved design quality due to the rigorous approach;
  + Especially useful for multi-variable systems;
  + Effect of "biased" decisions is eliminated;
  + Changes in design specifications are easily met;
  - Programming and debugging are difficult;
  - Optimization algorithms are not always effective;
  - Convergence problems may occur: no solution is found;
  - No insight into design sensitivity: only one design is obtained;
  - Inexperienced designers may produce and accept unrealistic results.

In the presented order, these optimization techniques involve a higher degree of design automation in which control of the search process is delegated to the computer. By comparing the pros and cons, it can be concluded that these 3 optimization techniques are complementary, hence, an effective design system should give the designer the freedom to choose a suitable combination for a given design problem.

PART 2: THE AIRCRAFT DESIGN AND ANALYSIS SYSTEM

In this part, the Aircraft Design and Analysis System (ADAS), a system for conceptual aircraft design, is introduced and described with respect to informatical and computer-technical aspects.

3. THE AIRCRAFT DESIGN AND ANALYSIS SYSTEM

3.1 Principal requirements for ADAS

Based on the considerations discussed in the previous Chapters, some principal requirements were formulated for the ADAS-system, with respect to its functionality and practical implementation, i.c.:

- As teaching and research in conceptual/preliminary design is traditionally the responsibility of the disciplinary group Aircraft Design/Flight Mechanics of the Faculty of Aerospace Engineering, ADAS is primarily intended for aircraft configuration development. However, an effective design system should include the capability to communicate with other design systems/programs to be able to carry-on the design to down-stream design levels (open-ended system).

- A design system must be flexible in handling a wide range of different design problems with little or no modifications required. This implies a modularly structured system architecture that simplifies future modifications and enhancements.

- The designer must be able to select sensitivity analysis and/or multivariate optimization as an optional feature, but an analysis study of a given configuration must remain possible.

- A conceptual design system is highly interactive, therefore a user-friendly operating environment is essential for the general acceptance and use of the system. This entails e.g. computer checking and reporting of logical and syntactical errors, use of default keywords where appropriate, an on-line help facility and elaborate documentation.

- Built-in design decisions within analysis programs must be avoided as much as possible, as they can hinder the interpretation of results and confuse the decisionmaking process. The primary task of a design system is to produce selected analysis results in an efficient way. Design decisions should be the sole responsibility of the designer, even when design control is delegated to the computer, e.g. in case of numerical optimization. Hence, design and analysis are two different functions that should be clearly separated.

- A design system can potentially reduce time and effort required particularly for data pre- and postprocessing. Substantial benefits can be gained in this area. The implementation of a database system, interactive geometry input through a CAD-system and options for interactive plotting of engineering diagrams are a logical consequence of this requirement.

- ADAS should logically be implemented on the ICI, as the most readily available system for interactive CAD-applications at the DUT, although this may impose some hard- and software limitations. As yet no attempt has been made to make ADAS completely hardware independent. However, hardware dependency is mainly restricted to the handling of global variables and system calls.

- Ample use should be made of available (commercial) software products in order to reduce development time and cost. In this respect, ADAS requires the MEDUSA-system for graphics (drafting and graph plotting) and the NAG-library for standard numerical routines although other equivalent software can in principle be used.

Note that the ADAS-version described in this dissertation, is a pilot-system: a starting point for further research and development. Currently, the analysis methods and geometry definition capabilities are geared toward conventional subsonic transport aircraft.

3.2 The ADAS-system architecture

On the basis of the functional requirements and available hard- and software described above, a system architecture has been developed for ADAS. This ADAS system architecture is schematically illustrated in Figure 3.1:

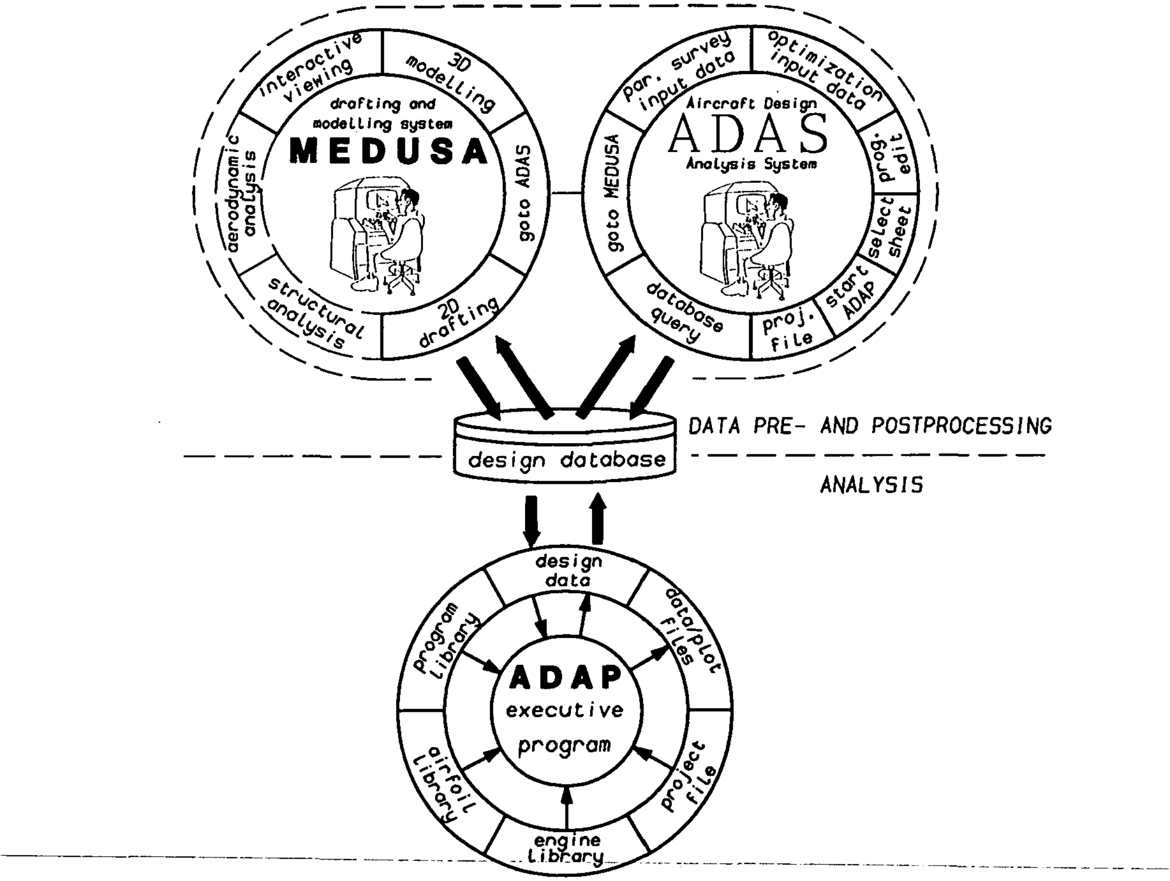


Figure 3.1: The ADAS-system general architecture.

The overall system comprises 3 self-contained programs which communicate through a common database system, i.c.:

- The ADAS program controls the user-system dialogue through a command-oriented language. This program also contains functions to interactively perform data pre- and postprocessing, e.g. database query, graph plotting, text editing, etc. The ADAS-program forms the nucleus of the system, any function or subsystem can be invoked from a central command level.

- The Aircraft Design and Analysis Program (ADAP) is a general executive program that controls the processing of a user-supplied analysis program. Optionally, ADAP can be run in parametric survey and/or optimization mode. ADAP is non-interactive, hence it can run independently once all the necessary input information has been prepared.

- The MEDUSA drafting and modelling system is used within ADAS to define a design configuration layout in the form of a schematic 3-view drawing. An interface has been developed to transfer geometry information between a MEDUSA sheet file to the ADAP-executive program for analysis and to place a newly computed (optimum) design configuration from ADAP back into a MEDUSA drawing.

  ADAS and MEDUSA are effectively integrated into one program. One invokes the MEDUSA 2D drafting module on any suitable workstation or alphanumeric terminal and can subsequently switch between the MEDUSA and ADAS command level as required. For example, plotting of engineering diagrams is done by ADAS directly into a MEDUSA drawing through the MEDUSA graphical interface. Subsequently, the designer can switch to MEDUSA command level and apply any of the available MEDUSA drafting commands to edit the plot or to make a high-quality copy.

All available options are input at ADAS or MEDUSA command level and control subsequently returns to this command level after the operation has been completed. The user is generally unaware of the underlying processes taking place, although knowledge thereof is useful for understanding the workings of the system.

In the following Chapters these individual system components will be described in more detail and their specific features will be highlighted by going through an imaginary design sequence as described in Section 2.2.

3.3 Project organization and data protection

When utilizing ADAS in a design project, it is likely that data will be generated that is to be retained for some period of time. Such data is placed in a file, under a given filename, and resides on a secondary storage device, usually a magnetic disk unit. The system manager will regularly make copies of all the disk files onto a magnetic tape for back-up purposes.

The file organization structure employed on the ICI is referred to as a hierarchical file structure, as shown in Figure 3.2:

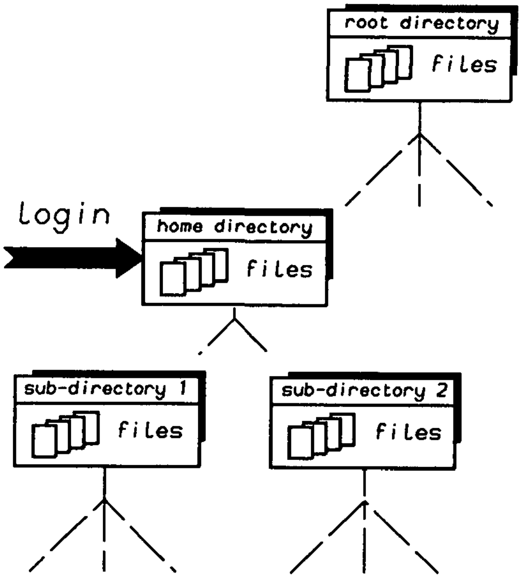


Figure 3.2: Organization structure of files and directories.

Logically-related files, e.g. database files, MEDUSA sheet files, source code, etc., may be placed into a User File Directory (UFD). A child-UFD is sub-ordinate to another UFD, referred to as the parent UFD. In this way, the file organization structure resembles an inverted tree, with each node representing a UFD. A filename must be unique within the UFD it resides in. To refer to a particular file, one must specify its complete pathname, i.c. all the UFD-names down to the filename itself. In a pathname, UFD-names are separated by the > symbol, e.g. VL>VO>ADAS>ENGINE>ALF-502D.

As typing of relatively long pathnames may become cumbersome, the system maintains a pointer to the current UFD: the working directory. When only a filename is given, the system automatically assumes that file to reside in the current UFD. Each user has an attach-UFD which automatically becomes the current UFD after signing on the ICI: the home directory. The user is free to create any sub-UFD as required. Before a new design project is initiated, the designer should set up a convenient UFD-structure, particularly if the project comprises several subjects or when a group of designers are working on one project. ADAS allows the user to store design data in any specified UFD and to apply any file management command provided by the operating system.

The user can also assign access rights either to a file or to a complete UFD and which may affect an individual user or a group of users. Access rights include for example delete, read, list, use and write. By setting access rights, an owner can protect or secure the files against e.g. unauthorized access or unintentional delete.

4. DESIGN DEFINITION

Design definition forms a logical first step in a typical design study with ADAS. This means that design data has to be acquired and entered into the system. These activities are generally interactive and are controlled by the ADAS program. The ADAS program forms the nucleus of the ADAS system and allows the user to access any available function or subsystem by means of a command-oriented language.

4.1 The user-system interface

As mentioned in Section 3.2, the MEDUSA and ADAS systems are integrated into one program. The user can switch between both command levels with only a simple instruction. This does not affect the state of the MEDUSA and ADAS programs or the information held in primary memory.

The advantages and disadvantages of this approach are:

+ Graphs generated by ADAS and stored in a MEDUSA sheet can be edited and plotted using the standard MEDUSA drafting options.
+ Use of the MEDUSA built-in graphical interface eliminates the need to adjust ADAS to different graphical interfaces for different terminal types.
- ADAS requires the availability of the MEDUSA software or an equivalent CAD-system.
- To start the ADAS system, one must invoke the MEDUSA system on a specially configured graphical workstation or alphanumeric terminal.

The ADAS/MEDUSA command processing control scheme is illustrated in Figure 4.1:

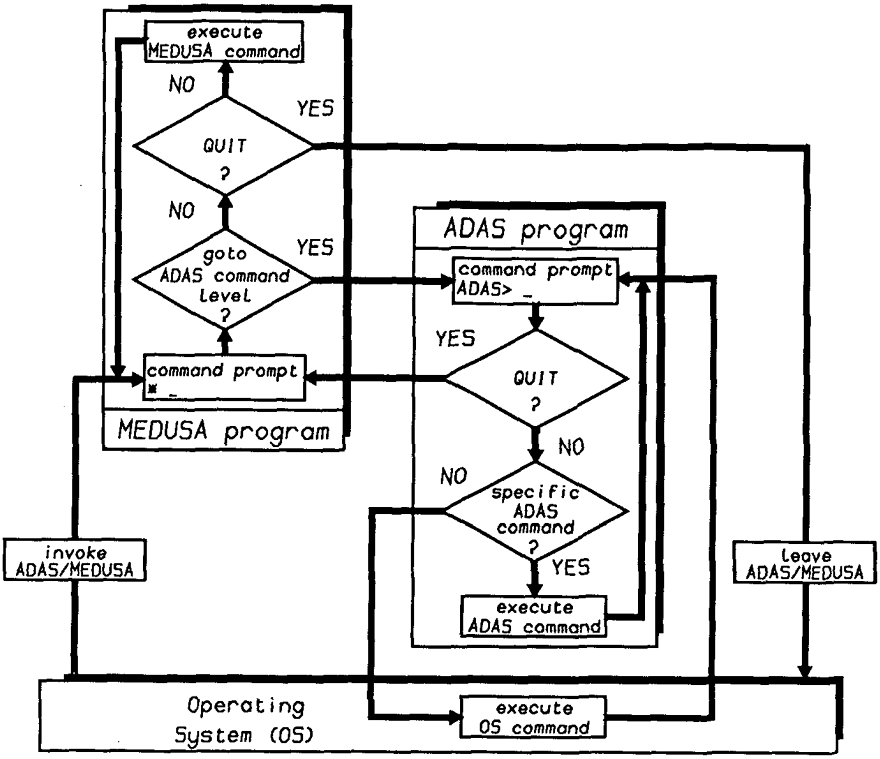


Figure 4.1: ADAS/MEDUSA command processing control flow.

At ADAS command level, the user is prompted for any of the commands described in the ADAS Command Reference Manual [Ref. 25]. A summary of available commands is given in Appendix A. Commands can be either specific ADAS commands or operating system commands. If a command name is not a valid ADAS command, it will be passed to the operating system as a system call. After a command is executed, control returns to ADAS command level. A built-in break-handler can be used to interrupt command processing at any time. The keyboard is the standard input device. Experiments have shown that ADAS can easily be adopted for menu-driven input, if so required. These provisions however, make the system extremely hardware-dependent and have therefore been omitted.

4.2 Configuration geometry definition

Geometry information constitutes a large part of the total collection of design data. A logical means of geometry data input is through a drafting system. With a drafting system, the configuration layout is directly visualized and errors are easily detected. In 1981, a graphics program (SKETCH) was developed to generate a schematic 3-view configuration drawing or a 3D wire-frame model from only a few, basic shape parameters [Ref. 23]. At a later stage, SKETCH was coupled to the ADAS system to display optimized design configurations. An example is shown in Figure 4.2:

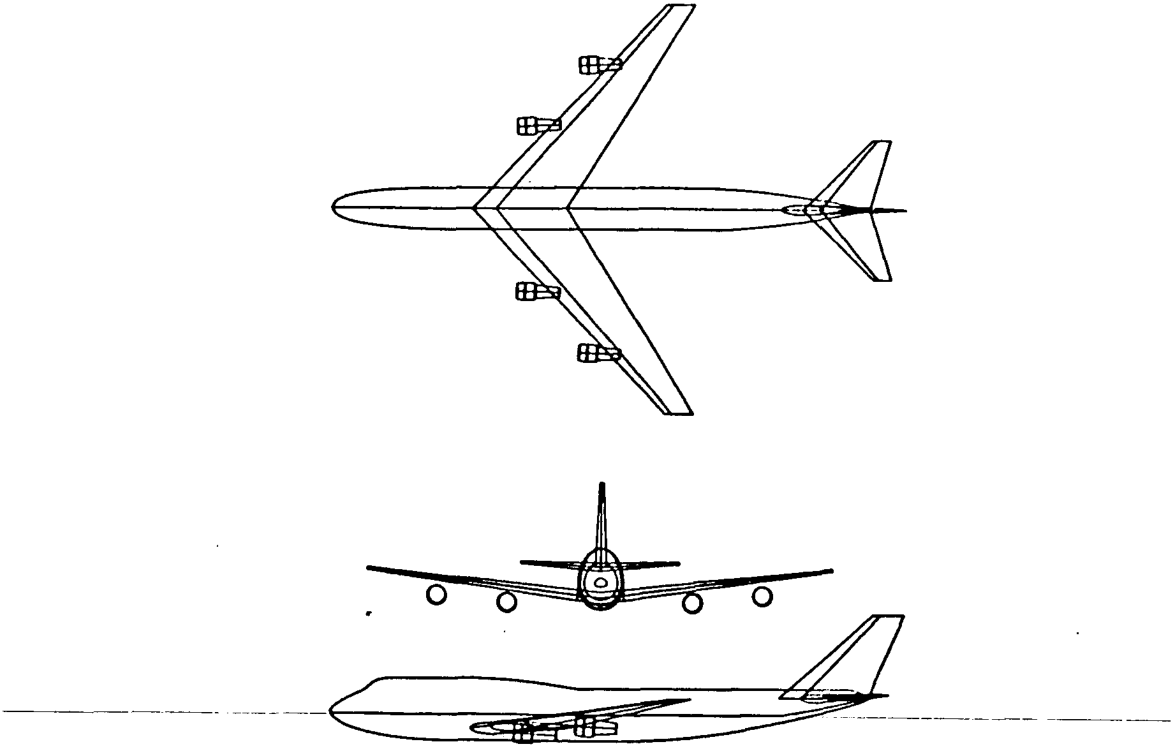


Figure 4.2: Sample 3-view configuration drawing from the SKETCH-program.

However, as it was only intended for postprocessing, the SKETCH program could not handle interactive drawing manipulation. After the implementation of the MEDUSA system on the ICI, development work was directed toward interfacing MEDUSA with ADAS, which rendered the SKETCH program obsolete.

4.2.1 The MEDUSA drafting and modelling system

The MEDUSA drafting and modelling system is a software product developed by Cambridge Interactive Systems (CIS) in the United Kingdom and it is offered as a "turnkey" CAD-system on several mini- and micro-computers, e.g. PRIME, VAX and SUN [Ref. 9]. The MEDUSA system architecture consists of a collection of separately available program modules configured around a general-purpose 2D drafting program, as shown in Figure 4.3:

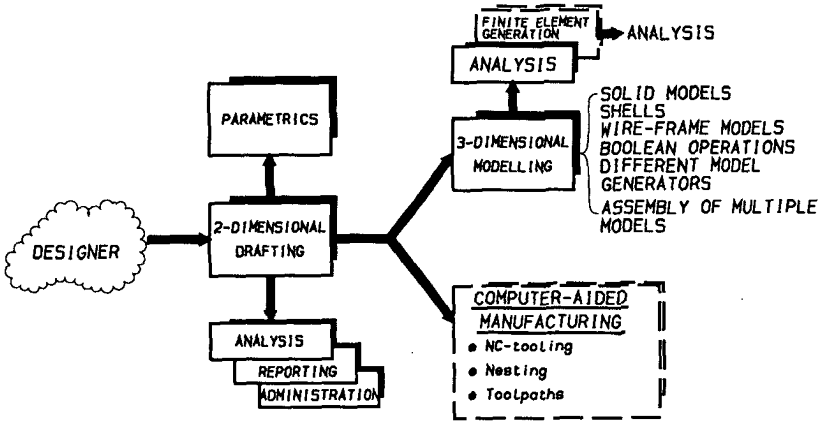


Figure 4.3: The MEDUSA-system modular configuration.

In general however, the ADAS-user will only be concerned with the 2D drafting and, occasionally, with the 3D modelling module. Therefore, only these programs will be briefly discussed.

4.2.1.1 2D drafting

The MEDUSA system can be invoked from ADAS command level or directly at operating system level. After start-up, the user will be at the drafting module command level. Command input is primarily menu-oriented, therefore MEDUSA can only be operated from an ICI-workstation with a data tablet.

At this point, an already existing drawing can be retrieved for modification (drawing editing) or a new drawing can be created. MEDUSA features numerous options to perform basic as well as complicated drawing operations [Ref. 11], e.g.:

- Automatic cross-hatching;
- Different line types;
- Different text types and fonts;
- Automatic dimensioning (DIN, ANSI or ISO standard);
- Transformation and duplication of drawing parts;
- Line editing ("rubber banding");
- Curve fitting (conics);
- Variable grids;
- Internal programming language (BaCIS2);
- Parametrics.

It is beyond the scope of this dissertation to discuss these options in detail, since they are standard for most CAD/CAM-systems.

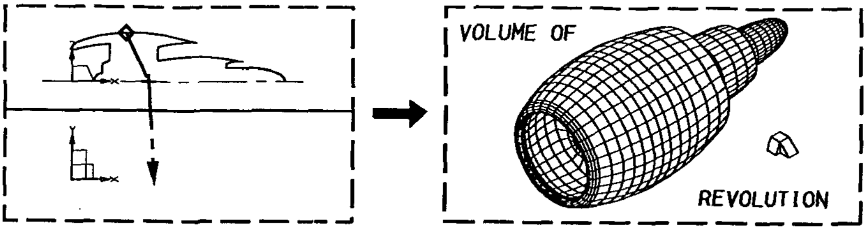
4.2.1.2 3D geometric modelling

The 3D module comprises a solid modelling and a viewing program [Ref. 10]. The solid modeller generates of 3-dimensional description of an object (model) while the viewer displays a projected image onto the graphics terminal.

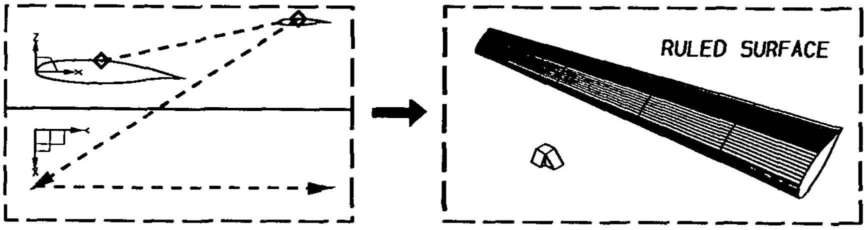
- The solid modeller

  The MEDUSA 3D module requires input of a specially prepared 2D drawing, referred to as a definition drawing. A definition drawing defines the object in terms of profile lines in one or more, orthogonal or oblique, views, interconnected by, so-called, link lines. The solid modeller generates a 3-dimensional description in terms of polygons (tiles), wire-lines and faces and stores the result in a model file. By default, the objects are modelled as solids, but special modelling commands are available to generate e.g. shells and wire-frame models. The solid modeller provides different model generators, which can be applied to objects with specific geometric properties, e.g.:

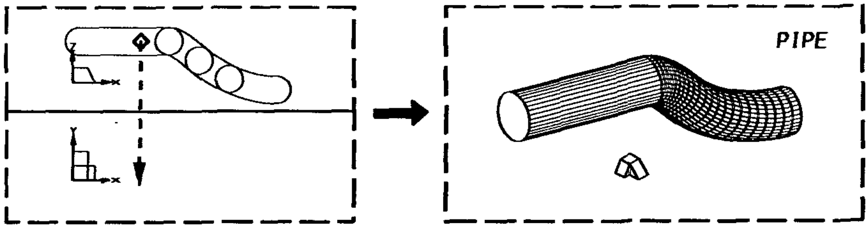
- **Volume of revolution: a profile line is rotated around a specified axis.**



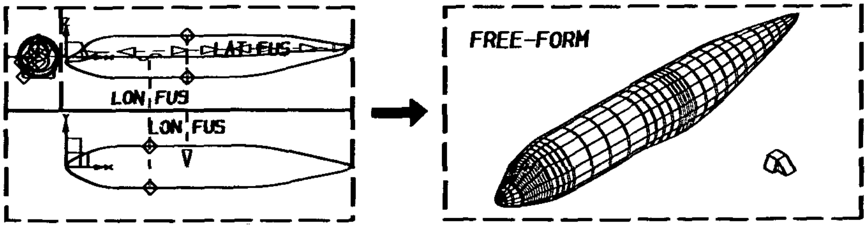
- **Ruled surface: linear lofting between two parallel endfaces.**



- **Pipe/slide: a cross-sectional profile line is extruded along a spatial curve.**



- **Free form modelling: the model is constructed from an arbitrary number of longitudinal and latitudinal profile lines.**



In addition, more complicated objects can be modelled with boolean operations. Boolean operations are used to join, subtract or take the complementary volume of two or more, separately defined objects. An object may be broken down into subcomponents which can in turn be modelled from separate definition drawings. The resulting object can subsequently be assembled in another drawing by indicating their relative locations and by referring to their corresponding model names (instancing).

- The (interactive) viewer and shader

  The viewer projects a model onto the graphics display according to a user-specified orientation. Typical viewer options are hidden line removal, display of tiles or boundary lines, isometric or perspective views, etc. The 3D viewer places generated views back into the definition drawing (reconstruction) so they can be saved in the drawing database or optionally sent to a line-plotter. The interactive viewer program module takes the viewing commands directly from the user rather than from a definition drawing. In this case, views cannot be saved or plotted, except for a local hardcopy. The functionality of the shader is similar to that of the interactive viewer, except that the shader is intended for rendering colour or greyscale images on a raster of bit-map terminal.

4.2.2 The ADAS → MEDUSA interface

To obtain better insight into the principles of the ADAS-MEDUSA interface a brief discussion will be given on the way MEDUSA organizes and stores information in a drawing. The information representing a MEDUSA drawing resides in a drawing database, stored under a user-specified name (sheet name). A drawing is a composition of graphical elements of which there are 3 basic classes, i.c.: LINE, TEXT and PRIM. A PRIM is a standard symbol, composed of LINE- and TEXT-elements, which is referenced as a single entity. The individual elements however can not be changed directly by the user. Each element class is associated with a number of specific attributes, shown in Table 4.1. These attributes define the shape, orientation and location of a graphical element in the drawing.

Table 4.1: MEDUSA graphical element classes and their associated attributes.

	ELEMENT CLASS		
	LINE	TEXT	PRIM
A	line type	text type	prim type
T	number of points	datum coordinates	datum coordinates
T	points coordinates	orientation	orientation
R	line functions	magnification	magnification
I	layer number	justification	layer number
B	mirror	mirror	mirror
U	layer number	layer number	layer number
T	data structural level	data structural level	data structural level
E	element marker	element marker	element marker
S	curve weight factors	text string	picture

The MEDUSA system provides the user with a set of tools in the form of FORTRAN subroutines to access a drawing database from an application program (DARS: Data Access RoutineS) [Ref. 12]. These routines can perform functions such as sequentially moving through the database, retrieve element attributes, delete and add elements. The ADAS-MEDUSA interface programs make use of these DARS-routines. To allow the interface program to identify graphical elements as particular aircraft components and to do subsequent computations, some simple rules have to be observed when creating a configuration layout drawing compatible with ADAS, i.c.:

- Elements are signified by placing them in predefined layer numbers (0 to 1023). Layer number is a common attribute for all element classes and arbitrarily assignable by the user. The function of layer numbers in MEDUSA is to simulate a stack of transparent sheets with selected elements. In addition, MEDUSA includes options to perform drawing operations on element(s) placed in the same layer.
- For each individual element, additional drawing instructions may be in effect, e.g. points sequence in a line, maximum number of points in a line, TEXT and PRIM orientation.

A complete list of aircraft components and their associated layer numbers and drawing conventions is given in Appendix B.

4.3 Design data storage and retrieval

4.3.1 Database management systems

An engineering database is a computer repository wherein all significant design data is stored in some organized manner. The contents of a database will be continuously modified and updated as the design evolves. In fact, a database represents the computer typification of a design and as such is often used synonymously.

Computer applications in design has consequently resulted in an increasing amount of design and engineering data generated. Database management systems (DBMS) form an invaluable tool for handling and manipulating this information in an efficient way. Moreover, there are additional aspects that make a database management system a key component in a computer-aided engineering environment, e.g.:

- All data is centrally stored as one set of logically related information and is no longer scattered among design offices in different forms, such as drawing, charts, tables, etc.
- Application programs can access data directly from the database, thus reducing the need for manual data preparation and formatting.
- Design data can more easily be accessed, sorted, verified, plotted, protected and approved. Also, data can be secured against ambiguity and loss of integrity.
- Project management can obtain immediate insight into the design status and progress.

Present commercial database management systems can be classified as hierarchical, network and relational by the logical way data is organized. Current research is mostly directed toward relational databases, as they are highly flexible in data representation. The basic concept of a relational database, is that data is archived under one or more keywords with additional attributes (schemes). Data with corresponding keywords are said to be related. Query commands allow logical and boolean operators to select and retrieve related data items. Examples of relational database management systems are the Relational Information Manager (RIM) which was developed under the IPAD-program [Refs. 27 and 55], AVID Relational Information System (ARIS) [Refs. 135 and 136], ORACLE, INGRES and RAPPORT.

As a suitable database management system was not implemented on the ICI at the time ADAS was conceived, a more specific method had to be developed to store ADAS design data. The ADAS database system will be discussed in the following Sections.

4.3.2 The ADAS design database

In the context of this dissertation, a database is a collection of design data that pertains to one design configuration. Obviously, a designer may own several of these databases. An ADAS database consists of a number of related files, as shown in Figure 4.4:

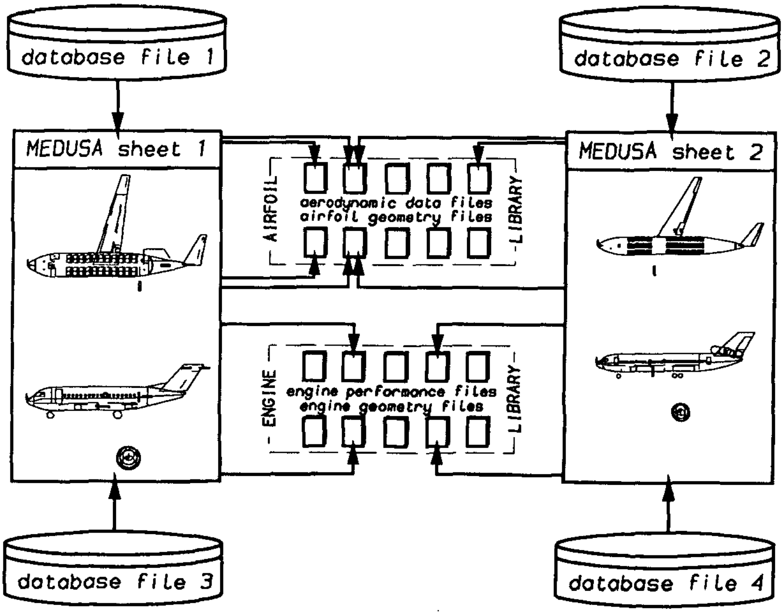


Figure 4.4: An ADAS design database composed of related data files.

- An ADAS database file typically contains non-geometric data, such as design speeds, altitudes, weights, aerodynamic polars, technology factors, etc. Also, a database file should contain additional input data, if the ADAP-program is to be run in a special control mode (see Section 2.4). The data structure of a database file will be described in Section 4.3.3. A database file may contain a MEDUSA sheet name which refers to:

- A MEDUSA sheet file, which in turn contains a geometrical description of the design configuration. A MEDUSA sheet file may be pertained to by different databases (see Figure 4.4). A configuration layout drawing is created and modified under control of the MEDUSA 2D drafting module (see Section 4.2.1). A description of the ADAS-MEDUSA interface is given in Section 4.2.2. A MEDUSA drawing may contain airfoil names and an engine name, which refer to:

- Airfoil files and engine file in the airfoil library and engine library respectively. The airfoil library contains files with geometry and aerodynamic data of standard airfoil sections. The engine library contains files with nacelle geometry and engine performance data of existing engine types. Engine and airfoil files may be pertained to by different databases (see Figure 4.4). A further discussion on the airfoil and engine library is given in Appendix C and D respectively.

Note that, although design data is distributed among several different files, a design is uniquely identified by only its associated database filename, simply referred to as the database name.

4.3.3 The database data structure

As mentioned in previous section, information in a database file consists of design data, a MEDUSA sheet name and, optionally, ADAP control input data.

Design data can be divided in general design data and aerodynamic polar, i.c.:

- General design data is represented by a vector, contained in a FORTRAN array data structure (1-dimensional). Currently the vector size is fixed at 10000 elements of single-precision real data type. Each element or entry can be referenced by its corresponding sequential index code. Although there is no specific relation among entries, particular by grouping certain items, e.g. weights, together in a sequence for efficiency.

The physical significance of each entry is defined in the Data Dictionary File (DDF). The DDF associates physical attributes with each entry, i.c.:

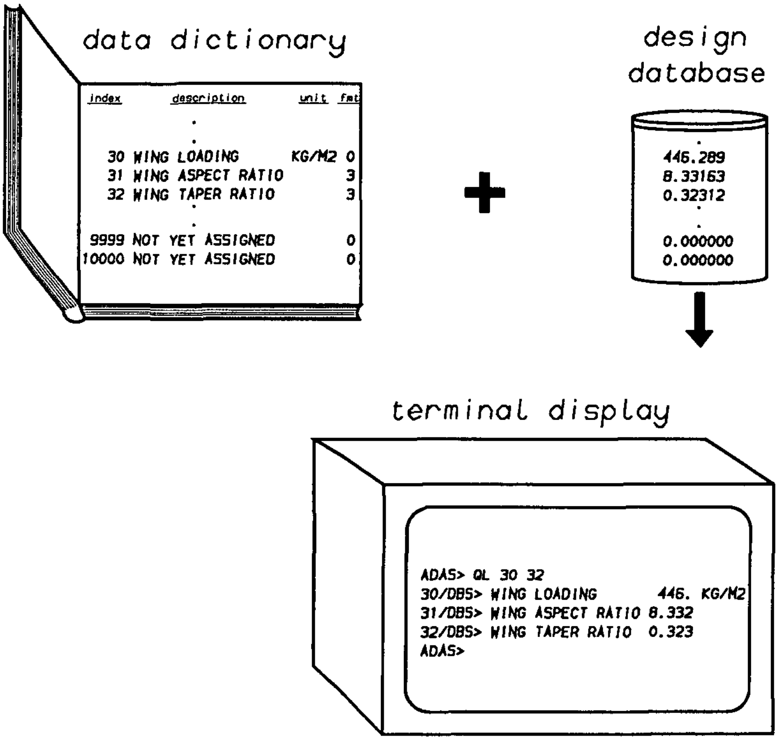
- a textual description (≤ 40 characters);
- physical unit (≤ 6 characters);
- print format (fmt), i.c. the number of significant decimal digits (0 ≤ fmt ≤ 9).

The DDF is a direct access method (DAM) file, i.c. records are identified by index which is particularly efficient for random access. The file can be accessed by many users at the same time. There is only one master DDF-file, thus the physical descriptions of database entries are common to all ADAS-users. Therefore, care must be taken that entries are not defined or redefined which may conflict with other users. The system supervisor may choose to write-protect the DDF for the general ADAS-user.

One disadvantage of the proposed database concept is that a database entry can only be referenced by its index number, rather then by some mneumonic name. However, the ADAS system provides a utility whereby the user can obtain a list of the DDF sorted either in alphabetical or in numerical order. The sorted list can subsequently be sent to the system text formatter (RUNOFF) which produces a print-ready layout. The DDF contains special control characters, such as tab characters, specifically for this purpose, e.g.:

```
 index          physical description          units  fmt
                         .
                         .
     44|WING AREA                            |M2      2
.*   45|NOT YET ASSIGNED                     |        0
                         .
                         .
```

The resulting database reference manuals can be used to cross-reference indices a their corresponding physical significance and vice-versa. When a query command is given at ADAS command level, the system will display the value and physical attributes of the selected entries on one line, as shown in Figure 4.5:



index	description	unit	fmt
30	WING LOADING	KG/M2	0
31	WING ASPECT RATIO		3
32	WING TAPER RATIO		3
9999	NOT YET ASSIGNED		0
10000	NOT YET ASSIGNED		0

```
ADAS> QL 30 32
30/DBS> WING LOADING        446. KG/M2
31/DBS> WING ASPECT RATIO 8.332
32/DBS> WING TAPER RATIO  0.323
ADAS>
```

Figure 4.5: Relation between the DDF and database entries.

- Aerodynamic polars refer to relationships between the lift, drag and pitching moment coefficient of the complete configuration versus flight condition and flap setting (Figure 4.6). These aerodynamic coefficients are represented in the form of 2-dimensional tables with up to 10 x 10 data points and can either be assigned indirectly by the designer, e.g. if this data comes from some external source, or they can be computed directly with an aerodynamic analysis program and then stored in the database.

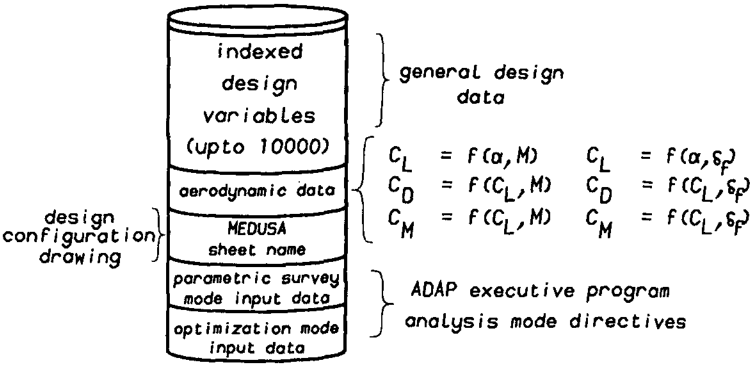

indexed design variables (upto 10000)
aerodynamic data
MEDUSA sheet name
parametric survey mode input data
optimization mode input data
general design data
$C_L = f(\alpha, M)$ $C_L = f(\alpha, \delta_f)$
$C_D = f(C_L, M)$ $C_D = f(C_L, \delta_f)$
$C_M = f(C_L, M)$ $C_M = f(C_L, \delta_f)$
design configuration drawing
ADAP executive program analysis mode directives

Figure 4.6: ADAS design database structure.

- As mentioned in Section 3.2, the designer has the option to run the ADAP executive program in parametric survey and/or optimization mode. In either case, additional information has to be included in the design database, i.e.:

  - <u>Parametric survey mode</u>

    In parametric survey mode (explicit optimization), the analysis program will be successively executed for different combinations of 1 or 2 design variables, referred to as survey variables. Within the analysis program, up to 20 dependent design characteristics, referred to as survey functions, may be computed. The survey function values will be evaluated and sampled for each survey point. When all surveys are completed, the results are saved in a standard data file in the form of parametric data tables. To select parametric survey mode, the user needs to specify the physical descriptions of the survey variable(s) and functions, i.c. a textual name, units and the number of significant decimal digits (print format). In addition, the user must specify the minimum and maximum range limits between which the survey variables are to be varied. Finally, the number of required survey points in each direction must be specified. The current limit is 10 x 10 survey points. If the number of survey points is not specified, parametric survey mode is assumed to be disabled, but the definitions of the survey variables and functions will be retained in the design database until explicitly erased.

- Optimization mode

  In optimization mode, i.c. implicit (automated) optimization, control is passed to a general-purpose numerical optimization algorithm for objective function optimization, i.c. minimization or maximization, by variation of selected free variables and subject to nonlinear constraints. Analoguous to survey variables and functions, the free variables, the objective function and constraints are defined by specifying their associated physical attributes. Also, the user must specify the upper and lower bounds for the free variables (side constraints) and for the general constraints. The general formulation of implicit inequality constraints is:

  $$B_l \leq \text{constraint function} \leq B_u$$

  If $B_u = B_l$, the constraint will be considered as an equality constraint. If some large, positive or negative, value is specified for either B_u or B_l , that bound is assumed not to be restrictive and is consequently ignored. Finally, the maximum number of iteration steps must be specified; otherwise optimization mode is assumed to be disabled, but the definitions of the free variables, the objective function and constraints will be retained in the design database until explicitly erased.

After all the necessary data have been entered, the design database is saved in the UFD under a user-specified filename, which will render all modifications to the database permanent. ADAS will request for user confirmation to overwrite an existing database or give a warning when the user attempts to terminate the ADAS session without having saved the modified database.

Although the ADAS database concept is rather specific and straightforward, it has proven to be adequate for conceptual design purposes. Moreover, database access times are negligible with only little demand on system resources. However, its rigid data structure inhibits easy interfacing with other programs and systems. Also, options to sort and compare data of different designs and to perform statistical analysis are not available.

Therefore, future implementation of a standard relational database system will be an advantage, provided its performance is acceptable.

4.4 Definition of design analysis methods

4.4.1 The analysis program

In principle, the ADAS system does not inherently contain design knowledge. The user is expected to provide this information in the form of a FORTRAN-program, simply referred to as an analysis program. An analysis program should contain the computer algorithm to perform the required analysis on a given design configuration. The user can apply structured programming to tailor computations and control logic to a specific design problem. Admittedly, some basic experience in FORTRAN-programming is required. However, considering the level of computer experience of potential ADAS users at the considerable gain in flexibility of application, this concept seems justified.

An analysis program itself is a subroutine with a fixed name: DSPROG. The general structure of the DSPROG-routine is as follows:

```
SUBROUTINE DSPROG (FV,OBJ,CO,SV,SF,ICALL)
DIMENSION FV(1:20),CO(1:20),SV(1:2),SF(1:20)
                   .
                   .
                   .
     any user-supplied FORTRAN-code,
     including calls to other subprograms
                   .
                   .
                   .
RETURN
END
```

The variables in the parameter list are significant only if the analysis program is processed in one of the special ADAP analysis modes:

- For <u>multivariate optimization</u>:
    - FV represents the values of the free variables (input);
    - OBJ is the value of the objective function computed within the analysis program (output);
    - CO represents the values of the constraints, if any, computed within the analysis program (output).

- For <u>parametric survey</u>:
    - SV represents the values of the survey variables (input);
    - SF are the values of the survey functions computed within the analysis program (output).

The significance of the ICALL-parameter will be explained in Section 5.5.

The data communication between subprograms is via the standard FORTRAN-mechanisms, i.e. variables are passed through a parameter list or through a named data area (common block). To write an ADAS-compatible analysis program, only some basic programming conventions have to be observed with respect to consistent variable names, data types and common block names.

After an analysis program has been coded, it can be input into the ADAS-system, which will then automatically compile and link it with the ADAP executive program for processing.

4.4.2 The program library

To avoid duplication of effort, standard and commonly used analysis methods can be (pre-)programmed and stored in the program library as FORTRAN-callable subprograms. The program library serves the function of a method base, from which designers can select subprograms and incorporate them in their analysis programs as "building blocks". As the program library does not constitute an integral part of the ADAS-system, the subprograms can be easily modified or replaced without affecting the generic ADAS-architecture (Figure 4.7):

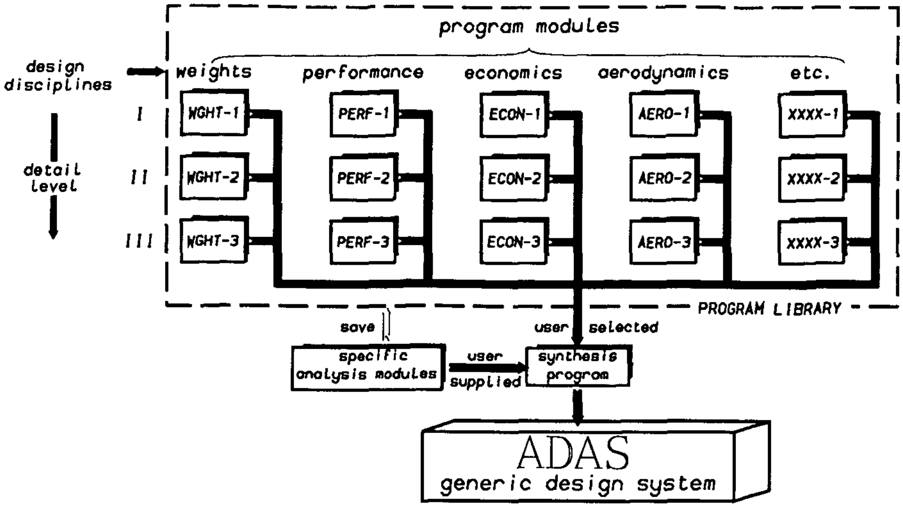


Figure 4.7: Composition of a design synthesis program from user-supplied modules and modules in the program library.

Ideally, the program library should contain subprograms covering a wide spectrum of design disciplines. Also, subprograms with different levels of accuracy and detail should be implemented to comply with the requirements at a particular design stage.

When properly tested and documented, user program modules can be added to the program library by the system supervisor to make them available for general use. Part 3 of this dissertation contains typical examples of analysis methods which are currently implemented in the program library.

5. DESIGN ANALYSIS

Once an aircraft configuration has been defined and an analysis program has been coded, the actual analysis computations can be initiated by running the ADAP executive program. The ADAP-program needs to know from which files to collect input data from and into which files to save analysis results. This information needs to be defined in a so-called project file before ADAP is run.

5.1 The project file

A project file is a global variable type file and it contains a list of global variable names and their corresponding values. The mechanism to define and maintain global variables is a feature provided by the operating system as a means to customize the user environment. The variables are global in the sense that they can be referenced at any level, i.e. at operating system command level, from a command program or from any user application program.

The ADAS-system requires the following global variables to be specified in a project file:

Global variable	Example value	Description
.NAME	C. Bil	User identification
.DATE	2 nov 1987/09:11:06	Date/time since last ADAS-session
.IDBS	T02250.DBS	Database filename
.DATA	T02250.DATA	File for alphanumeric output data
.PLOT	T02250.PLOT	File for graphical output data
.DSPR	WGHT.DSPR	Analysis program source code file
.PROJ	44 PASS. AIRLINER	Project identification

A user may own several project files for different design projects. A project file must always be active when ADAS or ADAP is used. This file is referred to as the current project file and remains active until explicitly closed or when another project file is selected.

Essentially, a project file contains an actual list of files with data pertaining to the design project the user is currently working on, hence the name. Although a project file is primarily intended as a means of communication between the ADAS and ADAP programs, most ADAS commands which require a file name will automatically select the appropriate filename from the current project file by default.

5.2 ADAP compile, load and run step

Before an analysis program can be processed, it must be compiled first and linked with the ADAP-(main)program and the necessary subroutine libraries, e.g. the program library, FORTRAN library, numerical routines (NAG), MEDUSA drawing interface routines (DARS), to resolve all external references. These steps will produce an executable file, which can subsequently be processed.

A command program U$ADAP.CPL is available in the ADAS-system which can perform this sequence of tasks automatically. The flow structure of this program is given in Figure 5.1. U$ADAP.CPL takes upto 3 arguments. A project filename may be given which must contain the names of input and output files for ADAP. If omitted, ADAS_GVAR is assumed. The flag -C specifies that a new executable file has to be created. U$ADAP.CPL takes the source file of the users analysis program specified in the project file, compiles it and, if successful, links it to with the ADAP main program and subprogram libraries. If the flag -GO is (also) given, the executable file is subsequently processed. If neither -C or -GO is specified, -GO is assumed. The link step in particular can take up a relatively long time. However, compilation and linkage is only required if changes are made to the analysis program source code, otherwise an executable file residing in the UFD can be executed directly with different input data. Alternatively, compilation and linkage can be submitted as a background process (phantom), leaving the terminal or workstation available for other tasks (see Section 5.3).

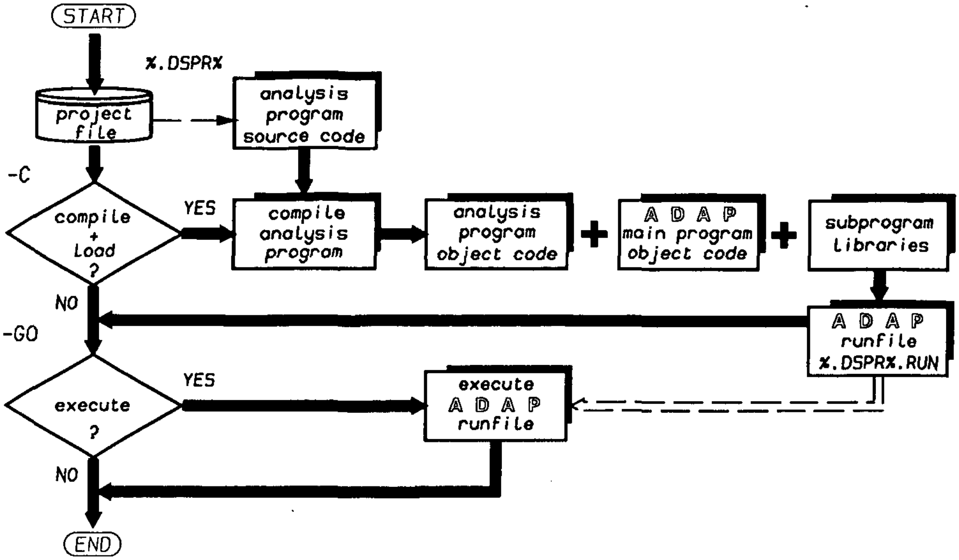


Figure 5.1: Control flow of the ADAP start-up utility program.

5.3 The ADAP executive program

Before ADAP can be executed, all the required input data must be defined and available from disk files specified in the project file. As the ADAP-program only communicates with these disk files and does not require any intermediate user action, ADAP can be run as a completely self-contained process. There are 3 processing modes available:

- Interactive processing: ADAP initiates processing instantly and all analysis results generated, appear on the terminal screen. Alphanumeric data is also automatically saved in the specified data file %.DATA% for reviewing at a later time. Plot data, if generated, will be saved in the specified MEDUSA sheet file %.PLOT%, but will not appear directly on the screen. However, plots saved in a MEDUSA sheet file can be regenerated and edited afterwards with the MEDUSA drafting system.

- Background processing: ADAP initiates processing instantly, but it does not tie up the terminal which remains available to the user for other purposes. All generated output will be automatically saved in the specified data and MEDUSA sheet file, %.DATA% and %.PLOT% respectively, for reviewing at a later time. A background process or phantom is treated by the system as an additional (virtual) user. Consequently, phantoms can reduce overall system performance. The system will notify the user when a phantom has terminated.

- <u>Batch processing</u>: Basically similar to background processing, except that the execution of ADAP is delayed until after some specified clock time, usually during the night time. Batch processing is controlled by a batch monitor. The batch monitor will not only suspend execution of the phantom until after a certain clock time, but it will also restrict available CPU-time and elapsed time. If either is exceeded, the batch monitor will terminate the job. Batch processing is particularly recommended for jobs with relatively long execution times, e.g. automated multivariate optimization with many free variables.

5.4 ADAP data storage and retrieval

Before ADAP initiates the processing of the user-supplied analysis program DSPROG, it first collects all the necessary design input data from the files specified in the project file and the MEDUSA sheet file. This sequence is schematically illustrated in Figure 5.2.

First, the design database %.IDBS%, specified in the project file, is accessed and all the design data are loaded into primary memory (common blocks). Next, if a valid MEDUSA sheet name is specified in the database, the MEDUSA → ADAS interface routine D$MEDG is activated. D$MEDG scans the drawing database and evaluates the graphical elements into significant geometrical properties and positional parameters of aircraft components. If the drawing contains an engine TEXT-element, engine performance and nacelle geometry data are collected from the UFD, c.q. from the engine library VL>VO>ADAS>ENGINE. If the drawing contains airfoil TEXT-elements, airfoil aerodynamic and geometry data are collected from the UFD, c.q. from the airfoil library VL>VO>ADAS>AIRFOIL.

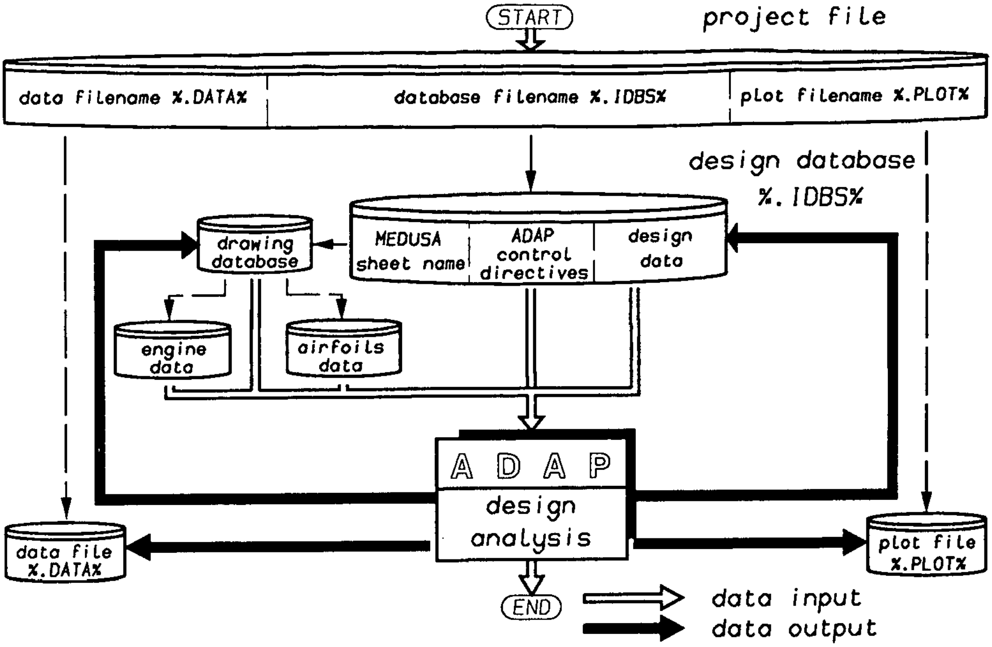


Figure 5.2: ADAP design data in- and output from and to selected files.

As shown in Section 4.3, a design database can only contain a fixed number of predefined data items, which have to be general and common to all ADAS-users. In some cases however, it may be necessary to store additional analysis results, which are specific and problem-dependent. In fact, the program library contains several print and plot modules. There are 2 user-specified files in which these analysis results can be stored. The files %.DATA% and %.PLOT% are used to save alphanumerical and graphical output data, respectively. Alphanumeric output is both printed at the user terminal and saved in the data file %.DATA%. This file must always be specified to assure that data is retained for reviewing at a later time. If plots are generated during the ADAP-run, the result will be saved in a MEDUSA sheet file %.PLOT% but will not be displayed on the user terminal until regenerated with the MEDUSA 2D-module.

ADAP will report the data input sequence so the user can verify that the correct input files and analysis program have been processed:

```
                    ****************************
                    *                          *
                    *  ADAP EXECUTIVE PROGRAM  *
                    *                          *
                    *    - P R O J E C T -     *
                    *                          *
                    *     44 PASS. AIRLINER    *
                    *                          *
                    ****************************

ADAP> ** Start ADAP at:            10:53:41
ADAP> ** By:                       C. Bil
ADAP> ** Output data file:         T02250.DATA
ADAP> ** Analysis program file:    WGHT.DSPR
ADAP> ** Read from ADAS database:  T02250.DBS
ADAP> ** Read from MEDUSA sheet:   S.T02250
ADAP> ** Airfoil file  1/ 1:       VL>VO>ADAS>AIRFOIL>FXXROOT
ADAP> ** Airfoil file  2/ 1:       VL>VO>ADAS>AIRFOIL>FXXTIP
ADAP> ** Airfoil file  1/ 2:       VL>VO>ADAS>AIRFOIL>N0012
ADAP> ** Airfoil file  2/ 2:       VL>VO>ADAS>AIRFOIL>N0012
ADAP> ** Airfoil file  1/ 3:       VL>VO>ADAS>AIRFOIL>N0012
ADAP> ** Airfoil file  2/ 3:       VL>VO>ADAS>AIRFOIL>N0012
ADAP> ** Airfoil file  1/ 5:       VL>VO>ADAS>AIRFOIL>N0012
ADAP> ** Airfoil file  2/ 5:       VL>VO>ADAS>AIRFOIL>N0012
ADAP> ** Engine file:              VL>VO>ADAS>ENGINE>CF-34

                           .
      references to the user-supplied analysis program (DSPROG)
                           .

ADAP> ** Write to ADAS database:  T02250.DBS
ADAP> ** ADAP ended at:           10:54:45

** STATISTICS **

CPU TIME:         3372 CSEC
DISK TIME:         645 CSEC
```

ADAP execution can be interrupted by the user at any time with the built-in break-handler.

5.5 ADAP analysis modes

After all input data have been collected from the specified files, the ADAP executive initiates the actual processing of the user-supplied analysis program, by calling the subroutine DSPROG. Depending on whether a particular analysis mode has been selected in the design database, a different control path is followed. The control flow in this part of ADAP is schematically illustrated in Figure 5.3:

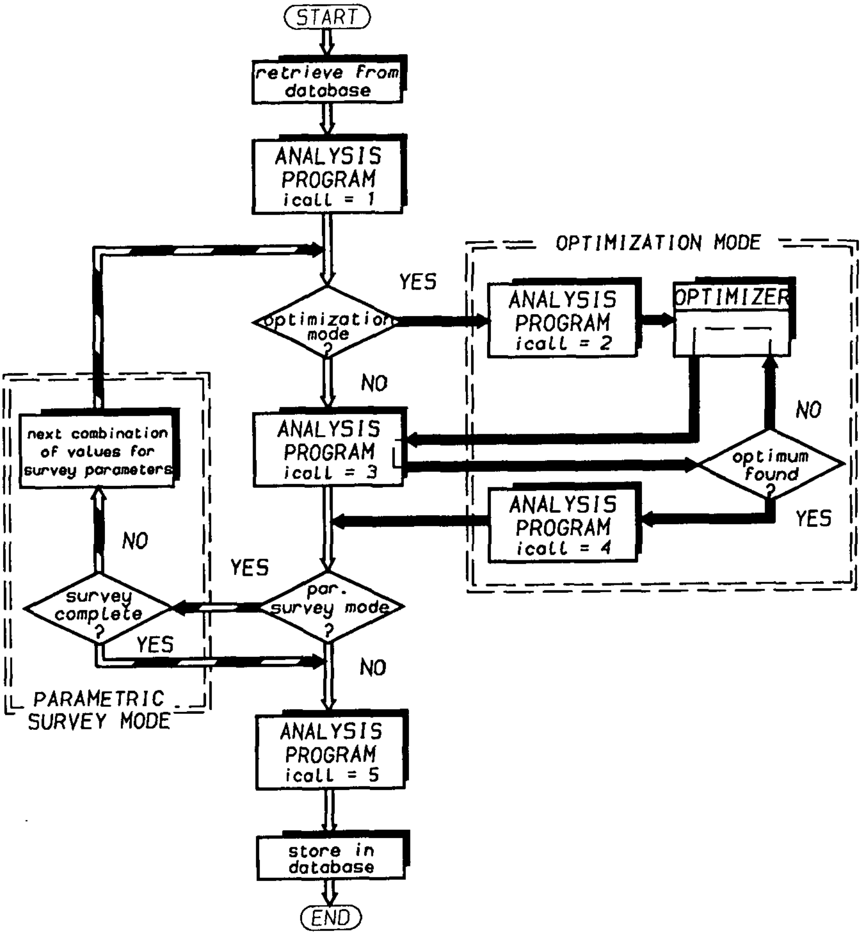


Figure 5.3: ADAP analysis control flow.

The DSPROG-routine is called at different points in the ADAP control flow, each time with a different value for the ICALL-parameter. The user can test on this parameter and use conditional statements to take selected computations outside the parameter survey loops or the iteration calls of the optimizer and so minimize CPU-time. In the following Sections the individual analysis modes will be discussed.

5.5.1 Parametric survey mode

If survey variables are defined in the design database (see Section 4.3.3), ADAP switches to parametric survey mode. The control scheme for parametric survey simply consists of 1 or 2 nested loops. For each survey point, the DSPROG-routine is called with different values for the survey variables, contained in the SV(1:2)-array, and the selected survey functions are output from DSPROG in the SF(1:20)-array, as shown in Figure 5.4:

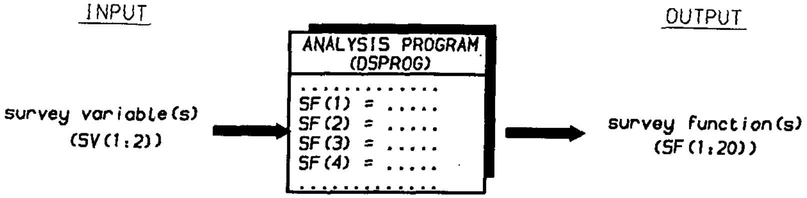


Figure 5.4: Input and output variables of DSPROG in parametric survey mode.

The values for the survey variables S_V are determined by dividing the range bounds V_u and V_l by a number of equidistant points N_s, as follows:

$$S_{V_{ix}} = V_{l_x} + \frac{V_{u_x} - V_{l_x}}{N_{s_x} - 1} (i - 1) \qquad \text{with } i = 1,N_{s_x} \qquad (5\text{-}1)$$

and

$$S_{V_{jy}} = V_{l_y} + \frac{V_{u_y} - V_{l_y}}{N_{s_y} - 1} (j - 1) \qquad \text{with } j = 1,N_{s_y} \qquad (5\text{-}2)$$

The control sequence is such that S_{V_x} changes most rapidly (inner loop) and S_{V_y} increases most slowly (outer loop). The direction of change in S_{V_x} is alternately reversed with every step for S_{V_y}, giving a "zigzag"-effect. This is to assure that, in the case parametric survey and implicit optimization are combined, the optimum solution of a survey point can be used as the starting point for the next. ADAP will sample these survey functions values until all surveys are completed. Subsequently, the results are printed as parametric data tables. An example is given in Figure 5.5:

```
Function:      MAXIMUM TAKEOFF MASS      KG      0
Table dim.:    10 x 10
X-variable:    WING LOADING              KG/M2   0
Y-variable:    WING ASPECT RATIO                 2
```

X Y	300.	333.	367.	400.	433.	467.	500.	533.	567.	600.
5.00	19322.	19071.	19066.	19249.	19606.	20121.	20819.	21748.	22933.	24423.
6.67	19550.	19141.	19004.	19066.	19304.	19690.	20224.	20939.	21868.	23062.
8.33	19904.	19363.	19122.	19097.	19257.	19567.	20019.	20645.	21458.	22480.
10.00	20322.	19663.	19333.	19238.	19336.	19590.	19992.	20558.	21306.	22257.
11.67	20809.	20015.	19597.	19436.	19482.	19690.	20048.	20582.	21289.	22182.
13.33	21336.	20406.	19896.	19673.	19670.	19837.	20168.	20671.	21340.	22212.
15.00	21914.	20832.	20227.	19941.	19892.	20025.	20328.	20798.	21453.	22320.
16.67	22540.	21291.	20585.	20235.	20140.	20240.	20511.	20964.	21617.	22458.
18.33	23222.	21783.	20968.	20551.	20412.	20474.	20719.	21165.	21804.	22608.
20.00	23962.	22310.	21376.	20889.	20702.	20724.	20951.	21388.	22002.	22775.

Figure 5.5: General parametric data table format.

This table format is compatible with the ADAS/MEDUSA plot facility and data can subsequently be plotted in, in a format compatible with the ADAS plot facility.

5.5.2 Optimization mode

If free variables are defined in the design database (see Section 4.3.3), ADAP is switched into optimization mode. In optimization mode, control is passed to an optimization algorithm for objective function optimization, subject to nonlinear constraints. The optimizer will call the DSPROG-routine as often as is required to reach the optimum design. The values of the free variables are passed through the FV(1:20)-array and the DSPROG-routine must return the corresponding values of the objective function and constraints, if any, in the OBJ-variable and CO(1:20)-array, respectively, as shown in Figure 5.6:

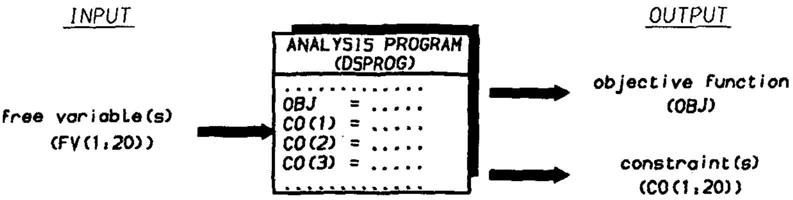


Figure 5.6: Input and output variables of DSPROG in optimization mode.

Multivariate optimization applied to aircraft design has been a research topic for about 10 years [Refs. 83 - 85]. A wide range of optimization algorithms are currently available, each with specific properties and applications. As in ADAS the objective function and constraints are defined by the user, a general-purpose optimization procedure had to be implemented. The particular optimization method currently in incorporated is referred to as Sequential Quadratic Programming (SQP), which has shown to be an efficient optimization procedure in comparative studies [Ref. 110]. The algorithm is available from the NAG-library (E04VCF) and has been slightly modified for scaling of variables and approximation of gradients [Ref. 14]. Although the designer usually employs an optimization routine as a tool, it is useful to have some understanding of the mathematical principles involved to appreciate its capabilities and limitations. This Section is only concerned with the description of the general strategy of the SQP-method. Special safeguards against ill-conditioning and degenerate problems are implemented but will not be discussed.

The SQP-method finds an optimum of a nonlinear objective function subject to nonlinear constraints by sequentially generating a quadratic and linear approximation to the objective function and constraint functions respectively, in the neighbourhood of the current iteration point $\bar{x}$ and solving the constrained optimum. For each major iteration step, referred to a the QP-phase, the problem can be mathematically formulated as:

$$\text{minimize } c^T\bar{x} + \frac{1}{2}\bar{x}^T H \bar{x} \qquad (5\text{-}3)$$

$$\text{with } \quad \bar{x} \in R^n \qquad (5\text{-}4)$$

subject to:

$$B_l \le \binom{\bar{x}}{Ax} \le B_u \tag{5-5}$$

where $\bar{x}$ is a vector representing the free variables at the current iteration point, c is a vector with first derivatives. As analytical gradient information is generally not available, the objective function and constraint functions gradients are approximated by taking small orthogonal steps Δh for the individual free variables (second-order differences):

$$\frac{\partial f}{\partial x} = \frac{f(x + \Delta h) - f(x - \Delta h)}{2\ \Delta h} \tag{5-6}$$

The step size Δh is related to the number of significant decimals fmt (print format), specified for the corresponding free variable, as follows:

$$\Delta h = \beta_h \cdot 10^{-fmt} \tag{5-7}$$

β_h is a user-specified proportionality factor and should be selected such that Δh is sufficiently small to give accurate gradients and sufficiently large to assure that machine round-off and truncation errors do not become dominant. Usually, $0.5 \le \beta_h \le 2$ gives reasonable results with $\beta_h = 1$ as a good first approximation. The second-order derivatives are represented by the Hessian-matrix H which are continuously updated with the Broyden-Fletcher-Goldfarb-Shanno approximation formula [Ref. 93]:

$$H^{k+1} = H^k + \frac{\tau^k \sigma^k (\sigma^k)^T - \sigma^k (y^k)^T H^k - H^k y^k (\sigma^k)^T}{(\sigma^k)^T y^k} \tag{5-8}$$

where

$$\tau^k = 1 + \frac{(y^k)^T H^k y^k}{(\sigma^k)^T y^k} \tag{5-9}$$

$$\sigma^k = x^{k+1} - x^k \tag{5-10}$$

and

$$y^k = \nabla f(x^{k+1}) - \nabla f(x^k) \tag{5-11}$$

The search strategy of the SQP-method can best be illustrated for an optimization problem with two free variables for which a geometric interpretation can be given in a contour plot (Figure 5.7):

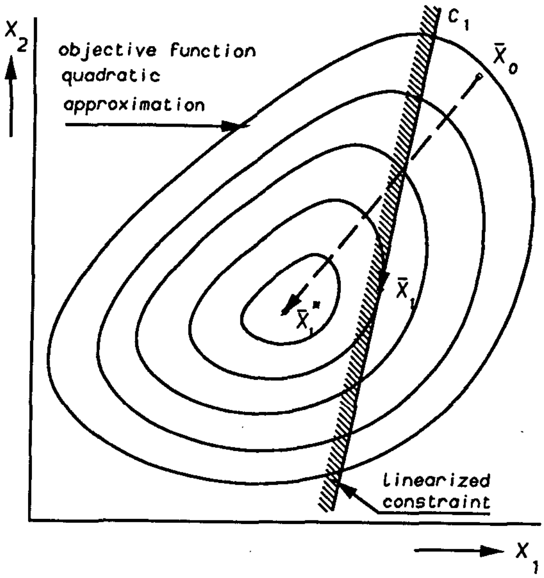


Figure 5.7: Geometric interpretation of the SQP-method for 2 free variables.

Assume at some iteration step the design is at point $\bar{x}_0$, which is feasible but not optimal. The objective function is locally approximated by a quadratic function and differentiated with respect to the free variables to obtain a new, unconstrained optimum $\bar{x}_1^*$. The design is checked for feasibility. If feasible, the process is repeated. If infeasible, a search is performed along the line $\bar{x}_0 \rightarrow \bar{x}_1^*$ until the design is just feasible. The particular constraint C_1 is said to be active and is added to the working set of constraints. Next the constrained optimum $\bar{x}_1$ is computed which becomes the new approximation for the optimal design.

This procedure works essentially similar for multiple free variables. During the optimization process, constraints which are violated are added to a working set of currently active constraints. As they are treated as equality constraints, the active set strategy has the favourable feature that the linearized constraints are not violated between iterations in the QP-phase: the SQP-method belongs to the category of feasible direction methods. Constraints are deleted from the working set only if the gradient vector points into the feasible region and if substantial improvement in the design can be obtained.

The user has control the amount and type of intermediate information that is to be displayed during each iteration. In this way the user can monitor the optimization process and, optionally, interrupt execution if the process seems to diverge or sufficient accuracy is obtained. Optionally, the user can include a plot routine available in the program library to draw a convergence history plot to verify convergence and to assess the performance of the optimizer. An example of a convergence history plot is given in Figure 5.8:

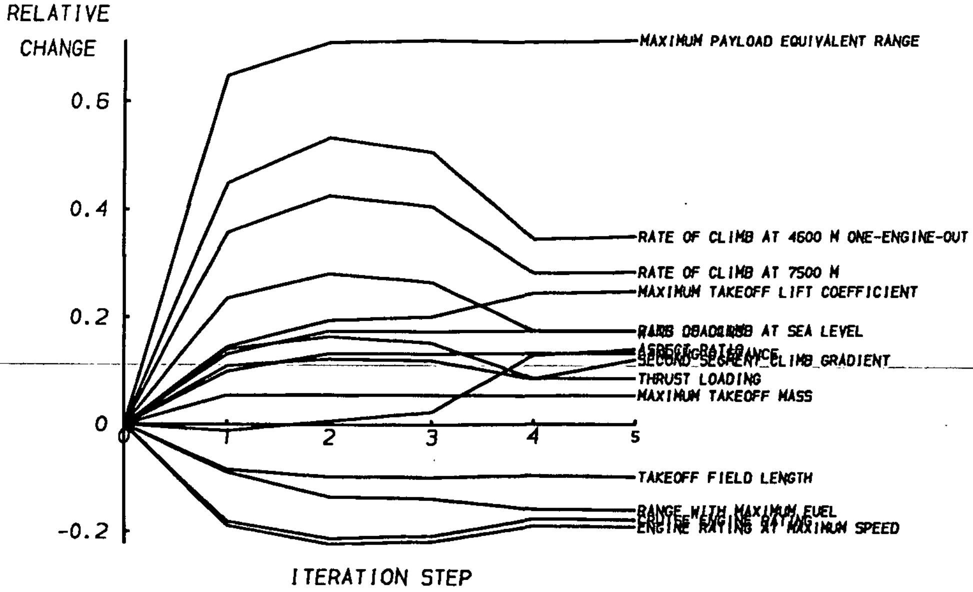


Figure 5.8: A convergence history plot in numerical optimization.

Note that the optimizer is contained within the parametric survey loops, thus both analysis modes can be selected simultaneously. An example of such an application is given in Part 4.

5.5.3 Design point analysis mode

If neither parametric survey or optimization is selected, ADAP switches to design point analysis mode by default. In this state, the DSPROG-routine is called only once and ADAP subsequently terminates. This option is typically selected if a design characteristics are to be evaluated for a given design configuration. For example, the designer can first perform design optimization, either by parameter variation or numerical optimization, and subsequently perform a more comprehensive analysis on the optimum design with ADAP in analysis mode. The program library contains several print and plot routines which can be included in the analysis program to represent design characteristics in suitable format for design documentation.

6. DESIGN EVALUATION

In the last phase of a typical design cycle, the analysis results are rendered and the design is evaluated. Although the decision to change or to accept the design is entirely the designers responsibility, a CAE-system should assist in this task by providing the necessary options to represent selected analysis results in a convenient format.

6.1 Data postprocessing with ADAS

The ADAS-system gives several alternatives for data postprocessing, in summary:

- Plots saved in sheet file(s) can be regenerated interactively on a graphics display for viewing with MEDUSA. Graphs can be edited as required using the standard MEDUSA 2D commands. Optionally, a sheet file can be sent to a remote plotter for high-quality copies.

- An alphanumeric data file can be listed or edited with the system text editor. Optionally, a data file can be spooled to a remote printer for high-quality copies.

- Simple database query commands are available to check the values of those database entries changed by the analysis program.

- ADAS includes facilities for graphical postprocessing of general parametric data. There are several options available to interactively generate different types of engineering diagrams and charts (see Section 6.1). Graphs are directly plotted into a MEDUSA sheet.

- The user can invoke the MEDUSA-system to display and inspect the completed 3-view configuration drawing. Optionally, this drawing can be submitted as a definition drawing for the solid modeller to create a 3D model description.

Because of their significance, these last two options will be discussed in more detail in the following Sections.

6.2 Graphical representation of parametric data

The ADAS graphics facility provides the user the capability to interactively create different types of engineering diagrams from general parametric data. Graphs are automatically drawn into a MEDUSA sheet. The user can subsequently edit, save and plot the drawing as required with the standard MEDUSA options. Parametric data is represented by discrete data points:

$$F_{ij}(x_i,y_j) \quad \text{for} \quad \begin{array}{l} i = 1,N_x \\ j = 1,N_y \end{array} \qquad (6\text{-}1)$$

with $2 \geq N_x$ and $1 \leq N_y \leq 10$. $N_y \leq 1$ for the 1-dimensional case. Parametric data tables should reside in a file, e.g. the standard data file %.DATA%. These tables may be embedded among other data, but must have the predefined format given in Figure 5.5. Note that when ADAP is run in parametric survey mode, the survey function values are automatically generated in the correct table format and saved in the data file %.DATA% (see Section 5.5.1). Airfoil and engine data are also represented in this format, so they can be readily plotted with ADAS. Before a graph can be made, the parametric data must be sampled from one or more specified files and placed into a logical plot stack. When the DATA-command is given, ADAS automatically scans the specified file for the selected data table(s) and subsequently copies the information in the next slot of the plot stack. The plotting facility in turn extracts plot data from the plot stack for a given index number, as illustrated in Figure 6.1:

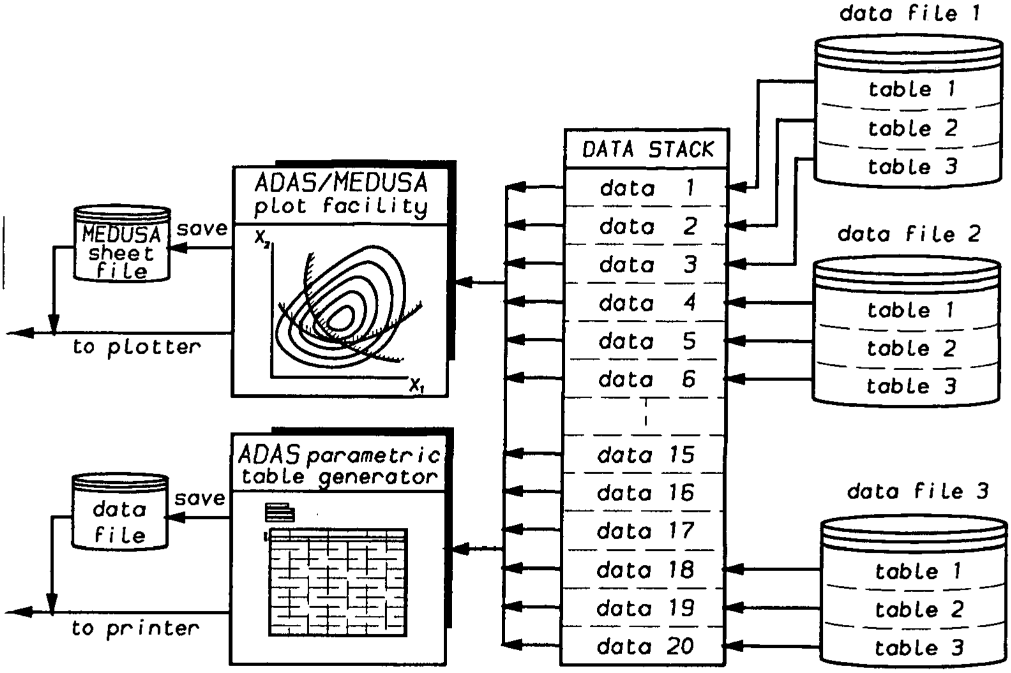


Figure 6.1: The ADAS plot stack.

The purpose of the plot stack is twofold:

- It speeds up plotting as file(s) do not have to be searched for parametric data tables each time a new plot is made as the data is already in main memory;
- Referencing plot data is much easier as only the corresponding index number is sufficient to identify data, instead of having to indicate which table should be selected from what file.

The plot stack can hold upto 20 different plot functions for plotting multiple functions in a contour plot. Consequently, these plot functions must dependent on the same two plot variables. When a table is copied from a file into the plot stack, the descriptive attributes of the plot functions and variables are also saved. However, with respect to the plot variables, ADAS only saves the descriptive attributes of the last table copied.

6.2.1 Types of engineering diagrams

Several options are available to the user to generate different types of engineering diagrams. They will be briefly discussed in the following Sections.

6.2.1.1 Carpet plots

The traditional way to graphically represent parametric data F = f(x) in a Cartesian axes-system, is to plot the plot variable x along the horizontal axis and the plot function F along the vertical. For 2-dimensional data F = f(x,y), several curves f(x) or f(y) are drawn for which y = constant or x = constant respectively. An example is given in Figure 6.2:

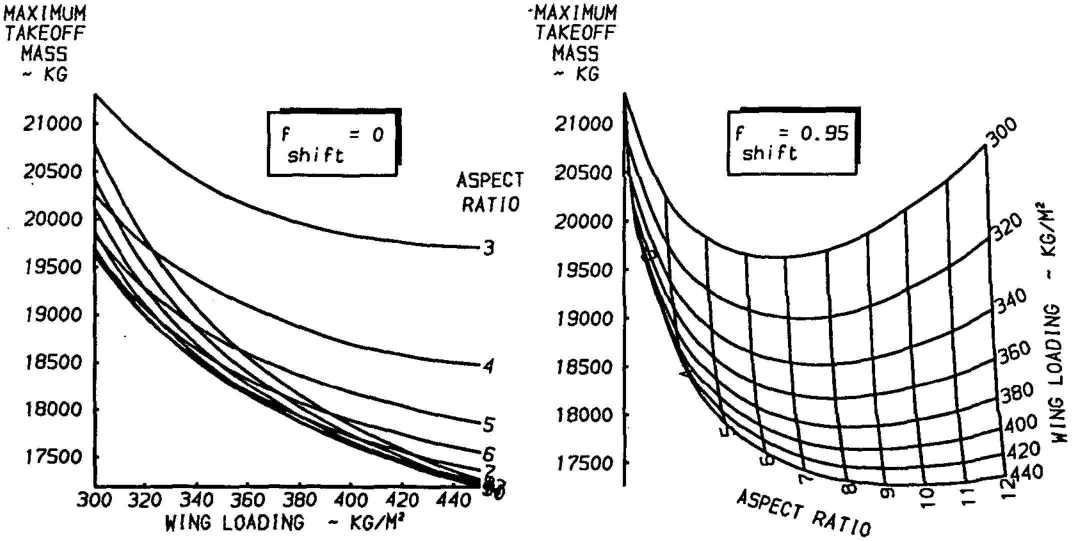


Figure 6.2: Example of a standard graph with 2 plot variables.

However, in the latter case, curves may intersect or overlap and the chart becomes difficult to interpret. A possible solution to this problem is to displace the origin of each curve proportionally to the right. In this case, curves f(x) and f(y) are drawn for which y = constant and x = constant respectively, giving a carpet plot its typical appearance. The horizontal axis has been eliminated.

In ADAS the amount of displacement is controlled by the user-specified shift factor. The shift factor f_{shift} follows from the relation:

$$\Delta O = \frac{f_{shift} \, N_x}{l_x} \qquad (6\text{-}2)$$

where ΔO is the origin displacement in absolute plot units, l_x is the available horizontal plot space in absolute plot units and N_x is the number of curves to be drawn. Consequently, the origin displacement for curves for which x = constant follows from Eq. (6-2) with the shift factor $1 - f_{shift}$. If the user selects $f_{shift} \leq 0$ or $f_{shift} \geq 1$, a horizontal axis is drawn for the x- or y-plot variable respectively.

A different type of carpet plot may be generated by plotting 2 plot functions F_1 and F_2 versus x or x and y. The 2 plot functions are plotted along the axes, while curves are drawn for which x = constant and y = constant. An example is given in Figure 6.3:

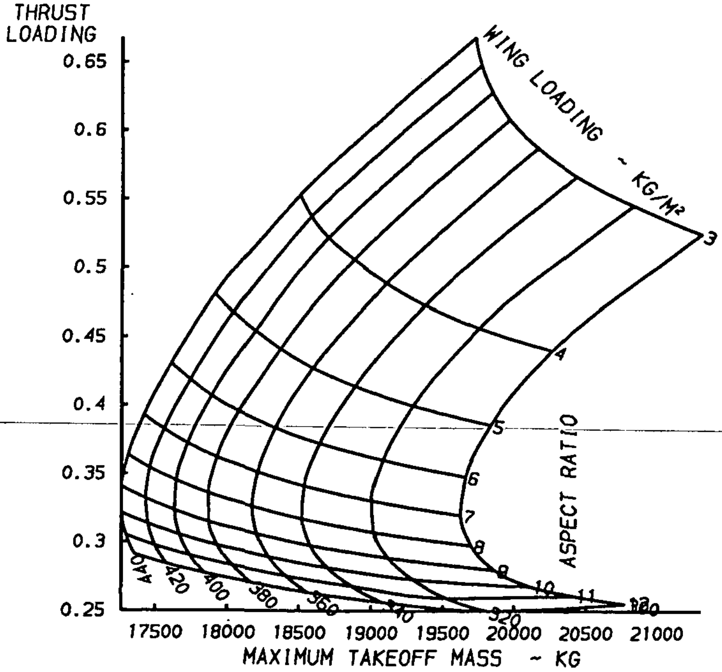


Figure 6.3: Example of a graph with 2 plot variables and 2 plot functions.

Note that curves may be multi-valued.

6.2.1.2 Surface plots

With surface plots, an isometric projection is displayed of 2-dimensional data F = f(x,y), as shown in Figure 6.4:

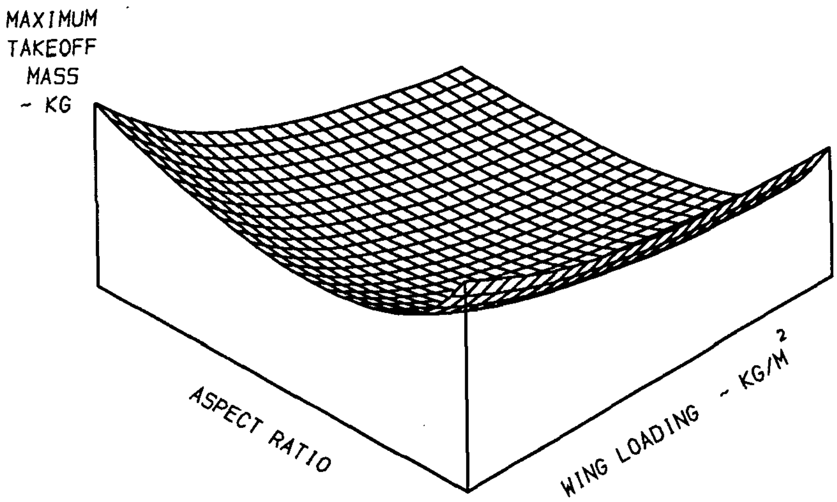


Figure 6.4: Example of an isometric projection of a function with 2 plot variables.

Hidden lines are automatically removed. The user has the option to rotate the surface about the vertical axis in steps of 90 degrees, thus pointing a different corner towards the viewer. Although surface plots are usually not suitable for reading off accurate function values due to the 3-dimensional character, this option has been included in ADAS to assess the general behaviour of a function.

6.2.1.3 Contour plots

Contour plots are a means to graphically represent 2-dimensional data F = f(x,y) by drawing curves in the xy-plane for which F = constant, as shown in Figure 6.5:

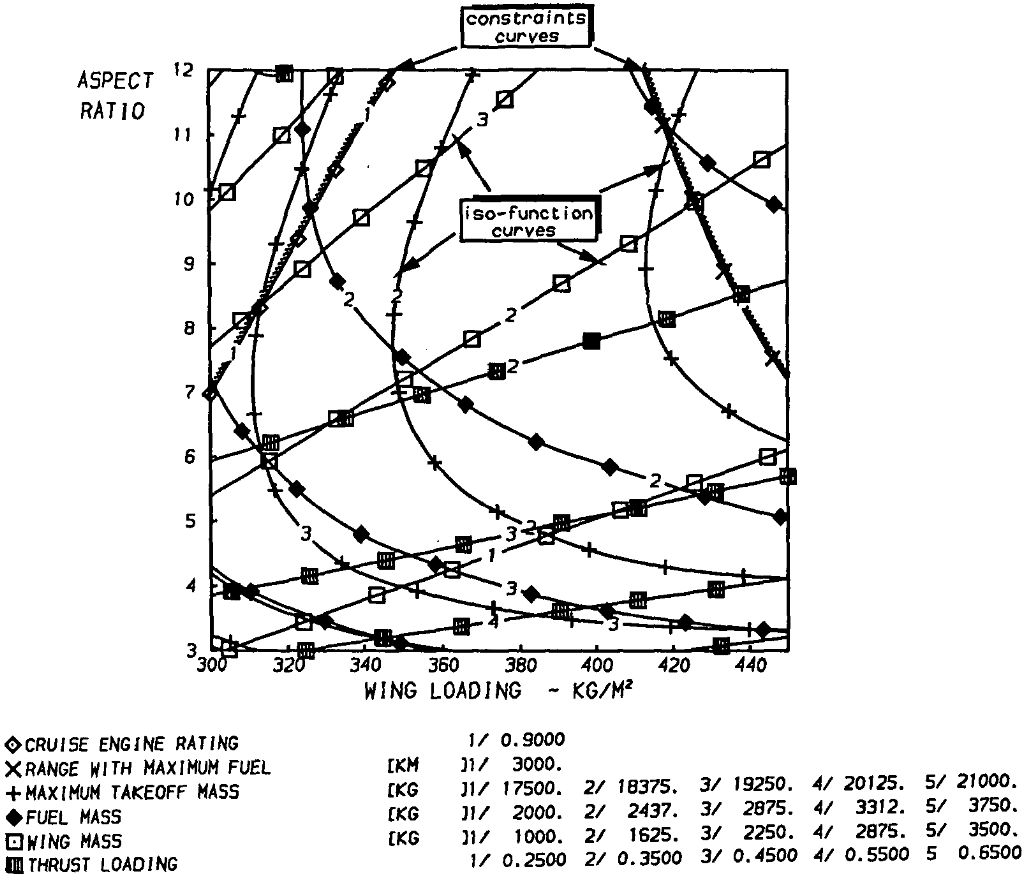


Figure 6.5: Example of a contour plot with multiple functions versus 2 plot variables.

The function values of the iso-function curves are specified by the user. Contour plots are generally convenient in representing parametric survey data for sensitivity studies. The user can display multiple functions in one chart, while constraints can be visualized by selecting hash-marks to be drawn on the high- or low-side of the curve. Finally, a pick function is available in ADAS by which the user can select a specific design point in the contour plot using the tracking-cross. For each design point thus indicated, the corresponding function values can be displayed either on the alphanumeric terminal or added to the plot itself.

6.2.2 Auto-scaling and annotation

By default, ADAS applies automatic scaling for annotation of axes and curves. This implies that a chart will exactly fit the physical plot region. The interval for tick-marks along the axes is determined as follows. Assume that the minimum and maximum value to be plotted are F_{min} and F_{max} respectively, then:

$$\Delta F = \frac{F_{max} - F_{min}}{N_s} \tag{6-3}$$

where N_s is a suitable value for the number of tick-marks required. The interval value can be written as mantissa x 10^{exp}, then the exponent is computed as follows:

$$\text{exp} = \text{entier}\ (\log \Delta F)$$

and the mantissa becomes:

$$\text{mantissa} = 2 \qquad \text{if } \frac{\Delta F}{10^{exp}} < 2 \tag{6-4}$$

$$\text{mantissa} = 2.5 \qquad \text{if } \frac{\Delta F}{10^{exp}} < 2.5 \tag{6-5}$$

$$\text{mantissa} = 5 \qquad \text{if } \frac{\Delta F}{10^{exp}} < 5 \tag{6-7}$$

otherwise

mantissa = 1 and the exponent becomes entier (log ΔF) + 1.

For annotation of axes N_s = 10. This usually results in approximately 10 tick-marks. For annotating curves, a similar procedure is followed, except that N_s may be specified by the user.

6.2.3 Curve fitting and interpolation techniques

Interpolation techniques are widely used in a computer-aided engineering environment, be it for drawing a smooth curve through a number of discrete data points or for general interpolation for a specific point. The ADAS-system offers two different curve fit techniques for plotting parametric data with at least 4 points, i.c.:

- Osculatory method, which gives a rather free flowing curve, but may produce unwanted oscillations and inflections;
- Butland's method, which results in a more tightly fitting curve, but oscillations are avoided.

Both methods perform local interpolation on a sequence of 4 adjacent data points (x_i, y_i) for $i = 1,4$. The mathematical principles of these methods will be explained for single-valued curves, i.c. x is monotonic. For multi-valued curves parameterization x(t) and y(t) has to be applied, but the theory is basically similar.

6.2.3.1 Osculatory method

Assume a sequence of 4 points (x_i, y_i), $i = 1,4$ as shown in Figure 6.6. To fit a cubic polynomial through point 2 and 3, the osculatory method proceeds as follows:

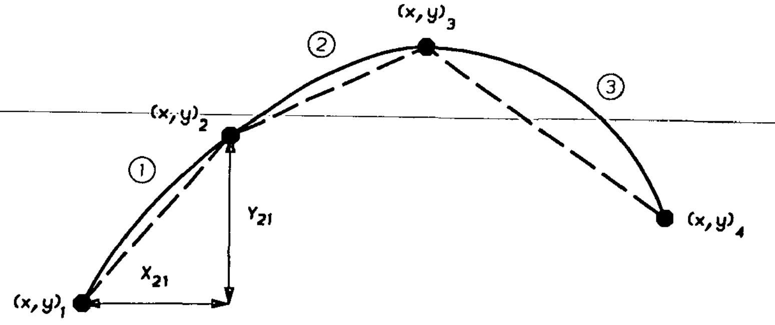


Figure 6.6: Principle of curve fitting with the osculatory method.

First, the first derivatives at point 2 and 3 are estimated by fitting a quadratic polynomial through points 1, 2 and 3 and point 2, 3 and 4 respectively:

$$q_i = \frac{\Delta x_{i,i-1} S_{i+1,i} + \Delta x_{i+1,i} S_{i,i-1}}{\Delta x_{i+1,i-1}} \qquad \text{for } i = 2,3 \qquad (6\text{-}8)$$

where

$$\Delta x_{i+1,i} = x_{i+1} - x_i \qquad \text{for } i = 2,3 \qquad (6\text{-}9)$$

$$S_{i+1,i} = \frac{y_{i+1} - y_i}{\Delta x_{i+1,i}} \qquad \text{for } i = 2,3 \qquad (6\text{-}10)$$

The cubic polynomial passing through point 2 and 3 and satisfying q_2 and q_3 can be written as:

$$y = y_2 + \Delta x(q_2 + \Delta x\ (C_2 + \Delta x\ C_3)) \qquad (6\text{-}11)$$

where

$$\Delta x = x - x_2 \quad \text{for } x_2 \leq x \leq x_3 \qquad (6\text{-}12)$$

$$C_2 = \frac{3\ (y_3 - y_2) - (2q_2 + q_3)(x_3 - x_2)}{(x_3 - x_2)^2} \qquad (6\text{-}13)$$

$$C_3 = \frac{(q_2 + q_3)(x_3 - x_2) - 2(y_3 - y_2)}{(x_3 - x_2)^3} \qquad (6\text{-}14)$$

In this way, first derivative continuity at the joints is assured in piecewise applications. If there are only three points available, e.g. at the edges of the data region, a fourth point is generated by quadratic extrapolation on the available 3 points and the process continues as above.

The osculatory method was originally developed by Ackland [Ref. 2] and is still widely used for curve drawing. Extensions to the osculatory method have been suggested for second derivative continuity at the joints, however these methods usually involve higher-order polynomials (quintics) and are less convenient computationally. Moreover, discontinuity in the second derivatives is generally not noticeable in the drawn curve.

6.2.3.2 Butland's method

Although the osculatory method generally results in a smooth, free flowing curve, unwanted oscillations or inflections may sometimes occur. If this is the case, the ADAS-user can select an alternative interpolation technique, based on Butland's method [Ref. 31]. Butland's method is attractive in combination with the osculatory method as the algorithms are basically identical, except for the estimation of the slopes q_2 and q_3. The slope at point i is computed from:

$$q_i = \frac{3(\Delta x_{i,i-1} + \Delta x_{i+1,i})}{(\Delta x_{i,i-1} + 2\Delta x_{i+1,i})/S_{i,i-1} + (2\Delta x_{i,i+1} + \Delta x_{i+1,i})/S_{i+1,i}}$$

for $S_{i,i-1}\, S_{i+1,i} > 0$, or (6-15)

$$q_i = 0 \qquad (6\text{-}15)$$

for $S_{i,i-1}\, S_{i+1,i} \leq 0$. The method proceeds as described above for the osculatory method.

6.3 The ADAS → MEDUSA interface

As mentioned in Section 2.2, a two-way interface has been developed to exchange geometric information between ADAS and MEDUSA. Data transfer from a MEDUSA sheet file to the ADAP executive program is automatically initiated when a valid sheet name is included in the design database. To add or change geometry information from ADAP in a MEDUSA sheet file automatically, the routine D$MEDP, available in the program library, must be explicitly included in the user-supplied analysis program.

The purpose of the D$MEDP-routine is twofold:

- In case the configuration is implicitly changed e.g. after ADAP has been run in either optimization or parametric survey mode, the D$MEDP-routine will automatically alter the original drawing according to the newly computed geometry. Thus, the designer can readily verify if the new configuration is acceptable.
- The D$MEDP-routine will also add 3D link lines to the drawing, as follows:

component	model generator
- Fuselage	Free-form model
- Lifting surfaces (per panel)	Ruled surface
- Engine nacelles	Volume of revolution
- Extended intake ducts	Pipe
- Undercarriage wheels	Volume of revolution

Thus, the drawing is suitable for input to the MEDUSA solid modeller in oder to generate a model definition. A typical example of such a definition drawing is given in Figure 6.8.

As it is not a priori known which geometry parameters are changed in a particular design study, the D$MEDP-routine must, in principle, redraw the complete configuration drawing. However, a more practical approach is used whereby only graphical elements placed in certain layers are regarded as 'changeable'. For example, the fuselage geometry and interior are considered as not 'changeable'. This is because fuselage design is primarily driven by payload and systems requirements and therefore numerical optimization is less practical. Currently, the layer numbers 200 to 1032 are designated as 'changeable'. When the D$MEDP-routine is invoked, it first erases all the elements in these layers and subsequently replaces them with the new geometry. All other elements are not erased and remain unchanged.

The drawing created by the D$MEDP-routine can be submitted to the MEDUSA solid modeller to generate a 3-dimensional model description. The conceptual design model defined in the definition drawing in Figure 6.8 is given in Figure 6.9. Once a model file has thus been created, the design can be displayed at any user-specified orientation with respect to the viewer.

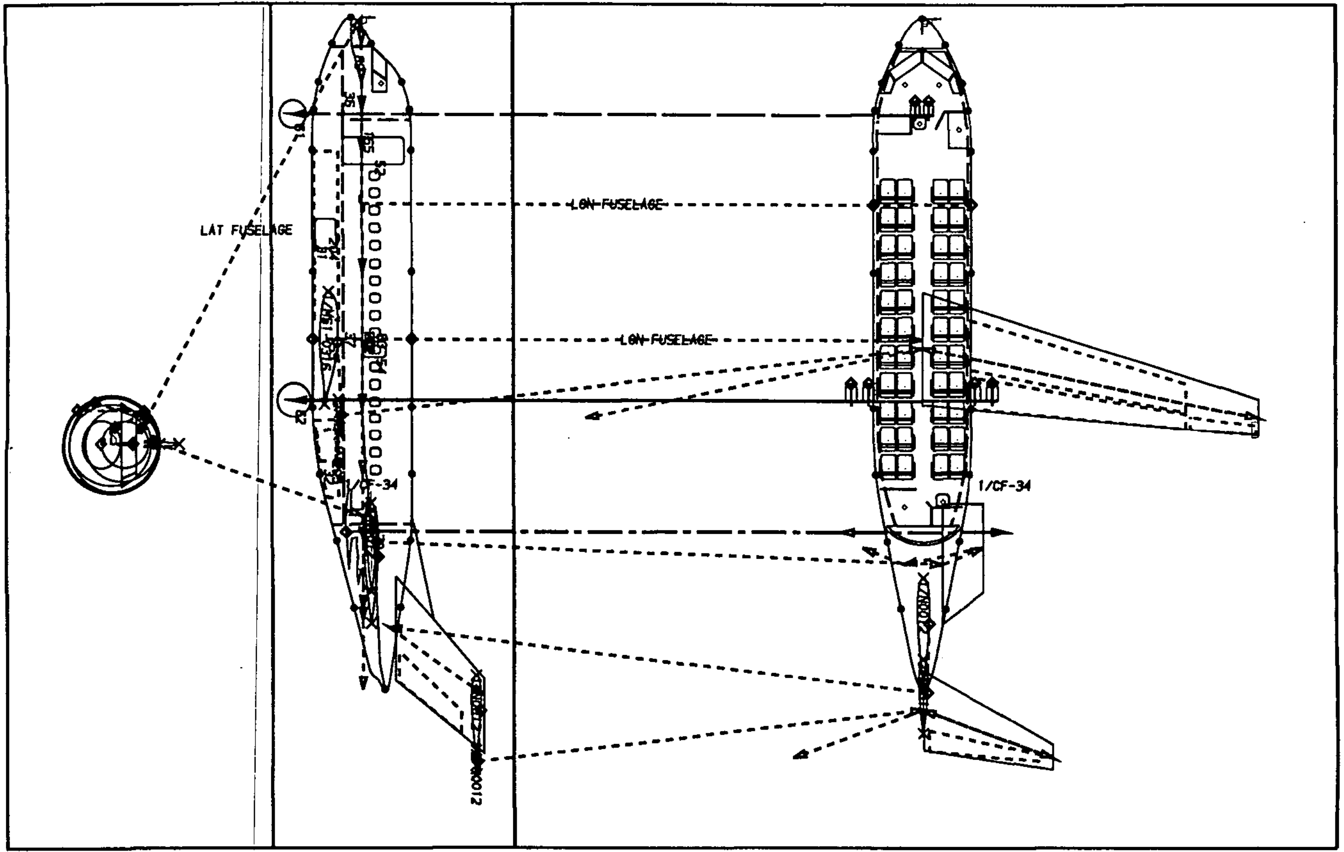


Figure 6.8: Typical 3-view drawing of a conceptual design configuration. The MEDUSA drawing, including the 3D link lines, is automatically created with the D$MEDP-routine.

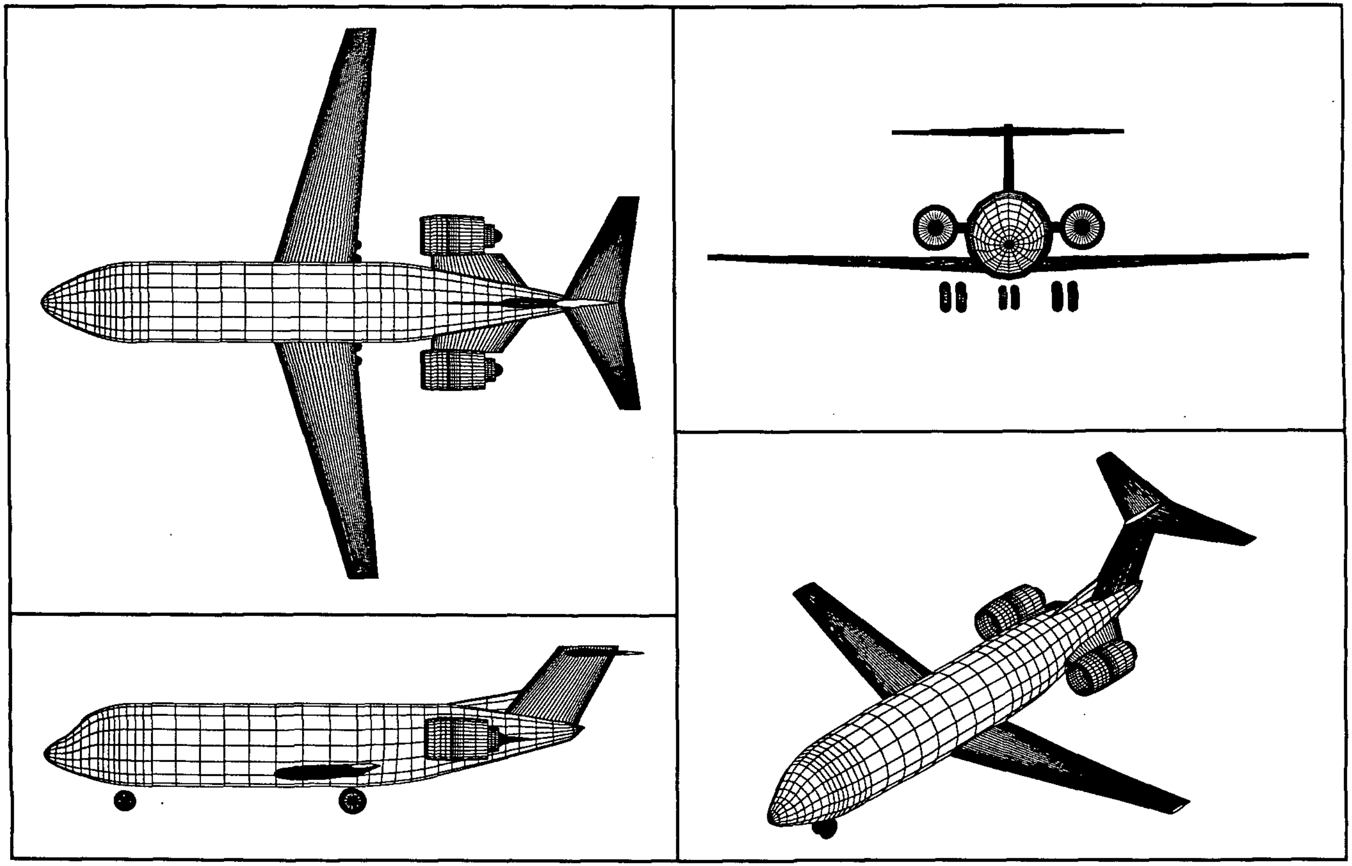


Figure 6.9: Conceptual design solid model representation. The model was constructed from the definition drawing given in Figure 6.8.

Typical viewing options are isometric and perspective views, hidden-line removal and creating cuts or sections. Although a conceptual design model is rather simplified in the sense that it lacks detail such as fairings, flap tracks and struts, it is a useful tool to assess the general configuration layout and to check clearances and geometric interferences. It should be noted however that the user can subsequently add any required detail with the standard MEDUSA 2D and 3D options, using the simplified model as a starting point. An example of such a detailed model is given on the cover of this dissertation.

PART 3: ANALYSIS METHODS FOR CONCEPTUAL AIRCRAFT DESIGN

In this part examples are given of typical analysis methods for conceptual aircraft design. Several alternative methods for design synthesis can be found in various textbooks, which can in principle be implemented in ADAS if so required. Methods and procedures are presented in a concise form, except for specific modifications and implementations. One is referred to the specified literature for a more detailed description.

7. WEIGHT ANALYSIS

In order to estimate the characteristic aircraft design weights, i.e. basic empty weight, operational empty weight, maximum zero-fuel weight and maximum takeoff weight, the aircraft is broken down into a number of weight groups, i.e.:

1	Fuselage structure	(D$FUSW)
2	Wing structure	(D$WNGW)
3	Empennage structure	(D$TALW)
4	Undercarriage	(D$UCWT)
5	Control surfaces	(D$CSWT)
6	Propulsion	(D$PRPW)
7	Engine nacelles and pylons	(D$NACW)
8	Airframe systems and instruments	(D$SYSW)
9	Furnishing and equipment	(D$FURW)
	--------------------------------- +	
	Basic empty weight	(D$WSUM)
10	Operational items	(D$OIWT)
	--------------------------------- +	
	Operational empty weight	(D$WSUM)
11	Maximum payload	(D$PAYW)
	--------------------------------- +	
	Maximum zero-fuel weight	(D$WSUM)
12	Fuel	(D$WSUM)
	--------------------------------- +	
	Maximum takeoff weight	(D$WSUM)

In turn each group is divided into a number of sub-components of which weight and center of gravity location are computed.

The following sections give examples of weight prediction methods currently implemented in the program library. Most of these methods have been derived from Refs. 21 and 124 and are based on statistical analysis applied to some population of existing aircraft designs. These methods have been implemented in such a way that the correlation coefficients can be assigned a value by the user through the design database. In this way, basic equations can be modified as required, provided the appropriate weight factors are known. Currently, upto 5 weight factors φ can be specified per weight item.

7.1 Fuselage weight

The fuselage weight prediction methods, derived from Ref. 124, are based on the methods first introduced by Burt [Ref. 32] and have been updated for modern aircraft fuselages. The total fuselage group weight is the summation of weight contributions for which equations are given in the following Sections.

Skin weight

Skin weight is determined by the critical load condition:

$$W_{sk} = \varphi_1 + \varphi_2 \left(\frac{l_t}{b_f + h_f}\right)^{\varphi_3} S_f^{\varphi_4} V_D^{\varphi_5} \qquad \text{(strength)} \qquad (7\text{-}1)$$

or

$$W_{sk} = \varphi_1 + \varphi_2 \, \Delta P^{\varphi_3} D_f^{\varphi_4} S_f^{\varphi_5} \qquad \text{(max. pressure differential)} \qquad (7\text{-}2)$$

or

$$W_{sk} = \varphi_1 + \varphi_2 \, S_f^{\varphi_3} \qquad \text{(minimum gauge)} \qquad (7\text{-}3)$$

where ΔP is the design pressure differential. Skin weight is the greater of the Eqs. (7-1), (7-2) and (7-3).

Stringers and longerons weight

The weight equation for stringers and longerons is:

$$W_{str} = \varphi_1 + \varphi_2 \left(\frac{l_t}{b_f + h_f}\right)^{0.75} S_f^{\varphi_3} \; V_D^{\varphi_4} \; n_{ult}^{\varphi_5} \tag{7-4}$$

Standard frames weight

The weight equation for standard frames is:

$$W_{fr} = \varphi_1 + \varphi_2 \left(W_{sk} + W_{str}\right)^{\varphi_3} \tag{7-5}$$

Flooring weight

The weight equation for fuselage floor structures is:

$$W_{fl_i} = \varphi_{1i} + \varphi_{2i} \, S_{fl_i}^{\varphi_{2i}} + \varphi_{4i} \, V_{co_i}^{\varphi_{5i}} \qquad \text{for } i = 1,10 \tag{7-6}$$

Bulkheads weight

The weight equation for fuselage (pressure) bulkheads is:

$$W_{bh_i} = \varphi_{1i} + \varphi_{2i} \, \Delta P^{\varphi_{3i}} \, S_{bh}^{\varphi_{4i}} \qquad \text{for } i = 1,5 \tag{7-7}$$

Wheelbays weight

The equation for the weight prediction of wheelbays for retractable undercarriages is:

$$W_{wb_i} = \varphi_{1i} + \varphi_{i2} \, n_{ult} \, W_{to}^{\varphi_{i3}} + \varphi_{i4} \, \Delta P^{\varphi_{i5}} \, W_G \qquad \text{for } i = 1,5 \tag{7-8}$$

where W_G is the fuselage gross shell structure weight:

$$W_G = W_{sk} + W_{str} + W_{fr} \tag{7-9}$$

Doors and windows weight

The weight equation for doors and windows is:

$$W_{dw_i} = \varphi_{1i} \sqrt{S_{dw_i}} + \varphi_{2i} \Delta P^{\varphi_{3i}} b_f^{\varphi_{4i}} V_D^{\varphi_{5i}} S_{dw_i} \quad \text{for } i = 1,5 \tag{7-10}$$

Eq. (7-10) is the net weight contribution, i.c. accounting for removed material, local strengthening and operating mechanism.

Wing/fuselage support structure weight

The weight equation for the wing/fuselage support structure is:

$$W_{wss} = \varphi_1 + \varphi_2 \, n_{ult}^{\varphi_3} \, W_{to}^{\varphi_4} \tag{7-11}$$

Engine support structure weight

The weight equation for the support structure of fuselage-mounted engines is:

$$W_{ess} = \varphi_1 + \varphi_2 \, W_e^{\varphi_3} + \varphi_4 \, \frac{l_e^2}{b_f} \, W_{to}^{\varphi_5} \tag{7-12}$$

where l_e is the distance between the quarter-chord point of the root chord and the engine front face.

Tail support structure weight

The weight equation for the tail support structure weight is:

$$W_{tss} = \varphi_1 + \varphi_2 \, W_t^{\varphi_3} + \varphi_4 \, W_{to}^{\varphi_5} \tag{7-13}$$

Fuel tank support structure weight

The weight equation for the centersection fuel tank support structure weight is:

$$W_{fss} = \varphi_1 + \varphi_2 \, V_{ft}^{\varphi_3} \qquad (7\text{-}14)$$

Additional items weight

The weight equation for additional items accounts for all other items not covered in the previous section, e.g. fairings, paint, redux, sealing and production joints:

$$W_{ai} = \varphi_1 + \varphi_2 \, W_G^{\varphi_3} + \varphi_4 \, S_f^{\varphi_5} \qquad (7\text{-}15)$$

7.2 Wing weight

The wing weight prediction method is derived from Ref. 124. The total wing group weight is the summation of basic wing weight, high-lift devices and spoilers/speedbrakes weights as presented in the following section:

Basic wing weight

The equation for basic wing weight is:

$$W_{w_b} = \varphi_1 \, k_{no} \, k_\lambda \left(n_{ult} \, W_{to}\right)^{\varphi_2} A^{\varphi_3} \, S^{\varphi_4} \, (t/c)_r^{\varphi_5} \, \cos^{-0.45} \Lambda_{0.5} \qquad (7\text{-}16)$$

where:

$$k_{no} = 1 + \sqrt{\frac{1.905}{b_s}} \qquad (7\text{-}17)$$

and

$$k_\lambda = (1 + \lambda)^{0.4} \qquad (7\text{-}18)$$

Eq. (7-16) is derived on the assumption that bending loads at the root are critical.

High-lift devices

The structural weight for trailing edge flap systems is:

$$W_{flp_i} = \varphi_{1i} + \varphi_{2i}\, S_{flp_i}^{\varphi_{3i}}\, b_{flp_i}^{\varphi_{4i}}\, \cos^{\varphi_{5i}} \Lambda_{flp_i} \qquad \text{for } i = 1,5 \tag{7-19}$$

where S_{flp_i} and b_{flp_i} refer to flap section i.

The structural weight of leading edge slat systems is:

$$W_{slt_i} = \varphi_{1i} + \varphi_{2i}\, W_{to}^{\varphi_{3i}} + \varphi_{4i}\, S_{slt_i}^{\varphi_{5i}} \quad \text{for } i = 1,5 \tag{7-20}$$

where S_{slt_i} refers to slat section i.

Spoilers and speed brakes

The structure weight of spoilers and speed brakes is:

$$W_{sp} = \varphi_1 + \varphi_2\, S_{sp}^{\varphi_3} + \varphi_4\, W_{w_b}^{\varphi_5} \tag{7-21}$$

The cumulate of Eqs. (7-16), (7-19), (7-20) and (7-21) yields the total wing group weight.

7.3 Empennage weight

The empennage weight prediction equations are derived from Ref. 124. The curves of Figure 8-5, Ref. 124, representing the relation between the specific tailplane weight and the parameter:

$$F = \frac{S_t^{\varphi_5} V_D}{\sqrt{\cos\Lambda_t}} \qquad (7\text{-}22)$$

are approximated by a cubic polynomial:

$$\frac{W_t}{S_t} = \varphi_1 + \varphi_2 F + \varphi_3 F^2 + \varphi_4 F^3 \qquad (7\text{-}23)$$

Basic horizontal tailplane weight

The basic weight of the horizontal tailplane structure for a given specific weight becomes:

$$W_{h_b} = \frac{W_h}{S_h} S_h \qquad (7\text{-}24)$$

Variable stabilizer structure weight

An additional weight contribution is provided for variable incidence stabilizers:

$$W_{vsw} = \varphi_1 + \varphi_2 W_{hb}^{\varphi_3} + \varphi_4 S_h^{\varphi_5} \qquad (7\text{-}25)$$

Basic vertical tailplane weight

Similar to the horizontal tailplane weight, the weight of the vertical tailplane structure is also related to the specific weight according to Eq. (7-24):

$$W_{v_b} = \frac{W_v}{S_v} S_v \qquad (7\text{-}26)$$

Optionally, for both surfaces different values may be specified for the weight technology factors in Eq. (7-23).

Fin-mounted stabilizer structure weight

A weight penalty must be applied to fin-mounted stabilizers:

$$W_{fmw} = \varphi_1 + \varphi_2 \left(\frac{S_h \Delta h}{S_v h_v}\right)^{\varphi_3} + \varphi_4 W_{v_b}^{\varphi_5} \tag{7-27}$$

7.4 Undercarriage weight

The prediction method for the undercarriage group is taken from Ref. 124. The weight per undercarriage assembly is:

$$W_{uc_i} = \varphi_{1i} + \varphi_{2i} W_{to}^{0.75} + \varphi_{3i} W_{to} + \varphi_{4i} W_{to}^{\varphi_{5i}} \quad \text{for } i = 1,5 \tag{7-28}$$

7.5 Control systems weight

Manoeuvering control system weight

$$W_{mcs} = \varphi_1 + \varphi_2 W_{to}^{\varphi_3} \tag{7-29}$$

Flap control system weight

$$W_{fcw} = \varphi_1 + \varphi_2 \left(S_{flp} \sin \delta_{flp}\right)^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \tag{7-30}$$

Slat control system weight

$$W_{scw} = \varphi_1 + \varphi_2 S_{slt}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \tag{7-31}$$

Variable incidence stabilizer control system weight

$$W_{vis} = \varphi_1 + \varphi_2 S_h^{\varphi_3} V_D^{\varphi_4} \Delta i_h^{\varphi_5} \tag{7-32}$$

where Δi is the range of incidence variation of the stabilizer.

Speed brakes control system weight

$$W_{sbc} = \varphi_1 + \varphi_2 \, S_{sb}^{\varphi_3} \, \sin^{\varphi_4} \delta_{sb} \qquad (7\text{-}33)$$

Spoilers control system weight

$$W_{spc} = \varphi_1 + \varphi_2 \, S_{sp}^{\varphi_3} \, \sin^{\varphi_4} \delta_{sp} \qquad (7\text{-}34)$$

Flight deck controls weight

$$W_{fdc} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \qquad (7\text{-}35)$$

If $W_{fdc} > \varphi_4$ then $W_{fdc} = \varphi_4$

Flight management system weight

$$W_{fms} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \qquad (7\text{-}36)$$

7.6 Propulsion group weight

Installed dry engine weight

$$W_{dry} = \varphi_1 + \varphi_2 \, W_e^{\varphi_3} \qquad (7\text{-}38)$$

where W_e is the uninstalled dry engine weight specified by the engine manufacturer.

Propellors weight

$$W_{prop_i} = \varphi_{1i} + \varphi_{2i} \, R_{prop_i}^{\varphi_{3i}} \qquad \text{for } i = 1,5 \qquad (7\text{-}37)$$

Fuel system weight

$$W_{fs} = \varphi_1 + \varphi_2 \, (N_e + N_{ft} - 1) + \varphi_3 \, N_{ft}^{\varphi_4} \, V_{ft}^{\varphi_5} \qquad (7\text{-}39)$$

where N_{ft} is the number of separate fuel tanks.

7.7 Engine nacelles and pylons weight

Engine cowlings weight

$$W_{cowl_i} = \varphi_{1i} + \varphi_{2i} S_{cowl_i}^{\varphi_{3i}} + \varphi_{4i} (S_{cowl_i} D_{cowl_i})^{\varphi_{5i}} \qquad (7\text{-}40)$$

for i = 1,10

Pylons weight

The weight prediction method for the pylons is derived from Ref. 21. This method relates the pylon weight to the bending moment at the pylon attachment points to the airframe:

$$W_{pyl} = \varphi_1 + \varphi_2 S_{pyl}^{\varphi_3} + \varphi_4 \left(W_i \frac{l_{pyl}}{h_{pyl}}\right)^{\varphi_5} \qquad (7\text{-}41)$$

where W_i is the installed engine weight. If the engine is fuselage-mounted, l_{pyl} and h_{pyl} are the pylon span and root thickness respectively. If the engine is wing-mounted, l_{pyl} is the distance between the engine/nacelle CG-position and the local quarter-chord point and h_{pyl} is the pylon height.

Extended air intake duct

$$W_{duct} = \varphi_1 + \varphi_2 l_{duct}^{\varphi_3} S_{duct}^{\varphi_4} \qquad (7\text{-}42)$$

7.8 Airframe systems and instruments weight

This section gives weight prediction methods for on-board systems for power supply, environmental control, anti-icing and instruments/avionics.

Auxilary Power Unit (APU)

Installed APU weight is dependent on the total airflow which is in turn can be related to the passenger cabin volume and/or maximum takeoff weight:

$$W_{APU} = \varphi_1 + \varphi_2 V_{pc}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \qquad (7\text{-}43)$$

Propulsion instruments weight

$$W_{pi} = \left(\varphi_1 + \varphi_2 T_{to}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5}\right) N_e \qquad (7\text{-}44)$$

Other instruments weight

$$W_{oi} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \qquad (7\text{-}45)$$

Avionics weight

$$W_{av} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \qquad (7\text{-}46)$$

Hydraulics system weight

$$W_{hs} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \qquad (7\text{-}47)$$

Pneumatics system weight

$$W_{ps} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \qquad (7\text{-}48)$$

Electrical system weight

$$W_{el} = \varphi_1 + \varphi_4 P_e \left(1 - \varphi_5 \sqrt{P_e}\right) \qquad (7\text{-}49)$$

where P_e is the electrical power required:

$$P_e = \varphi_2 V_{pc}^{\varphi_3} \qquad (7\text{-}50)$$

Airconditioning system weight

$$W_{acs} = \varphi_1 + \varphi_2\, N_{pass}^{\varphi_3} + \varphi_4\, V_{pc}^{\varphi_5} \tag{7-51}$$

Anti-icing system weight

$$W_{ais} = \varphi_1 + \varphi_2\, S_w^{\varphi_3} + \varphi_4\, W_{to}^{\varphi_5} \tag{7-52}$$

7.9 Furnishing and equipment weight

Flight deck accommodations weight

$$W_{fda} = \varphi_1 + \varphi_2\, N_{fc}^{\varphi_3} + \varphi_4\, W_{to}^{\varphi_5} \tag{7-53}$$

Passengers seats weight

$$W_{seat_i} = \varphi_{1i} + \varphi_{2i}\, N_{pass_i}^{\varphi_{3i}} \qquad \text{for } i = 1,3 \tag{7-54}$$

Galley structure and provisions weight

$$W_{gal} = \varphi_1 + \varphi_2\, N_{gal}^{\varphi_3} \tag{7-55}$$

Lavatory provisions and water system weight

$$W_{lpw} = \varphi_1 + \varphi_2\, N_{toilet}^{\varphi_3} \tag{7-56}$$

Floor covering weight

$$W_{fco} = \varphi_1 + \varphi_2\, S_{pcf}^{\varphi_3} \tag{7-57}$$

Soundproofing and insulation weight

$$W_{spi} = \varphi_1 + \varphi_2\, V_{pc}^{\varphi_3} \tag{7-58}$$

Oxygen system weight

$$W_{os} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} \qquad (7\text{-}59)$$

Fire detection and extinguishing system

$$W_{fde} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \qquad (7\text{-}60)$$

Escape provisions weight

$$W_{ep} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} + \varphi_4 \, (N_{fc} + N_{cc})^{\varphi_5} \qquad (7\text{-}61)$$

Cabin crew seats weight

$$W_{ccs} = \varphi_1 + \varphi_2 \, N_{cc}^{\varphi_3} \qquad (7\text{-}62)$$

7.10 Operational items weight

Flight crew provisions weight

$$W_{fcpr} = \varphi_1 + \varphi_2 \, N_{fc}^{\varphi_3} + \varphi_4 \, W_{to}^{\varphi_5} \qquad (7\text{-}63)$$

Cabin crew provisions weight

$$W_{ccpr} = \varphi_1 + \varphi_2 \, N_{cc}^{\varphi_3} + \varphi_4 \, W_{to}^{\varphi_5} \qquad (7\text{-}64)$$

Passenger cabin supplies weight

$$W_{pcs} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} + \varphi_4 \, W_{to}^{\varphi_5} \qquad (7\text{-}65)$$

Potable water and toilet chemicals weight

$$W_{pwtc} = \varphi_1 + \varphi_2 \, N_{toilets}^{\varphi_3} + \varphi_4 \, N_{pass}^{\varphi_5} \qquad (7\text{-}66)$$

Safety equipment weight

$$W_{se} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \qquad (7\text{-}67)$$

Residual fuel and oil weight

$$W_{rfo} = \varphi_1 + \varphi_2 V_{tk}^{\varphi_3} + \varphi_4 W_{to}^{\varphi_5} \qquad (7\text{-}68)$$

7.11 Payload weight

The payload is the sum of passengers weight and cargo/freight weight.

Passengers weight

The total passengers weight, including hand luggage, is:

$$W_{pass_i} = \varphi_{1i} + \varphi_{2i} N_{pass_i}^{\varphi_{3i}} \qquad \text{for } i = 1,3 \qquad (7\text{-}69)$$

Cargo/freight weight

The total cargo/freight weight is related to the available cargo/freight hold volume:

$$W_{car_i} = \varphi_{1i} + \varphi_{2i} V_{car_i}^{\varphi_{3i}} \qquad \text{for } i = 1,5 \qquad (7\text{-}70)$$

Thus, the total payload weight is:

$$W_{pay} = \sum_{i=1}^{i=3} W_{pass_i} + \sum_{i=1}^{i=5} W_{car_i} \qquad (7\text{-}71)$$

7.12 Fuel weight

Aircraft are usually designed for a given range with design payload (harmonic range). In this case, fuel weight and maximum takeoff weight depend on the required range:

$$W_{to} = W_{mzf} + W_{fuel} \tag{7-72}$$

However, most weight items are directly or indirectly related to W_{to}, hence Eq. (7-72) has to be solved numerically to obtain the so-called "converged" aircraft".

However, to size the aircraft for a given range is a design decision and in contradiction with the philosophy to use only methods which analyze a given design configuration. Therefore, the routines in the program library take a different approach for estimating fuel weight and range. First, maximum zero-fuel weight W_{mzf} is computed for a given W_{to} and the weight limited fuel weight follows from:

$$W_{fuel} = W_{to} - W_{mzf} \tag{7-73}$$

Evidently, a subsequent check must be made to determine if the available fuel tank capacity is sufficient. This is usually represented by the fuel volume ratio (FVR), i.c. the ratio fuel volume required/fuel tank capacity available. Thus, in general, FVR $\leq$ 1.

8. AERODYNAMIC ANALYSIS

To accurately assess aircraft performance it is imperative to avail of relationships between the principal aerodynamic coefficients versus flight condition and flap angle. The program library currently provides routines for the aerodynamic analysis of subsonic aircraft in the en-route configuration, i.e. with flaps retracted and at low angles of attack. Total lift, drag and pitching moment are computed by summation of the respective contributions of the major aircraft components. The aerodynamic analysis program D$POLH computes total lift coefficient, drag coefficient and pitching moment coefficient for a range of angles of attack and Mach numbers. Most of the methods discussed in this Chapter are derived from Refs. 77 and 124. Analoguous to the weight prediction methods, aerodynamic factors ϕ can be assigned a value by the user to adapt basic equations to a particular application.

8.1 Lifting surfaces

The methods currently implemented in the program library assume lifting surfaces to have a relatively high aspect ratio, such that root and tip effects do not become predominant. Also, the methods are only applicable for moderate sweep angles, so that cross-flow effects in the boundary layer can be neglected. For each lifting surface, the basic and additional spanwise lift distribution is computed using Diederich's method [Ref. 42]:

$$c_l(\eta) = c_{l_b}(\eta) + C_L\, c_{l_{a1}}(\eta) \tag{8-1}$$

where $c_{l_{a1}}$ is the additional lift coefficient at $C_L = 1$, which follows from:

$$c_{l_{al}} = C_1 + \frac{c_g}{c} \left(C_2 \frac{4}{\pi} \sqrt{1 - \eta^2} + C_3 F_A\right) \qquad (8\text{-}2)$$

The factors C_1, C_2 and C_3 follow from Diederich's method and are approximated by polynomials as follows:

$$\begin{aligned} C_1 = {} & 0.1454\ 10^{-2} + 0.5558\ 10^{-1}\ F_A - 0.7405\ 10^{-3}\ F_A^2 - \\ & 0.2789\ 10^{-4}\ F_A^3 + 0.1056\ 10^{-5}\ F_A^4 - 0.9738\ 10^{-8}\ F_A^5 \end{aligned} \qquad (8\text{-}3)$$

$$\begin{aligned} C_2 = {} & 1.001 - 0.1035\ F_A - 1.538\ 10^{-3}\ F_A^2 + 9.366\ 10^{-4}\ F_A^3 - \\ & 8.390\ 10^{-5}\ F_A^4 + 2.614\ 10^{-6}\ F_A^5 \qquad \text{if } F_A \leq 15 \end{aligned} \qquad (8\text{-}4)$$

$$C_2 = 0 \qquad \text{if } F_A > 15 \qquad (8\text{-}5)$$

$$C_3 = 1 - C_1 - C_2 \qquad (8\text{-}6)$$

where

$$F_A = \frac{2\ \pi\ A}{c_{l_\alpha}\ \cos\ \Lambda_{0.25}} \qquad (8\text{-}7)$$

The lift distribution function F_A accounts for non-ellipticity and depends on the corrected sweep angle:

$$\Lambda_\beta = \arctan\left(\frac{\tan\ \Lambda}{\beta}\right) \qquad \text{with } \beta = \sqrt{1 - M^2} \qquad (8\text{-}8)$$

This function is also approximated by a polynomial:

$$F_A = D_0 + D_1 \eta + D_2 \eta^2 + D_3 \eta^3 + D_4 \eta^4 + D_5 \eta^5 \qquad (8\text{-}9)$$

where the coefficients D_0, D_1, D_2, D_3, D_4 and D_5 are a function of Λ_β as follows:

$$\begin{aligned} D_0 = {} & 1.2795 - 0.6474\ \Lambda_\beta + 0.2959\ \Lambda_\beta^2 - 0.9683\ \Lambda_\beta^3 - 0.1763\ \Lambda_\beta^4 + \\ & 0.7445\ \Lambda_\beta^5 \end{aligned} \qquad (8\text{-}10)$$

$$D_1 = -0.6375 - 1.0218\,\Lambda_\beta + 3.7507\,\Lambda_\beta^2 + 5.4653\,\Lambda_\beta^3 - 2.412\,\Lambda_\beta^4 - 3.655\,\Lambda_\beta^5 \tag{8-11}$$

$$D_2 = 5.6945 + 15.406\,\Lambda_\beta - 16.656\,\Lambda_\beta^2 - 18.795\,\Lambda_\beta^3 + 9.487\,\Lambda_\beta^4 + 12.37\,\Lambda_\beta^5 \tag{8-12}$$

$$D_3 = 21.27 - 31.817\,\Lambda_\beta + 48.342\,\Lambda_\beta^2 + 21.563\,\Lambda_\beta^3 - 40.73\,\Lambda_\beta^4 - 2.882\,\Lambda_\beta^5 \tag{8-13}$$

$$D_4 = 28.849 + 29.261\,\Lambda_\beta - 76.76\,\Lambda_\beta^2 + 3.3595\,\Lambda_\beta^3 + 78.678\,\Lambda_\beta^4 - 32.06\,\Lambda_\beta^5 \tag{8-14}$$

$$D_5 = -13.91 - 11.179\,\Lambda_\beta + 39.956\,\Lambda_\beta^2 - 9.543\,\Lambda_\beta^3 - 44.053\,\Lambda_\beta^4 + 24.62\,\Lambda_\beta^5 \tag{8-15}$$

The basic lift is defined as the lift distribution in the condition where $C_L = 0$ and depends primarily on the geometric twist ε:

$$c_{l_b} = c_{l_{al}}\, c_{l_\alpha}\, \beta\, C_4 \cos \Lambda_\beta \left(\varepsilon + \alpha_{0_1} \varepsilon_t\right) \tag{8-16}$$

where α_{0_1} represents the change in the zero-lift angle of attack per unit twist:

$$\alpha_{0_1} = -\int_0^1 \frac{\varepsilon}{\varepsilon_t}\, c_{l_{al}}\, \frac{c}{c_g}\, dn \tag{8-17}$$

where c_g is the mean geometric chord and C_4 is:

$$C_4 = \frac{F_A}{F_A\sqrt{1 + \frac{36}{F_A^2}} + 6} \tag{8-18}$$

For each panel the geometric twist is assumed to vary linearly along the span.

The lift-curve slope for a lifting surface in compressible flow is predicted with the DATCOM-method [Ref. 77]:

$$C_{L_\alpha} = \frac{2\pi\beta}{\frac{2}{\beta A} + \sqrt{\frac{1}{k^2 \cos^2 \Lambda_\beta} + \left(\frac{2}{\beta A}\right)^2}} \qquad (8\text{-}19)$$

where $k = \frac{c_{l_\alpha}}{2\pi}$. Generally, $k \approx 1$.

The total lift coefficient for a given angle of attack follows from:

$$C_L = C_{L_\alpha}\left(\alpha + i - \alpha_{0_r} - \alpha_{0_1}\varepsilon_t\right) \qquad (8\text{-}20)$$

where α is the angle of attack relative to the horizontal reference plane, i is the incidence of the root section and α_{0_r} is the zero-lift angle of attack of the root airfoil.

For a given total lift coefficient, the additional lift coefficient at spanwise station η follows from:

$$c_{l_a}(\eta) = C_L\, c_{l_{a1}}(\eta) \qquad (8\text{-}21)$$

In the current application of this routine, the lift distribution is used to estimate profile drag and compressibility drag of a lifting surface, through the so-called strip theory. The current approach is developed on the basis that aerodynamic data of airfoils, if available, must be incorporated in the analysis methods to reflect airfoil technology, e.g. supercritical airfoils, natural laminar flow, etc.

In this approach, the span is divided into a number of streamwise strips, as shown in Figure 8.1:

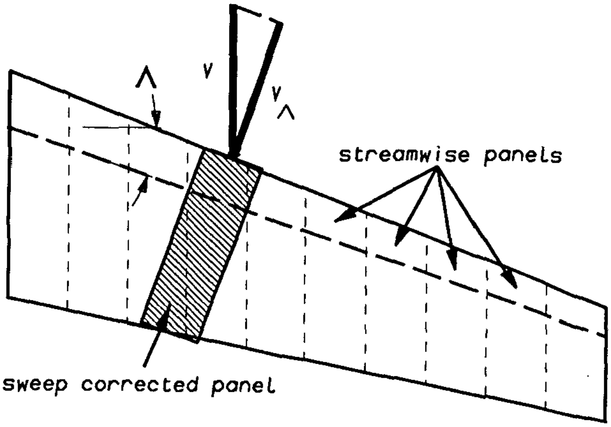


Figure 8.1: Partitioning of lifting surfaces into streamwise strips.

A correction for the wing sweep is applied according to the simple sweep theory. The free-stream velocity is decomposed into a velocity component perpendicular and parallel to the quarter-chord line. For each strip, the local c_l-value is available from Eq. (8-1).

This c_l-value is corrected for the sweep angle:

$$c_{l_\Lambda} = \frac{c_l}{\cos^2 \Lambda_{0.25}} \tag{8-22}$$

The local section drag coefficient is related to the drag coefficient of the root and tip airfoil section of the panel, derived from the corresponding airfoil data:

$$c_{d_r} = \overbrace{f_{R_r}(c_l, R_e)}^{\text{section drag}} + \overbrace{f_{M_r}(c_{l_\Lambda}, M_\Lambda)}^{\text{compressibility drag}} \cos^3 \Lambda_{0.25} \tag{8-23}$$

and

$$c_{d_t} = f_{R_t}(c_l, R_e) + f_{M_t}(c_{l_\Lambda}, M_\Lambda) \cos^3 \Lambda_{0.25} \tag{8-24}$$

where M_Λ is the Mach number perpendicular to the quarter-chord line M_Λ = Mcos $\Lambda_{.25}$ and R_e is the Reynolds number based on the strip chord. The section drag coefficient at the strip is obtained from linear interpolation on c_{d_r} and c_{d_t} according to t/c:

$$c_d = c_{d_r} + (c_{d_t} - c_{d_r}) \frac{t/c - t/c_r}{t/c_t - t/c_r} \tag{8-25}$$

where both t and c are assumed to vary linearly with η.

The total C_D of the lifting surface is obtained by summation of c_d along the span:

$$C_D S = b \int_0^1 c_d(\eta)\ c(\eta)\ d\eta \tag{8-26}$$

For a given lift distribution, the induced drag follows from (for the wing):

$$C_{D_i} = \int_0^1 \frac{c_l c}{c_g} \alpha_i(\eta)\ d\eta \tag{8-27}$$

where α_i is the local induced angle of attack. It can be computed using Prandtl's equation:

$$\alpha_i = \frac{1}{4\pi A} \int_0^1 \frac{d}{d\eta'} \left(\frac{c_l c}{c_g}\right) \frac{d\eta'}{\eta-\eta'} \tag{8-28}$$

The induced drag for a given lift coefficient follows from:

$$C_{D_i} = \frac{C_L^2}{\pi\ A\ e} \tag{8-29}$$

where e is the well-known Oswald factor, which is used to account for the non-ellipticity of the lift distribution. As in subsonic flow e is independent of α, e needs only to be computed for one angle of attack.

The pitching moment coefficient of a lifting surface is composed of a basic pitching moment and a pitching moment contribution due to lift about some reference point, e.g. the aerodynamic center of the root airfoil:

$$C_m = \frac{2}{S\bar{c}} \int_0^{b/2} \{c_{m_{ac}} c^2 + c_l c\, y \tan \Lambda_{ac}\}\, dy \tag{8-30}$$

The aerodynamic center location and the basic pitching moment of each airfoil section is known from the aerodynamic data in the airfoil library. Between 2 airfoil sections, the aerodynamic centers are assumed to be on a straight line with a sweep angle Λ_{ac}. Variation of $c_{m_{ac}}$ along the span is obtained from interpolation with Eq. (8-25).

8.1.1 The wetted area of lifting surfaces

In the case airfoil data is not available, a more simplified method can be applied. For example, the minimum profile drag drag can be derived from the flat-plate analogy (see 8.1.3) and related to the total wetted area of the lifting surface, as follows:

$$(C_D S)_s = C_F\{1 + 2.7\,\frac{t}{c} + 100\,\frac{t}{c}^4\}\,\frac{S_{wet}}{S} \tag{8-31}$$

The ADAS-routine D$LSWT in the program library automatically computes the total wetted area by summation of the wetted areas of the respective panels. The wetted area of a panel can be derived from:

$$S_{wet_p} = b_p \int_0^1 C(\eta)\, d\eta \tag{8-32}$$

where C is the circumference of the local airfoil section:

$$C = k(\eta)\, c(\eta) \tag{8-33}$$

where the proportionality factor k is:

$$k = \sum_{i=1}^{i=30} \sqrt{\left(\frac{t}{c}_{i+1} - \frac{t}{c}_{i}\right)_u^2 + \left(\frac{x}{c}_{i+1} - \frac{x}{c}_{i}\right)^2} + \sqrt{\left(\frac{t}{c}_{i+1} - \frac{t}{c}_{i}\right)_l^2 + \left(\frac{x}{c}_{i+1} - \frac{x}{c}_{i}\right)^2} \tag{8-34}$$

and c is the airfoil chord. For a straight-tapered panel, c varies linearly according to:

$$c(\eta) = c_{r_p} \left(1 + \eta(\lambda_p - 1)\right) \tag{8-35}$$

for $0 \leq \eta \leq 1$. Assuming further that k also varies linearly with η:

$$k(\eta) = k_r \left(1 + \eta(\tau_\rho - 1)\right) \tag{8-36}$$

where $\tau_p = \frac{k_t}{k_r}$, then substitution of Eqs. (8-35) and (8-36) in Eq. (8-33) and solving the integral of Eq. (8-32) yields:

$$S_{wet_p} = b_p \, C_r \, c_r \left(\frac{(\lambda_p - 1)(\tau_p - 1)}{3} + \frac{\lambda_p + \tau_p - 2}{2} + 1 \right) \tag{8-37}$$

8.1.2 The mean aerodynamic chord

Many aerodynamic and geometric coefficients are related to the mean aerodynamic chord (MAC) of the wing. The spanwise location of the MAC of a lifting surface follows from:

$$\bar{y} = \frac{1}{S} \int_0^b y \, c \, dy \tag{8-38}$$

where b and S are the span and area of the (semi-)lifting surface. The ADAS-system computes $\bar{y}$ by solving Eq. (80-38) per panel, as follows.

The distance y from the lifting surface root chord for a given panel spanwise station η_p $(0 \leq \eta \leq 1)$ is:

$$y = y_p + \eta_p b_p \tag{8-39}$$

where y_p is the spanwise station of the panel root chord. Substitution of Eqs. (8-35) and (8-39) in Eq. (8-38) yields:

$$\bar{y} = \frac{1}{S} \sum_{i=1}^{N_p} b_{p_i} c_{r_i} \left(y_{p_i} + \frac{b_{p_i} - (\lambda_{p_i} - 1)\, y_{p_i}}{2} + \frac{b_{p_i} (\lambda_{p_i} - 1)}{3} \right) \qquad (8\text{-}40)$$

The corresponding mean aerodynamic chord length follows from:

$$\bar{c} = \frac{1}{S} \int_0^1 c^2 \, dy \qquad (8\text{-}41)$$

A similar expression as Eq. (8-40) can be derived for the MAC:

$$\bar{c} = \frac{1}{S} \sum_{i=1}^{N_p} c_{r_i}^2 \, \frac{\lambda_{p_i}^2 + \lambda_{p_i} + 1}{3} \qquad (8\text{-}42)$$

8.1.3 The flat-plate friction coefficient

To compute friction drag of aircraft components, the so-called flat-plate analogy principle can be used, whereby friction drag is related to the equivalent flat-plate friction coefficient, multiplied by a form factor to account for body thickness and overspeeds:

$$C_{D_f} = F_f \, C_F \, \frac{S_{wet}}{S_{ref}} \qquad (8\text{-}43)$$

The method implemented in ADAS to estimate the flat-plate friction coefficient for mixed laminar and turbulent flow is based on the assumption that momentum thickness at the given transition point does not change significantly. Thus, the friction coefficient can be considered as a summation of friction coefficients for partly laminar and partly turbulent flow.

The friction coefficient of a flat-plate in incompressible flow with a fully turbulent boundary layer is given by the well-known Prandtl/Schlichting equation. This equation has been adapted for compressible flow by applying correction functions t and f as follows [Ref. 109]:

$$C_{F_{turb}} = 0.43\ t\ f^{2}\ \log^{-2.56} \left(R_{e_1}\ x\ t^{n+1}\ f\right) \tag{8-44}$$

where x is a characteristic length and R_{e_1} is the Reynolds number per unit length. For an adiabatic wall condition, the functions t and f can be written as:

$$t = \frac{T_\infty}{T_{aw}} = \frac{1}{1 + r\,\frac{\gamma - 1}{2}\,M^{2}} \tag{8-45}$$

and

$$f = 1 + 0.044\ r\ M^{2}\ t \tag{8-46}$$

respectively. Ref. 109 suggests for the recovery factor r = 0.89 and for the viscosity power-law exponent n = 0.67.

For the laminar portion of the flow the Blasius relation is used, also corrected for compressibility effects:

$$C_{F_{lam}} = \frac{1.328}{(1 + 0.1256\,M^{2})^{0.12}\,\sqrt{R_{e_1}\,x}} \tag{8-47}$$

The Reynolds number R_e per unit length is dependent on the surface roughness as follows:

$$R_{e_1}\ c = \min \left|\begin{array}{l} \frac{M\ a\ c}{\nu} \\ (37.587 + 4.615\,M + 2.949\,M^{2} + 4.132\,M^{3})\ \left(\frac{c}{k}\right)^{1.0489} \end{array}\right. \tag{8-48}$$

where k is the equivalent grain size and c the characteristic length.

The momentum thickness of the laminar and fully turbulent boundary layer at the given transition point T are matched:

$$C_{F_{lam}}\ x_T = C_{F_{turb}}\ \Delta x \tag{8-49}$$

where Δx is the distance ahead of the transition point of a ficticious point A from where the development of the turbulent boundary layer originates, as shown in Figure 8.2:

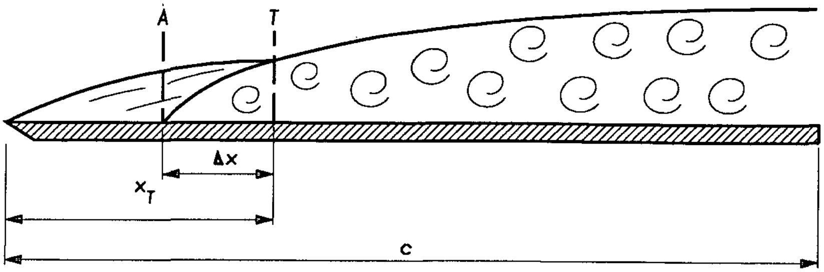


Figure 8.2: Principle of mixed laminar and turbulent flow.

As $C_{F_{turb}}$ also depends on Δx, an iterative procedure is used to solve Δx. The total flat-plate friction coefficient for a mixed laminar and turbulent flow for a given Δx follows from:

$$C_F = \frac{(c - x_T + \Delta x)\, C_{F_{turb}}}{c} \tag{8-50}$$

with $C_{F_{turb}}$ based on $c - x_T + \Delta x$.

8.2 Fuselage

The method to estimate fuselage vortex-induced drag is taken from Ref. 77. Upto a certain fuselage station x_s the flow around the fuselage is assumed to be potential flow and beyond this station the flow is predominantly vortex flow. This station of transition is a function of the location $i_{(\frac{dA}{dx})_{min}}$ where the change in cross-sectional area $\frac{dA}{dx} = \frac{A_{i+1} - A_i}{x_{i+1} - x_i}$ is most negative. Currently, the fuselage is divided into 51 equidistant station. The station of transition i_T follows from:

$$i_T = 0.5\, i_{(\frac{dA}{dx})_{min}} + 20.5 \tag{8-51}$$

The corresponding cross-sectional area is approximated by:

$$A_{i_T} = \frac{A_{i_T} + A_{i_T + 1}}{2} \tag{8-52}$$

The fuselage lift-curve slope is:

$$C_{L_{\alpha_f}} = 2\,k\,\frac{A_{i_T}}{S_{ref}}, \tag{8-53}$$

where k is the apparent mass factor developed by Munk:

$$2\,k = 0.53625 + 0.0786\,\lambda_f - 0.00481\,\lambda_f^2 + 0.0001\,\lambda_f^3, \tag{8-54}$$

where $\lambda_f = \frac{l_f}{D_f}$. For individual fuselages, $V_f^{2/3}$ is usually taken for S_{ref}.

For a given angle of attack α, the fuselage induced drag (potential flow) follows from:

$$C_{D_{i_f}} = C_{L_{\alpha_f}}\,\alpha^2 \tag{8-55}$$

The fuselage friction drag is computed from:

$$(C_D S)_{F_f} = C_F\,S_{wet_f}\,F_f \tag{8-56}$$

where C_F is based on the fuselage length. The form factor F_f follows from:

$$F_f = 1 + \frac{\varphi_1}{\lambda_{eff}^{\varphi_2}} + \frac{\varphi_3}{\lambda_{eff}^{\varphi_4}} \tag{8-57}$$

where λ_{eff} is the effective fuselage slenderness ratio:

$$\lambda_{eff} = \text{minimum} \left| \begin{array}{l} \dfrac{l_f}{D_{f_{eff}}} \\ \dfrac{l_N + l_A}{D_{f_{eff}}} + 2 \end{array} \right. \qquad \text{for } \frac{l_A}{D_{f_{eff}}} \geq 2 \tag{8-58}$$

or

$$\lambda_{eff} = \text{minimum} \left| \begin{array}{l} \dfrac{l_N + l_C}{D_{f_{eff}}} + 2 \\ \dfrac{l_N}{D_{f_{eff}}} + 4 \end{array} \right. \qquad \text{for } \frac{l_A}{D_{f_{eff}}} < 2 \tag{8-59}$$

and

$$D_{f_{eff}} = \sqrt{\frac{4}{\pi} A_c} \tag{8-60}$$

This method is only valid if the flight Mach number is below the fuselage critical Mach number and compressibility effects can be ignored.

For fuselage shapes with blunt and/or short afterbodies an additional drag contribution is due to the afterbody pressure drag increment:

$$\Delta(C_D S)_A = C_{D_{b_{ref}}} F_{D_A} \frac{\pi}{4} D_{f_{eff}}^{\,2} \tag{8-61}$$

where F_{D_A} is a function dependent on the afterbody length/diameter ratio and is derived from Figure F-11 of Ref. 124:

$$F_{D_A} = \varphi_1 + \varphi_2 \frac{l_A}{D_{f_{eff}}} + \varphi_3 \left(\frac{l_A}{D_{f_{eff}}}\right)^2 + \varphi_4 \left(\frac{l_A}{D_{f_{eff}}}\right)^3 \tag{8-62}$$

For a fuselage bare drag, the following equation is used.

$$\Delta(C_D S)_{base} = \left(\varphi_1 + \varphi_2 M^{\varphi_3}\right) S_{base} \tag{8-63}$$

Due to angle of attack and tail upsweep, cross-flow causes an additional viscous drag contribution. The procedure to estimate this effect is based on Allen's cross-flow principle and is illustrated in Figure 8.3:

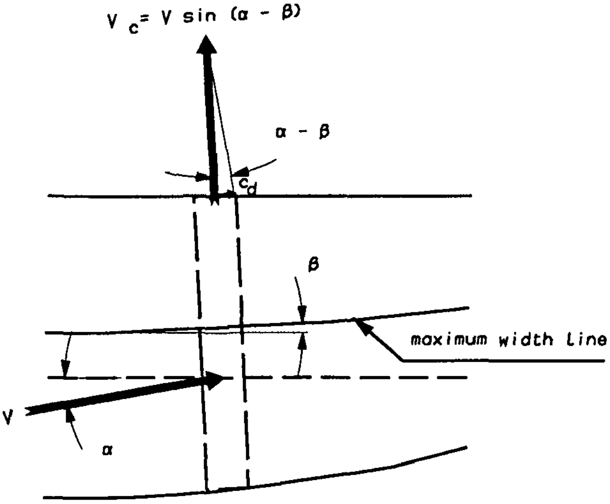


Figure 8.3: Cross-flow due to angle of attack and fuselage upsweep.

For each fuselage segment, the local velocity component perpendicular to the maximum width line is computed:

$$V_c = V \sin(\alpha - \beta) \tag{8-64}$$

The local fuselage shape is approximated by a cylindrical segment with an equivalent drag coefficient c_{d_c}. For a circular fuselage cross-section $c_{d_c} \approx 1$, while for more rectangular cross-sections $1.5 \le c_{d_c} \le 2$. The total drag coefficient follows from the summation of the drag contributions of each segment:

$$(C_D S)_{\alpha\beta} = \int_{x_0}^{l_f} |\sin' (\alpha - \beta)| \, c_{d_c} \, b_f \frac{dx}{\cos \beta} \tag{8-65}$$

The integral of Eq. (8-65) is solved numerically.

For a non-streamlined nose shape with windshield panels, an additional drag penalty has to be considered:

$$(C_D S)_{ws} = \varphi_1 + \varphi_2 \, A_{ws}^{\varphi_3} \tag{8-66}$$

where A_{ws} is the frontal windshield area.

Similar to drag, the fuselage lift is composed of a contribution due to viscous and potential flow.

$$C_{L_f} = C_{L_{\alpha_f}} \alpha + \int_{x_0}^{l_f} \sin^2 (\alpha - \beta) \; c_{d_c} \; b_f(x) \frac{dx}{\cos \beta} \tag{8-67}$$

The pitching moment slope $C_{m_{\alpha_f}}$ at low angles of attack follows from:

$$C_{m_{\alpha_f}} = \frac{2 \, k}{S_{ref} \, \bar{c}} \int_0^{x_0} \frac{dS_x}{dx} \left(x_m - x\right) dx \tag{8-68}$$

where x_m is pitching moment reference point. For individual bodies, $C_{m_{\alpha_f}}$ is generally based on the fuselage volume V_f.

8.3 Engine nacelles

The nacelle friction drag is computed separately for each cowling:

$$(C_D S)_{F_n} = \sum_{i=1}^{i=N_{cowl}} \varphi_{1i} + \varphi_{2i} \, C_{F_i} \, S_{wet_i}^{\varphi_{3i}} \tag{8-69}$$

where C_{F_i} is the flat-plate friction coefficient based on the cowling length and for a given surface roughness and flow transition location.

The afterbody pressure drag component can be written as:

$$(C_D S)A_n = \sum_{i=1}^{i=N_{cowl}} \varphi_{1i} + \varphi_{2i} D_{n_i}^{\varphi_{3i}} \qquad (8\text{-}70)$$

8.4 Interference and miscellaneous drag

In the previous Sections, methods were given to compute drag of isolated aircraft components. In the following Section, methods will be given to correct for interference effects, i.c. changes in the aerodynamic characteristics due to changes in the flow by the presence of other components.

A large contribution is represented by the wing/fuselage interference. The following methods correct for these effects.

The presence of the fuselage affects the local wing lift distribution and hence produces an increase in the induced drag:

$$\Delta C_{D_i} = \varphi_1 + \varphi_2 \frac{\eta_f}{1 + \lambda} (2 - \pi \eta_f) \frac{C_{L_o}^2}{\pi A} \qquad (8\text{-}71)$$

where η_f is the spanwise station of the structural root chord.

Change is the profile drag follows from:

$$\Delta C_{D_p} = \varphi_1 + \varphi_2 C_F t_r C_r \cos^2 \Lambda_{0.5} + \varphi_3 C_F C_L c_r D_f, \qquad (8\text{-}72)$$

where C_r is the circumference at the wing/fuselage intersection.

Additional drag penalties due to air intakes, scoops, antennas and other 'surface imperfections' have to be accounted for by adjusting the factors φ is the respective equations.

8.5 Trim drag

With respect to the induced horizontal tailplane drag, the user has the option to compute C_{D_h} for a given stabilizer setting or for C_{L_h} corresponding with the trimmed condition ($C_M = 0$), i.c.:

$$C_{L_h} = \frac{C_{m_{ac}} + C_{L_{wf}} (x_{cg} - x_{ac}) / \bar{c}}{\dfrac{S_h \, l_h \, q_h}{S \, \bar{c} \, q}} \qquad (8\text{-}73)$$

where $\frac{q_h}{q}$ is the dynamic pressure ratio at the horizontal tailplane.

8.6 Aerodynamic curves generation

Performance computations in ADAS are usually solved numerically, i.e. in a step-by-step fashion. However, this requires the aerodynamic properties of the aircraft to be known for each different flight condition. A more practical approach is to generate explicit aerodynamic relationships, i.c. to systematically evaluate the principal aerodynamic characteristics for a range of practical flight conditions and to apply interpolation for a specific flight condition.

The module D$POLH, implemented in the program library, performs this task automatically. The aerodynamic data is represented in 2-dimensional tables and are stored in the design database (see Section 4.3.3). D$POLH evaluates the aerodynamic characteristics for a specified range of α- and M-values using the analysis methods described in the previous Sections. The angles of attack and Mach numbers follow from:

$$\alpha_i = \alpha_l + \frac{(i - 1)(\alpha_u - \alpha_l)}{N_\alpha - 1} \qquad \text{for } i = 1,N_\alpha \qquad (8\text{-}74)$$

and

$$M_i = M_l + \frac{(j - 1)(M_u - M_l)}{N_M - 1} \qquad \text{for } i = 1,N_M \qquad (8\text{-}75)$$

where α_l, α_u, N_α, M_l, M_u and N_M are specified by the user. The following aerodynamic relationships are computed:

$$C_L = f(\alpha,M) \qquad (8\text{-}76)$$

$$C_D = f(\alpha,M) \qquad (8\text{-}77)$$

$$C_M = f(\alpha,M) \tag{8-78}$$

As it is more convenient to represent C_D and C_M as a function of C_L, i.e. $C_D = f(C_L,M)$ and $C_M = f(C_L,M)$, backward interpolation is performed on C_D for a range of C_L-values:

$$C_{L_i} = C_{L_1} + \frac{(i-1)(C_{L_u} - C_{L_1})}{N_{C_L} - 1} \qquad \text{for } i = 1,N_{C_L} \tag{8-79}$$

As it is not a priori known what the resulting C_L-values will be for the given range of α-values, it may be difficult to select C_{L_1} and C_{L_h} so that extrapolation is avoided. To overcome this problem, D$POLH automatically selects C_{L_1} and C_{L_h} as shown in Figure 8.4:

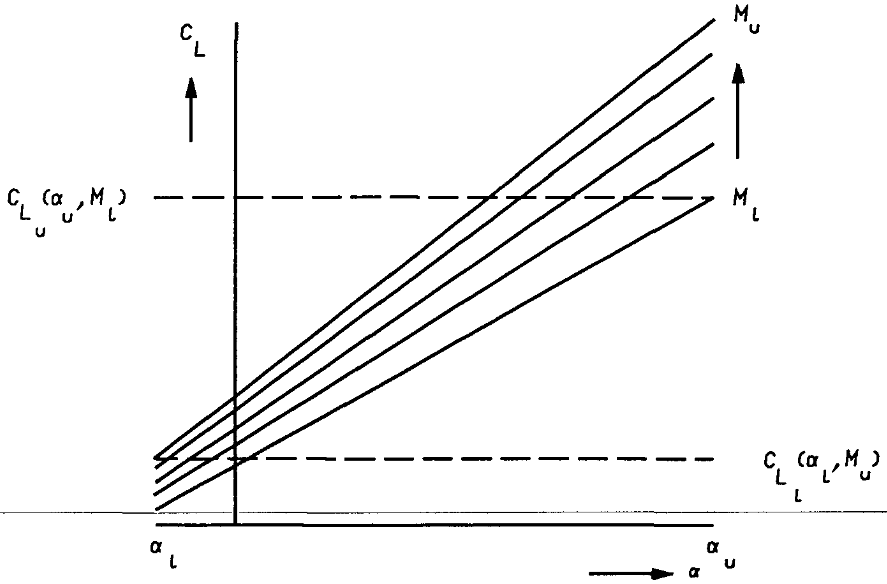


Figure 8.4: Selection of C_L-range for interpolation on C_D.

The number of C_L-values N_{C_L} can be assigned by the user.

Another routine D$POLL generates aerodynamic curves for the low-speed cases. It computes increments in C_L, C_D and C_M due to flap deflection. It subsequently adds these changes to the respective curves for the en-route configuration and for the lowest Mach number. However, the analysis methods to estimate aerodynamic characteristics for the configuration with high-lift devices deflected are not yet implemented in the program library.

The aerodynamic relationships are automatically stored in the design database (see Section 4.4.3) so they do not have to be re-computed, unless the configuration geometry is changed. In addition, the user can optionally include special print routines in the analysis program to represent these data in the form of parametric data tables in the standard data file %.DATA%. Subsequently the user can represent this data in graphical form with the standard ADAS postprocessing options (see Chapter 6).

9. ATMOSPHERIC PROPERTIES

The module D$ATMO in the program library computes ambient atmospheric properties for a given (pressure) altitude. The atmospheric model is based on the "U.S. Standard Atmosphere 1962" [Ref. 6] for $0 \leq h \leq 700$ km. The lower part $0 \leq h \leq 32$ km is adopted as the "ICAO Standard Atmosphere" [Ref. 7]. Currently, the D$ATMO-routine only covers $0 \leq h \leq 20$ km.

The following equations are used to estimate the ambient atmospheric properties at a given altitude h:

<u>For $0 \leq h \leq 11000$ m (troposphere)</u>

$$T = 288.15 - 0.0065\ h \qquad \text{(temperature in K}^{o}\text{)} \qquad (9\text{-}1)$$

$$p = 1.1915.10^{-8}\ T^{5.2559} \qquad \text{(pressure in N/m}^{2}\text{)} \qquad (9\text{-}2)$$

$$\rho = 4.1710\ 10^{-11}\ T^{4.2559} \qquad \text{(density in kg/m}^{3}\text{)} \qquad (9\text{-}3)$$

<u>For $11000 < h \leq 20000$ m (stratosphere)</u>

$$T = 216.65 \qquad (9\text{-}4)$$

$$p = 22632.0\ e^{1.7347\ -\ 1.577\ 10^{-4}\ h} \qquad (9\text{-}5)$$

$$\rho = 0.36392\ e^{1.7347\ -\ 1.577\ 10^{-4}\ h} \qquad (9\text{-}6)$$

The following additional properties are computed for all altitudes $0 \le h \le 20000$ m:

$$g = \frac{9.80665}{\left(1 + \frac{h}{6371020}\right)^2} \quad \text{(acceleration due to gravity in m/s}^2\text{)} \qquad (9\text{-}7)$$

$$a = \sqrt{401.87\ T} \quad \text{(speed of sound in m/s)} \qquad (9\text{-}8)$$

$$\nu = \frac{1.458\ 10^{-6}}{(T + 110.4)}\ \frac{T^{1.5}}{p} \quad \text{(kinematic viscosity in m}^2\text{/s)} \qquad (9\text{-}9)$$

10. RANGE PERFORMANCE

10.1 The equivalent range concept

The program module F$CRSE included in the program library, computes range for a given fuel weight, initial aircraft weight and for constant speed and altitude. Range follows from the basic equation:

$$R = \int_{W_{init} - W_{f_{cr}}}^{W_{init}} \frac{V}{F_{fuel}} \, dW \qquad (F_{fuel} = - \frac{dW}{dt}) \qquad (10\text{-}1)$$

where W_{init} is the initial aircraft weight at the start of the cruise and $W_{f_{cr}}$ is the available fuel weight for cruise. The routine F$CRSE computes range by solving Eq. (10-1) numerically using a weight step:

$$\Delta W = \frac{W_{fuel}}{N_s} \qquad (10\text{-}2)$$

where N_s is the number of steps ($N_s \leq 20$) specified by the user. Thus, the instantaneous aircraft weight at the start of segment i is:

$$W_i = W_{init} - (i - 1)\Delta W \qquad \text{for } i = 1, N_s \qquad (10\text{-}3)$$

From horizontal and vertical equilibrium of forces in quasi-stationary flight, it follows that:

$$C_L = \frac{2\,W_i}{\rho V_{cr}^2\,S} , \tag{10-4}$$

$$C_D = f(C_L, M_{cr}) , \tag{10-5}$$

$$T_{req} = \frac{C_D}{C_L} W_i , \tag{10-6}$$

$$\Gamma = \frac{T_{req}}{N_e\,T_{max}\,(M_{cr},h_{cr})} , \tag{10-7}$$

$$F_{fuel} = N_e \cdot f(\Gamma, M_{cr},h_{cr}) \tag{10-8}$$

Hence, the distance travelled during a segment is:

$$\Delta R_i = V_{cr} \frac{\Delta W}{F_{fuel}} \tag{10-9}$$

and the elapsed time is:

$$\Delta t_i = \frac{\Delta W}{F_{fuel}} = \frac{\Delta R_i}{V_{cr}} \tag{10-10}$$

Total range and endurance follow from the summation of Eqs. (10-9) and (10-10) respectively for all segments, i.e.:

$$R = \sum_{i=1}^{i=N_s} \Delta R_i \tag{10-11}$$

$$t = \sum_{i=1}^{i=N_s} \Delta t_i \tag{10-12}$$

Although speed and altitude are assumed constant in the range prediction procedure described above, the so-called step climb cruise technique can be simulated by dividing the total fuel weight in a number of portions and running F$CRSE for each fuel portion with different speed and/or altitude.

Note that R is an equivalent range, i.c. with all the specified fuel burnt in cruise condition. To obtain the actual range, either a reduction can be applied to the total available fuel weight to allow for climb, descent and reserve fuel ('lost fuel') or, alternatively, the equivalent range can be corrected for 'lost range' [Ref. 13]. The routine F$CRSE stores basic flight parameters vs. elapsed time in a buffer. A subsequent call to the routine D$PR10 will print out this information in a tabular form (Table 10.1):

Table 10.1: Flight profile analysis.

```
--------------------------------------------------------------------------------------------------------
|                                          FLIGHT PROFILE                                              |
|------------------------------------------------------------------------------------------------------|
|      |        SPEED       |  C  |    |SPEC.| ETA |       ENGINE        | TRAVELLED  | FUEL  |   AC   |
|  ALT |MACH   TAS    EAS   |   L |L/D |RANGE|x L/D| RAT  THRUST   FF    |TIME  RANGE |BURNED |  MASS  |
|------------------------------------------------------------------------------------------------------|
|   m  |      | m/s | m/s |       |    |m/kg |     |  %  |   N    |kg/hr| hrs |  km   |  kg   |   kg   |
|------------------------------------------------------------------------------------------------------|
|10058.|0.730|218.4|126.3|0.464|15.1| 376.|3.708| 79.0|28730.|2090.| 0.11|   84.|   223.| 43145.|
|10058.|0.730|218.4|126.3|0.461|15.1| 377.|3.699| 78.8|28650.|2084.| 0.21|  168.|   447.| 42921.|
|10058.|0.730|218.4|126.3|0.459|15.0| 378.|3.691| 78.5|28560.|2078.| 0.32|  253.|   670.| 42698.|
|10058.|0.730|218.4|126.3|0.457|15.0| 379.|3.683| 78.3|28480.|2072.| 0.43|  338.|   894.| 42474.|
|10058.|0.730|218.4|126.3|0.454|15.0| 381.|3.674| 78.1|28390.|2066.| 0.54|  423.|  1117.| 42251.|
|10058.|0.730|218.4|126.3|0.452|14.9| 382.|3.665| 77.8|28310.|2060.| 0.65|  508.|  1340.| 42027.|
|10058.|0.730|218.4|126.3|0.450|14.9| 383.|3.657| 77.6|28230.|2054.| 0.75|  593.|  1564.| 41804.|
|10058.|0.730|218.4|126.3|0.447|14.9| 384.|3.648| 77.4|28140.|2048.| 0.86|  679.|  1787.| 41581.|
|10058.|0.730|218.4|126.3|0.445|14.8| 385.|3.639| 77.2|28060.|2042.| 0.97|  765.|  2011.| 41357.|
|10058.|0.730|218.4|126.3|0.442|14.8| 386.|3.629| 76.9|27980.|2036.| 1.08|  852.|  2234.| 41134.|
|10058.|0.730|218.4|126.3|0.440|14.7| 387.|3.620| 76.7|27900.|2031.| 1.19|  938.|  2458.| 40910.|
|10058.|0.730|218.4|126.3|0.438|14.7| 388.|3.611| 76.5|27820.|2025.| 1.30| 1025.|  2681.| 40687.|
|10058.|0.730|218.4|126.3|0.435|14.7| 389.|3.601| 76.3|27740.|2019.| 1.41| 1112.|  2904.| 40464.|
|10058.|0.730|218.4|126.3|0.433|14.6| 391.|3.591| 76.1|27660.|2013.| 1.52| 1199.|  3128.| 40240.|
|10058.|0.730|218.4|126.3|0.430|14.6| 392.|3.582| 75.8|27580.|2008.| 1.64| 1287.|  3351.| 40017.|
|10058.|0.730|218.4|126.3|0.428|14.5| 393.|3.572| 75.6|27500.|2002.| 1.75| 1374.|  3575.| 39793.|
|10058.|0.730|218.4|126.3|0.426|14.5| 394.|3.562| 75.4|27430.|1997.| 1.86| 1462.|  3798.| 39570.|
|10058.|0.730|218.4|126.3|0.423|14.5| 395.|3.551| 75.2|27350.|1991.| 1.97| 1551.|  4021.| 39346.|
|10058.|0.730|218.4|126.3|0.421|14.4| 396.|3.541| 75.0|27270.|1986.| 2.08| 1639.|  4245.| 39123.|
|10058.|0.730|218.4|126.3|0.418|14.4| 397.|3.530| 74.8|27200.|1980.| 2.20| 1728.|  4468.| 38900.|
|10058.|0.730|218.4|126.3|0.416|14.3| 398.|3.520| 74.6|27120.|1975.| 2.31| 1817.|  4692.| 38676.|
|10058.|0.730|218.4|126.3|0.414|14.3| 399.|3.509| 74.4|27050.|1970.| 2.42| 1906.|  4915.| 38453.|
|10058.|0.730|218.4|126.3|0.411|14.3| 400.|3.498| 74.2|26980.|1964.| 2.54| 1995.|  5139.| 38229.|
|10058.|0.730|218.4|126.3|0.409|14.2| 401.|3.487| 74.0|26900.|1959.| 2.65| 2085.|  5362.| 38006.|
|10058.|0.730|218.4|126.3|0.407|14.2| 402.|3.476| 73.8|26830.|1954.| 2.77| 2175.|  5585.| 37782.|
--------------------------------------------------------------------------------------------------------
```

The flight profile table also includes parameters which represent the quality of the design with respect to cruise performance, e.g. the lift/drag ratio L/D and the range parameter $\eta \frac{L}{D}$.

10.2 Payload versus range

Range capability for different payload + fuel combinations is an important quality in aircraft design as it relates to the profit potential and operating flexibility for a particular airline network with a mixture of routes and destinations.

The routine D$PRNG in the program library computes equivalent range for different combinations of payload and fuel, considering limitations on zero-fuel weight, takeoff weight and fuel tank capacity.

The significant points considered in a payload-range diagram are shown in Figure 10.1:

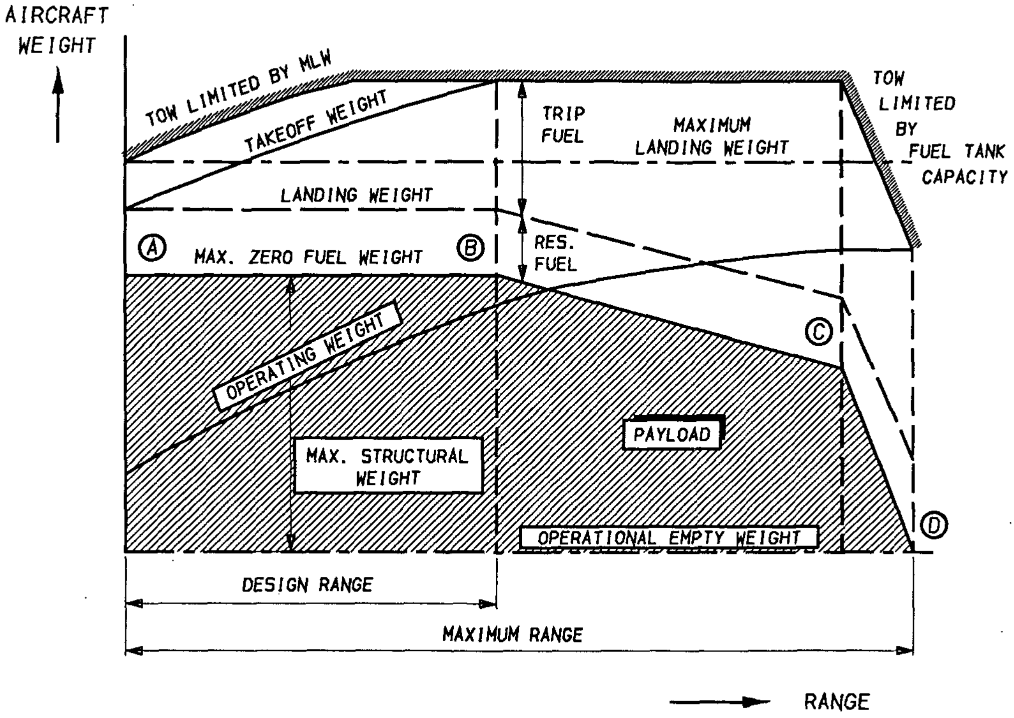


Figure 10.1: The payload - range diagram.

A → Range with design payload and reduced fuel weight

B → Range with design payload at maximum takeoff weight

C → Range with maximum fuel at maximum takeoff weight and reduced payload

D → Range with maximum fuel at reduced takeoff weight (no payload).

Figure 10.2 gives an example of a payload-range diagram computed and plotted with ADAS:

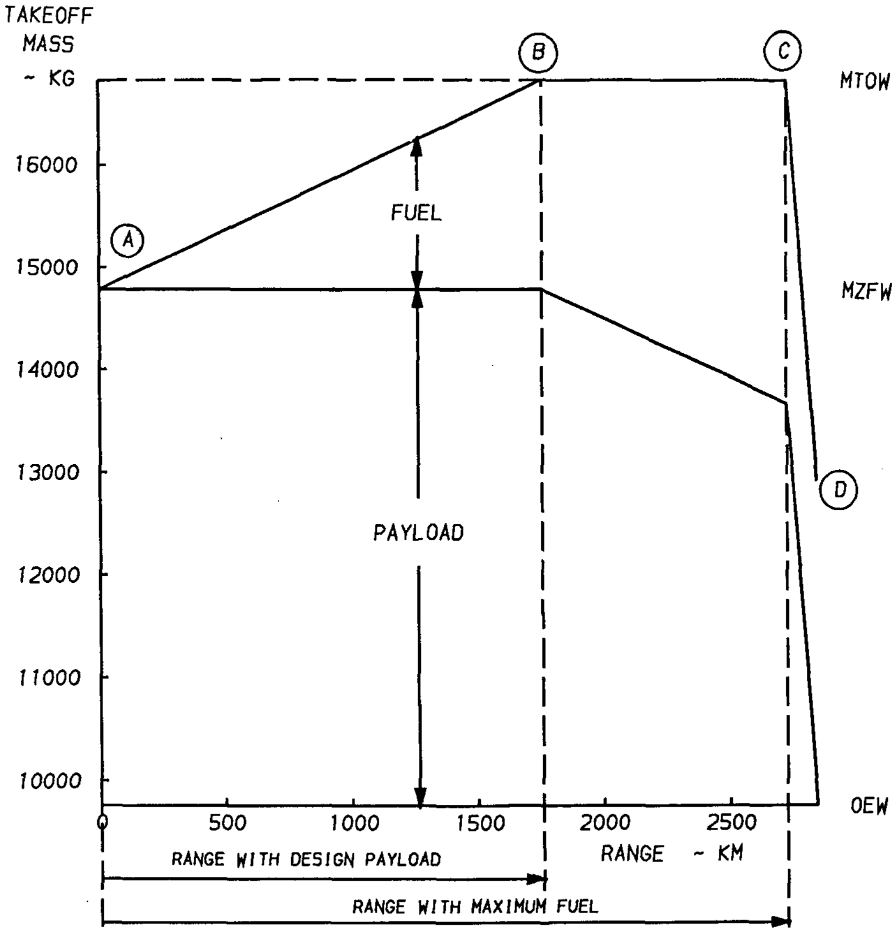


Figure 10.2: Payload-range representation with ADAS.

Upto point B, range is computed with fuel increasing from A upto W_{fuel}, where W_{fuel} is the weight-limited fuel load, i.e. $W_{fuel} = W_{to} - W_{mzf}$. Between point B and C, payload is exchanged for fuel, until either the maximum fuel tank capacity becomes limiting or no payload remains. In the latter case point C and D are equal. Between point C and D payload is further reduced with fixed maximum fuel weight and reduced takeoff weight. A weight step $\Delta W_{fuel} = \frac{W_{fuel}}{5}$ is used to account for the effect of the initial aircraft weight on range. In the case for a short-haul aircraft however, the relation is practically linear (see Figure 10.2).

11. FIELD PERFORMANCE

11.1 Takeoff performance

Takeoff performance constitutes an important requirement in many aircraft design studies. Often certain limits are posed on e.g. takeoff distance, second segment climb potential and takeoff safety speed in relation to the stalling speed. The prediction of takeoff performance is a complicated task as it involves the computation of aerodynamic characteristics with high-lift devices deflected, undercarriage extended, and the influence of ground effect has to be taken into account. In addition, the airworthiness requirements specify that with respect to the takeoff reference speeds, certain safety margins have to be considered relative to the stalling speed. Another important aspect is the takeoff performance with engine failure. In this case, an increase in drag has to be considered due to asymmetric flight and an inoperative engine. This Chapter gives a description of two methods available in the program library to assess takeoff performance.

If little design information is available, a first-order estimate for the takeoff distance can be obtained from an analytical relationship given in Ref. 124. Assuming that the takeoff with engine failure is the most critical case, the takeoff distance follows from:

$$S_{to} = \frac{0.863}{1 + 2.3\Delta\gamma_2}\left[\frac{W_{to}/S}{\rho g C_{L_2}} + h_{to}\right]\left\{\frac{1}{T/W_{to} - \mu} + 2.7\right\} + \frac{\Delta S_{to}}{\sqrt{\sigma}} \qquad (11\text{-}1)$$

where

$$\mu = 0.01\, C_{L_{max}} + 0.02 \qquad (11\text{-}2)$$

$$C_{L_2} = 0.694\, C_{L_{max}} \quad \text{for } V_2 = 1.2\, V_S \qquad (11\text{-}3)$$

and

$$\Delta\gamma_2 = \gamma_2 - \gamma_{2_{min}} \qquad (11\text{-}4)$$

Assuming a parabolic approximation for the drag polar with flaps and undercarriage extended:

$$C_D = C_0 + \frac{C_L^2}{\pi A E} \qquad (11\text{-}5)$$

the second segment climb gradient follows from:

$$\gamma_2 = \frac{N_e - 1}{N_e} \left(\frac{T}{W}\right)_{V_2} - \left(\frac{C_D}{C_L}\right)_{V_2} \qquad (11\text{-}6)$$

If the aerodynamic characteristics $C_L = f(\alpha, \delta_f)$ and $C_D = f(C_L, \delta_f)$ are known, a more elaborate method can be used whereby the takeoff performance is analyzed by solving the equations of motion in a step-by-step fashion. For the analysis of takeoff performance it is practical to divide the takeoff sequence into a number of subsequent phases, as shown in Figure 11.1:

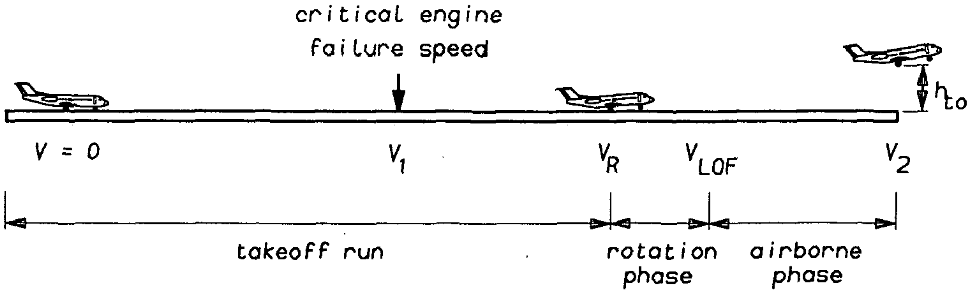


Figure 11.1: Typical phases in the takeoff sequence.

In the following Sections these phases will be discussed separately.

11.1.1 The takeoff run

For a given end speed, i.c. rotation speed V_R, the takeoff run is divided into a given number of segments N_s during which acceleration is assumed constant. The forces during the ground run are indicated in Figure 11.1:

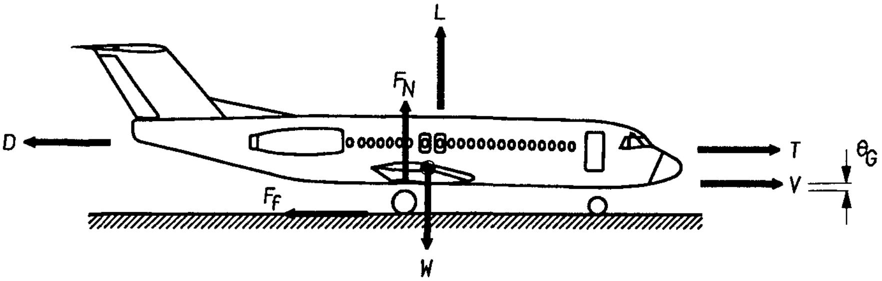


Figure 11.2: Forces during the ground run.

With a constant speed increment $\Delta V = \frac{V_R}{N_s}$, the speed at the beginning of each segment i is:

$$V_i = (i - 1)\Delta V \qquad \text{with } i = 1, N_s \tag{11-7}$$

Neglecting runway slope, wind effects and thrustvector tilt, the instantaneous forces acting on the aircraft are:

$$L = C_{L_G} \frac{1}{2} \rho V_i^2 S \qquad \text{with } C_{L_G} = f(\theta_G, \delta_f) \tag{11-8}$$

$$D = C_{D_G} \frac{1}{2} \rho V_i^2 S \qquad \text{with } C_{D_G} = f(C_{L_G}, \delta_f) \tag{11-9}$$

$$F_N = W - L - T \sin \theta_G \tag{11-10}$$

$$F_W = \mu F_N \tag{11-11}$$

and

$$T = N_e \, \Gamma_{to} \, T_{max} \, (M,h) \tag{11-12}$$

The forward acceleration is:

$$a_i = \frac{T(\cos \theta_G + \mu \sin \theta_G) - (C_{D_G} - \mu \, C_{L_G}) \, \frac{1}{2} \, \rho \, V_i^2 \, S - \mu \, W}{m} \tag{11-13}$$

where μ is the runway friction coefficient. For each segment, a_i can be computed for a given V_i and the increment in elapsed time and distance is:

$$\Delta t_i = \frac{\Delta V}{a_i} \tag{11-14}$$

and

$$\Delta S_i = V_i \Delta t_i + \frac{1}{2} \, a_i \, \Delta t_i^2 \tag{11-15}$$

repsectively. Summation of Eqs. (11-14) and (11-15) gives the total elapsed time and distance for the takeoff run:

$$t_{run} = \sum_{i=1}^{i=N_s} \Delta t_i \tag{11-16}$$

and

$$S_{run} = \sum_{i=1}^{i=N_s} \Delta S_i \tag{11-17}$$

respectively.

11.1.2 The rotation phase

In the rotation phase, the aircraft is assumed to be rotated at a constant rate of pitch $(\frac{d\theta}{dt})_R$. The system of forces during rotation is basically similar to that during the ground run (see Figure 11.3), except that for a given time step Δt, the instantaneous pitch angle follows from:

$$\theta_{R_i} = \theta_G + (\frac{d\theta}{dt})_R \, t_i \tag{11-18}$$

with:

$$t_i = (i - 1)\Delta t \tag{11-19}$$

where t is the time after rotation is initiated.

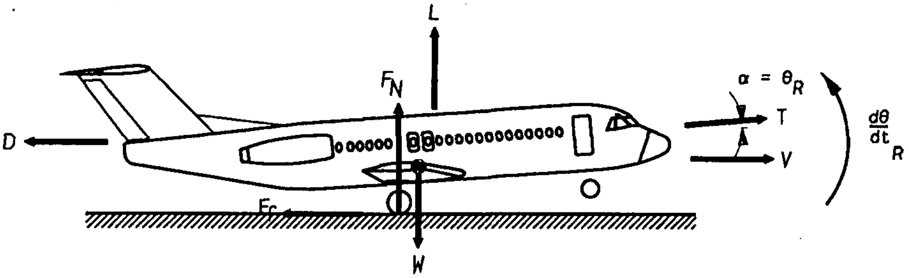


Figure 11.3: Forces during the rotation phase.

As the end speed, i.c. lift-off speed, is not known explicitly, computations are performed for a given time step Δt. The pitch angle is increased according to Eq. (11-18) until $F_N = 0$ and the aircraft becomes airborne (lift-off).

11.1.3 The airborne phase

The airborne distance is the most difficult part of the takeoff to assess, as it depends strongly on the piloting technique. The method implemented in ADAS assumes that after lift-off the aircraft is rotated further at a (different) constant rate of pitch $(\frac{d\theta}{dt})_A$, thus the pitch angle is:

$$\theta_i = \theta_{to} + (\frac{d\theta}{dt})_A \, t_i \tag{11-20}$$

with:

$$t_i = (i - 1)\,\Delta t \tag{11-21}$$

Analoguous to the rotation phase, the time step Δt must be specified by the user as the endspeed V_2 is not known in advance. The system of forces acting on the aircraft during the airborne phase is schematically illustrated in Figure 11.4:

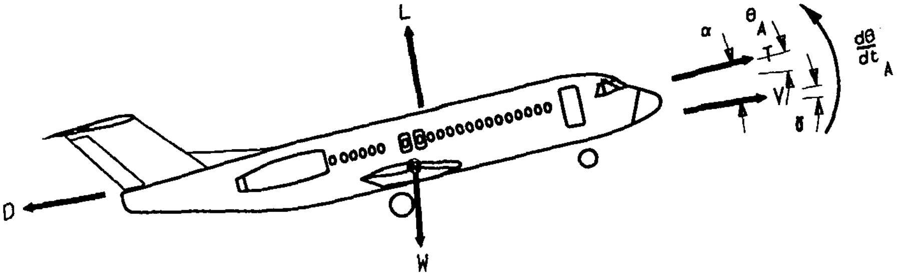


Figure 11.4: Forces during the airborne phase.

These forces will result in a horizontal and vertical acceleration:

$$a_h = \frac{T\cos\theta - D\cos\gamma - L\sin\gamma}{m} \tag{11-22}$$

and

$$a_v = \frac{L\cos\gamma + T\sin\theta - D\sin\gamma}{m} - g \tag{11-23}$$

respectively. Also, the following relations hold:

$$\alpha = \theta - \gamma \tag{11-24}$$

$$\gamma = \arctan\left(\frac{C}{V_G}\right) \tag{11-25}$$

$$V = \sqrt{C^2 + V_G^2} \tag{11-26}$$

The rate of climb C and the ground speed V_G are solved numerically by stepwise integration:

$$\Delta C_i = a_{v_i}\,\Delta t \tag{11-27}$$

and

$$\Delta V_{G_i} = a_{h_i} \Delta t \qquad (11\text{-}28)$$

The gain in altitude and horizontal distance per segment follow from:

$$\Delta h = C_i \Delta t_i + \frac{1}{2} a_{v_i} \Delta t_i^2 \qquad (11\text{-}29)$$

and

$$\Delta S_a = V_{G_i} \Delta t_i + \frac{1}{2} a_{h_i} \Delta t_i^2 \qquad (11\text{-}30)$$

respectively. The initial state parameters are taken from the final condition after rotation (lift-off), i.c. $\theta = \theta_{to}$, $\gamma = 0$ and $C = 0$. The pitch angle is increased according to Eq. (11-20) until a specified climb angle is attained, after which the pitch angle is held constant. The takeoff climb proceeds until the screen height h_{to} is passed.

11.1.4 The stop distance

The mechanics of motion during decelleration, e.g. after engine failure, is basically similar to that during the ground run, except for the following:

- Thrust reversal at a given engine rating is modelled by applying a factor as follows:

$$T_{stop} = f_{rev} T \qquad (11\text{-}31)$$

  where $f_{rev} \leq 0$. In the case of engine failure, all engines are usually shut down, thus $f_{rev} = 0$ is an reasonable assumption.

- Deflection of spoilers and speedbrakes cause a reduction in lift and an increase in drag. In addition, the increase in weight on the wheels will improve the braking effectiveness. These effects are modelled by specifying correction factors as follows:

$$C_{L_{stop}} = f_{C_L} C_{L_G} \qquad (11\text{-}32)$$

and

$$C_{D_{stop}} = f_{C_D} C_{D_G} \qquad (11\text{-}33)$$

where f_{C_L} and f_{C_D} represent a percentual reduction in lift and an increase in drag respectively.

The program library contains separate routines for each takeoff phase. The takeoff performance can be assessed by referencing these modules in the proper sequence. ADAS automatically stores the primary takeoff data in a buffer. A subsequent call to a special routine (D$PR11) will produce a print-out this information in the standard data file %.DATA%. Table 11.1 gives an example of a print-out for a continued takeoff after engine failure.

11.1.5 The Balanced Field Length

In order to determine the takeoff distance after engine failure, the corresponding engine failure speed needs to be known. The decision speed (V_1) is defined as the speed during the ground run after which, in the case of engine failure, the takeoff must be continued and before which the takeoff must be aborted. Therefore, the decision speed V_1 corresponds with the condition at which:

$$S^{N-1}_{continued} = S^{N-1}_{aborted} \qquad (11\text{-}34)$$

The takeoff distance is referred to as the Balanced Field Length (BFL). V_1 may be determined by computing $S^{N-1}_{continued}$ and $S^{N-1}_{aborted}$ for a range of decision speeds, i.e. $0.5\, V_R \leq V_1 \leq V_R$. Figure 11.1 gives a typical result of this procedure, using the methods discussed previously.

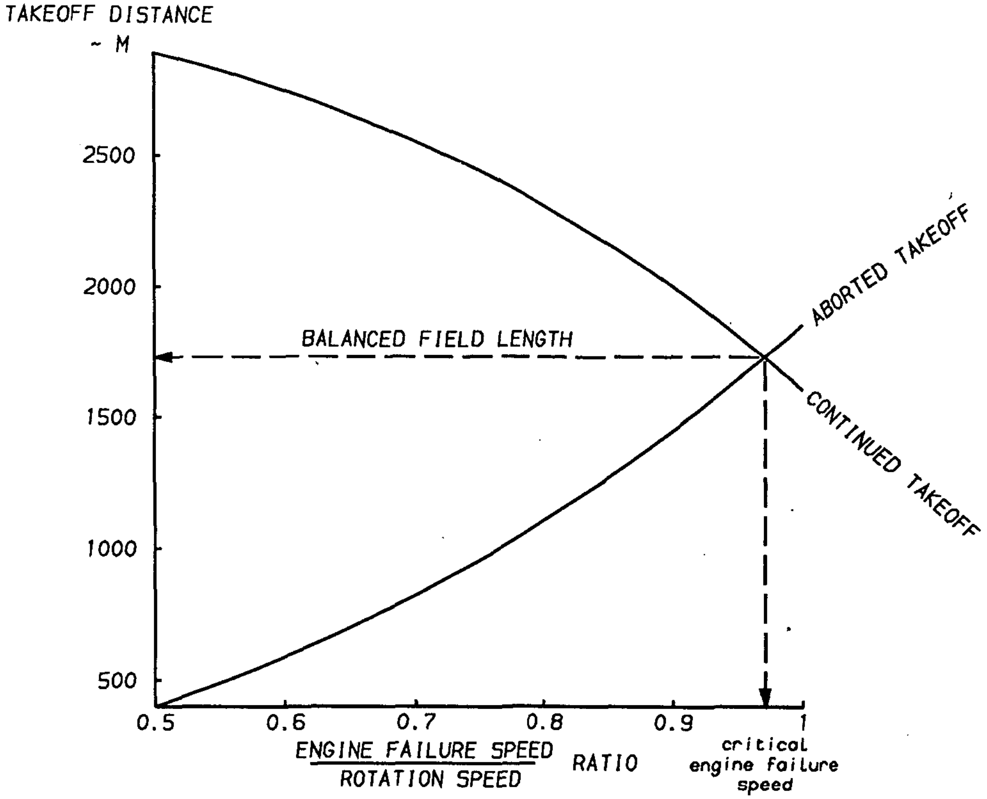


Figure 11.6: Graphical assessment of the Balanced Field Length.

The intersection of the two curves represents the decision speed.

11.2 Landing performance

Compared to the the takeoff distance, a more simplified method is used to determine the landing distance. Except for the landing flare, the landing procedure is generally less complicated as engine failure and asymmetric flight does not have to be considered. Moreover, in many design cases, the takeoff distance proves to be the most critical performance item. Nevertheless, the landing distance, which is closely related to the minimum speed requirement, must be estimated.

Table 11.1: Procedure for continued takeoff after engine failure.

TAKEOFF PERFORMANCE

ALT	SPEED	CLIMB		PITCH	C		ENGINE		ELAPSED		AC	FLAP	UC	EF
		SPEED	ANGLE	ANGLE	L	ER	THRUST	FUEL	TIME	DIST	MASS			
m	m/s	m/s	degr	degr		%	N	kg/hr	sec	m	kg	degr		
TAKEOFF RUN ALL ENGINES OPERATIVE														
0.0	0.0	0.0	0.0	-0.9	0.775	100.0	126900	5468.	0.0	0.	43547.	10.	1	0
0.0	3.6	0.0	0.0	-0.9	0.775	100.0	125740	5476.	1.3	2.	43545.	10.	1	0
0.0	7.1	0.0	0.0	-0.9	0.775	100.0	124610	5484.	2.7	10.	43543.	10.	1	0
0.0	10.7	0.0	0.0	-0.9	0.775	100.0	123510	5494.	4.1	22.	43541.	10.	1	0
0.0	14.3	0.0	0.0	-0.9	0.775	100.0	122460	5506.	5.5	39.	43539.	10.	1	0
0.0	17.8	0.0	0.0	-0.9	0.775	100.0	121440	5518.	6.9	62.	43537.	10.	1	0
0.0	21.4	0.0	0.0	-0.9	0.775	100.0	120450	5531.	8.3	89.	43534.	10.	1	0
0.0	25.0	0.0	0.0	-0.9	0.775	100.0	119510	5546.	9.7	123.	43532.	10.	1	0
0.0	28.5	0.0	0.0	-0.9	0.775	100.0	118590	5561.	11.2	162.	43530.	10.	1	0
0.0	32.1	0.0	0.0	-0.9	0.775	100.0	117720	5578.	12.6	206.	43528.	10.	1	0
0.0	35.7	0.0	0.0	-0.9	0.775	100.0	116880	5596.	14.1	257.	43525.	10.	1	0
0.0	39.3	0.0	0.0	-0.9	0.775	100.0	116080	5615.	15.7	314.	43523.	10.	1	0
0.0	42.8	0.0	0.0	-0.9	0.775	100.0	115320	5635.	17.2	377.	43521.	10.	1	0
0.0	46.4	0.0	0.0	-0.9	0.775	100.0	114590	5657.	18.8	447.	43518.	10.	1	0
0.0	50.0	0.0	0.0	-0.9	0.775	100.0	113900	5679.	20.4	524.	43516.	10.	1	0
0.0	53.5	0.0	0.0	-0.9	0.775	100.0	113250	5703.	22.0	608.	43513.	10.	1	0
0.0	57.1	0.0	0.0	-0.9	0.775	100.0	112630	5727.	23.6	699.	43510.	10.	1	0
0.0	60.7	0.0	0.0	-0.9	0.775	100.0	112040	5753.	25.3	798.	43508.	10.	1	0
0.0	64.2	0.0	0.0	-0.9	0.775	100.0	111480	5780.	27.0	905.	43505.	10.	1	0
0.0	67.8	0.0	0.0	-0.9	0.775	100.0	110950	5808.	28.8	1021.	43502.	10.	1	0
TAKEOFF RUN ONE ENGINE OUT														
0.0	67.8	0.0	0.0	-0.9	0.775	100.0	55470	2904.	28.8	1021.	43502.	10.	1	1
0.0	67.9	0.0	0.0	-0.9	0.775	100.0	55460	2905.	29.0	1035.	43502.	10.	1	1
0.0	68.1	0.0	0.0	-0.9	0.775	100.0	55450	2905.	29.2	1049.	43502.	10.	1	1
0.0	68.2	0.0	0.0	-0.9	0.775	100.0	55440	2906.	29.4	1063.	43502.	10.	1	1
0.0	68.4	0.0	0.0	-0.9	0.775	100.0	55430	2906.	29.6	1077.	43501.	10.	1	1
0.0	68.5	0.0	0.0	-0.9	0.775	100.0	55420	2907.	29.8	1091.	43501.	10.	1	1
0.0	68.7	0.0	0.0	-0.9	0.775	100.0	55410	2907.	30.0	1105.	43501.	10.	1	1
0.0	68.8	0.0	0.0	-0.9	0.775	100.0	55400	2908.	30.2	1119.	43501.	10.	1	1
0.0	68.9	0.0	0.0	-0.9	0.775	100.0	55390	2909.	30.4	1133.	43501.	10.	1	1
0.0	69.1	0.0	0.0	-0.9	0.775	100.0	55380	2909.	30.6	1147.	43501.	10.	1	1
0.0	69.2	0.0	0.0	-0.9	0.775	100.0	55370	2910.	30.8	1162.	43500.	10.	1	1
0.0	69.4	0.0	0.0	-0.9	0.775	100.0	55360	2910.	31.0	1176.	43500.	10.	1	1
0.0	69.5	0.0	0.0	-0.9	0.775	100.0	55350	2911.	31.2	1191.	43500.	10.	1	1
0.0	69.6	0.0	0.0	-0.9	0.775	100.0	55340	2911.	31.5	1205.	43500.	10.	1	1
0.0	69.8	0.0	0.0	-0.9	0.775	100.0	55330	2912.	31.7	1220.	43500.	10.	1	1
0.0	69.9	0.0	0.0	-0.9	0.775	100.0	55320	2913.	31.9	1235.	43500.	10.	1	1
0.0	70.1	0.0	0.0	-0.9	0.775	100.0	55310	2913.	32.1	1249.	43499.	10.	1	1
0.0	70.2	0.0	0.0	-0.9	0.775	100.0	55300	2914.	32.3	1264.	43499.	10.	1	1
0.0	70.4	0.0	0.0	-0.9	0.775	100.0	55290	2914.	32.5	1279.	43499.	10.	1	1
0.0	70.5	0.0	0.0	-0.9	0.775	100.0	55280	2915.	32.7	1294.	43499.	10.	1	1
ROTATION PHASE														
0.0	70.5	0.0	0.0	-0.9	0.711	100.0	55280	2915.	32.7	1294.	43499.	10.	1	1
0.0	70.6	0.0	0.0	0.0	0.799	100.0	55280	2915.	32.9	1308.	43499.	10.	1	1
0.0	70.8	0.0	0.0	0.9	0.885	100.0	55270	2916.	33.1	1322.	43499.	10.	1	1
0.0	70.9	0.0	0.0	1.9	0.971	100.0	55260	2917.	33.3	1337.	43498.	10.	1	1
0.0	71.0	0.0	0.0	2.8	1.056	100.0	55250	2917.	33.5	1351.	43498.	10.	1	1
0.0	71.2	0.0	0.0	3.7	1.140	100.0	55240	2918.	33.7	1365.	43498.	10.	1	1
0.0	71.3	0.0	0.0	4.6	1.223	100.0	55230	2918.	33.9	1379.	43498.	10.	1	1
0.0	71.4	0.0	0.0	5.5	1.306	100.0	55220	2919.	34.1	1393.	43498.	10.	1	1
0.0	71.5	0.0	0.0	6.4	1.389	100.0	55220	2919.	34.3	1408.	43498.	10.	1	1
0.0	71.6	0.0	0.0	7.4	1.470	100.0	55210	2920.	34.5	1422.	43497.	10.	1	1
AIRBORNE PHASE														
0.0	71.6	0.0	0.0	7.4	1.476	100.0	55210	2920.	34.5	1422.	43497.	10.	1	1
0.1	71.9	0.2	0.2	8.1	1.528	100.0	55190	2921.	35.2	1472.	43497.	10.	1	1
0.4	72.1	0.7	0.6	8.1	1.488	100.0	55170	2922.	35.9	1523.	43496.	10.	1	1
1.1	72.3	1.1	0.9	8.1	1.459	100.0	55150	2922.	36.6	1573.	43496.	10.	1	1
2.0	72.5	1.4	1.1	8.1	1.438	100.0	55140	2923.	37.3	1624.	43495.	10.	1	1
3.0	72.7	1.6	1.3	8.1	1.422	100.0	55120	2923.	38.0	1675.	43495.	10.	1	1
4.2	72.8	1.8	1.4	8.1	1.410	100.0	55100	2923.	38.7	1726.	43494.	10.	1	1
5.5	72.9	1.9	1.5	8.1	1.401	100.0	55090	2923.	39.4	1777.	43493.	10.	1	1
6.9	73.1	2.0	1.6	8.1	1.393	100.0	55070	2923.	40.1	1828.	43493.	10.	1	1
8.4	73.2	2.1	1.7	8.1	1.387	100.0	55050	2923.	40.8	1879.	43492.	10.	1	1
9.9	73.3	2.2	1.7	8.1	1.381	100.0	55040	2923.	41.5	1930.	43492.	10.	1	1
11.5	73.4	2.3	1.8	8.1	1.377	100.0	55020	2923.	42.2	1981.	43491.	10.	1	1

The method to assess landing performance, currently implemented in the program library, has been derived from Ref. 124, which gives an analytical expression for the landing distance:

$$\frac{S_{land}}{h_{land}} = \frac{1}{\gamma} + 1.69\,\frac{W_{land}/S}{h_{land}\rho g C_{L_{max}}}\left(\frac{1}{a/g}\left(1 - \frac{\gamma^2}{\Delta n}\right) + \frac{\gamma}{\Delta n}\right) \qquad (11\text{-}36)$$

where a/g is the average decelleration, taking into account braking effectiveness, time delay for spoilers and speedbrakes deployment, etc. Δn is the load factor increment during the flare-out. The approach speed is assumed to be 1.3 V_S.

12. OPERATIONAL CLIMB PERFORMANCE

The program library contains routines to assess climb performance. Routines are available to estimate point climb performance, e.g. maximum rate of climb for a given altitude and engine rating, or to assess a climb procedure between two specified altitudes.

12.1 Angle of attack and climb angle for quasi-stationary climb

The forces acting on the aircraft in quasi-stationary climb are illustrated in Figure 12.1:

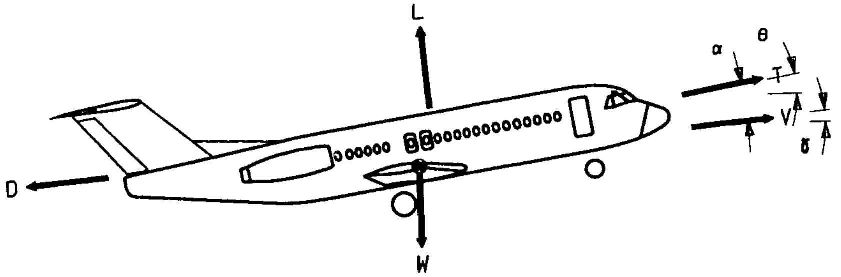


Figure 12.1: Forces during quasi-stationary climb.

For a given altitude h, speed V and thrust T, the angle of attack α and climb angle γ can be solved from the equations for equilibrium of forces along and normal to the flight path in quasi-stationary flight, i.c.:

$$T \cos \alpha - D - W \sin \gamma = 0 \qquad (12\text{-}1)$$

and

$$T \sin \alpha + L - W \cos \gamma = 0 \qquad (12\text{-}2)$$

respectively.

The routine F$GAMA solves this system of equations numerically, using an iterative procedure as follows:

$$\alpha = \alpha_i \text{ (with } \alpha_i = 0 \text{ as a first estimate)}, \qquad (12\text{-}3)$$

$$L = C_L \frac{1}{2} \rho V^2 S \qquad \text{with } C_L = f(\alpha, M), \qquad (12\text{-}4)$$

$$D = C_D \frac{1}{2} \rho V^2 S \qquad \text{with } C_D = f(C_L, M), \qquad (12\text{-}5)$$

$$\gamma = \arcsin \left(\frac{T \cos \alpha - D}{W}\right), \qquad (12\text{-}6)$$

and

$$\alpha_i = \arcsin \left(\frac{L + W \cos \gamma}{T}\right) \qquad (12\text{-}7)$$

This process returns to Eq. (12-3) and repeats until $|\alpha - \alpha_i| < 0.001$ rad.

To obtain maximum rate of climb at a given altitude and engine rating, a numerical procedure is used to optimize rate of climb C:

$$C = V \sin \gamma \qquad (12\text{-}8)$$

by varying speed V and by solving the corresponding angle of attack for quasi-stationary climb with the procedure outlined above. Evaluating the rate of climb at different speeds and altitudes gives the maximum rate of climb at a given altitude (Figure 12.1):

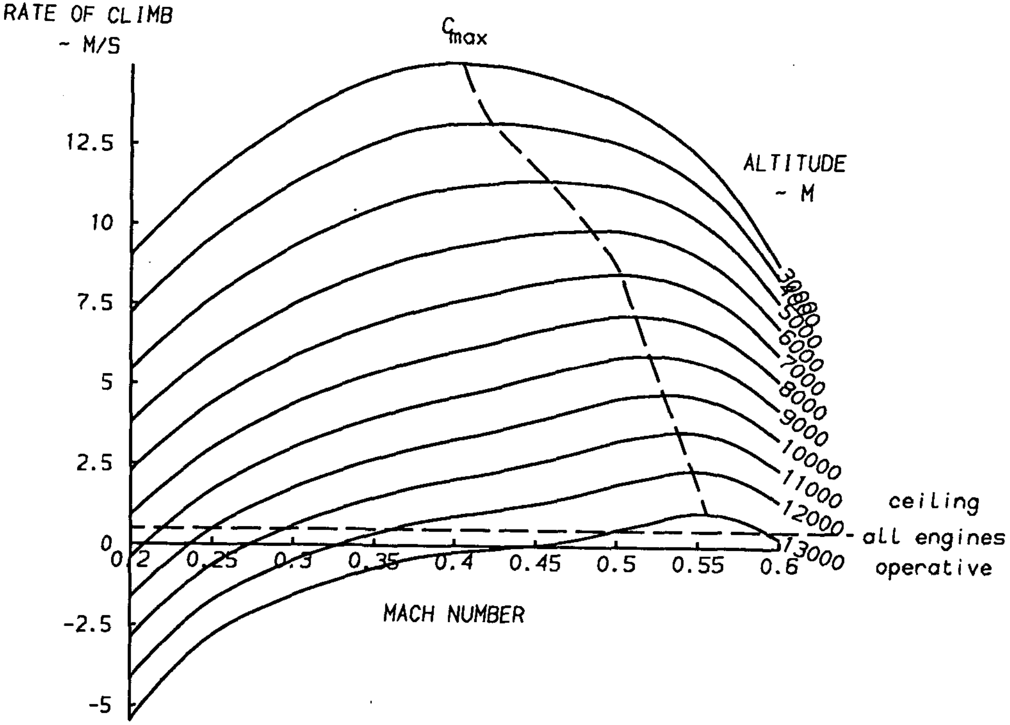


Figure 12.1: Maximum rate of climb and ceiling for all engines operative.

From this Figure, the maximum, thrust-limited altitude (ceiling) can be derived.

12.2 Operational climb procedure

The routine D$CLMB utilizes F$GAMA to compute a climb procedure from and to a given altitude, h_b and h_e respectively, for a constant engine rating r_{cl}.

The climb procedure is solved in a step-by-step fashion. The step size follows from dividing the altitude difference into a specified number of segments N_s, i.e.:

$$\Delta h = \frac{h_e - h_b}{N_s} \qquad \text{with } N_s \leq 20 \qquad (12\text{-}9)$$

Hence, the altitude h_i at the start of each segment i is:

$$h_i = h_b + \Delta h(i - 1) \qquad \text{with } i = 1, N_s \tag{12-10}$$

For each segment, a stationary climb is assumed and the equations of motion given in Section 12.1 are solved.

The initial climb is performed at a constant, specified equivalent airspeed. The true airspeed follows from:

$$V = \sqrt{\frac{\rho_0}{\rho}}\, V_{EAS} \tag{12-11}$$

The Mach number increases with increasing altitude. If at some altitude a specified Mach number limitation is exceeded, the climb continues at a constant Mach number. Table 12.1 gives an example of a print-out for a typical climb procedure:

Table 12.1: Climb performance output data.

```
-------------------------------------------------------------------------------------
|                                  CLIMB PERFORMANCE                                |
|-----------------------------------------------------------------------------------|
|      |            SPEED      |  CLIMB       | C    |      ENGINE      | TRAVELLED   | FUEL    |   AC   |
|  ALT |MACH    TAS     EAS    |SPEED ANGLE|  L    | RAT  THRUST  FF  |TIME  RANGE |BURNED |  MASS  |
|-----------------------------------------------------------------------------------|
|   m  |      | m/s | m/s | m/s |degr |      |  %  |   N   |kg/hr| min |  km  |  kg   |   kg   |
|-----------------------------------------------------------------------------------|
|    0. |0.441|150.0|150.0| 19.0|  7.3|0.290| 95.0| 35880|1996.|  0.5|   4.1|    15.| 16810.|
|  526. |0.455|153.9|150.0| 18.2|  6.8|0.290| 95.0| 34420|1932.|  0.9|   8.6|    31.| 16794.|
| 1053. |0.470|157.9|150.0| 17.4|  6.3|0.290| 95.0| 32930|1868.|  1.4|  13.3|    47.| 16778.|
| 1579. |0.485|162.0|150.0| 16.5|  5.8|0.290| 95.0| 31390|1805.|  2.0|  18.4|    63.| 16762.|
| 2105. |0.501|166.4|150.0| 15.4|  5.3|0.290| 95.0| 29760|1743.|  2.5|  24.1|    79.| 16746.|
| 2632. |0.518|170.9|150.0| 14.3|  4.8|0.290| 95.0| 28110|1682.|  3.2|  30.4|    96.| 16729.|
| 3158. |0.535|175.5|150.0| 13.1|  4.3|0.290| 95.0| 26580|1620.|  3.8|  37.4|   114.| 16711.|
| 3684. |0.554|180.4|150.0| 11.8|  3.8|0.290| 95.0| 25140|1556.|  4.6|  45.4|   134.| 16691.|
| 4211. |0.573|185.5|150.0| 10.4|  3.2|0.289| 95.0| 23750|1491.|  5.4|  54.8|   155.| 16671.|
| 4737. |0.593|190.8|150.0|  8.7|  2.6|0.289| 95.0| 22380|1425.|  6.4|  66.2|   178.| 16647.|
| 5263. |0.600|191.7|146.5|  7.8|  2.3|0.303| 95.0| 21130|1351.|  7.5|  79.1|   204.| 16621.|
| 5789. |0.600|190.4|141.3|  7.4|  2.2|0.325| 95.0| 19960|1275.|  8.7|  92.6|   229.| 16596.|
| 6316. |0.600|189.1|136.3|  7.0|  2.1|0.348| 95.0| 18880|1203.| 10.0| 106.9|   254.| 16571.|
| 6842. |0.600|187.8|131.4|  6.6|  2.0|0.374| 95.0| 17840|1134.| 11.3| 121.9|   279.| 16546.|
| 7368. |0.600|186.4|126.6|  6.1|  1.9|0.402| 95.0| 16850|1068.| 12.8| 137.9|   305.| 16520.|
| 7895. |0.600|185.1|121.9|  5.7|  1.8|0.433| 95.0| 15910|1003.| 14.3| 155.1|   331.| 16494.|
| 8421. |0.600|183.8|117.4|  5.2|  1.6|0.467| 95.0| 15020| 941.| 16.0| 173.7|   357.| 16468.|
| 8947. |0.600|182.4|112.9|  4.7|  1.5|0.504| 95.0| 14170| 881.| 17.8| 193.9|   384.| 16441.|
| 9474. |0.600|181.0|108.5|  4.3|  1.3|0.544| 95.0| 13370| 824.| 19.9| 216.3|   413.| 16413.|
|10000. |0.600|179.7|104.3|  3.8|  1.2|0.588| 95.0| 12610| 768.| 22.2| 241.3|   442.| 16383.|
-------------------------------------------------------------------------------------
```

From Table 12.1 the total climb time, range and fuel burn can be derived.

12.3 Maximum speed in horizontal flight

From the forces of equilibrium, a numerical procedure (F$VMAX) is used to obtain the maximum speed for a given altitude and engine rating Γ, as follows:

$$V_i = M_i\ a, \qquad \text{(with } M_i = 0.4 \text{ as a first estimate)} \tag{12-12}$$

$$V = V_i, \tag{12-13}$$

$$C_L = \frac{m\ g}{\frac{1}{2}\ \rho\ V^2\ S}, \tag{12-14}$$

$$C_D = f(C_L, M), \tag{12-15}$$

$$T = N_e\ \Gamma\ T_{max}(M,h), \tag{12-16}$$

$$V_i = \sqrt{\frac{T}{C_D\ \frac{1}{2}\ \rho\ S}} \tag{12-17}$$

This procedure repeats at Eq. 12-13 until $|V - V_i| \le 0.01$ m/s.

In the case a minimum requirement is imposed on V_{max}, it is more convenient to solve the corresponding engine rating Γ_{max} directly from:

$$\Gamma_{max} = \frac{C_D\ \frac{1}{2}\ \rho\ V_{max}^2\ S}{T_{max}} \tag{12-18}$$

and verify that $\Gamma_{max} \le 1$.

13. LOAD AND BALANCE ANALYSIS

Load and balance considerations are important with regard to e.g. wing positioning and tailplane sizing. A loading diagram may be a useful tool in this respect. It gives the CG-location versus different loading combinations of passengers, cargo and fuel. From this diagram, the most forward and rearward CG-location to be expected in operational service can be depicted. It can also show the necessity for certain loading restrictions and regulations.

In ADAS there are 2 routines available in the program library to generate a loading diagram. D$CGSH computes the CG-shift for a given CG-location at W_{oe} and D$PL09 subsequently plots the diagram.

13.1 Center of gravity at Operational Empty Weight

The CG-location of the complete configuration at W_{oe} is computed concurrent with the computation of the weight items (cf. Chapter 7). Each weight item is associated with a CG-position relative to the geometric axis system. The resultant CG-location follows from:

$$X_{cg} = \frac{1}{W_{oe}} \sum X_{cg_i} W_i \tag{13-1}$$

$$Z_{cg} = \frac{1}{W_{oe}} \sum Z_{cg_i} W_i \tag{13-2}$$

The D$CGSH-routine computes the CG-location for different loading conditions of payload and fuel (useful load).

13.2 Center of gravity travel

13.2.1 Passengers

The position of passengers in the passenger cabin is derived from the seat locations defined in the MEDUSA drawing. Seats are first sorted into rows, i.c. seats at equal fuselage station, and the number of seats abreast for each row. As it is unpractical to determine the CG-location for every possible passenger seating arrangement, some schematisation is applied in the form of the 'window seating rule'. Passengers are assumed to occupy the window seats first, then the adjacent seat, etc. Seating is considered from the front to the rear and vice-versa, which results in the typical closed curves, as shown in Figure 13.1:

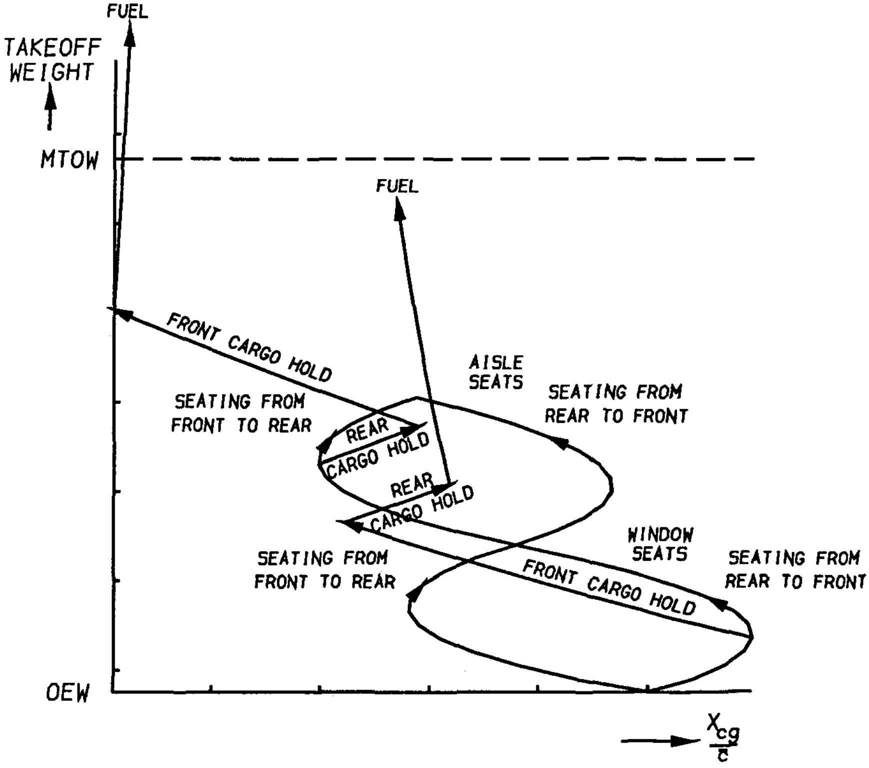


Figure 13.1: The general load and balance diagram.

The CG-location of the complete configuration can be computed from:

$$X_{cg_{i+1}} = \frac{X_{cg_i} W_i + X_{cg_p} W_p}{W_{i+1}} \tag{13-3}$$

where

$$W_{i+1} = W_i + W_p \tag{13-4}$$

W_p is the weight per passenger, including hand luggage. W_p may be different for each cabin class.

13.2.2 Cargo

Cargo is assumed to be located in the fuselage compartments, defined in the MEDUSA drawing, and designated for cargo storage (cargo holds). The cargo center of gravity $X_{cg_{cargo}}$ is assumed to be independent of the cargo weight. The total cargo weight is divided into 5 portions and the resultant center of gravity due to cargo follows from:

$$X_{cg_{i+1}} = \frac{X_{cg_i} W_i + X_{cg_{cargo}} \Delta W_{cargo}}{W_{i+1}} \tag{13-5}$$

where

$$W_{i+1} = W_i + \Delta W_{cargo} \tag{13-6}$$

and

$$\Delta W_{cargo} = \frac{W_{cargo}}{5} \tag{13-7}$$

This procedure is repeated for cargo loading from the front to the rear cargo hold and vice-versa.

13.2.3 Fuel

The fuel center of gravity is computed concurrent with the estimation of the fuel tank volume. An accurate prediction of the available fuel tank capacity is essential as it must be shown that all required fuel can be contained in the available fuel tanks. Usually, the maximum fuel tank capacity determines the maximum range. The routine D$TANK computes the volumes of the fuel tanks defined in the MEDUSA drawing.

To compute the tank volume, the cross-sectional areas of the tank at the kinks are calculated. First, the thickness distribution at the local chord is derived by linear interpolation between the root and tip airfoil section of the corresponding panel:

$$t_i = c_r \left(\frac{t}{c_r} + \eta_c\left(\frac{t}{c_t}\lambda_p - \frac{t}{c_r}\right)\right) \tag{13-8}$$

where c_r and λ_p are the root chord and taper ratio of the wing panel and η_c is a non-dimensional, spanwise station with respect to the panel span:

$$\eta_c = \frac{l_{ft}}{b_p} \tag{13-9}$$

To compute the fuel tank cross-sectional area at a given spanwise station, the local airfoil thickness distribution is integerated in chordwise direction:

$$S_{tk} = \frac{c}{2} \sum_{i = i_1}^{i = i_2} (t_i + t_{i+1})(\eta_{i+1} - \eta_i) \tag{13-10}$$

where i_1 and i_2 are the chordwise front and rear tank station respectively, e.g. between the front and rear spar, and c is the local chord:

$$c = c_r \left(1 + \eta_c(\lambda_p - 1)\right) \tag{13-11}$$

In this way, the tank cross-sectional areas are computed at all the indicated sections and the volume of each section is calculated by assuming the tank geometry to be prismoidal shaped:

$$V_t = \frac{1}{3} \sum_{i=1}^{i=N_s} l_{s_i} \left(S_{r_i} + S_{t_i} + \sqrt{S_{r_i} S_{t_i}} \right) \quad \text{per wing-half} \qquad (13\text{-}12)$$

Note that this equation gives the gross fuel tank volume and the user should correct this value to allow for internal structure.

The lateral distance of the center of gravity from plane S_1 for each fuel tank follows from:

$$y_{cg_{fuel}} = \frac{l_s}{4} \frac{S_1 + 3\,S_2 + 2\sqrt{S_1\,S_2}}{S_1 + S_2 + \sqrt{S_1\,S_2}} \qquad (13\text{-}13)$$

The center of gravity per fuel tank $X_{cg_{fuel}}$ is assumed to be independent of the fuel weight. The total fuel weight per tank is divided into 5 portions and the resulting center of gravity follows from:

$$X_{cg_{i+1}} = \frac{X_{cg_i} W_i + X_{cg_{fuel}} \Delta W_{fuel}}{W_{i+1}} \qquad (13\text{-}14)$$

where

$$W_{i+1} = W_i + \Delta W_{fuel} \qquad (13\text{-}15)$$

and

$$\Delta W_{fuel} = \frac{f_{fuel} V_{ft}}{5} \qquad (13\text{-}16)$$

f_{fuel} is the specific fuel weight. This procedure is repeated for all fuel tanks, assuming that fuel tanks are filled from inboard to outboard.

14. DIRECT OPERATING COST

This Section gives basic equations to predict aircraft Direct Operating Cost (DOC) in the conceptual design stage. The method currently implemented in the program library, is based on Ref. 95 which gives the DOC per trip. Similar to the methods for weights and aerodynamic analysis (see Chapters 7 and 8), cost factors (φ) are included in the equations which can be specified by the designer through the database.

Utilization

Aircraft utilization is defined as the number of departures (trips) per year and is related to the flight time as follows:

$$U = \varphi_1 + \frac{\varphi_2}{\varphi_3 \, t_{fl}^{\varphi_4} + \varphi_5} \qquad (14\text{-}1)$$

where t_{fl} is the flight time.

Aircraft purchase price (exluding engines and spare parts)

A difficult item to estimate in DOC computations during design is the aircraft price. Especially the impact of new technology such as digital avionics, fly-by-wire, large scale use of composite materials and new production techniques are difficult to predict. Also, the non-recurring cost per aircraft, and thus the price, is determined by the total number of aircraft to be produced. Finally, company policy and sales conditions also have an effect on aircraft price. A very simple prediction method relates aircraft airframe price to the maximum takeoff weight or operational empty weight (minus engine weight), as follows:

$$P_{af} = \varphi_1 + \varphi_2 W_{to}^{\varphi_3} + \varphi_4 \left(W_{oe} - W_{en}\right)^{\varphi_5} \qquad (14\text{-}2)$$

Airframe spare parts

$$P_{afsp} = \varphi_1 + \varphi_2 P_{af}^{\varphi_3} \qquad (14\text{-}3)$$

Engine price (excl. spare parts)

$$P_{eng} = \varphi_1 + \varphi_2 T_{to}^{\varphi_3} \qquad (14\text{-}4)$$

Engines spare parts

$$P_{esp} = \varphi_1 + \varphi_2 P_{eng}^{\varphi_3} \qquad (14\text{-}5)$$

The total aircraft price, including spare parts, is:

$$P_{ac} = P_{af} + P_{afsp} + N_e \left(P_{eng} + P_{esp}\right) \qquad (14\text{-}6)$$

Depreciation and interest

$$C_{depr} = \varphi_1 + \varphi_2 \frac{P_{ac}^{\varphi_3}}{U^{\varphi_4}} \qquad (14\text{-}7)$$

Insurance

$$C_{ins} = \varphi_1 + \varphi_2 \frac{P_{ac}^{\varphi}}{U^{\varphi_4}} \qquad (14\text{-}8)$$

Airtraffic control fee

$$C_{ATC} = \varphi_1 + \varphi_2 W_{to}^{\varphi_3} \tag{14-9}$$

Landing fee

$$C_{land} = \varphi_1 + \varphi_2 W_{to}^{\varphi_3} \tag{14-10}$$

Aircraft servicing

$$C_{serv} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} \tag{14-11}$$

Cabin crew pay

$$C_{ccp} = \varphi_1 + N_{pass}^{\varphi_2} \left(\varphi_3 t_{fl} + \varphi_4 t_{fl}^{\varphi_5}\right) \tag{14-12}$$

Flight crew pay

$$C_{fcp} = \varphi_1 + \varphi_2 t_{fl}^{\varphi_3} + W_{to} \left(\varphi_4 + \varphi_5 t_{fl}\right) \tag{14-13}$$

Fuel cost

$$C_{fuel} = \varphi_1 + \varphi_2 W_{fuel}^{\varphi_3} \tag{14-14}$$

Maintenance cost

Maintenance cost of airframe and engines is another contribution to the DOC which is difficult to predict. Factors like accessibility, inspectibility, engine modularity, built-in test facilities, system complexity, etc. have an effect on maintenance cost. The method given in Ref. 95 divides maintenance cost in a number of sub-contributions, where a distinction is made between labour and material cost. For the sake of simplicity, the latter two are added in one equation.

Inspection and miscellaneous cost

$$C_{insp} = \varphi_1 + \varphi_2 \, W_e^{\varphi_3} \tag{14-15}$$

Airconditioning system

$$C_{acs} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \tag{14-16}$$

Flight management system

$$C_{fms} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \tag{14-17}$$

Communications

$$C_{comm} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} \tag{14-18}$$

Electrical system

$$C_{els} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} \tag{14-19}$$

Furnishing and equipment

$$C_{feq} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} \tag{14-20}$$

Fire protection system

$$C_{fdes} = \varphi_1 + \varphi_2 \, (N_e + 1)^{\varphi_3} \tag{14-21}$$

Flight controls system

$$C_{fcs} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \tag{14-22}$$

Fuel system

$$C_{fs} = \varphi_1 + \varphi_2 \, V_{tk}^{\varphi_3} \tag{14-23}$$

Hydraulics system

$$C_{hs} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \tag{14-24}$$

Anti-icing system

$$C_{ais} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \tag{14-25}$$

Instruments

$$C_{instr} = \varphi_1 + \varphi_2 \, W_{e}^{\varphi_3} \tag{14-26}$$

Undercarriage

$$C_{uc} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \tag{14-27}$$

Lighting system

$$C_{ls} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} \tag{14-28}$$

Navigation system

$$C_{nav} = \varphi_1 + \varphi_2 \, W_{to}^{\varphi_3} \tag{14-29}$$

Oxygen system

$$C_{os} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} \tag{14-30}$$

Pneumatics system

$$C_{ps} = \varphi_1 + \varphi_2 W_{to}^{\varphi_3} + \varphi_4 T_{to}^{\varphi_5} \qquad (14\text{-}31)$$

Water/waste system

$$C_{wws} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} \qquad (14\text{-}32)$$

Auxilary Power Unit (APU)

$$C_{APU} = \varphi_1 + \varphi_2 W_{to}^{\varphi_3} + \varphi_4 N_{pass}^{\varphi_5} \qquad (14\text{-}33)$$

Structures

$$C_{struc} = \varphi_1 + \varphi_2 W_{e}^{\varphi_3} \qquad (14\text{-}34)$$

Doors

$$C_{doors} = \varphi_1 + \varphi_2 N_{pass}^{\varphi_3} \qquad (14\text{-}35)$$

Fuselage

$$C_{fus} = \varphi_1 + \varphi_2 W_{e}^{\varphi_3} \qquad (14\text{-}36)$$

Nacelles

$$C_{nac} = \varphi_1 + \varphi_2 N_{e}^{\varphi_3} \qquad (14\text{-}37)$$

Stabilizer

$$C_{stab} = \varphi_1 + \varphi_2 W_{to}^{\varphi_3} \qquad (14\text{-}38)$$

Windows

$$C_{wind} = \varphi_1 + \varphi_2 \, N_{pass}^{\varphi_3} \tag{14-39}$$

Wing

$$C_w = \varphi_1 + \varphi_2 \, S_w^{\varphi_3} \tag{14-40}$$

Engines

The method given in Ref. 95 requires some detail information on engine hardware and thermodynamics for the prediction of engine maintenance cost. A simpler method is given in Ref. 8 which relates engine maintenance cost primarily to the takeoff thrust and the number of engines as follows:

$$C_{eng} = (\varphi_1 + \varphi_2 \, T_{to})N_e \, t_{fl} + (\varphi_3 + \varphi_4 \, T_{to})N_e \tag{14-41}$$

The total DOC/trip is obtained by summation of the above contributions.

PART 4: APPLICATION EXAMPLE OF ADAS

This part illustrates how ADAS can be applied to solve a typical design problem. The example aircraft is a 44 passenger, short-haul airliner with 2 fuselage-mounted turbofans and is based on a hypothetical design specification. Starting with an existing baseline design, selected design parameters are optimized with ADAS using both implicit and explicit optimization. In conclusion, some design characteristics of the optimum design are presented.

15. DESIGN EXAMPLE OF A SHORT-HAUL PASSENGER AIRLINER

This Chapter gives an example of how ADAS can be applied in a typical design optimization study. Starting with a given baseline design, based on a hypothetical design specification, selected design parameters are optimized using a combination of implicit and explicit optimization. The objective of this exercise is to illustrate some options in ADAS and to demonstrate how they can be applied in real design practice. The analysis methods used are for the larger part described in Part 3. As the optimum design solution is dependent on the accuracy of the methods and on the selection of technology factors and other 'guesstimates', emphasis has been put on a qualitative assessment and to signal certain design trends.

15.1 The design specification

Aircraft synthesis design constitutes an important part of the curriculum at many aeronautical faculties [Refs. 98 and 108]. In the 5th semester of the curriculum, the aeronautical students at the Faculty of Aerospace Engineering with interest in aircraft design, can participate in the preliminary design exercise. They can choose from about 25 design specifications in different subsonic aircraft categories, such as general aviation, executive/business, trainers, agricultural aircraft, seaplanes, sailplanes, cargo and passenger transports. Because of the given time frame - nominally 200 hrs - configuration development is necessarily restricted to one design cycle, with little or no optimization aspects involved.

The example discussed in this chapter will refer to a hypothetical aircraft based on one of these design specifications. The major operational and performance requirements are enumerated in Table 15.1:

Table 15.1. Design specification for a hypothetical aircraft design.

Number of passengers (standard)	44	
Passenger cabin volume	≈ 55 m^3	(942 ft^3)
Seat pitch	0.8 m	(31.5 inch)
Minimum aisle width	0.46 m	(18.1 inch)
Cargo capacity	1200 kg	(2646 lbs)
Minimum cargo holds volume	6 m^3	(212 ft^3)
Cruise speed	700 km/h	(378 knots)
Cruise altitude	7500 m	(24606 ft)
Maximum speed at 6000 m	750 km/h	(405 knots)
Minimum speed in landing configuration	170 km/h	(92 knots)
Rate of climb at sealevel	15 m/s	(49 ft/s)
Rate of climb at cruise altitude	1.5 m/s	(5 ft/s)
One-engine-out ceiling	4600 m	15092 ft)
Takeoff distance	1350 m	(4429 ft)
Landing distance	1200 m	(3937 ft)
Range with maximum payload	1200 km	(648 nm)
Range with maximum fuel (Reserves according to FAR 121.639 with 370 km (200 nm) to alternate)	2500 km	(1350 nm)

15.2 The baseline design

As mentioned in Section 2.2, a typical design process with ADAS starts off with the conception and definition of an intuitive design configuration, referred to as the baseline design (TO-2250). This design is primarily based on designer's experience, study of other, comparable designs, literature audit, and relatively simple calculations. However, at this time it is unknown whether the proposed design is optimal or even feasible as any thorough analysis has not yet been carried out. The 3-view configuration drawing of the TO-2250 baseline design is given in Figure 15.1. Fuselage design is generaly a suitable starting point for configuration development. The fuselage shape and dimensions are primarily dependent on the payload to be accommodated. For passenger transports, factors such as number of seats abreast, seat dimensions, aisles width and requirements for above floor cargo compartments affect the fuselage dimension and shape. For the TO-2250 the fuselage geometry and internal arrangement is given in Figure 15.2.

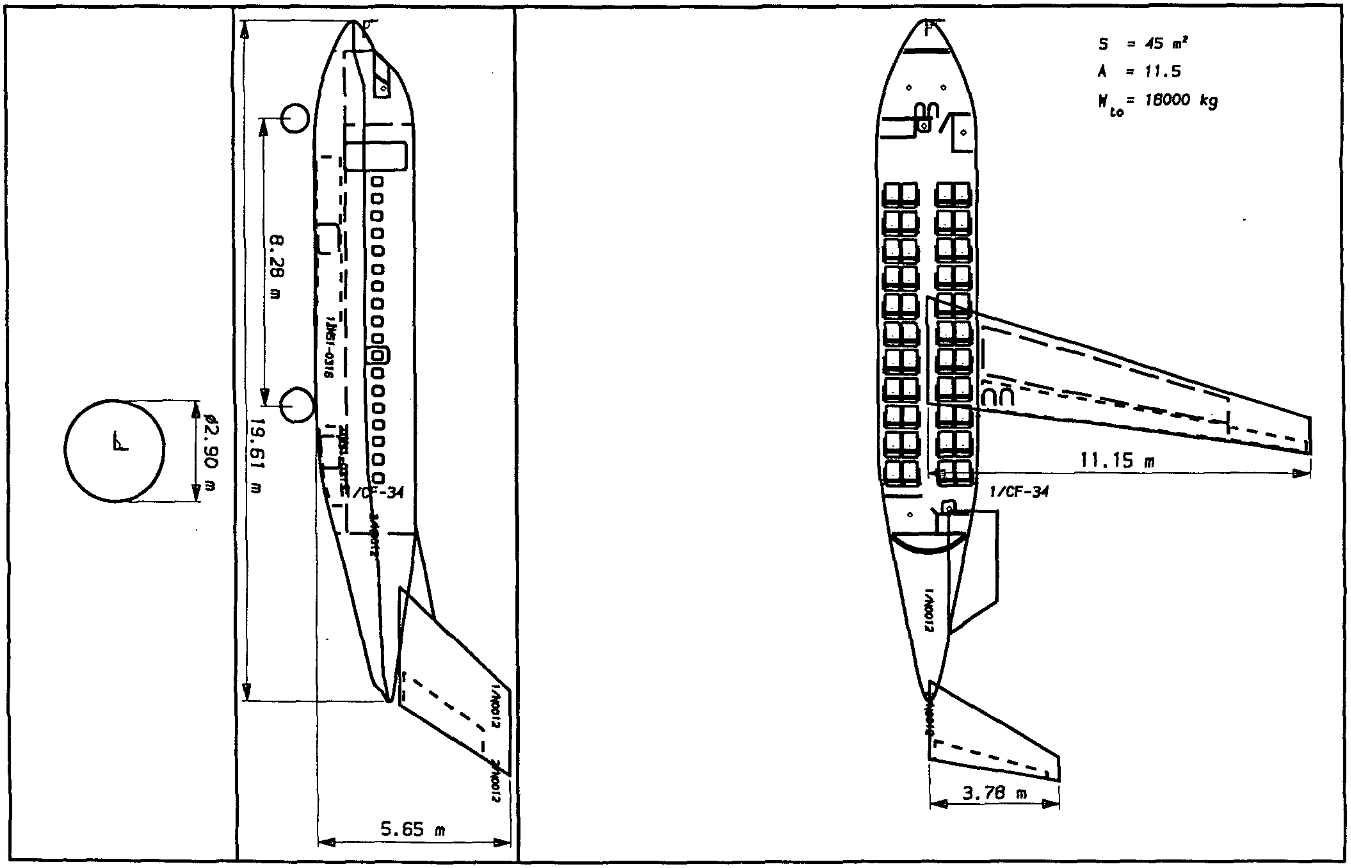


Figure 15.1: Baseline design configuration of a hypothetical passenger airliner.

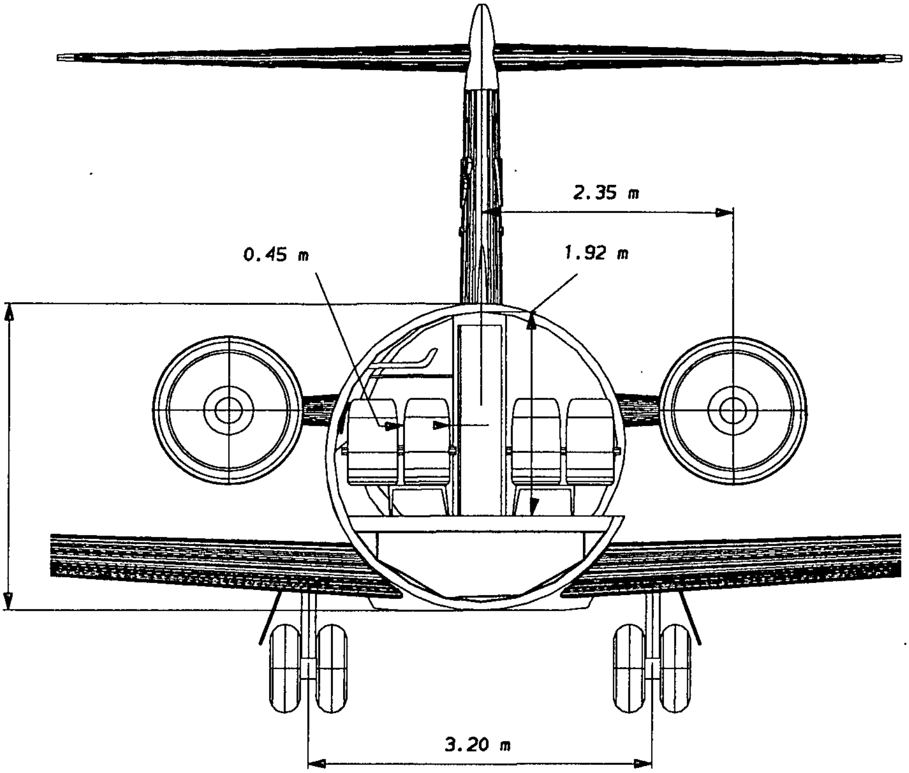


Figure 15.2: Fuselage cross-sectional shape.

The fuselage external diameter (2.90 m), relatively large for this type of aircraft, was due to the requirement that all cargo must be stored in underfloor (belly) cargo holds. This also presented the opportunity to increase the seat dimensions and aisle width to improve passenger comfort. The digitization of the fuselage geometry by ADAS into 51 equidistant stations is given in Table 15.2. The fuselage volume and wetted area, computed according to Eqs. (B-6) and (B-7) respectively, have been compared with the results from the MEDUSA model properties module with 10 x 10 facets per patch. The relative error is only 0.1 %.

Table 15.2: Fuselage dimensions and principal geometric properties.

FUSELAGE GEOMETRY

cross-sectional shape factors 0.70711/ 0.70711

fus sta	max width	dimensions height		dimensions width		cross-sectional area		cross-sectional circumference	
m	% h/2	m	% max	m	% max	m²	% max	m	% max
0.00	0.1	0.01	0.3	0.01	0.3	0.00	0.0	0.03	0.3
0.39	1.6	0.76	26.2	0.93	32.1	0.56	8.4	2.66	29.2
0.78	-0.4	1.14	39.4	1.37	47.3	1.23	18.6	3.96	43.4
1.18	-1.0	1.62	56.0	1.75	60.2	2.22	33.7	5.29	58.1
1.57	-0.8	2.11	72.9	2.09	72.0	3.47	52.5	6.60	72.5
1.96	3.4	2.44	84.1	2.39	82.3	4.57	69.2	7.58	83.2
2.35	1.4	2.66	91.7	2.61	90.2	5.46	82.7	8.28	90.9
2.75	0.3	2.81	96.8	2.78	95.8	6.12	92.7	8.77	96.3
3.14	0.0	2.88	99.4	2.87	99.0	6.50	98.4	9.04	99.2
3.53	0.0	2.90	100.0	2.90	100.0	6.61	100.0	9.11	100.0
3.92	0.0	2.90	100.0	2.90	100.0	6.61	100.0	9.11	100.0
4.31	0.0	2.90	100.0	2.90	100.0	6.61	100.0	9.11	100.0
10.59	0.0	2.90	100.0	2.90	100.0	6.61	100.0	9.11	100.0
10.98	0.0	2.90	100.0	2.90	100.0	6.61	100.0	9.11	100.0
11.37	0.0	2.90	100.0	2.90	100.0	6.61	100.0	9.11	100.0
11.77	0.6	2.89	99.8	2.90	100.0	6.59	99.8	9.10	99.9
12.16	1.0	2.87	98.8	2.90	100.0	6.53	98.8	9.06	99.4
12.55	0.8	2.82	97.3	2.89	99.8	6.42	97.2	8.98	98.6
12.94	0.3	2.77	95.5	2.86	98.6	6.22	94.2	8.84	97.1
13.33	0.4	2.71	93.5	2.79	96.2	5.94	90.0	8.64	94.9
13.73	0.6	2.64	90.9	2.69	92.9	5.58	84.5	8.38	91.9
14.12	-0.1	2.54	87.4	2.58	88.9	5.13	77.7	8.03	88.1
14.51	-0.1	2.43	83.8	2.44	84.2	4.66	70.6	7.65	84.0
14.90	-0.3	2.31	79.5	2.29	79.1	4.16	62.9	7.23	79.3
15.30	-0.7	2.17	74.8	2.14	73.7	3.64	55.1	6.76	74.2
15.69	-0.7	2.02	69.6	1.98	68.1	3.13	47.4	6.27	68.9
16.08	-0.4	1.86	64.2	1.82	62.6	2.65	40.2	5.77	63.4
16.47	-0.4	1.70	58.5	1.65	57.1	2.21	33.4	5.27	57.8
16.86	-0.5	1.54	53.0	1.49	51.5	1.80	27.3	4.76	52.2
17.26	-0.5	1.37	47.4	1.33	46.0	1.44	21.8	4.25	46.7
17.65	-0.7	1.21	41.8	1.17	40.5	1.12	16.9	3.75	41.1
18.04	-1.2	1.05	36.2	1.01	34.9	0.84	12.7	3.24	35.6
18.43	-1.8	0.89	30.6	0.85	29.4	0.60	9.0	2.74	30.0
18.83	-2.7	0.73	25.1	0.68	23.4	0.39	5.9	2.21	24.2
19.22	3.7	0.50	17.2	0.47	16.1	0.18	2.8	1.52	16.6
19.61	0.1	0.01	0.3	0.01	0.3	0.00	0.0	0.03	0.3

maximum width	2.90	m
maximum height	2.90	m
wetted area	143.74	m²
volume	93.39	m³

15.3 Optimum aspect ratio and wing loading for minimum MTOW

The objective of this study is to optimize the aircraft configuration, i.c. aspect ratio and wing loading, such that all requirements are met and Maximum TakeOff Weight (MTOW) is minimum.

Tailplane sizing is approximated by keeping the respective tail volume parameters constant, i.c.:

horizontal tailplane $$\frac{S_h l_h}{S \bar{c}} = 1.20$$

vertical tailplane $$\frac{S_v l_v}{S b} = 0.065$$

The effect of engine size is assessed by matching engine thrust to a given performance requirement. The following cases were considered:

case I: Engines fixed, engine size remains equal to the reference engine (Section 15.3.1).

case II: Engines sized for cruise, thrust is selected such that at the given cruise condition, the cruise engine rating is the recommended cruise engine rating (Section 15.3.2).

case III: Engines sized for takeoff, thrust is selected such that the maximum takeoff distance and the minimum second segment climb gradient are met for an optimum flap setting (Section 15.3.3).

In Section 15.5 the effect of range variation on the optimum design is assessed by assuming that the proposed design is to be convertable to an executive/business jet with transatlantic capability with the same design payload. In Section 15.6, the effect of alternative merit functions on the optimum design configuration is investigated, by optimizing for minimum Direct Operating Cost (DOC). In both cases engines were sized for takeoff (case III).

These case studies have been selected as it constitutes a classical design problem while at the same time presents the opportunity to demonstrate the combined use of implicit and explicit optimization with ADAS. The procedure used to solve these three design problems is to apply parameter variation (explicit optimization) and to evaluate 10 x 10 designs with different values of A and $(W/S)_{to}$. All other design parameters are derived from the baseline design and are held constant. For each design the performance requirements listed in Table 15.1 are computed. The input data to run ADAP in parametric survey mode is:

```
                         ****************************
                         *                          *
                         *  PARAMETRIC SURVEY MODE  *
                         *                          *
                         *       - I N P U T -      *
                         *                          *
                         ****************************

No      Survey Parameters                  From       To        Step      Unit
                                                                               2
 1 WING LOADING                             300.       450.        17. kg/m
 2 ASPECT RATIO                            3.000     12.000      1.000

Table              Survey Functions                        Unit          Fmt

  1     CRUISE ENGINE RATING                                              4
  2     ENGINE RATING AT MAXIMUM SPEED                                    4
  3     RANGE WITH MAXIMUM PAYLOAD                          km            0
  4     TAKEOFF FIELD LENGTH                                m             0
  5     SECOND SEGMENT CLIMB GRADIENT                                     5
  6     LANDING DISTANCE                                    m             0
  7     RATE OF CLIMB AT SEA LEVEL                          m/s           2
  8     RATE OF CLIMB AT 7500 M                             m/s           2
  9     RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT              m/s           2
 10     RANGE WITH MAXIMUM FUEL                             km            0
 11     MAXIMUM TAKEOFF MASS                                kg            0
 12     TAKEOFF MAXIMUM LIFT COEFFICIENT                                  3
 13     THRUST/WEIGHT RATIO                                               4
```

CRUISE ENGINE RATING (r_{cr}) is the T/T_{max} ratio at the specified cruise speed (700 km/hr) and altitude (7500 m). One could also compute cruise speed at a given altitude and engine rating, but this would require a more complicated procedure. The recommended cruise engine rating is 90 %. Similarly, ENGINE RATING AT MAXIMUM SPEED (r_{max}) is the T/T_{max} ratio at maximum speed (750 km/hr) at the specified altitude (6000 m). The maximum continuous engine rating is 95 %. RANGE WITH DESIGN PAYLOAD (R_d) is the equivalent range with maximum payload. To account for reserve fuel and fuel required for takeoff and landing, this value is set at 2000 km. RANGE WITH MAXIMUM FUEL (R_{max}) is the equivalent range that can be flown with maximum fuel load (tank volume or weight limited) at maximum takeoff weight and reduced payload. For this design the requirement is 3000 km. When ADAP is run in parametric survey mode, these design characteristics are computed for 100 different design configurations and output in parametric table format. Subsequently, these results are graphically represented in the form of contour plots and the performance requirements are plotted as explicit constraint boundaries.

15.3.1 Case I: Engines fixed

This situation occurs when it is decided to consider only one engine type with a given takeoff thrust. In this case, powerplant related drag and weight are not affected. Also, the maximum lift coefficient at takeoff is fixed. As mentioned in Section 7.12, the range module F$CRSE computes range for a given fuel weight. The available fuel weight at maximum payload follows from the relationship $W_{fuel} = W_{to} - W_{mzf}$ where the maximum takeoff weight W_{to} is given. Thus, to determine W_{fuel} for a specified range the optimizer can be used with W_{to} specified as a free variable and an implicit equality constraint on range:

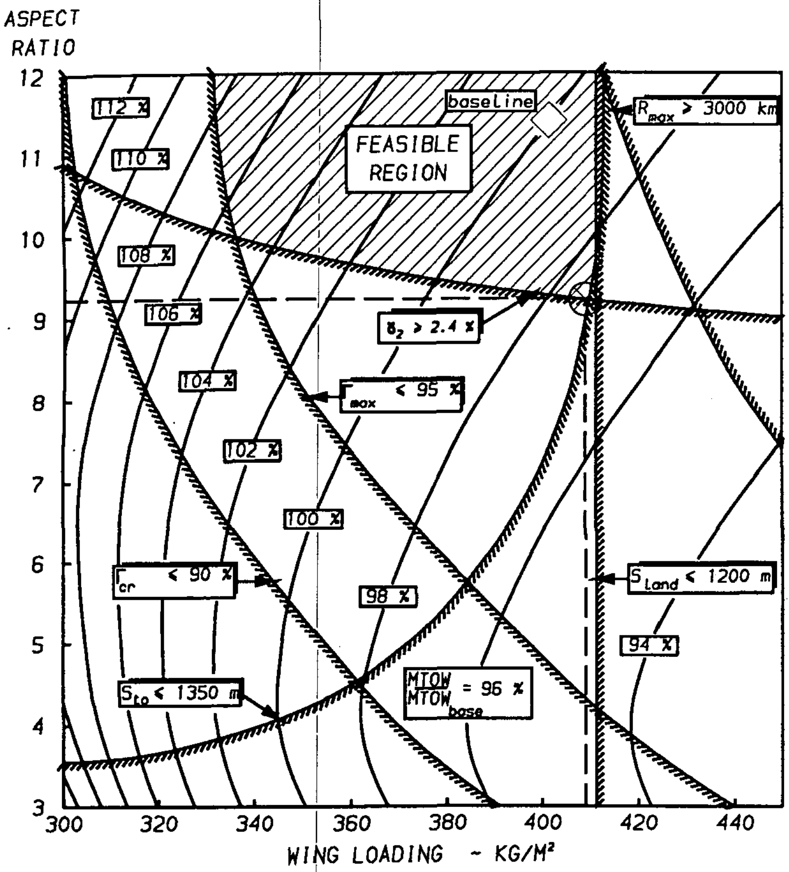
```
                              ****************************
                              *                          *
                              *    OPTIMIZATION MODE     *
                              *                          *
                              *       - I N P U T -      *
                              *                          *
                              ****************************

Lower Bound                     Free Variable                      Upper Bound  Unit

    14000.  ≤  MAXIMUM TAKEOFF MASS                            ≤       30000. kg
------------------------------------------------------------------------------------

Lower Bound                      Constraint                        Upper Bound  Unit

     2000.  ≤  RANGE WITH DESIGN PAYLOAD                       ≤        2000. km

                              Objective Function

MAXIMUM TAKEOFF MASS                                           MINIMIZATION     kg
```

Note that, because the number of free variables is equal to the number of equality constraints, this entails a root finding problem rather than an optimization problem. The graphical results for the case with engines fixed is given in Figure 15.3. Both the baseline and the optimum design are indicated. From Figure 15.3 it can be concluded that the baseline design is a feasible concept, but a lower aspect ratio (A = 9.2) would result in an improvement in MTOW of about 2.5 %. The total required CPU-time was 3.5 hrs. with 10 sec. disk access time.

Engine scaling is a technique which is applied when a specific engine type is not available or has not yet been selected. With ADAS, engine "rubberizing" can be easily applied by defining a suitable parameter, e.g. thrust/weight ratio $(T/W)_{to}$, as a free variable and to use the optimizer to match engine thrust to the aircraft performance requirements. The engine scaling module D$RUBB should be included in the analysis program to scale engine thrust, and all related engine/nacelle properties and dimensions, relative to the reference engine according to the required takeoff thrust. Obviously, if no constraints are specified, the optimizer will set $(T/W)_{to}$ to its lower bound to achieve the smallest engine and the lowest MTOW. A more realistic result can be obtained by defining performance requirements, e.g. the takeoff distance or cruise speed/altitude, as implicit constraints. This will be illustrated in the following Sections for two different engine sizing criteria: engines sized for cruise and engines sized for takeoff respectively.

15.3.2 Case II: Engines sized for cruise

To match for example engine thrust to the cruise condition, i.e. the required engine thrust for a given cruise speed and altitude should be equal to the available thrust for a given engine rating, $(T/W)_{to}$ can be defined as a free variable and cruise engine rating can be specified as an implicit constraint, as follows:



DESIGN POINT ⊗

WING LOADING	410. KG/M²
ASPECT RATIO	9.220
CRUISE ENGINE RATING	0.7288
ENGINE RATING AT MAXIMUM SPEED	0.8404
RANGE WITH MAXIMUM PAYLOAD	1999. KM
TAKEOFF FIELD LENGTH	1350. M
SECOND SEGMENT CLIMB GRADIENT	0.02400
LANDING DISTANCE	1197. M
RATE OF CLIMB AT SEA LEVEL	21.57 M/S
RATE OF CLIMB AT 7500 M	8.94 M/S
RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT	3.33 M/S
RANGE WITH MAXIMUM FUEL	3233. KM
MAXIMUM TAKEOFF MASS	17533. KG
TAKEOFF MAXIMUM LIFT COEFFICIENT	2.500
THRUST LOADING	0.3582

ENGINES FIXED

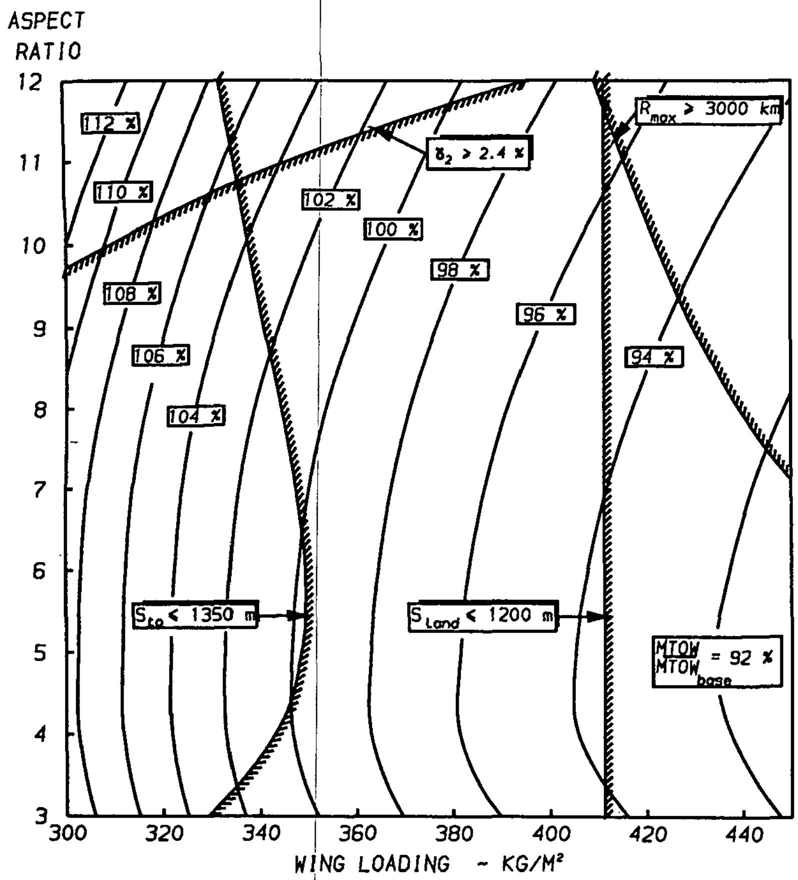
Figure 15.3: Optimization of wing loading and aspect ratio for minimum MTOW and with fixed engines. The design equivalent range is 2000 km.

```
                        ****************************
                        *                          *
                        *    OPTIMIZATION MODE     *
                        *                          *
                        *      - I N P U T -       *
                        *                          *
                        ****************************

Lower Bound                  Free Variable                   Upper Bound  Unit

   14000.  ≤  MAXIMUM TAKEOFF MASS                     ≤         30000. kg
   0.200   ≤  THRUST/WEIGHT RATIO                      ≤          0.800

Lower Bound                   Constraint                     Upper Bound  Unit

   2000.   ≤  RANGE WITH MAXIMUM PAYLOAD               ≤          2000. km
   0.000   ≤  CRUISE ENGINE RATING                     ≤          0.900

                           Objective Function

MAXIMUM TAKEOFF MASS                                  MINIMIZATION      kg
```

The results for engines sized for cruise are given in Figure 15.4. For all designs the engine rating was set at the specified upper bound, i.c. 90 %, which was to be expected as it represents the smallest possible engine for cruise. However, due to the relatively low $(T/W)_{to}$, there is no feasible region in the indicated design space as the maximum speed requirement can not be met for any of the evaluated designs. Hence, the cruise condition is not a convenient engine sizing criteria for this type of aircraft. The total required CPU-time for this case was about 7.2 hrs. and 22 sec. of disk-access time.

15.3.3 Case III: Engines sized for takeoff

An alternative criteria for engine sizing may be the takeoff field performance. For typical design configurations, this requirement is generally the critical factor for engine size, especially the climb requirements after engine failure. To investigate the effect of this criteria on the optimum design the takeoff distance and the second segment climb requirement were defined as implicit constraints. Also, the maximum lift coefficient at takeoff was defined as a free variable to compute optimal flap setting for each design:



ENGINES SIZED
FOR
CRUISE

Figure 15.4: Optimization of wing loading and aspect ratio for minimum MTOW and with engines sized for cruise. The design equivalent range is 2000 km.

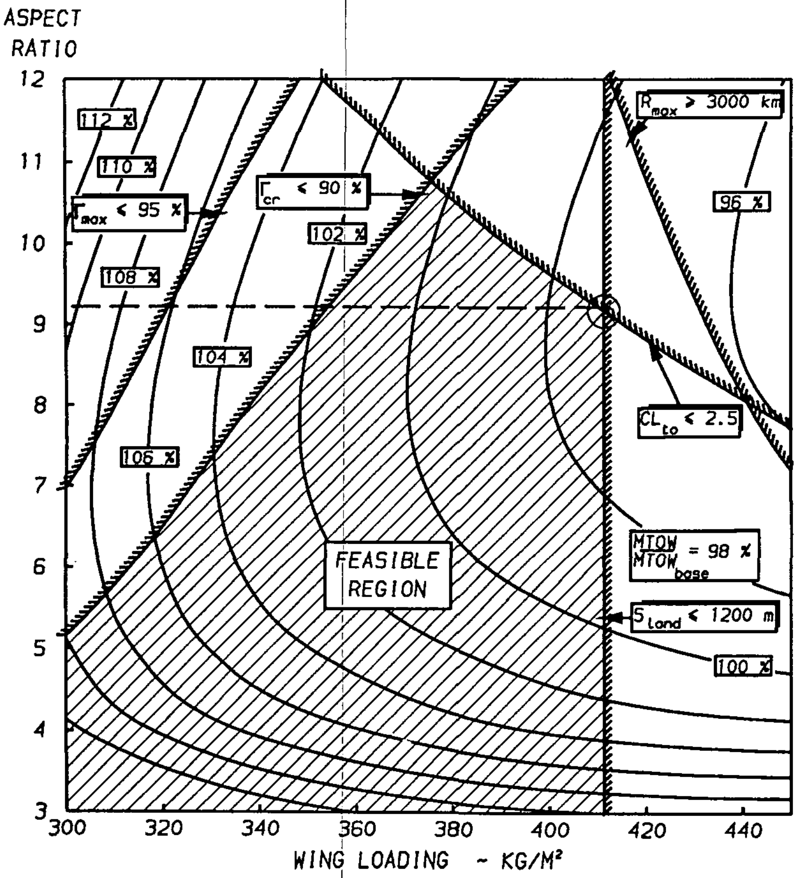
```
*****************************
*                           *
*     OPTIMIZATION MODE     *
*                           *
*       - I N P U T -       *
*                           *
*****************************


Lower Bound              Free Variable                     Upper Bound  Unit

   14000.  ≤  MAXIMUM TAKEOFF MASS                     ≤      30000. kg
   0.200   ≤  THRUST/WEIGHT RATIO                      ≤      0.800
   1.000   ≤  TAKEOFF MAXIMUM LIFT COEFFICIENT         ≤      3.000

Lower Bound               Constraint                       Upper Bound  Unit

    2000.  ≤  RANGE WITH MAXIMUM PAYLOAD               ≤       2000. km
       0.  ≤  TAKEOFF FIELD LENGTH                     ≤       1350. m
   0.0240  ≤  SECOND SEGMENT CLIMB GRADIENT            ≤      0.1000

                      Objective Function

MAXIMUM TAKEOFF MASS                               MINIMIZATION      kg
```

The resulting contour plot for this design case is given in Figure 15.5. As a simplified takeoff prediction method was used, an explicit constraint had to be imposed on the maximum takeoff lift coefficient ($C_{L_{to}} \leq 2.5$), as the method would be to inaccurate for higher values.

The total CPU-time required to solve this design problem was 9.5 hrs. and 30 sec. of disk-access time.

15.3.4 Design with multivariate optimization

In the design examples discussed previously, it has been illustrated how a combined use of implicit and explicit optimization can be applied in a design study. However, limited use was made of the optimization algorithm only to match the fuel load to the required range and for engine "rubberizing". Alternatively, the designer can define additional free variables and implicit constraints. Note that with implicit optimization, the objective function and constraint bounds have to be defined in advance,



DESIGN POINT ⊗

WING LOADING	411. KG/M²
ASPECT RATIO	9.165
CRUISE ENGINE RATING	0.7252
ENGINE RATING AT MAXIMUM SPEED	0.8362
RANGE WITH MAXIMUM PAYLOAD	2000. KM
TAKEOFF FIELD LENGTH	1350. M
SECOND SEGMENT CLIMB GRADIENT	0.02400
LANDING DISTANCE	1200. M
RATE OF CLIMB AT SEA LEVEL	21.73 M/S
RATE OF CLIMB AT 7500 M	9.02 M/S
RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT	3.36 M/S
RANGE WITH MAXIMUM FUEL	3222. KM
MAXIMUM TAKEOFF MASS	17519. KG
TAKEOFF MAXIMUM LIFT COEFFICIENT	2.500
THRUST LOADING	0.3250

ENGINES
SIZED FOR
TAKEOFF

Figure 15.5: Optimization of wing loading and aspect ratio for minimum MTOW and with engines sized for takeoff and for optimum flap angle. The design equivalent range is 2000 km.

while with explicit optimization the objective function and constraints can be specified afterwards, i.c. at the time the contour plots are made.

For example, the optimum design point in Figure 15.5 can also be found by defining wing loading and aspect ratio as free variables and by selecting all the performance requirements listed in Table 15.1 as constraints:

```
                         ***************************
                         *                         *
                         *    OPTIMIZATION MODE    *
                         *                         *
                         *      - I N P U T -      *
                         *                         *
                         ***************************

Lower Bound                  Free Variable                      Upper Bound  Unit

    14000.  ≤  MAXIMUM TAKEOFF MASS                        ≤      30000.  kg
      200.  ≤  WING LOADING                                ≤        500.  kg/m²
     5.000  ≤  ASPECT RATIO                                ≤      15.000
     0.100  ≤  THRUST/WEIGHT RATIO                         ≤       0.800
     0.100  ≤  MAXIMUM TAKEOFF LIFT COEFFICIENT            ≤       3.000

Lower Bound                  Constraint                         Upper Bound  Unit

     0.000  ≤  CRUISE ENGINE RATING                        ≤       0.900
     0.000  ≤  ENGINE RATING AT MAXIMUM SPEED              ≤       0.950
     2000.  ≤  RANGE WITH MAXIMUM PAYLOAD                  ≤       2000.  km
        0.  ≤  TAKEOFF FIELD LENGTH                        ≤       1350.  m
     0.024  ≤  SECOND SEGMENT CLIMB GRADIENT               ≤       1.000
        0.  ≤  LANDING DISTANCE                            ≤       1200.  m
     15.00  ≤  RATE OF CLIMB AT SEA LEVEL                  ≤       50.00  m/s
      1.50  ≤  RATE OF CLIMB AT 7500 M                     ≤       50.00  m/s
      0.50  ≤  RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT      ≤       50.00  m/s
     3500.  ≤  RANGE WITH MAXIMUM FUEL                     ≤       5000.  km

                              Objective Function

MAXIMUM TAKEOFF MASS                                  MINIMIZATION        km
```

Running this problem in optimization mode gives the following numerical results on the optimum design:

```
                          *****************************
                          *                           *
                          *    OPTIMIZATION MODE      *
                          *                           *
                          *   - S O L U T I O N -     *
                          *                           *
                          *       error = 0/ 5        *
                          *                           *
                          *****************************

                                Free Variables

MAXIMUM TAKEOFF MASS                                           17583. kg
WING LOADING                                                     411. kg/m²
ASPECT RATIO                                                    9.112
THRUST/WEIGHT RATIO                                             0.330
MAXIMUM TAKEOFF LIFT COEFFICIENT                                2.476


                              Objective Function

MAXIMUM TAKEOFF MASS                                           17583. kg


                                 Constraints

CRUISE ENGINE RATING                                              0.713
ENGINE RATING AT MAXIMUM SPEED                                    0.785
RANGE WITH DESIGN PAYLOAD                                         2000. km
TAKEOFF FIELD LENGTH                                              1350. m
SECOND SEGMENT CLIMB GRADIENT                                     0.024
LANDING DISTANCE                                                  1200. m
MAXIMUM RATE OF CLIMB AT SEA LEVEL                                22.21 m/s
MAXIMUM RATE OF CLIMB AT 7500 M                                    9.29 m/s
RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT                             3.49 m/s
RANGE WITH MAXIMUM FUEL                                           3309. km
```

Within acceptable tolerances the optimum design is the same as found in Figure 15.5. This example clearly illustrates the principal difference between implicit and explicit optimization. With implicit optimization, only one (optimum) design point is obtained, with little information on the sensitivity of the objective function and constraints with respect to the free variables. However, the CPU-time was only 7 min. compared to the 9.5 hrs. required to generate Figure 15.11. Especially in those design studies where many design variables, say more than 10, are to be optimized, implicit optimization seems to be the only practical alternative.

15.4 Optimum design evaluation

With the use of a tracking-cross device, the designer can indicate the desired optimum design point in the contour plot. ADAS will print-out the corresponding values of the survey variables and functions, as derived from interpolation. After an optimum design configuration has been selected, additional design and engineering information on the optimum design can be generated by running the ADAP executive program in point design analysis mode for the particular design configuration. The analysis program may include any selected plot and print routine available from the program library or the designer may define his own specific output routines. As it is not practical to present all possible information on the optimum design here in detail, only some examples of data representation will be given as an illustration.

For example, in this design study, the wing location remained invariant with respect to the baseline design. Therefore, a check has to be made afterwards to assure that the most forward and aft CG-position are within acceptable limits. A load and balance diagram can be made which gives the CG-location for various loading conditions of passengers, cargo and fuel in operational use (see Chapter 10). The D$PL09 routine in the program library draws a loading diagram automatically into a MEDUSA-sheet. As an example, the loading diagram of the optimum design configuration is presented in Figure 15.6. Another common print-out is a detailed weight breakdown statement, which tabulates the weight and CG-location of subcomponents and the corresponding design weights. Table 15.3 gives the weight statement for the optimum design configuration.

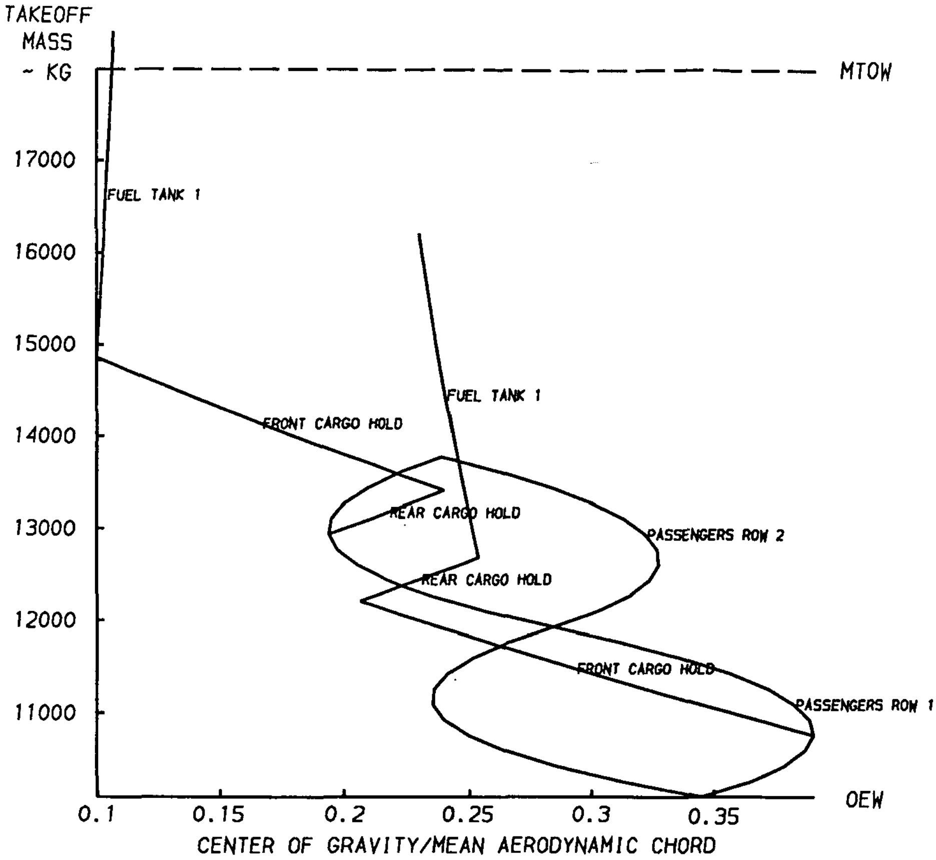


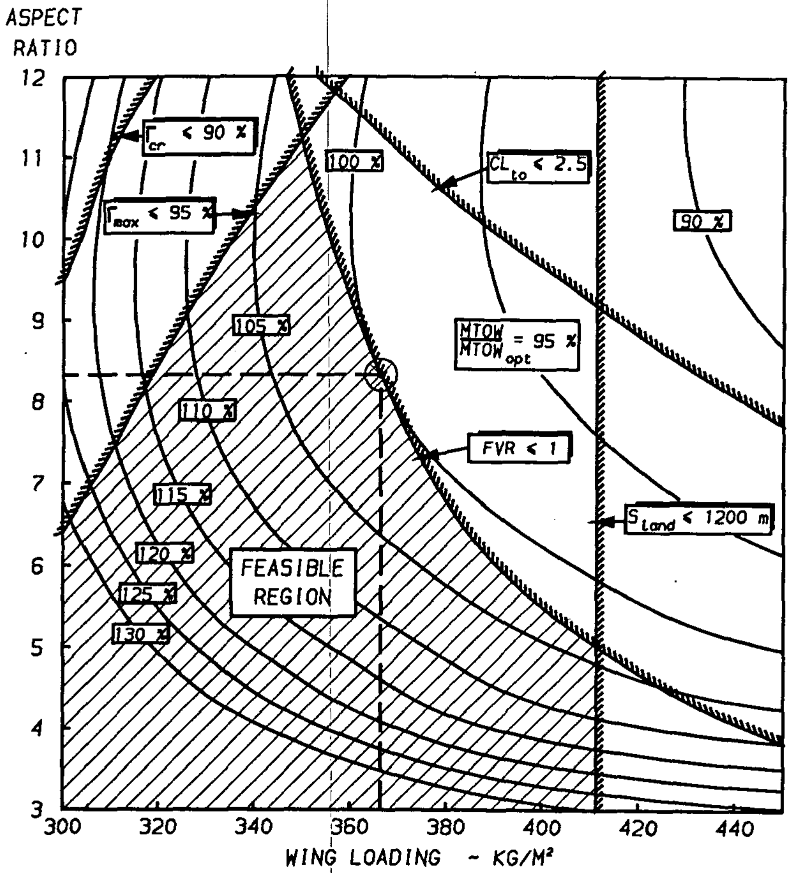
Figure 15.6: Load and balance diagram for the TO-2250 aircraft.

15.5 Effect of design range.

In general, the optimum design configuration depends on the selection of the free variables, constraints and objective function. If, for example, the proposed aircraft is to be convertable to an executive/business jet version with transatlantic range capability, a different optimum design is found. This is illustrated in Figure 15.7, where for case III the design range was increase to 5000 km. The active constraint for this design is the fuel volume ratio (FVR) which represents the required fuel capacity/available fuel tank capacity.

Table 15.3: Detailed mass breakdown for the TO-2250 aircraft.

MASS BREAKDOWN STATEMENT										
ITEM	MASS	CENTER OF GRAVITY				TECHNOLOGY FACTORS				
		X		Z						
	KG	M	KGM	M	KGM	1	2	3	4	5
1/WING PRIMARY STRUCTURE MASS	1835.	10.69	19615.	-0.74	-1355.	0.0078	0.5500	1.6750	-0.4500	-1.3250
2/WING FLAP SECTION 1 STRUCTURE MASS	129.	10.74	1388.	-0.57	-74.	129.2000	0.0000	1.0000	0.0000	1.0000
11/HORIZONTAL TAILPLANE BASIC MASS	187.	20.89	3898.	4.18	781.	2.8906	3.0213	193.7487	********	0.2000
12/FIN-MOUNTED STABILIZER MASS	19.	21.19	395.	4.26	79.	0.0000	0.1000	1.0000	0.0000	0.0000
21/VERTICAL TAILPLANE BASIC MASS	91.	18.64	1692.	2.38	216.	2.8906	3.0213	193.7487	********	0.2000
22/T-TAIL STRUCTURE MASS	29.	19.21	561.	3.20	93.	0.0000	0.0000	1.0000	0.1500	1.0000
31/FUSELAGE COMPARTMENT 1 FLOOR MASS	49.	6.41	315.	-1.40	-69.	0.0000	0.0000	1.0450	13.2800	0.7000
32/FUSELAGE COMPARTMENT 2 FLOOR MASS	34.	12.90	438.	-1.19	-41.	0.0000	0.0000	1.0450	20.0000	0.7000
36/FUSELAGE COMPARTMENT 6 FLOOR MASS	18.	1.96	35.	-0.56	-10.	0.0000	3.6960	1.0450	0.0000	0.7000
37/FUSELAGE COMPARTMENT 7 FLOOR MASS	164.	8.93	1464.	-0.56	-92.	0.0000	4.6200	1.0450	0.0000	0.7000
41/FUSELAGE STRINGERS AND LONGERONS MASS	160.	9.07	1456.	0.00	0.	0.0117	1.0000	1.4500	0.3900	0.3160
42/FUSELAGE STANDARD FRAMES MASS	119.	9.07	1077.	0.00	0.	0.0911	1.1300	0.0911	1.1300	0.0000
43/FUSELAGE SKIN MASS	449.	9.07	4069.	0.00	0.	0.0543	1.0700	0.7430	1.5950	2.1730
44/UNDERCARRIAGE 1 WHEELBAY MASS	53.	9.07	479.	0.00	0.	3.1800	0.0028	0.0000	0.0000	1.0000
45/UNDERCARRIAGE 2 WHEELBAY MASS	66.	9.07	602.	0.00	0.	0.0000	0.0000	0.5000	0.1250	0.8000
49/WING/FUSELAGE CONNECTION MASS	91.	9.57	868.	-0.69	-62.	20.4000	0.0009	1.0000	1.0000	0.0000
50/ENGINE/FUSELAGE CONNECTION MASS	58.	15.35	888.	0.23	13.	0.0000	0.0250	1.0000	0.0007	1.0000
51/DOORS AND WINDOWS 1 MASS	137.	9.07	1241.	0.00	0.	0.0000	4.3100	0.2500	0.5000	0.5000
52/DOORS AND WINDOWS 2 MASS	71.	3.94	282.	0.34	24.	22.3000	44.2000	0.5000	0.0000	0.0000
53/DOORS AND WINDOWS 3 MASS	45.	8.97	404.	0.37	17.	0.0000	23.9000	0.0000	0.0000	0.5000
54/DOORS AND WINDOWS 4 MASS	10.	9.72	95.	0.35	3.	14.9000	9.7650	0.0000	0.0000	0.0000
55/DOORS AND WINDOWS 5 MASS	47.	9.32	439.	-1.05	-49.	26.8000	32.2000	0.5000	0.0000	0.0000
61/UNDERCARRIAGE ASSEMBLY 1 MASS	87.	2.88	251.	-2.02	-176.	5.4000	0.0490	0.0000	0.0000	1.5000
62/UNDERCARRIAGE ASSEMBLY 2 MASS	*435.*	*11.56*	*5029.*	*-2.00*	*-870.*	*15.0000*	*0.0330*	*0.0210*	*0.0000*	*1.5000*
69/FUSELAGE BULKHEAD 4 MASS	12.	0.88	11.	-0.27	-3.	9.1000	7.2250	0.8000	1.2000	0.0000
70/FUSELAGE BULKHEAD 5 MASS	32.	14.83	468.	0.26	8.	9.1000	12.4800	0.8000	1.2000	0.0000
71/ADDITIONAL ITEMS (PAINT, JOINTS) MASS	18.	9.07	163.	0.00	0.	0.0000	0.0250	1.0000	0.0000	1.0000
81/MANEUVERING CONTROL SYSTEM MASS	183.	9.07	1659.	0.00	0.	0.0000	0.3180	0.6500	0.0000	0.0000
82/FLAP CONTROL SYSTEM MASS	20.	6.73	135.	-0.69	-14.	0.0000	11.0200	0.9200	0.0000	0.0000
84/STABILIZER CONTROL SYSTEM MASS	65.	3.21	207.	-0.70	-45.	0.0000	1.5200	0.8800	0.4400	0.8800
85/SPEEDBRAKES CONTROL SYSTEM MASS	336.	5.34	1796.	-0.37	-124.	40.4000	0.9200	0.6000	0.0000	0.0000
86/SPOILERS CONTROL SYSTEM MASS	134.	5.34	717.	-0.40	-54.	20.2000	0.9200	0.9200	0.0000	0.0000
87/FLIGHTDECK CONTROLS MASS	50.	5.34	267.	-0.09	-5.	0.0460	0.7500	50.0000	0.0000	0.0000
101/NACELLE COWLING 1 STRUCTURE MASS	61.	15.02	916.	0.24	15.	0.0000	12.2060	1.0000	9.1497	1.0000
102/NACELLE COWLING 2 STRUCTURE MASS	47.	15.02	703.	0.24	11.	0.0000	9.2766	1.0000	0.0000	1.0000
103/NACELLE COWLING 3 STRUCTURE MASS	73.	15.02	1096.	0.24	18.	0.0000	12.2060	1.0000	13.9553	1.0000
104/NACELLE COWLING 4 STRUCTURE MASS	12.	15.02	180.	0.24	3.	0.0000	9.2766	1.0000	0.0000	1.0000
114/PYLON 1 MASS	390.	15.02	5853.	0.24	94.	0.0000	39.0000	1.0000	0.0144	1.0000
121/DRY ENGINES MASS	983.	15.02	14773.	0.24	236.	0.0000	1.0000	1.0000	0.0000	0.0000
122/GEARBOXES AND DRIVES MASS	22.	15.02	332.	0.24	5.	0.0000	0.0100	1.0000	0.0000	0.0000
126/OIL SYSTEM AND COOLER MASS	10.	15.02	148.	0.24	2.	0.0000	0.0000	0.0000	0.0000	0.0000
128/FUEL SYSTEM MASS	209.	15.02	3139.	0.24	50.	0.0000	36.3000	43.6600	0.5000	0.3333
130/ENGINE CONTROLS MASS	21.	15.02	312.	0.24	5.	0.0000	0.0000	0.0000	0.0000	0.0000
131/STARTING SYSTEM MASS	23.	15.02	346.	0.24	6.	23.0000	0.0000	0.7600	0.7200	0.0000
141/AUXILARY POWER UNIT MASS	150.	19.61	2941.	0.64	96.	150.0000	0.0000	0.6000	0.0000	1.0000
142/FUEL INDICATION INSTRUMENTS MASS	5.	2.01	10.	0.00	0.	5.0000	0.0000	0.0000	0.0000	0.0000
143/PROPULSION INSTRUMENTS MASS	19.	2.01	38.	0.00	0.	9.5000	0.0000	0.0000	0.0000	0.0000
144/OTHER INSTRUMENTS MASS	41.	2.01	82.	0.00	0.	41.0000	0.0000	0.0000	0.0000	0.0000
145/AVIONICS MASS	60.	2.01	121.	0.00	0.	60.0000	0.0000	0.0000	0.0000	0.0000
146/HYDRAULIC SYSTEM MASS	153.	9.07	1384.	0.00	0.	91.0000	0.0035	1.0000	0.0000	0.0000
148/ELECTICAL SYSTEM MASS	423.	9.07	3834.	0.00	0.	0.0000	0.5650	1.0000	16.3000	0.0330
149/AIRCONDITIONING SYSTEM MASS	235.	9.07	2135.	0.00	0.	0.0000	5.3500	1.0000	0.0000	1.0000
150/ANTI-ICING SYSTEM MASS	94.	9.07	855.	0.00	0.	0.0000	2.2000	1.0000	0.0000	1.0000
1/MANUFACTURER'S EMPTY MASS	8263.	11.09	91605.	-0.15	-1268.	W/WTO = 46.9 %		CGX 63.9 %		CGZ -10.6 %
161/FLIGHTDECK ACCOMMODATIONS MASS	268.	2.01	538.	0.00	0.	0.0000	16.5000	0.2850	0.0000	0.0000
162/FLOOR COVERING MASS	64.	9.07	576.	0.00	0.	0.0000	1.2500	1.1500	0.0000	0.0000
163/SOUNDPROOFING AND INSULATION MASS	462.	9.07	4187.	0.00	0.	0.0000	6.1700	1.0700	0.0000	0.0000
165/GALLEY STRUCTURE AND PROVISIONS MASS	45.	3.29	148.	0.00	0.	0.0000	45.0000	1.0000	0.0000	0.0000
166/LAVATORY AND TOILET PROVISIONS MASS	75.	14.28	1071.	0.00	0.	0.0000	75.0000	1.0000	0.0000	0.0000
167/OXYGEN SYSTEM MASS	38.	9.07	341.	0.00	0.	13.6000	0.5440	1.0000	0.0000	0.0000
168/FIRE DETECTION AND EXTINGUISHING MASS	21.	9.07	192.	0.00	0.	0.0000	0.0012	1.0000	0.0000	0.0000
169/ESCAPE PROVISIONS MASS	22.	9.07	197.	0.00	0.	0.0000	0.4530	1.0000	0.0000	0.0000
171/ATTENDANT SEATS MASS	*13.*	*8.62*	*110.*	*0.00*	*0.*	*0.0000*	*6.4000*	*1.0000*	*0.0000*	*0.0000*
173/PASSENGER SEATS MASS (CLASS II)	440.	9.15	4028.	0.00	0.	0.0000	10.0000	1.0000	0.0000	0.0000
2/FURNISHING AND EQUIPMENT MASS	1446.	7.88	11388.	0.00	0.	W/WTO = 8.2 %		CGX -65.8 %		CGZ 0.0 %
3/BASIC EMPTY MASS	9709.	10.61	102993.	-0.13	-1268.	W/WTO = 55.1 %		CGX 44.6 %		CGZ -9.0 %
181/FLIGHT CREW PROVISIONS MASS	186.	2.01	374.	0.00	0.	0.0000	93.0000	1.0000	0.0000	0.0000
182/PASSENGER CABIN SUPPLIES MASS	279.	9.07	2535.	0.00	0.	0.0000	6.3500	1.0000	0.0000	0.0000
183/TOILET CHEMICALS MASS	21.	14.28	300.	0.00	0.	0.0000	21.0000	1.0000	0.0000	0.0000
184/SAFETY EQUIPMENT MASS	40.	9.07	362.	0.00	0.	0.0000	0.9070	1.0000	0.0000	0.0000
185/RESIDUAL FUEL/OIL MASS	40.	9.07	360.	0.00	0.	0.0000	15.1000	0.6667	0.0000	0.0000
186/CABIN CREW PROVISIONS MASS	130.	8.62	1121.	0.00	0.	0.0000	65.0000	1.0000	0.0000	0.0000
4/OPERATIONAL ITEMS MASS	696.	7.26	5050.	0.00	0.	W/WTO = 4.0 %		CGX -90.8 %		CGZ 0.0 %
5/OPERATIONAL EMPTY MASS	10405.	10.38	108044.	-0.12	-1268.	W/WTO = 59.1 %		CGX 35.5 %		CGZ -8.4 %
202/PASSENGERS MASS (CLASS II)	3300.	9.15	30211.	0.00	0.	0.0000	75.0000	1.0000	0.0000	0.0000
204/CARGO MASS HOLD 1	1298.	6.41	8323.	-1.05	-1361.	0.0000	200.0000	1.0000	0.0000	0.0000
205/CARGO MASS HOLD 2	427.	12.90	5502.	-0.95	-406.	0.0000	200.0000	1.0000	0.0000	0.0000
6/PAYLOAD MASS	5025.	8.76	44036.	-0.35	-1767.	W/WTO = 28.5 %		CGX -29.9 %		CGZ -24.3 %
7/MAXIMUM ZERO FUEL MASS	15430.	9.86	152080.	-0.20	-3035.	W/WTO = 87.6 %		CGX 14.2 %		CGZ -13.6 %
221/FUEL MASS WING TANK 1	3402.	9.88	33624.	-0.90	-3055.	0.0000	0.0000	0.0000	0.0000	0.0000
8/FUEL MASS	2153.	9.88	21502.	-0.90	-1953.	W/WTO = 12.4 %		CGX 15.4 %		CGZ -61.9 %
9/MAXIMUM TAKEOFF MASS	17583.	9.86	173610.	-0.22	-3868.	W/WTO = 100.0 %		CGX 14.4 %		CGZ -15.3 %



DESIGN POINT ⊗

WING LOADING	366. KG/M²
ASPECT RATIO	8.417
CRUISE ENGINE RATING	0.7217
ENGINE RATING AT MAXIMUM SPEED	0.8351
RANGE WITH MAXIMUM PAYLOAD	5000. KM
TAKEOFF FIELD LENGTH	1350. M
SECOND SEGMENT CLIMB GRADIENT	0.02400
LANDING DISTANCE	1095. M
RATE OF CLIMB AT SEA LEVEL	21.44 M/S
RATE OF CLIMB AT 7500 M	9.06 M/S
RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT	3.52 M/S
FUEL VOLUME RATIO	1.000
MAXIMUM TAKEOFF MASS	25860. KG
TAKEOFF MAXIMUM LIFT COEFFICIENT	2.247
THRUST LOADING	0.3200

ENGINES
SIZED FOR
TAKEOFF

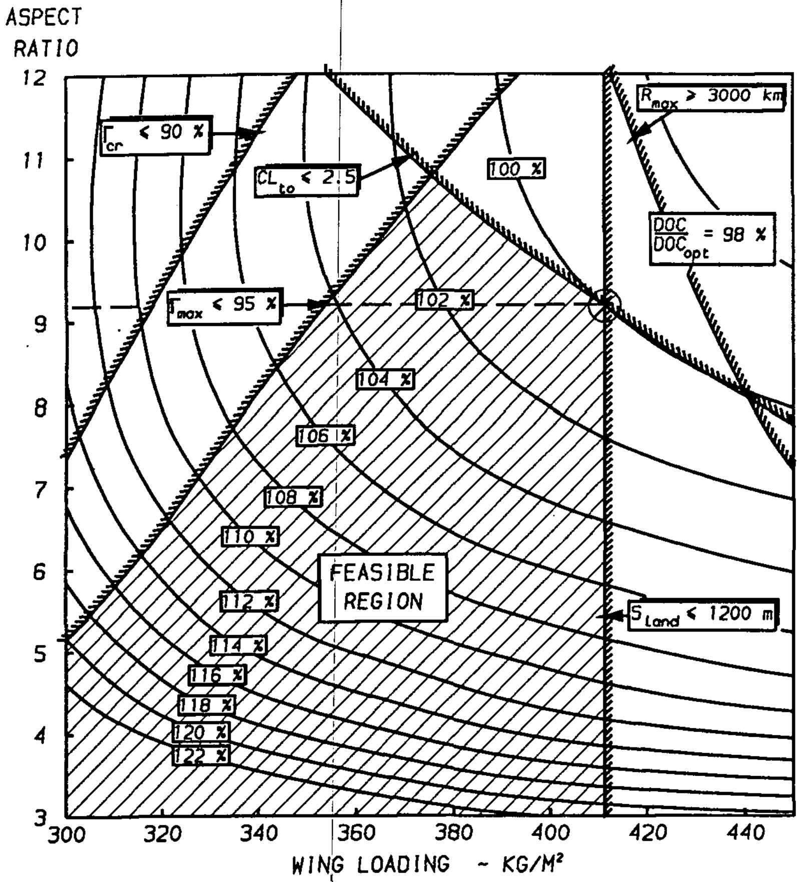
Figure 15.7: Optimization of wing loading and aspect ratio for minimum MTOW and with engines sized for takeoff and for optimum flap angle. The design equivalent range is 5000 km.

15.6 Effect of alternative merit functions.

In the design studies discussed previously, Maximum TakeOff Weight was selected as the merit function on the basis of which the design quality is evaluated. In general however several alternative merit functions can be defined and each may give a different optimum design point [Ref. 125]. Therefore, the definition of the merit function can be critical.

Direct Operating Cost (DOC) is one of the major aspects on which airlines compare competing design proposals as it reflects the aircraft's economic potential. However, DOC is difficult to assess, especially in the per-design stage. Contributions, such as development cost, production series and maintenance cost are still uncertain factors. Therefore, DOC estimation methods for conceptual design have to be simplified, as shown in Chapter 14.

To illustrate the effect of different merit functions on the optimum design, the proposed design has also been optimized for minimum Direct Operating Cost. Figure 15.8 and 15.9 give the results for the short (R_d = 2000 km) and long (R_d = 5000 km) range variant respectively. For short-haul aircraft, optimization for minimum MTOW or DOC gives approximately the same optimum design solution. The contribution of fuel cost in the total DOC is relatively minor in comparison to ownership and maintenance cost which are related to W_{to}. For long ranges, the fuel cost becomes dominant and the optimum design has a higher aspect ratio to reduce drag and fuel consumption.

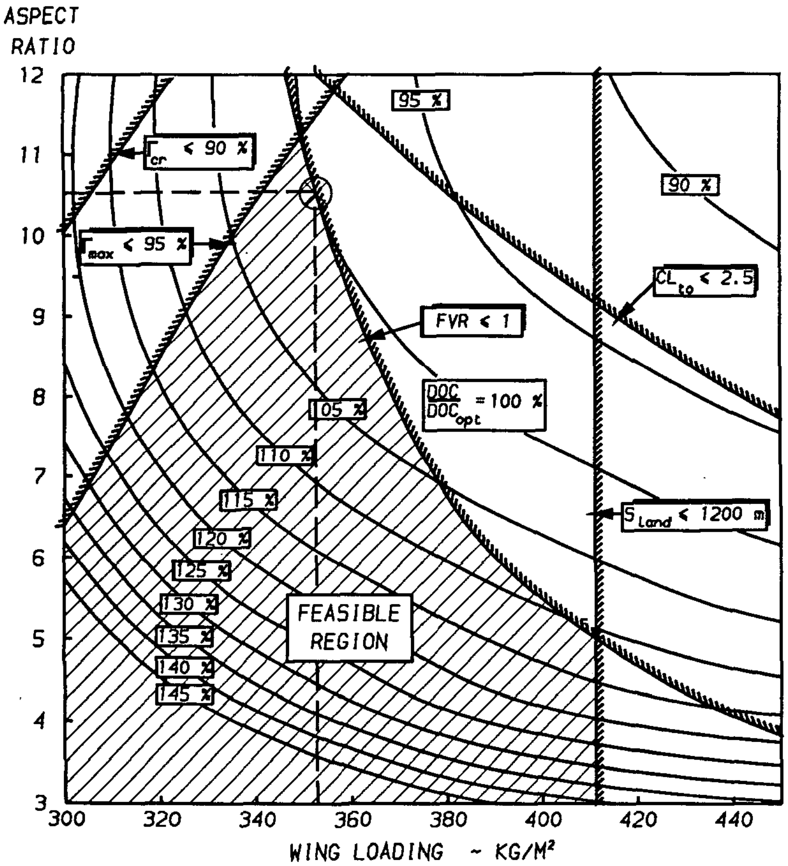


DESIGN POINT ⊗

WING LOADING	411. KG/M²
ASPECT RATIO	9.175
CRUISE ENGINE RATING	0.7256
ENGINE RATING AT MAXIMUM SPEED	0.8368
RANGE WITH MAXIMUM PAYLOAD	2000. KM
TAKEOFF FIELD LENGTH	1351. M
SECOND SEGMENT CLIMB GRADIENT	0.02400
LANDING DISTANCE	1200. M
RATE OF CLIMB AT SEA LEVEL	21.70 M/S
RATE OF CLIMB AT 7500 M	9.01 M/S
RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT	3.35 M/S
RANGE WITH MAXIMUM FUEL	3221. KM
MAXIMUM TAKEOFF MASS	17518. KG
TAKEOFF MAXIMUM LIFT COEFFICIENT	2.500
THRUST LOADING	0.3247
DIRECT OPERATING COST	2877.18 $/TRIP

ENGINES
SIZED FOR
TAKEOFF

Figure 15.8: Optimization of wing loading and aspect ratio for minimum DOC and with engines sized for takeoff and for optimum flap angle. The design equivalent range is 2000 km.



DESIGN POINT ⊗

WING LOADING	353. KG/M²
ASPECT RATIO	10.556
CRUISE ENGINE RATING	0.7957
ENGINE RATING AT MAXIMUM SPEED	0.9252
RANGE WITH MAXIMUM PAYLOAD	5000. KM
TAKEOFF FIELD LENGTH	1350. M
SECOND SEGMENT CLIMB GRADIENT	0.02400
LANDING DISTANCE	1066. M
RATE OF CLIMB AT SEA LEVEL	17.93 M/S
RATE OF CLIMB AT 7500 M	7.24 M/S
RATE OF CLIMB AT 4600 M ONE-ENGINE-OUT	2.70 M/S
FUEL VOLUME RATIO	1.000
MAXIMUM TAKEOFF MASS	26289. KG
TAKEOFF MAXIMUM LIFT COEFFICIENT	2.384
THRUST LOADING	0.2786
DIRECT OPERATING COST	6343.97 $/TRIP

ENGINES
SIZED FOR
TAKEOFF

Figure 15.9: Optimization of wing loading and aspect ratio for minimum DOC and with engines sized for takeoff and for optimum flap angle. The design equivalent range is 5000 km.

16. CONCLUSION

A pilot-system for computer-aided conceptual aircraft design, designated as the Aircraft Design and Analysis System (ADAS), has been developed and implemented on the Interfaculty CAD-Installation of the Delft University of Technology. Although this system is intended for design education and research in a university environment, it incorporates basic features and capabilities that are considered as representative for computer-aided engineering applications in industrial practice.

The development of ADAS is based on the philosophy that the experience, intuition and creativity of the designer form the important attributes that can produce an innovative design. Therefore, ADAS assigns a central role to the designer with respect to process control and decisionmaking, while computer applications are restricted to automation of standard and routine tasks and providing optional features that are usually not available in the traditional design environment, e.g. multivariate optimization. Essentially, ADAS provides the designer with a working environment, a computer infrastructure of hard- and software tools that can be employed to suit a specific purpose. In principle, ADAS is an 'empty shell' that can be 'filled' with design information and programmed to solve a particular design problem.

The ADAS-system has been successfully utilized in several student graduation projects. Subjects included propfan propulsion, canard/three-surface aircraft design and engine cycle analysis. Chapter 15 illustrates how ADAS can be applied in a typical design optimization study. Using a combination of explicit and implicit optimization, a baseline design was optimized with respect to a selected merit function. In addition, the effect of different design requirements and merit functions on the optimum design was also illustrated.

Although an acceptable, optimum design configuration was obtained, verification and validation remains difficult. This is inherent to the use of conceptual design methods. The analysis methods currently implemented in the program library are generally referred to as class I methods. One can find these types of methods in several textbooks on aircraft design. They are usually based on statistical/empirical considerations and are generally quite suitable for obtaining absolute 'ball park' answers with very little input information. However, caution must be taken when using these methods in automated design:

- These methods are generally inaccurate in predicting the sensitivity of individual parameters. Optimization algorithms in particular tend to capitalize on these deficiencies and may produce relatively large errors [Ref. 128];
- They are very dependent on a particular class of aircraft and are usually only valid for small variations in the design parameters;
- They are generally based on data of existing aircraft and so reflect old technology;
- They are generally not available for unconventional aircraft design concepts.

These disadvantages induce the tendency to apply more fundamental analysis methods earlier in the design process. However, these methods, such as panel methods, require far more CPU-time, which makes design optimization only practical if supercomputers are available. Instead, these methods can be used to verify the optimum design obtained in conceptual design and to adjust the statistical methods if necessary and to do a new optimization study, etc.

Hence, the first priority in the further development of ADAS is to conceive analysis methods that not only cover a wide spectrum of design disciplines and aircraft types, but also have different levels of accuracy, ease of usage, demand on CPU-time, etc. The concept of separating analysis methods and the generic design system presents the opportunity to modify the program library, while leaving the ADAS-system itself practically unchanged.

The implementation of a professional, relational database mangement system should be considered in the future. Although the ADAS-system operates sufficiently well with the simplified, special-purpose database system as currently implemented, a general-purpose database management system will improve interfacing with other, commercial systems and programs. Also, data presentation could be more uniform compared to the present situation where special-purpose print and plot routines have to be used. The development of an aircraft catalog with design data of existing aircraft will improve data lookup and design comparison.

Although the ICI-hardware with the MEDUSA CAD-software provided the necessary environment for the development of ADAS, it proved to be unsuited for running processes with heavy demand on CPU and disk I/O. With the availability of network technology, several special-purpose computers can be linked in a network. For example, personal computers and alphanumeric terminals for text editing, database query, etc. One level of computers with moderate computing power for interactive graphics applications and for running relatively small jobs, e.g. engineering workstations, and a top-level of (super)computers dedicated to the (batch)processing of CPU-intensive programs.

The current ADAS-version is a pilot-system that will be enhanced and expanded when new options and capabilities have to be accommodated. As new technologies emerge and alternative configurations are conceived to improve airvehicle design, so must the development of a design system keep equal pace and work is never done.

17. REFERENCES

[1] Abbott, I.H. and von Doenhoff, A.E.
Theory of wing sections.
Dover Publications, Inc., New York, June 1958.

[2] Ackland, T.G.
On osculatory interpolation, where the given values of the function are at unequal interval.
J. Inst. Actuar., 49, 369-375, 1915

[3] Afimiwala, K.A. and Mayne, R.W.
Evaluation of optimization techniques for applications in engineering design.
Journal of Spacecraft, Vol. 11, No. 10, Oct. 1974.

[4] Afimiwala, K.A. and Mayne, R.W.
Interactive computer methods for design optimization.
Computer Aided Design, Vol. 11, No. 4, July 1979.

[5] Alsina, J, Fielding, J.P. and Morris, A.J.
Progress towards an aircraft design expert system.
Computer Applications in Aircraft Design and Operation, Computational Mechanics Publications, Springer-Verlag, 1987.

[6] An.
U.S. Standard Atmosphere, 1962.
U.S. Government Printing Office, Washington D.C., December 1962.

[7] An.
Manual of the ICAO Standard Atmosphere, extended to 32 kilometers (105000 ft).
ICAO-Doc. 7488/2, 1964.

[8] An.
Standard method for estimating direct operating costs of turbine powered transport airplanes.
Air Transport Association of America, December 1967.

[9] An.
Introduction to MEDUSA.
MED9002/1/0, Cambridge Interactive Systems Ltd., March 1984.

[10] An.
MEDUSA modelling system user guide.
MED2001/4/0, Cambridge Interactive Systems Ltd., February 1985.

[11] An.
MEDUSA drafting system commands.
MED1002/4/0, Cambridge Interactive Systems Ltd., July 1985.

[12] An.
Reference manual for the MEDUSA DARS library.
MED5010/1/0, Cambridge Interactive Systems Ltd., August 1985.

[13] An.
Lost range, fuel and time due to climb and descent: Aircraft with turbojet and turbofan engines.
ESDU Data Sheets 74018, August 1974.

[14] An.
The NAG Fortran library Manual - Mark 12.
Numerical Algorithms Group Ltd., Oxford, United Kingdom, March 1987.

[15] Ardema, M.D. and Williams, L.J.
Automated synthesis of transonic transports.
AIAA paper 72-794, 1972.

[16] Arora, J.S. and Baenziger, G.
Use of AI in design optimization.
AIAA paper 85-0801, 1985.

[17] Ashley, H.
On making things the best - Aeronautical use of optimization.
AIAA paper 81-1738, 1981.

[18] Baker, A.N.
Distributed graphics system for computer augmented design and manufacturing.
AGARD CP-280, September 1979.

[19] Batdorf, W.J., Holliday, J.F. and Peed, J.L.
A graphics program for aircraft design-GPAD system.
AIAA paper 75-136, 1975.

[20] Baxter, T.G.
Aircraft interactive system management - From pencils to computers.
AIAA paper 86-2631, October 1986.

[21] Beltramo, M.N., Trapp, D.L., Kimoto, B.W. and Marsh, D.P.
Parametric study of transport aircraft systems cost and weight.
NASA CR-151970, April 1977

[22] Bernard, F.
CATIA: A computer-aided design and manufacturing tridimensional system.
ICAS paper 82-1.7.1., Seattle, 1982.

[23] Bil, C.
SKETCH: A computer program for plotting a transport airplane configuration in conceptual design.
LR-388, Faculty of Aerospace Engineering, Delft University of Technology, June 1983.

[24] Bil, C.
Applications of computer-aided engineering to subsonic aircraft design in a university environment.
AIAA paper ICAS-86-3.1.1., London, 1986.

[25] Bil, C.
ADAS command reference manual.
LR-98, Faculty of Aerospace Engineering, Delft University of Technology, November 1986.

[26] Bishop, A.W. and Page A.N.
An approach to design integration.
AGARD CP-147, October 1973.

[27] Blackburn, C.L., Storaasli, O.O. and Fulton, R.E.
The role and application of database management in integrated computer-aided design.
AIAA paper 82-0681, 1982.

[28] Boehm, B.W.
Developing interactive computer graphics for aerospace applications.
AIAA paper 69-954, September 1969.

[29] Boyles, R.Q.
Aircraft design augmented by a man-computer graphic system.
Journal of Aircraft, Vol. 5, No. 5, 1968.

[30] Britton, C.L. and Jenkinson, W.W.
A computer-aided design system for airplane configuration.
Computer Applications in Aircraft Design and Operation, Computational Mechanics Publications, Springer-Verlag, 1987.

[31] Brodlie, K.W.
Mathematical methods in computer graphics and design.
Academic Press, 1980.

[32] Burt, M.E. and Philips, J.
Prediction of fuselage and hull structure weight.
RAE Report Structures 122, April 1952.

[33] Carmichael R. and Gregory, T.
Interactive computer graphics why's, wherefore's and examples.
Astronautics and Aeronautics, April 1983.

[34] Carmichael R. and Gregory, T.
Interactive computer graphics getting the picture through computer graphics.
Astronautics and Aeronautics, April 1983.

[35] Chacksfield, J.E.
Multivariate optimization as applied to aircraft project design.
ICAS paper 84-4.8.2, 1984.

[36] Cheeseman, P. and Gevarter, W.
Introduction to artificial intelligence.
AIAA paper 86-0163, January 1986.

[37] Coen, P.G. and Willard, E.F. Jr.
Computer sizing of fighter aircraft.
Journal of Aircraft, Vol. 23, No. 5, 1986.

[38] Coles, W.A.
Use of graphics in an aircraft design office.
Computer Aided Design, Vol. 9, No. 1, January 1977.

[39] Crookes, W.A., Cooper, P.J. and Webb, F.
CAPS: A computer-based aid to aircraft project studies.
Computer-aided design, Vol. 6, No. 2, April 1974.

[40] Daues, J.J.
Project management issues & lessons learned from computer-aided design applications.
AIAA paper 87-2912, September 1987.

[41] De Filippo, R.
Aircraft synthesis using numerical optimization methodology.
AIAA paper 83-2458, October 1983.

[42] Diederich, F.W.
A simple approximate method for calculating spanwise lift distributions and aerodynamic influence coefficients at subsonic speeds.
NACA TN 2751, 1952.

[43] Divan III, P.E.
Aerodynamic optimization and analysis as part of the computer aided design process.
AIAA paper 78-97, January 1978.

[44] Dixit, C.S. and Patel, T.S.
Multivariate optimum design of a subsonic jet passenger airplane.
Journal of Aircraft, Vol. 17, No. 6, June 1980.

[45] Dongarra, J.J
Performance of various computers using standard linear equations software in a Fortran environment.
Technical Memorandum No. 23, Argonne National Laboratory, Mathematics and Computer Science Division, October 1987.

[46] Dror, B.
Development and application of computer-aided design technology at Israel Aircraft Industries.
Comput. & Graphics, Vol. 3, 1978.

[47] Dukes, R.
On the Symbolics artificial intelligence programming environment.
AIAA paper 86-0417, 1986.

[48] Eason, E.D. and Fenton, R.G.
A comparison of numerical optimization methods for engineering design.
Journal of Engineering for Industry, February 1974.

[49] Eckard, G.J., Rettie, I.H., Healy, M.J. and Kowalik, J.S.
Preliminary airplane design aided by mathematical optimization.
Mathematical programming study 11, North-Holland Publishing Company, 1979.

[50] Edwards, B.
The use of computer based optimisation methods in aircraft studies.
AGARD CP-280, September 1979.

[51] Elias, A.L.
Computer-aided engineering: the AI connection.
Aeronautics and Astronautics, July/August 1983.

[52] Feder, A.
Computer graphics create the new wave of design.
Astronautics and Aeronautics, February 1979.

[53] Fielding, J.P.
The teaching of aircraft design computer applications.
Computer Applications in Aircraft Design and Operation, Computational Mechanics Publications, Springer-Verlag, 1987.

[54] Fraher, M.J. and Monro, M.A.
A manufacturer's use of computer-aided design.
AIAA paper 68-1014, October 1968.

[55] Fulton, R.E., Sobieszczanski, J. and Landrum, E.J.
An integrated computer system for preliminary design of advanced aircraft.
AIAA paper 72-796, 1972.

[56] Fulton, R.E.
CAD/CAM approach to improving industry productivity gathers momentum.
Astronautics and Aeronautics, February 1982.

[57] Galloway, T.L. and Waters, M.H.
Computer-aided parametric analysis for general aviation aircraft.
SAE paper 73-0332, 1973.

[58] Galloway, T.L. and Smith, M.R.
General aviation design synthesis utilizing interactive computer graphics.
SAE paper 76-0476, 1976.

[59] Gerend, R.P and Roundhill, J.P.
Correlation of gasturbine engine weights and dimensions.
AIAA paper 70-669, June 1970.

[60] Gill, P.E. and Murray, W.
Numerical methods for constrained optimization.
Academic Press, 1974.

[61] Gill, P.E., Murray, W. and Wright, M.H.
Practical optimization.
Academic Press, 1981.

[62] Gregory, T.J. and Roberts, L.
An acceptable role of computers in the aircraft design process.
AGARD CP-280, September 1979.

[63] Gregory, T.J.
Computerized preliminary design at the early stages of vehicle definition.
AGARD CP-147, October 1973.

[64] Haberland, C., Thorbeck, J. and Fenske, W.
A computer augmented procedure for commercial aircraft preliminary design and optimization.
ICAS-84-4.8.1, Toulouse, 1984.

[65] Haberland, C. and Fenske W.
A computer-augmented procedure for commercial aircraft configuration development and optimization.
Journal of Aircraft, Vol. 23, No. 5, May 1986.

[66] Haberland, C., Fenske, W. and Grabert, M.
Konfigurationsoptimierung von verkehrsflugzeugen.
DGLR Jahrbuch paper 87-081, Institut fur Luft- und Raumfahrt, Technische Universität Berlin, 1987.

[67] Hague, D.S., Glatt, C.R. and Jones R.T.
Integration of aerospace vehicle performance and design optimization.
AIAA paper 72-948, September 1972.

[68] Hammond, D.L.
Graphics in conceptual design, a designer's viewpoint.
AIAA paper 86-2733, 1986.

[69] Harris, D.H.W.,
Applying computer aided design (CAD) to the 767.
Astronautics and Aeronautics, September 1980.

[70] Hatvany, J., Newman, W.M. and Sabin, M.A.
World survey of computer-aided design.
Computer Aided Design, Vol. 9, No. 2, April 1977.

[71] Healy, M.J., Kowalik, J.S. and Ramsay, J.W.
Airplane engine selection by optimization on surface fit approximations.
Journal of Aircraft, Vol. 12, No. 7, 1975.

[72] Hebb, H.S.
Artificial intelligence applications in aerospace.
SAE paper 871540, August 1987.

[73] Heldenfels, R.R.
Automating the design process: progress, problems, prospects, potential.
AIAA paper 73-410, March 1973.

[74] Heldenfels, R.R.
Integrated, computer-aided design of aircraft.
AGARD CP-147, October 1973.

[75] Herbst, W. and Ross, H.G.
Application of computer-aided design programs for the technical management of complex fighter development projects.
AIAA paper 70-364, 1970.

[76] Hitch, H.P.Y.
Computer-aided aircraft project design.
Aeronautical Journal, No. 469, 1977.

[77] Hoak, D.E. and Ellison, D.E.
USAF Stability and Control DATCOM.
Wright Patterson Air Force Base, October 1960 (Rev. August 1968).

[78] Inman, J.P.
Evolving business practices in a computerized design environment.
AIAA paper 86-2708, 1986.

[79] Jenkinson, L. and Wright, I.
Developments of teaching methods for computer-aided engineering (CAE) courses at undergraduate degree level.
AIAA paper 86-2634, October 1986.

[80] Jensen, S.C., Rettie, I.H. and Barber, E.A.
Role of figures of merit in design optimization and technology assessment.
Journal of Aircraft, Vol. 18, No. 2, February 1981.

[81] Kidwell, G. and Eskey, M.
Expert systems and their use in augmenting design optimization.
AIAA paper 85-3095, October 1985.

[82] Kidwell, G.H.
Workstations take over conceptual design.
Aerospace American, January 1987.

[83] Kirkpatrick, D.L.I. and Peckham, D.H.
Multivariate analysis applied to aircraft optimization - some effects of research advances on the design of future subsonic aircraft.
RAE Technical Memorandum Aero 1448, September 1972.

[84] Kirkpatrick, D.L.I.
Review of two methods of optimizing aircraft design.
AGARD LS-56, April 1972.

[85] Kirkpatrick, D.L.I. and Larcombe, M.J.
Initial design optimisation on civil and military aircraft.
AGARD CP-147, October 1973.

[86] Klomp, C.W., Braithwaite, W.W. and Gern, R.A.
The concept of an interactive graphic design system (IGDS) with distributed computing.
AIAA paper 75-966, August 1975.

[87] Kossira, H. and Pohl, H.
Der einfluss von tenerungsraten und kraftstoffpreisen auf den optimierten flugzugentwurf.
DGLR Jahrbuch paper 86-110, Institut für Flugzeugbau und Leichtbau, Technische Universität Braunschweig, 1986.

[88] Ladner, F.K. and Roch, A.J.
A summary of the design synthesis.
SAWE paper 907, 1972.

[89] Landgraf, S.K.
Some applications of performance optimization techniques to aircraft.
Journal of Aircraft, Vol. 2, No. 2, 1965.

[90] Lee, V.A. and Ball, H.G.
Parametric aircraft synthesis and performance analysis.
AIAA paper 66-795.

[91] Lee, V.A., Ball, H.G., Wadsworth, E.A., Moran, W.J. and McLeod, J.D.
Computerized aircraft design.
Journal of Aircraft, Vol. 4, No. 5, 1967.

[92] Littlefield, J.E. and Howard, R.L.
Computer graphics win over engineers.
Aerospace America, January 1987.

[93] Lootsma, F.A.
Algorithms for non-linear optimization.
Department of Mathematics and Informatics, Delft University of Technology, 1981 (lecture notes).

[94] Marinopolous, S., Jackson, D., Shupe, J. and Mistree, F.
Compromise: An effective approach for conceptual aircraft design.
AIAA paper 87-2965, September 1987.

[95] Maddalon, D.V.
A new method for estimating current and future transport aircraft operating economics.
NASA CR-145190, January 1978.

[96] McIntire, W.L. and Beam Jr., P.E.
Engine and airplane - will it be a happy marriage ?.
SAWE paper 910, May 1972.

[97] Mitchell, A.R.
Market supremacy through engineering automation.
Aerospace America, January 1987.

[98] Newberry, C.F. and Lord, P.A.
Aerospace vehicle design at California State Polytechnic University, Pomona.
AIAA paper 86-2637, 1986

[99] Newman, W.M. and Sproull, R.F.
Principles of interactive computer graphics.
McGraw-Hill, 1981.

[100] Palmer, D.
Influence of computer aids on engineering productivity.
AIAA paper 85-3099

[101] Peyton, R.S.
An aerodynamics model applicable to the synthesis of conventional fixed-wing aircraft.
SAWE paper 908, May 1972.

[102] Piggot, B.A.M. and Taylor, B.E.
Application of numerical optimization techniques to the preliminary design of transport aircraft.
RaeS, Technical Report 71-074, 1971.

[103] Purser, K.
Interactive computer graphics.
AIAA paper 80-1889, August 1980.

[104] Rau, T.R. and Decker, J.P.
ODIN - Optimal design integration system for synthesis of aerospace vehicles.
AIAA paper 74-72, 1974.

[105] Raymer, D.P.
CDS grows new muscles.
Aeronautics and Astronautics, June 1982.

[106] Raymer, D.P. and Albrecht S.K.
CDS - The designer's media, the analyst's model.
ICAS paper 82-1.7.2, Seattle, 1982.

[107] Raymer, D.P.
Developing an aircraft configuration using a minicomputer.
Aeronautics and Astronautics, November 1979.

[108] Roskam, J.
Aircraft design education at the University of Kansas.
AIAA paper 86-2636, 1986.

[109] Schemensky, R.T.
Development of an empirically based computer program to predict the aerodynamic characteristics of aircraft.
AFFDL-TR-73-144, Vol. I and II, November 1973.

[110] Schittkowski, K.
Nonlinear optimization codes.
Springer-Verlag, Berlin, 1980.

[111] Schuberth, E.R. and Celniker, L.
Synthesis sizing aircraft design.
Space/Aeronautics, April 1969.

[112] Silver, B. and Ashley, H.
Optimization techniques in aircraft configuration design.
SUDAAR 406, Department of Aeronautics and Astronautics, Stanford University, Stanford, California, June 1970.

[113] Silver, B.W.
Optimization studies in aircraft design.
SUDAAR No. 426, Department of Aeronautics and Astronautics, Stanford University, Stanford, California, June 1971.

[114] Simmons, D.M.
Nonlinear programming for operations research.
Prentice-Hall, 1976.

[115] Sliwa, S.M.
Use of constrained optimization in the conceptual design of a medium-range subsonic transport.
NASA TP-1762, December 1980.

[116] Sliwa, S.M.
Sensitivity of the optimal preliminary design of a transport to operational constraints and performance index.
AIAA paper 80-1095, 1980.

[117] Smyth, S.J.
CADAM data handling from conceptual design through product support.
Journal of Aircraft, Vol. 17, No. 10, 1980.

[118] Stack, S.H.
Computer-aided design system geared toward conceptual design in a research environment.
Journal of Aircraft, Vol. 19, No. 2, February 1982.

[119] Stepnieuwski, W.Z. and Kalmbach, C.F.
Multivariable search and its application to aircraft design optimization.
The Aeronautical Journal of the Royal Aeronautical Society, May 1970.

[120] Straub, W.L.
Managerial implications of computerized aircraft design synthesis.
Journal of Aircraft, Vol. 11, No. 3, March 1974.

[121] Talcott Jr., N.A.
A computer graphics display technique for the examination of aircraft design data.
AIAA paper 81-0370, January 1981.

[122] Thorbeck, J.
Ein beitrag zum rechner gestützten entwurf von verkehrsflugzeugen.
D-83, Fachbereich für Verkehrswesen, Technischen Universität Berlin, 1984.

[123] Tong, S.S.
Coupling artificial intelligence and numerical computation for engineering design.
AIAA paper 86-0242, 1986.

[124] Torenbeek, E.
Synthesis of subsonic airplane design.
Martinus Nijhoff/Delft University Press, 1976.

[125] Torenbeek, E.
Fundamentals of conceptual design optimization of subsonic transport aircraft.
LR-292, Faculty of Aerospace Engineering, Delft University of Technology, August 1980.

[126] Torenbeek, E.
Computer Aided Design as a discipline in aeronautical teaching and research at the Delft University of Technology.
M-472, Faculty of Aerospace Engineering, Delft University of Technology, 1983.

[127] Torenbeek, E.
Teaching and learning conceptual design in CAD era: generating problems and integrating solutions.
AIAA paper 85-3081, 1985.

[128] Vanderplaats, G.N.
Automated optimization techniques for aircraft synthesis.
AIAA paper 76-909, 1976.

[129] Vanderplaats, G.N.
Numerical optimization techniques for engineering design.
McGraw-Hill, 1984.

[130] Victor, K. and Weir, T.
Application of computers in the design process.
AIAA paper 85-3044, October 1985.

[131] Wallace, R.E.
Parametric and optimisation techniques for airplane design synthesis.
AGARD LS-56, April 1972.

[132] Wallace, R.E.
A computerized system for the preliminary design of commercial airplanes.
AIAA paper 72-793, August 1972.

[133] Wehrman, M.D.
CAD produced aircraft drawings.
Journal of Aircraft, Vol. 18, No. 7, July 1981.

[134] Wennagel, G.J., Mason, P.W. and Rosenbaum, J.D.
IDEAS, integrated design and analysis system.
SAE paper 68-0728, October 1968.

[135] Wilhite, A.
The Aerospace Vehicle Interactive Design System.
AIAA paper 81-0233, 1981.

[136] Wilhite, A. and Johnson, S.C.
Integrating computer programs for engineering and design.
AIAA paper 83-0597, 1983.

[137] Wittenberg, H.
Prediction of off-design performance of turbojet and turbofan engines.
AGARD CP-242, 1978.

[138] Whitaker, R.
The computer draughtsman.
Flight International, No. 19, June 1982.

[139] Wood, J.R., Kaib, C.E. and Packard, C.C.
The computer revolution.
AIAA paper 81-0876, May 1981.

[140] Zadarnowski, J.H.
CADD on the F-18 program.
Astronautics and Aeronautics, March 1980.

APPENDIX A: ADAS COMMAND SUMMARY

This Appendix gives a list of all available ADAS commands with their function descriptions.

* denotes specific ADAS commands which are only recognized at ADAS command level. + denotes primary MEDUSA 2D drafting commands. All other commands are operating system commands which can be specified at either command level. At MEDUSA command level an operating system command must be prefixed with an exclamation mark (!).

	Command	Description
	ABBREV	Define abbreviations for PRIMOS commands
	ATTACH	Select current working directory (UFD)
*	CARI	Draw a carpet plot with 2 plot functions
*	CARP	Draw a carpet plot with 1 plot function
	CHANGE_PASSWORD	Define login password
	CLOSE	Close a file
	CNAME	Change the name of a file
	COMINPUT	Execute a command input file
	COMOUTPUT	Open a command output file
*	CONM	Specify attributes of constraints
*	CONS	Specify bounds of constraints
*	CONT	Draw a contour plot
	COPY	Copy files
	CPL	Execute a CPL-program
	CREATE	Create a new sub-directory (UFD)

Command	Description
* DATA	Load plot data from a file
DATE	Display current time and data
* DBST	Draw database entries in a plot
* DDF	Change descriptions of database entries
DEFINE_GVAR	Activate a global variable file
DELETE	Delete a file or directory
DELETE_VAR	Delete a global variable
* DUMP	Display database entries in compact format (2 columns)
* DVAR	Specify bounds of plot variables
ED	Invoke the system editor
AERO	Invoke NLRAERO postprocessing program (0270)
F77	Invoke the FORTRAN 77 compiler
* FIT	Select curve fit technique
* FMON	Select optimization monitoring option
* FNAM	Specify attributes of survey functions
* FRAN	Specify bounds of plot functions
* FVAL	Specify contour values
* FVAR	Specify bounds of free variables
* FVNM	Specify attributes of free variables
* HELP	Display command usage descriptions
INDATA	Import files from the CDI to the ICI
JOB	Submit a CPL-program for batch processing
LD	List directory
LIST_ACCESS	List access rights
LIST_QUOTA	List disk quotes
LIST_VAR	List global variables
* LOAD	Load a database
LOGIN	Log into ICI
LOGOUT	Log out from ICI
LON	Select phantom logout notification option
+ MACRO	Execute a MEDUSA command file
* MEDF	Specify MEDUSA sheet name
MEDUSA	Invoke the MEDUSA system
MESSAGE	Send a message to another user

Command	Description
* OBJ	Specify attributes of objective function
* OPT	Select optimization mode
ORIGIN	Select home directory
OUTDATA	Export files from ICI to CDI
* PAR	Select parametric survey mode
PHANTOM	Execute a phantom (background) process
* PICK	Indicate a point in a contour plot
* POLAR	Copy a parametric table into aerodynamic polar
* PVAR	Specify bounds of survey parameters
* PVNM	Specify attributes of survey parameters
* QF	Display database entry by name
* QL	List database entries by index
* QN	Relative move in database
* QU	Change database entry value
* QUIT	Leave ADAS system
RESUME	Invoke an executable file (EPF)
RSPOOL	Spool a file to CDI-printer
+ SAV	Save a MEDUSA drawing in a file
* SAVE	Save the current database
SET_ACCESS	Set access rights
SET_QUOTA	Set disk quotes
SET_VAR	Set global variables
+ SHE	Retrieve an existing MEDUSA drawing for editting
SIZE	Display file size
SLIST	List a file
SPOOL	Spool a file to the ICI-printer
STATUS	List system status
* SURF	Draw an isometric surface plot
* TABLE	Print a parametric data table
TERM	Select terminal characteristics
TIME	Display time-used statistics
* U$ADAP.CPL	Invoke the ADAP program
* U$LIFT.BAS	Draw a lifting surface planform into MEDUSA sheet
* U$MEDB.CPL	Submit MEDUSA modeller and/or viewer for batch processing

Command	Description
* U$SEAT.BAS	Draw passenger seats into a MEDUSA sheet
USAGE	Display system usage statistics
USERS	Display number of users currently logged into ICI

APPENDIX B: THE MEDUSA - ADAS INTERFACE PROTOCOL

To create a new ADAS-compatible drawing, the user can select a standard empty MEDUSA-sheet. The drawing area is divided into 3 separate viewboxes in which the configuration layout can be defined by drawing 3 orthogonal views. The geometric coordinate-system is defined as follows (Figure B-1):

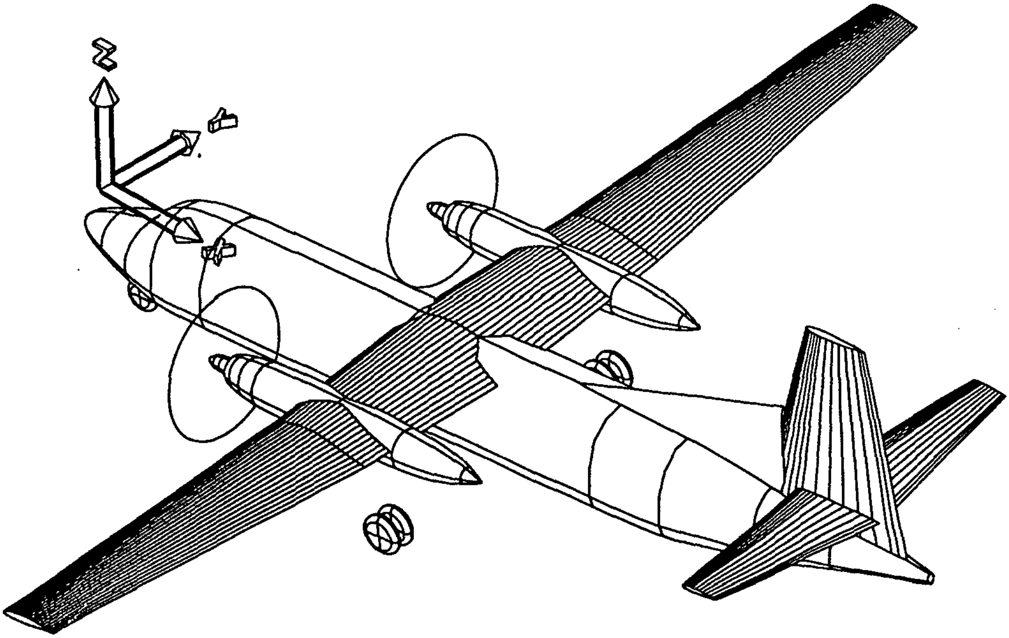


Figure B-1: Aircraft geometric axis system definition.

x-coordinate: positive rearward with x = 0 at fuselage nose

y-coordinate: positive starboard with y = 0 at center line

z-coordinate: positive upward with z = 0 at waterline through half fuselage height.

The recommended arrangement of views is the front, side and top view in the left, middle and right viewbox respectively. The origin position (0,0) of each view is indicated by placing the appropriate orientation prim in the corresponding viewbox, as shown in Figure B-2:

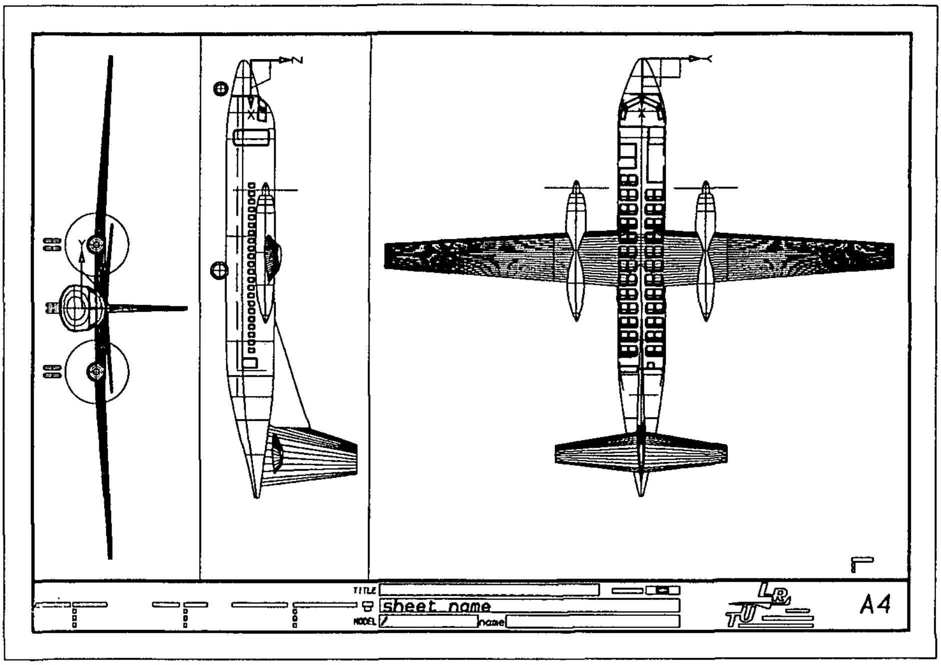


Figure B-2: Recommended viewbox arrangement for configuration drawings.

The viewbox dimensions and the sheet scale factor should be selected such that the views fit in their respective viewboxes. Although drawing coordinates are usually defined in millimeters (default), the internal representation of lengths and distances in ADAS is in meters, hence:

$$F_{scale_{ADAS}} = F_{scale_{MEDUSA}}/1000 \qquad \text{(B-1)}$$

After a convenient sheet layout has been prepared, the configuration geometry can be drawn using the standard MEDUSA drafting commands [Ref. 11]. However, in order for ADAS to interpret the drawing correctly, some simple drawing conventions have to be observed. These drawing rules will be discussed in the following Sections for each individual aircraft component.

B-1: Lifting surfaces planforms

A lifting surface is composed of one or more planar panels with trapzoidal planform and with airfoil-shaped cross-sections. Lifting surfaces are e.g. wings, tailplanes, canards and pylons.

A horizontal or vertical lifting surface is defined by drawing the planform outline in the top or side view respectively. The first point is at the apex and the point sequence is clockwise (Figure B-3):

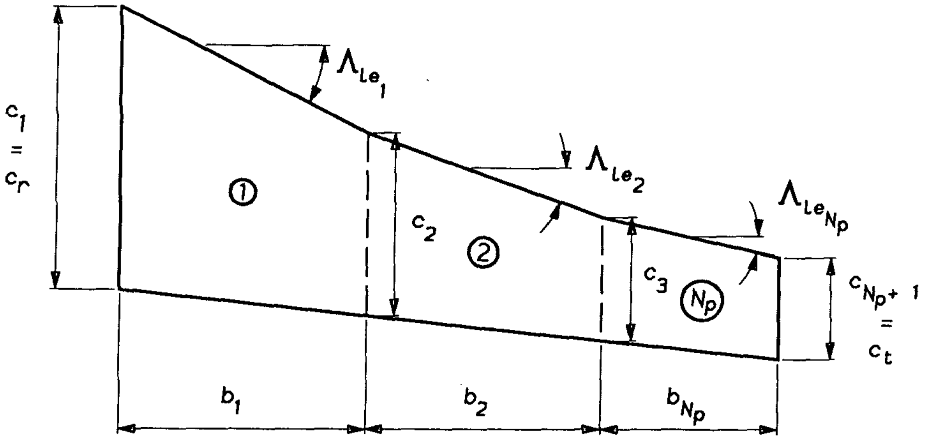


Figure B-3: Lifting surfaces planform definition.

Each point in the leading-edge requires a corresponding point in the trailing edge, hence there must always be an even number of points in the planform outline. The current limitations in ADAS are fixed at 10 lifting surfaces and upto 5 panels per lifting surface.

B-2: Airfoil selection and positioning

A lifting surface must be associated with airfoil sections for the root, tip and each intermediate kink. An airfoil section is defined by placing a TEXT-element representing the airfoil name in the appropriate viewbox, i.c. in the side viewbox for the wing and horizontal tailplane and in the top viewbox for the vertical tailplane. An airfoil designation must be prefixed with an integer value indicating its sequence number, increasing from root to tip, in the lifting surface, e.g. 1/NACA-63-415. Incidence and geometric twist (ϵ) can be defined by rotating a TEXT-element about its datum point. The relative locations of the TEXT-elements determine dihedral or anhedral angles (Γ), as shown in Figure B-4:

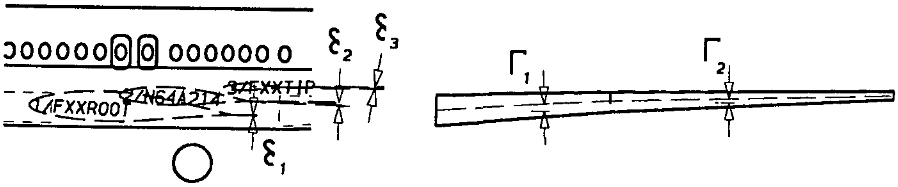


Figure B-4: Airfoil selection and positioning.

If the interface program encounters a valid airfoil name in the drawing, it will subsequently load the corresponding aerodynamic and geometric data files from the UFD, c.q. the airfoil library VL>VO>ADAS>AIRFOIL (see Appendix D). Obviously, these files must exist at run-time.

In order to obtain a convenient distribution of points along the airfoil contour, an automatic rearrangement of contour points is performed by interpolation according to the cosine-distribution for the non-dimensional chord-stations:

$$\frac{x_i}{c} = \frac{1 - \cos\frac{\pi(i-1)}{30}}{2} \qquad \text{with } i = 1,31 \qquad \text{(B-2)}$$

for 31 points in the upper and lower airfoil contour.

This distribution is specifically recommended for aerodynamic calculations because of the higher density of points at the leading- and trailing-edge, as shown in Figure B-5:

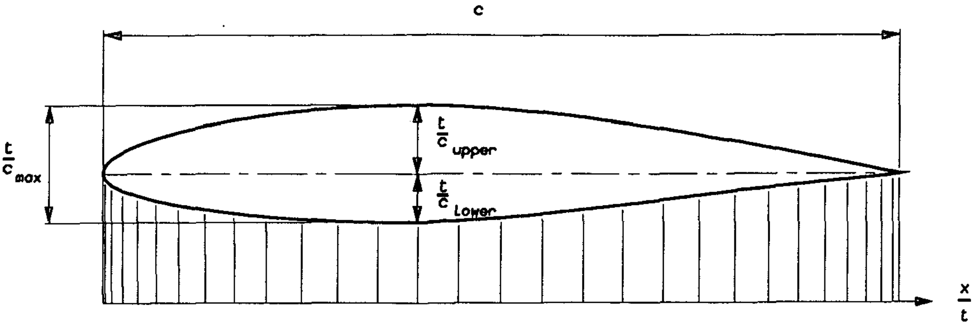


Figure B-5: Airfoil contour digitization.

In addition, data reduction is achieved as the chord-stations according to Eq. (B-2) are equal for all airfoil sections in the design. Assuming linear line segments between the contour points, the circumference and enclosed area can be readily computed, e.g. for calculating wetted area and fuel tank volume respectively (see Section 8.1.1 and 13.2.3).

If the D$MEDP-routine is used, the airfoil contour line, as defined by Eq. (B-2), is automatically drawn into the MEDUSA sheet, scaled to the correct chord length. The root and tip airfoils of each panel are connected by a LRS link line to obtain a ruled surface model from the MEDUSA solid modeller.

B-3: High-lift devices and control surfaces

Currently defined control surfaces are ailerons, elevator and rudder. High-lift devices are wing trailing-edge and leading-edge flaps. High-lift devices may be divided into separate flap sections and each flap section may in turn be composed of upto 5 flap elements. The geometry of high-lift devices and control surfaces are defined by drawing the surface outline in the corresponding lifting surface planform as shown in Figure B-6:

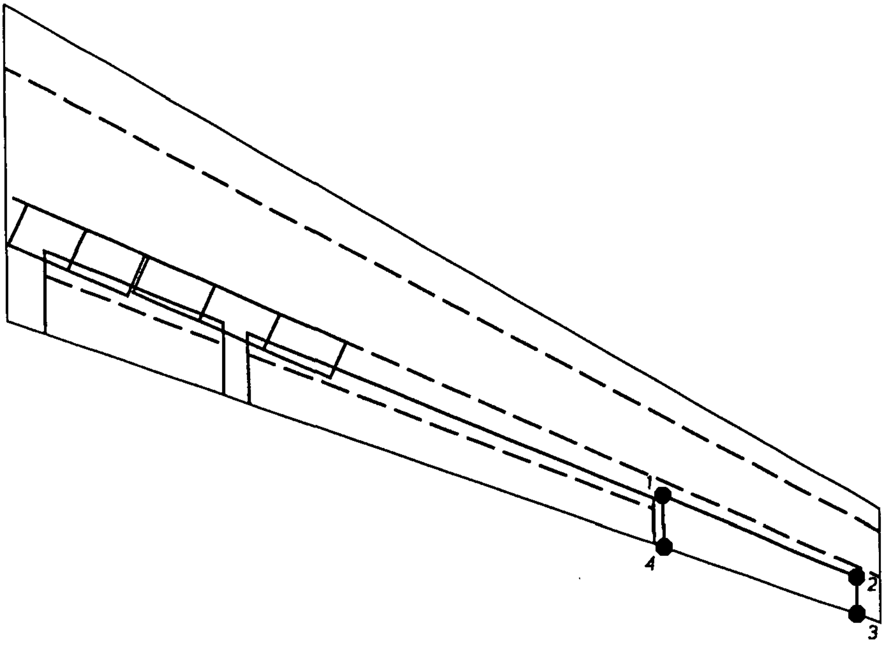


Figure B-6: Control surfaces and high-lift devices geometry definition.

Each surface is defined by 4 points, which must be orientated in the clockwise direction and starting at the left upper corner.

B-4: Wing fuel tanks geometry

Wing fuel tanks are defined by drawing the tank outline into the wing planform in the top viewbox. Upto 3 separate fuel tanks may be defined, each may in turn be composed of upto 4 trapzoidal shaped tanks sections. An example is given in Figure B-7.

ADAS determines the location of the fuel tanks within the wing planform and can calculate e.g. the fuel tank volume and fuel center of gravity (see Chapter 13). Each point in the tank outline may be arbitrarily placed within the wing planform. If the tank endface is not placed in the streamwise direction, an average spanwise location is used.

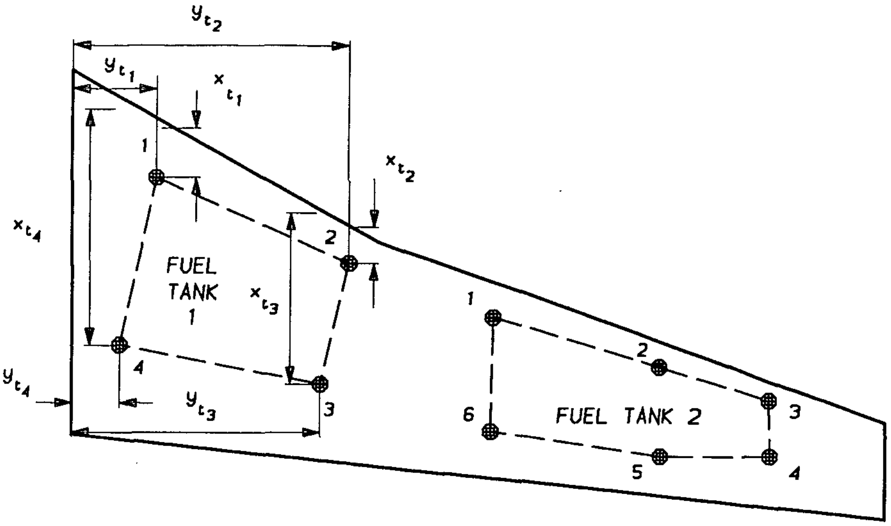


Figure B-7: Wing fuel tank geometry definition

If ADAP is used in either parametric survey mode or optimization mode and the configuration geometry is changed in an automatic fashion, provisions have to be incorporated to assure that aircraft components remain logically related. For example, if the wing location is changed, wing fuel tanks, wing-mounted undercarriage and engines must also be moved to assure that the geometric model remains coherent. This can be achieved by including relationships which connect the position and dimensions of a component to those of another. Proportional scaling rules may be developed by making dimensions and locations non-dimensional. These equations however, have to be included in the analysis program.

B-5: Fuselage geometry

The fuselage shape and dimensions are defined by drawing 4 profile lines in top and side view, as shown in Figure B-8:

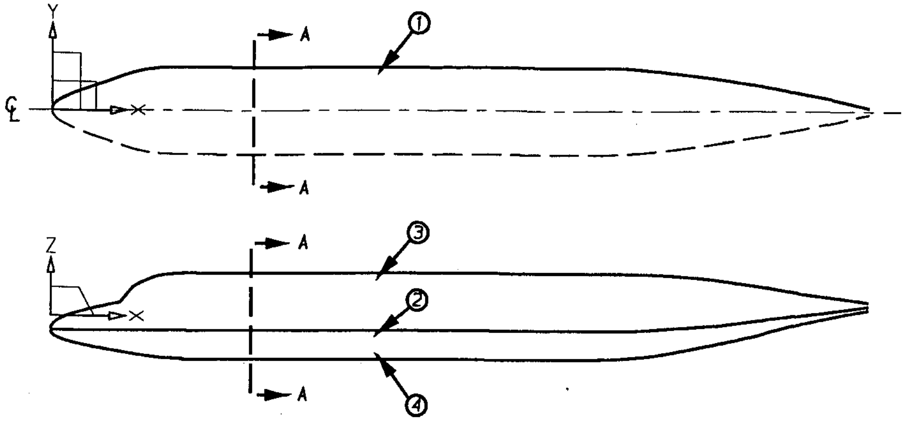


Figure B-8: Fuselage geometry definition

①② maximum width line in top and side view respectively, represents the maximum width location at any fuselage station;

③ upper line in side view, represents the distance between the reference line and the back line of the fuselage;

④ lower line in side view, represents the distance between the reference line and the keel line of the fuselage.

A fuselage profile line runs downwards, i.c. from the fuselage nose to the tail, and may contain linear as well as conical line segments (max. 100 points).

A conical equation is used by MEDUSA to fit a smooth curve through a sequence of points. A conical line segment is a curve between 3 consecutive points (x_i, y_i) i = 1,3, where the outer points are the endpoints of the curve and the middle point is a control point, used to define the shape of the curve, as shown in Figure B-9:

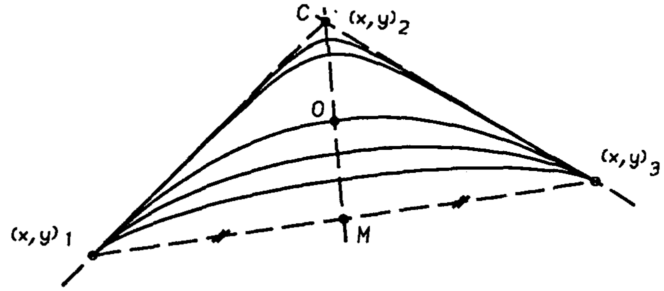


Figure B-9: Conical curve construction.

The mathematical representation of the conical curve in its parametric form is:

$$\binom{x}{y} = \frac{\binom{y_1}{y_1} t^2 + 2 \binom{x_2}{y_2} t(1 - t)W + \binom{x_3}{y_3}(1 - t)^2}{t^2 + 2t(1 - t)W + (1 - t)^2} \tag{B-3}$$

In geometrical terms, the control point C is the intersection of the tangential lines through the endpoints. The associated weight factor $W > 0$ defines the intersection point 0 along the line $\overline{CM}$ thereby manipulating the shape of the curve. The corresponding weight factor $W = \frac{\overline{OM}}{\overline{CO}}$. To interpolate for a given x-value $x_1 \leq x \leq x_3$ the correspondinge can be obtained by solving the quadratic polynomial, as given in Eq. (B-4):

$$(x_1 - x)t^2 + 2t(1 - t)W(x_2 - x) + (x_3 - x)(1 - t^2) = 0 \tag{B-4}$$

This yields two answers for t, but only $0 \leq t \leq 1$ is valid. The interpolant follows by substitution of t into Eq. (B-3):

$$y = \frac{y_1 t^2 + 2y_2 \, t(1 - t)W + y_3(1 - t)^2}{t^2 + 2t(1 - t)W + (1 - t)^2} \tag{B-5}$$

The ADAP-program does not retain the fuselage geometry description in this form. Instead, the fuselage length is divided into 50 equidistant fuselage stations, for which the local cross-sectional dimensions are computed with the interpolation technique described above.

At each fuselage station, the intersection points of the 4 profile lines with the cross-sectional YZ-plane for x = constant, define the maximum local dimensions, but the actual cross-sectional shape still needs to be determined. This is done by representing the outer periphery of the cross-section by 2 different conical curves for the upper and lower lobe respectively, as illustrated in Figure B-10:

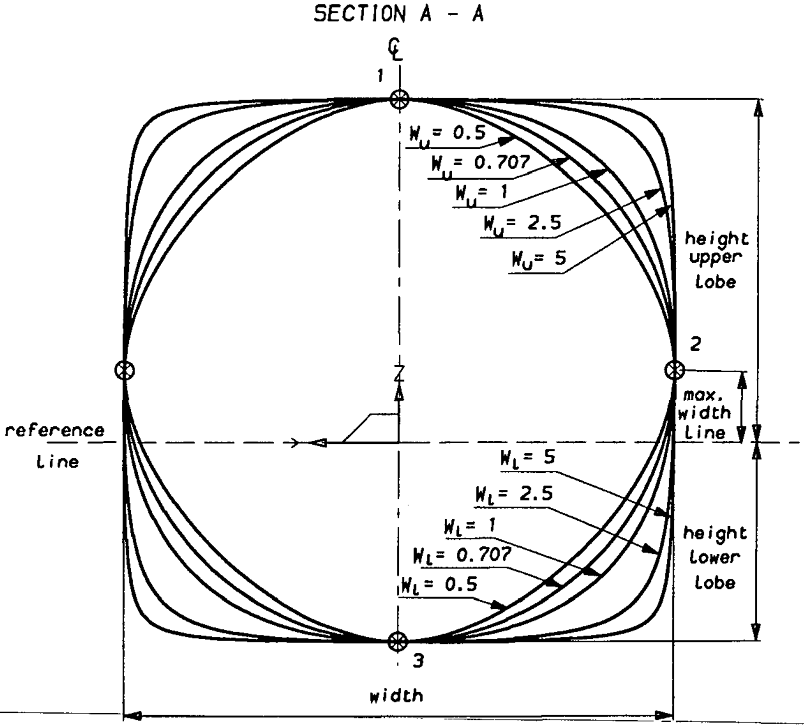


Figure B-10: Fuselage cross-sectional shape definition.

The cross-section is assumed symmetrical with respect to the Z-axis, i.c. plane of symmetry.

The position of the control points is fixed as the tangents at points 1, 3 and point 2 are assumed horizontal and vertical respectively. The only means available to control the cross-sectional shape are the 2 weight factors W_u and W_1.

The designer can draw a representative fuselage cross-section in the front viewbox, but the interface program will only read the two respective weight factors. These weight factors are assumed constant for all the cross-sections. This may cause some differences in the estimated and desired fuselage model, especially near the flight deck windshield, but this is considered acceptable for conceptual design purposes. Moreover, additional and arbitrary cross-sections may be defined afterwards with MEDUSA to obtain a more accurate geometry description with the solid modeller.

Of each cross-section the local circumference C_i and area A_i are numerically computed. The total fuselage volume and wetted area follow from summation, i.c.:

$$V_f = \sum_{i=1}^{i=50} \frac{(A_i + A_{i+1})\ \Delta x}{2} \qquad \text{(B-6)}$$

and

$$S_f = \sum_{i=1}^{i=50} \frac{(C_i + C_{i+1})\ \Delta x}{2} \qquad \text{(B-7)}$$

B-6: Doors and windows

Apertures in the fuselage hull, such as for doors and windows, are defined by a closed line in the side viewbox, including those on the starboard side of the aircraft. Different types of doors an windows, e.g. entrance doors, escape hatches, emergency exits, service doors, etc., may be logically grouped together by placing them in the same layer.

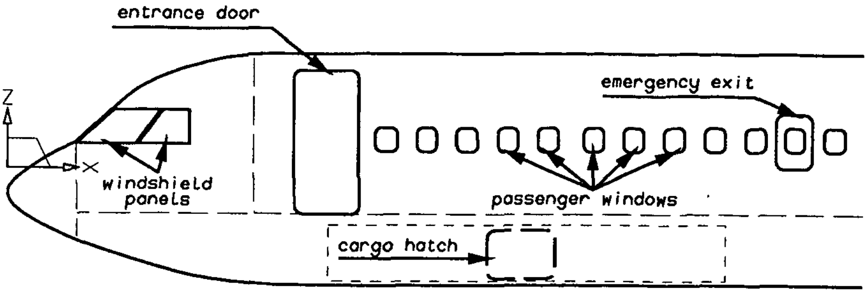


Figure B-11: Doors and windows geometry definition.

Upto 5 of such groups may be defined. For each group, the total (projected) area and the centre of area is computed.

A somewhat different procedure is followed for the flight deck windshield. A windshield may be composed of upto 5 separate panels on each side, drawn in the side viewbox, e.g. Figure B-12:

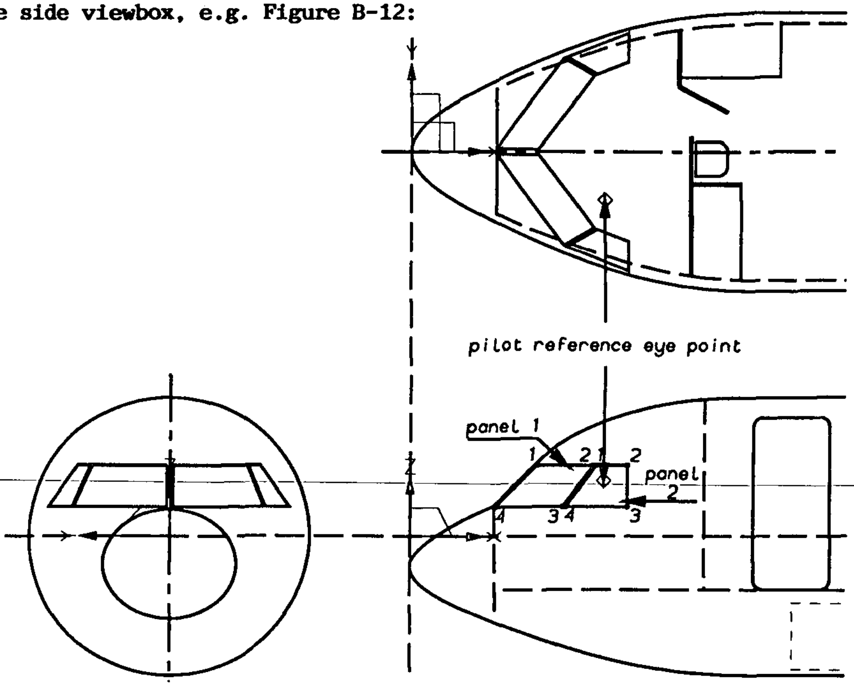


Figure B-12: Flightdeck windshield geometry definition.

Each panel consists of 4 points and is assumed to be a flat surface, hence these points must lie in one plane in the 3-dimensional space.

The interface program uses the windshield definition in side view and the local shape of the fuselage, to construct a 3-dimensional geometrical description of the panels. Figure B-12 also gives the top view and front view representation of the windshield. These are automatically drawn in the MEDUSA sheet if the D$MEDP-routine is used.

The procedure proceeds as follows. The program assumes a support strut in the plane of symmetry with a fixed width, say 3 inches. This fixes the position of points 1 and 4 of the first panel. Next, the Y-coordinate of point 3 is computed from the local fuselage geometry. Points 1, 3 and 4 define a plane in space. Finally, the y-coordinate of point 2 can be computed as it must also lie in this plane:

$$y_2 = y_4 + R_1 \left(y_3 - y_4\right) + R_2 \left(y_1 - y_4\right) \tag{B-8}$$

where

$$R_1 = \frac{\Delta z_1 \left(x_2 - x_4\right) + \Delta x_1 \left(z_4 - z_2\right)}{\Delta x_2 \Delta z_1 - \Delta x_1 \Delta z_2} \tag{B-9}$$

$$R_2 = \frac{z_2 - z_4 - R_1 \Delta z_2}{\Delta z_1} \tag{B-10}$$

$$\Delta x_1 = x_1 - x_4 \tag{B-11}$$

$$\Delta x_2 = x_2 - x_4 \tag{B-12}$$

$$\Delta z_1 = z_1 - z_4 \tag{B-13}$$

and

$$\Delta z_2 = z_3 - z_4 \tag{B-14}$$

For any following panel $i = 2, N_{ws}$, both y_3 and y_4 are derived from the local fuselage geometry. The lateral location of point 1 is estimated from the location of point 3 and 4 of the preceeding panel:

$$y_{1i} = y_{4i} - (y_3 - y_2)_{i-1} \qquad \text{(B-15)}$$

Finally, point 3 is computed with Eq. (B-8). This procedure repeats for all panels in the drawing.

The total and frontal area of each panel are computed as follows. The total area A_t is:

$$A_t = \sqrt{E_{21}E_{32} - \frac{(E_{21} + E_{32} - E_{31})^2}{4}} + \sqrt{E_{41}E_{43} - \frac{(E_{41} + E_{43} - E_{31})^2}{4}} \qquad \text{(B-16)}$$

where $\sqrt{E_{ij}}$ is the Euclidean distance between point i and j:

$$E_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 \qquad \text{(B-17)}$$

To compute the frontal area, the same procedure is used, except that $x_i = 0$ for $i = 1,4$.

The procedure outlined above generally gives an acceptable windshield geometry representation for transport type aircraft. For general aviation and executive/business aircraft the result may be unrealistic as these aircraft usually have curved windshields. In ADAS, this problem can be overcome by approximating a curved panel with a number of smaller flat panels.

B-7: Fuselage internal arrangement

The designer can define the internal fuselage arrangement by indicating fuselage compartments, e.g. passenger cabin, flight deck, cargo holds, and cabin furnishings, such as passenger seats, attendant seats, flight crew seats, lavatories and gallies.

Fuselage compartments can be defined by drawing the compartment outline in side view. Upto 10 separate compartments may be defined and each line must be placed in a different layer. An outline in side view is a closed line with upto 20 points. There must be a pair of points at each selected fuselage station, as shown in Figure B-13:

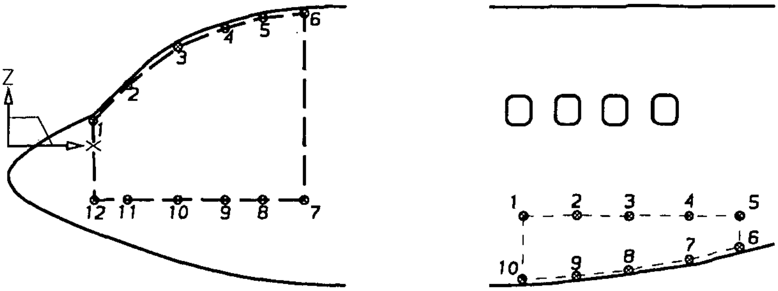


Figure B-13: Fuselage compartment geometry definition (full fuselage width).

ADAS assumes compartments thus defined to cover the total aircraft width. If this is not the case, e.g. for above-floor cargo holds, an additional line has to be drawn in top view. An example is given in Figure B-14:

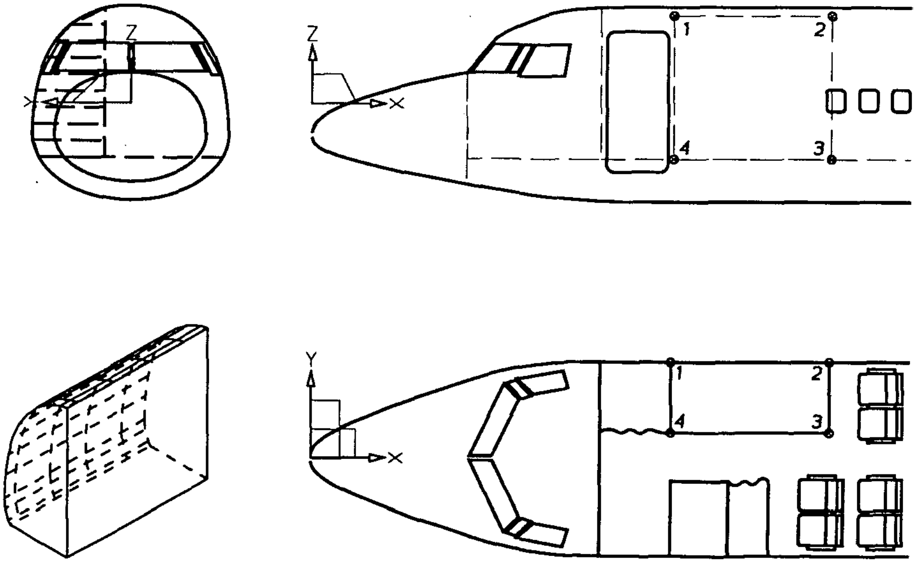


Figure B-14: Fuselage compartment geometry definition (partial fuselage width).

The MEDUSA - ADAS interface will automatically compute the dimensions, volume and floor area of each compartment.

Flight deck and passenger cabin furnishings can be defined by placing a PRIM- or TEXT-element in the top viewbox. Each item type must be placed in a separate layer. An example is given in Figure B-15:

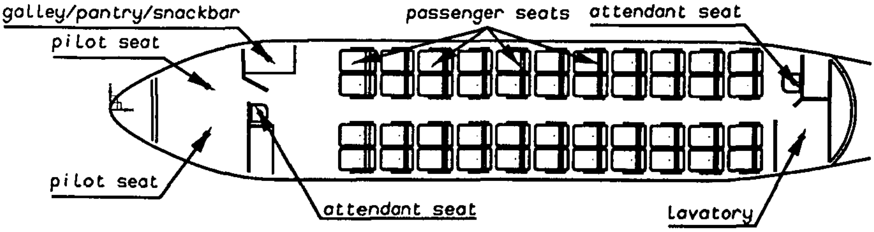


Figure B-15: Fuselage furnishings and equipment.

A procedure U$SEAT.BAS has been developed which can be used under MEDUSA to automatically draw a group of seats into the sheet. The use of this procedure is particularly useful if a large number of seats have to be drawn. U$SEAT.BAS is interactive and prompts the user for information, i.c.:

- The name of the symbol file Y.<seat> wherein the seat top view drawing is contained;
- The number of seats abreast;
- Seat spacing, i.c. the lateral distance between single seats. For dual or triple seats, the user should specify half-seat width + armrest. This will cause one, joined armrest for two seats.
- The number of seat rows;
- The seat pitch.

After this information has been specified, the procedure will load the given file Y.<seat> and sequentially draw the seats, starting from a given starting point, as shown in Figure B-16.

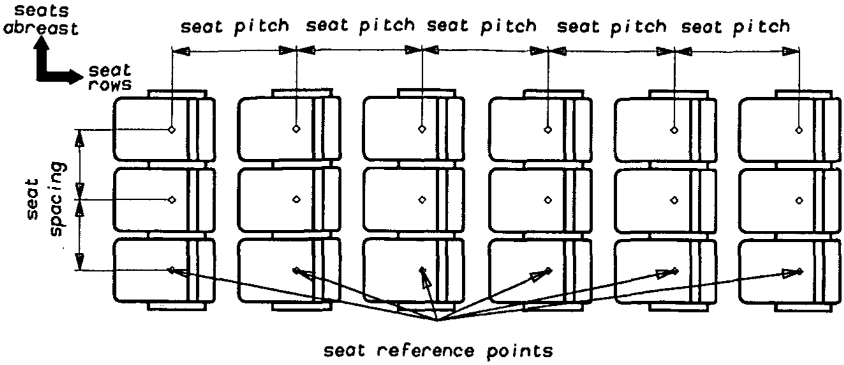


Figure B-16: Passenger seat arrangement.

Note that at present, ADAS does not use the actual dimensions of the furnishing equipment,only the relative location of the corresponding PRIM- or TEXT-elements. The user must make sure however, that the dimensions of the selected furnishing equipment are such that they fit within the fuselage internal contours.

B-8: Engine selection and positioning

Analoguous to airfoils (see Section B-1), the engine nacelles geometry reside outside ADAS in a special directory VL>VO>ADAS>ENGINE, referred to as the engine library. An engine is associated with a design by placing a TEXT-element with the engine designation into the top and side viewbox of the drawing. The text datum point defines the location of the engine highlight center point, while the text may be rotated to define engine tilt and toe angles. The user can define upto 3 engine groups. An engine group is either 2 engines symmetrically placed with respect to the XZ-plane, which is automatically assumed if an engine is placed off-set to the XZ-plane, or 1 engine placed in the plane of symmetry. Thus, upto 6 engines may be fitted to an aircraft, but at this time they have to be of the same type.

Engines may be mounted either to the wing or to the fuselage. This can be indicated by setting the FLAG-attribute of the engine TEXT-element as follows. If the FLAG-attribute is UNSET the corresponding engine group is assumed to be wing-mounted. Figure B-17 gives a typical layout of wing-mounted engines:

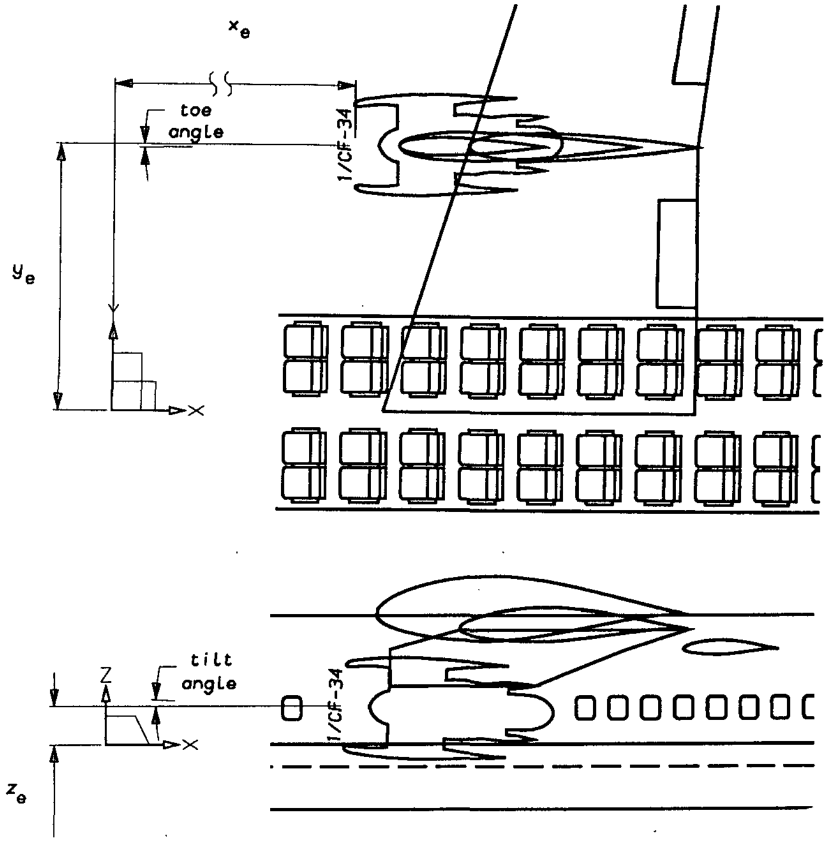


Figure B-17: Definition of wing-mounted engines.

If the FLAG-attribute is SET the engine group is assumed to be fuselage-mounted, as shown in Figure B-18. For both configurations the engine location is computed relative to the origin of the geometric axis-system. Setting the FLAG-attribute does not effect the engine positioning as drawn, but it can be used as an indicator in e.g. weight analysis programs.

The D$MEDP-routine will automatically draw the nacelle profile line into the MEDUSA sheet. Also, a LVR link line is drawn to obtain a volume of revolution representation for the engines from the MEDUSA solid modeller.

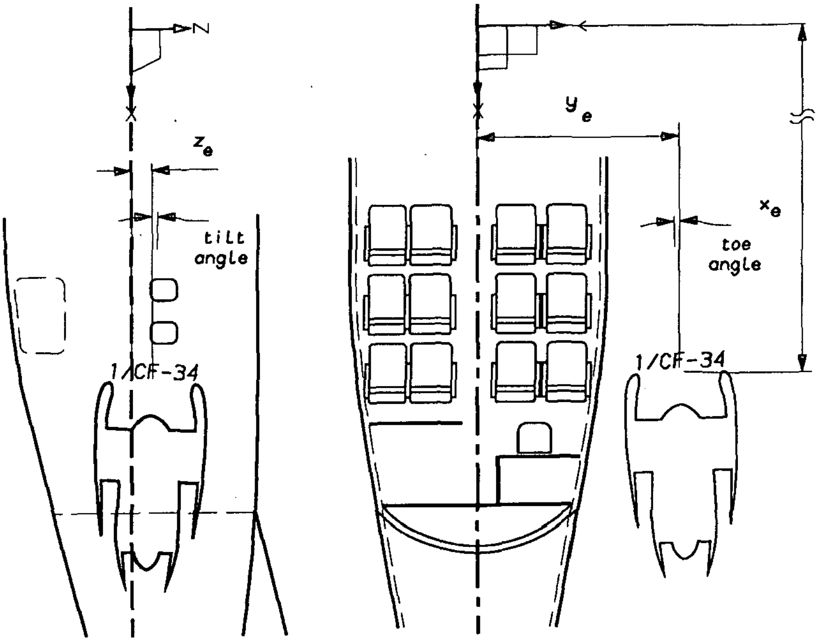


Figure B-18: Definition of fuselage-mounted engines.

B-9: Extended air intake ducts

In the previous section it was assumed that the engines are podded and fixed on pylons to the airframe. An alternative engine location is to bury the engine in the fuselage, e.g. B-727, L-1011 Tristar, Trident, or in the vertical tailplane, e.g. DC-10. This is usually the alternative if there are an odd number of engines. However, special provisions have to be made with respect to the engine air intake.

In ADAS, air intake ducts can be associated with an engine group. ADAS assumes these ducts to be of tubular shape and located parallel to the plane of symmetry. Thus, to define a duct, the designer needs to draw the duct centerline in side view, as shown in Figure B-19:

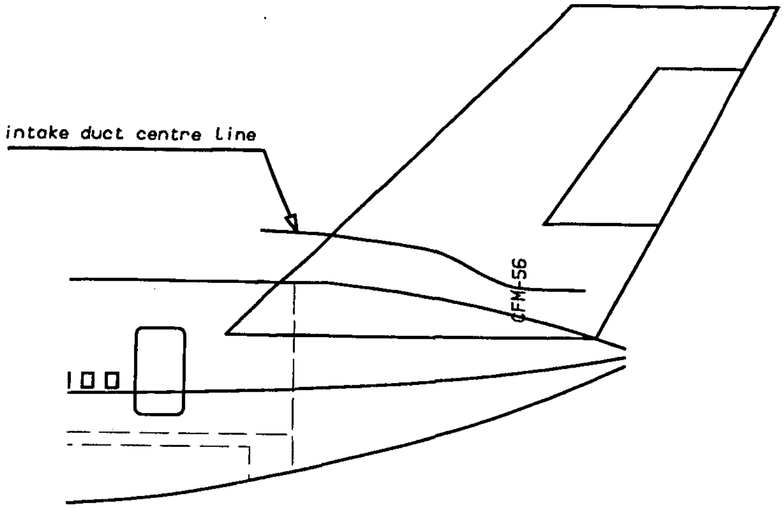


Figure B-19: Extended engine intake duct geometry definition.

The intake diameter is automatically taken equal to the maximum diameter of the engine nacelle. Also, ADAS translates, but does not deform, the centerline in such a way that the end of the duct connects with the engine nacelle, irrespective of where the designer places the centerline in the drawing.

If the D$MEDP-routine is used, the centerline will be represented by a LPS-line, a so-called superline, with the width attribute set equal to the engine diameter. A LTU link line is automatically drawn to obtain a pipe-model from the MEDUSA solid modeller.

B-10: Undercarriage geometry and disposition

An undercarriage layout is composed of a number of wheel assemblies, which in turn can consist of either 1, 2 (dual or tandem) or 4 (boogie) tires mounted on one support strut. A tire geometry is defined by a line of at least 4 points drawn in top view. This line represents the outline of a quarter tire segment, as shown in Figure B-20:

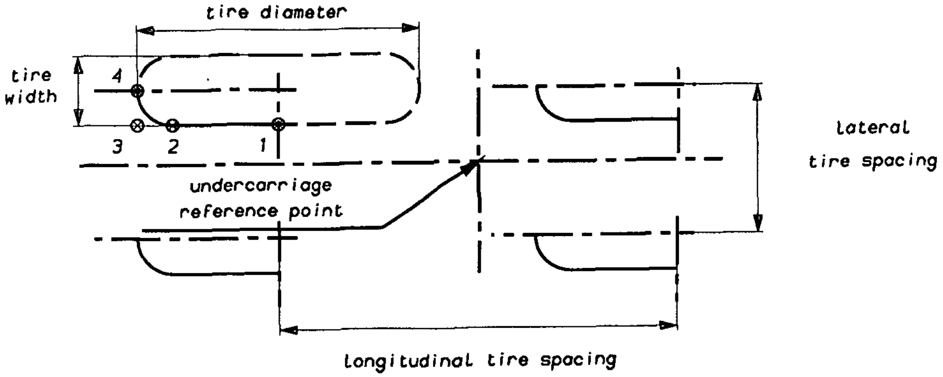


Figure B-20: Tire assembly geometry definition.

From these 4 points, the tire diameter and width are derived. The location of each assembly is defined with respect to the assembly datum point. To define an undercarriage layout all wheel assemblies must be defined in the top viewbox. As the undercarriage disposition is assumed symmetrical with respect to the XZ-plane, an assembly only needs to be defined on one side. If the assembly datum point is placed in the plane of symmetry, mirroring is automatically disabled, as shown in Figure B-21.

The Z-location is defined by the center point of a circle drawn in the side viewbox. ADAS assumes the strut location to be in the center of the wheel configuration. The top view defines the tire dimensions, i.e. diameter and width, and strut XY-location, while the Z-location is defined in side view.

If the D$MEDP-routine is used, LVR link lines are automatically drawn into the MEDUSA sheet to obtain a volume of revolution model representation for each tire, by rotating the tire profile line about the wheel axis.

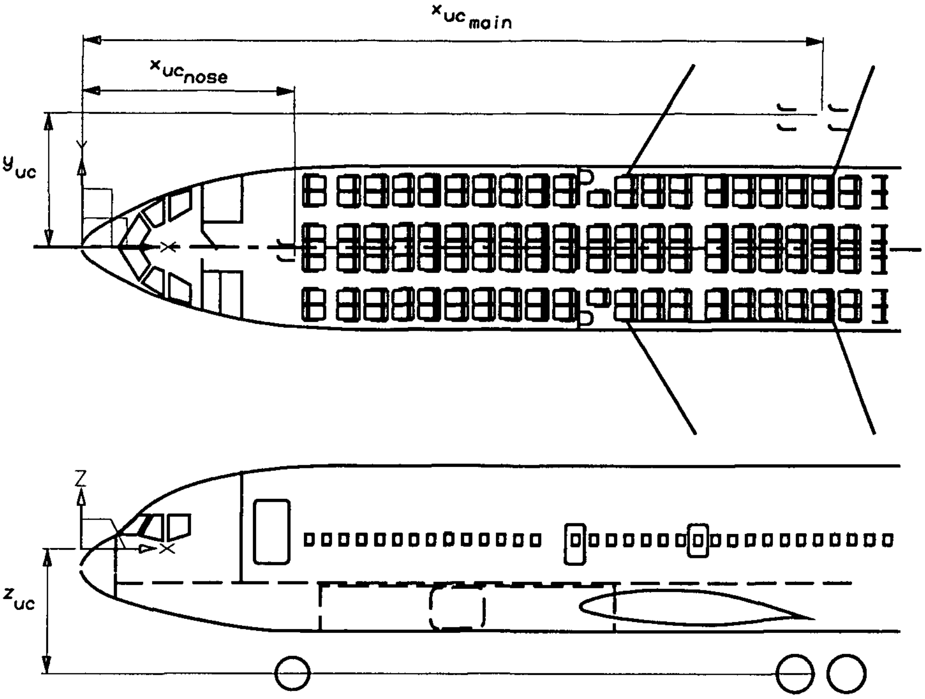


Figure B-21: Undercarriage disposition.

B-11: Center of gravity locations

Each weight item is assigned an internal reference number, related to the corresponding database index code: item number = database index - 6000. The designer can define the CG-location of each weight item by placing the item number in the side viewbox as a TEXT-element. The text datum point is used to define the XZ-coordinates of the center of gravity. Some weight prediction modules will also estimate the CG-location of a particular component if it is not specified in the MEDUSA drawing.

B-12: ADAS significant layer numbers

layer	description	view	remarks
145	Doors and windows 1	side	A closed line of any type. Orientation is not important.
146	Doors and windows 2	side	
147	Doors and windows 3	side	
148	Doors and windows 4	side	
149	Doors and windows 5	side	
150	Doors and windows 6	side	
151	Doors and windows 7	side	
152	Doors and windows 8	side	
153	Doors and windows 9	side	
154	Fuselage compartment 1	top	Any LINE-element with clockwise orientation. 2 points for station. Stations must correspond with stations in side view.
155	Fuselage compartment 2	top	
156	Fuselage compartment 3	top	
157	Fuselage compartment 4	top	
158	Fuselage compartment 5	top	
159	Fuselage compartment 6	top	
160	Fuselage compartment 7	top	
161	Fuselage compartment 8	top	
162	Fuselage compartment 9	top	
163	Fuselage compartment 10	top	
164	Pilot reference eye point	top	Any PRIM- or TEXT-element allowed.
165	Pilot reference eye point	side	Any PRIM- or TEXT-element allowed. Only Z-coordinate is used.
166	Fuselage compartment 3	side	Any LINE-element with clockwise orientation. 2 points for each station.
167	Fuselage compartment 4	side	
168	Fuselage compartment 5	side	
169	Cabin crew seats	top	Any PRIM- or TEXT-element allowed.
170	Flight crew seats	top	
171	Windshield panel 1	side	A closed line of any type with linear line segments. First point at
172	Windshield panel 2	side	
173	Windshield panel 3	side	

layer	description	view	remarks
174	Windshield panel 4	side	right top corner.
175	Windshield panel 5	side	Clockwise orientation.
176	Not used	side	
177	Not used	side	
178	Not used	side	
179	Not used	side	
180	Not used	side	
181	Fuselage max width line	side	LP-line. Top-down orientation. Only linear and conical line segments allowed.
182	Passenger seats class I	top	Any PRIM or TEXT is allowed.
183	Passenger seats class II	top	
184	Passenger seats class III	top	
185	Fuselage compartment 6	side	
186	Fuselage compartment 7	side	
187	Fuselage bulkhead 1	side	Any LINE-element of 2 points.
188	Fuselage bulkhead 2	side	
189	Fuselage bulkhead 3	side	
190	Fuselage bulkhead 4	side	
191	Fuselage bulkhead 5	side	
192	Toilets/Lavatories	top	Any PRIM- or TEXT-element.
193	Gallies/Pantries/Snackbars	top	Any PRIM- or TEXT-element.
194	Top view origin	top	Any PRIM- or TEXT-element.
195	Side view origin	side	Any PRIM- or TEXT-element.
196	Front view origin	front	Any PRIM- or TEXT-element.
197	Fuselage top line	side	See layer 181.
198	Fuselage bottom line	side	See layer 181.
199	Fuselage side line	top	See layer 181.
200	Not yet used		
201	Lifting surface 1	any	LINE of any type with linear
202	Lifting surface 2	any	segments.
203	Lifting surface 3	any	First point at apex.
204	Lifting surface 4	any	Clockwise orientation.

layer	description	view	remarks
205	Lifting surface 5	any	Each point in the
206	Lifting surface 6	any	leading edge must have
207	Lifting surface 7	any	a corresponding point in
208	Lifting surface 8	any	the trailing edge, so
209	Lifting surface 9	any	number of points must be even.
210	Lifting surface 10	any	
211	Airfoils lifting surface 1	any	TEXT-element of any type with
212	Airfoils lifting surface 2	any	airfoil designation.
213	Airfoils lifting surface 3	any	Rotation represents
214	Airfoils lifting surface 4	any	geometric twist or
215	Airfoils lifting surface 5	any	incidence. Only Z-
216	Airfoils lifting surface 6	any	coordinate of datum is
217	Airfoils lifting surface 7	any	used (V-angle).
218	Airfoils lifting surface 8	any	Airfoil name must begin
219	Airfoils lifting surface 9	any	with sequence number
220	Airfoils lifting surface 10	any	e.g. 1/NACA-64-415.
221	Undercarriage assembly 1	top	LINE-element of any type. Only
222	Undercarriage assembly 2	top	first 4 points are used.
223	Undercarriage assembly 3	top	Clockwise orientation.
224	Undercarriage assembly 4	top	
225	Undercarriage assembly 5	top	
226	Engine designation	top	Any TEXT-element with engine designation. Rotation represents toe angle. Engine designation must be prefixed with sequence number e.g. 1/ALF-502D.
227	Engine designation	side	Any PRIM- or TEXT-element. Rotation represents tilt angle. Only Z-coordinate is used.
228	Engine duct 1	side	Any LINE-element representing
229	Engine duct 2	side	the duct center line.
230	Engine duct 3	side	Top-down orientation.

layer	description	view	remarks
231	Fuselage cross-section	front	Any LINE-element representing the upper and lower cross-sectional lobe. Orientation is not important. Only the second and fourth point (weight factors) are used.
232	Flaps sections	top	Any LINE-element.
233	Aileron	side	Any LINE-element.
234	Elevator	top	Any LINE-element.
235	Rudder	top	Any LINE-element.
236	Dorsal fin	side	A straight line of any type (2 points) with top-down orientation.
237	Undercarriage wheel 1	side	Any LINE-element. Only the first 2 points are used representing the center and radius respectively.
238	Undercarriage wheel 2	side	
239	Undercarriage wheel 3	side	
241	Undercarriage wheel 4	side	
242	Undercarriage wheel 5	side	
243	Wing fuel tank 1	top	Any LINE-element.
244	Wing fuel tank 2	top	Any LINE-element.
245	Wing fuel tank 3	top	Any LINE-element.
246	Fuselage compartment 1	side	Any LINE-element.
247	Fuselage compartment 2	side	line
248	CG locations	any	Any TEXT-element representing the item number with 3 digits. Text must be right justified. Text datum point represents the CG-location.
1000	Link lines, TMG text, etc.		Link lines and other 3D-elements generated by the D$MEDP-routine are automatically placed in this layer.

layer	description	view	remarks
1001	Other aircraft components		Other components are graphical elements which are automatically added to the MEDUSA drawing if the D$MEDP-routine is used, e.g. mirroring of lifting surfaces, engines and airfoil profile lines, etc.

APPENDIX C: THE ENGINE LIBRARY

In ADAS, an engine can be associated with a design by selecting an engine type from the engine library and placing the engine name <engine> as a TEXT-element into the corresponding MEDUSA-drawing. When the ADAP-program is executed, the MEDUSA → ADAS interface routine will automatically search the UFD or the engine library VL>VO>ADAS>ENGINE for the engine performance data file <engine> and the engine/nacelle geometry file Y.<engine> respectively.

The designer may also define a new engine type by preparing the data files and optionally placing them in the engine library for general use. The source of this information is immaterial to ADAS. Data of existing engines is usually obtained from engine manufacturers brochures and other references, while for study engines, engine performance may be derived from engine cycle analysis, e.g. with the method outlined in Ref. 137. Provisions are included in ADAS to scale a reference engine according to a required static takeoff thrust (see Section C-3). The following Sections illustrate how engine data is represented in ADAS.

C-1: Engine nacelle geometry

The geometry of engine nacelles is defined with the MEDUSA-system. ADAS assumes an engine nacelle to be an axisymmetric body, represented by a single profile line, as shown in Figure C-1:

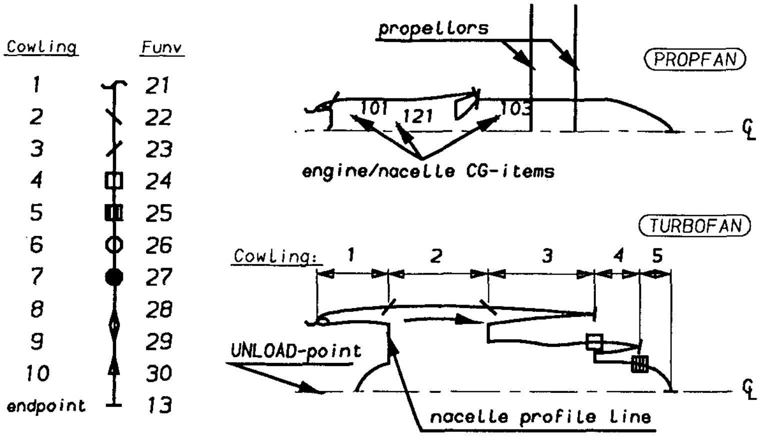


Figure C-1: Engine/nacelle geometry definition.

This approximation usually holds well for turbofan and turbojet engines and, to a lesser extend, also for propellor engines. In some cases however, it may be neccesary to define an 'equivalent' engine, which approximates the original nacelle in terms of e.g. overall dimensions, wetted areas, etc.

C-2: Engine performance data

Engine performance, i.c. thrust, fuel flow and airflow, is dependent on the operating conditions, i.e. engine rating, Mach number and altitude:

$$T = f(\Gamma, M, h) \tag{C-1}$$

$$F_{fuel} = f(\Gamma, M, h) \tag{C-2}$$

$$F_{air} = f(\Gamma, M, h) \tag{C-3}$$

There are several alternative ways in which engine performance data can be represented in a computer system, e.g.:

- Analytical methods are available to estimate off-design engine performance based on 1-dimensional thermodynamic cycle analysis and given component efficiencies [e.g. Ref 137]. These types of methods are typically useful to assess the influence of engine technology on the overall aircraft design. However, the effect of changes in the engine cycle on e.g. engine weight and geometry, remain very difficult to predict. An interesting approach is proposed by the TU Berlin and incorporated in the CAPDA-system [Refs. 64 - 66]. A parametric engine study was conducted, applying a cycle analysis method to a modern reference engine (CF6-50 C2), while varying by-pass ratio, Mach number and altitude for takeoff, climb and cruise engine rating respectively. The engine performance data thus obtained were approximated by a polynomial fit.

- The most commonly used method for engine performance representation is interpolation on a set of data tables:
  - The performance of an existing engine is accurately modelled in this way, even in the operating limitation regions i.c. high Mach number (pressure recovery) and low speeds (Reynolds number effects);
  - Engine scaling and flat rating can be easily incorporated;
  - The method is general and flexible enough to accommodate several types of propulsion systems, e.g. jet engines, propellor engines and propfans, without effecting the propulsion modules;
  - The evaluation of engine performance for a given operating condition is fairly straightforward and takes relatively little computer time, although memory requirements may be significant.

Both methods are generally employed in configuration development and a design system should ideally include a combination of both methods. Prominent design systems, such as Northrop's Advanced Configuration Analysis and Design system (ACAD) and Boeing's Computer-aided Preliminary Design System (CPDS), include both engine cycle analysis methods and engine data tables. Although scheme are required for configuration development, the current pilot-version of ADAS only allows engine performance data representation in the form of data tables, although experience has been obtained with data tables generated by an analytical approach.

The parameter generally used to define engine rating or power setting is the rotational speed of the fan/compressor, expressed in RPMs. For Rolls-Royce aero-engines, the pressure ratio, e.g. TGT or TET is used. However, for conceptual aircraft design and engine sizing purposes, an alternative definition for engine rating is used in ADAS. For aircraft performance calculations, it is often necessary to interpolate on engine performance for either a given engine rating or a given thrust. This would mean that for a given thrust, engine rating has to be determined from eq. (C-1) by backward interpolation which is inconvenient computationally.

A more logical way is to relate engine rating directly to the thrust available:

$$\Gamma = \frac{T}{T_{max}} \tag{C-4}$$

where T_{max} and T are the maximum available thrust and derated thrust respectively, for a given Mach number and altitude. Hence, engine rating thus defined is typically $0 \leq \Gamma \leq 1$.

The engine performance data now becomes:

$$T_{max} = f(M,h) \qquad \text{with } \Gamma = 1 \tag{C-5}$$

$$F_{fuel} = f(\Gamma,M,h) \tag{C-6}$$

$$F_{air} = f(\Gamma,M,h) \tag{C-7}$$

Note that the new definition for engine rating also entails a data reduction for the thrust table, as engine rating is now eliminated. The format of these engine performance data tables is compatible with the ADAS plotting facility (see Figure 5.5). Thus, engine performance can be graphically displayed with MEDUSA for example to check input data for errors. As an example, Figure C-2 gives the engine performance representation of the Avco Lycoming ALF-502D turbofan engine for 8 different altitudes.

C-3: Engine scaling techniques

The development of aero-engines and airframes are usually the responsibility of separate industries. Because of the critical nature in matching engines and airframes, there is a close interaction between what the airframe manufacturers require and engine manufacturers can offer. Often, aircraft are designed based on an existing engine type (off-the-shelf).

However, sometimes designs are considered with an engine with upgraded or derated thrust levels with respect to an available (reference) engine. This technique of engine scaling or "rubberizing" is usually applied if a suitable engine is not (yet) available. However, engine scaling requires the effect of engine size on engine weight, dimensions and performance to be taken into account. The ADAS program library provides a routine D$RUBB which contains simple scaling rules for engine "rubberizing".

The ADAS engine scaling module D$RUBB applies simple scaling laws to the reference engine for a given static take-off thrust required, as follows.

If

$$\sigma_T = \frac{T_{to_{req}}}{T_{to_{avail}}} \tag{C-8}$$

then

$$\sigma_{FF} = T_{FF_{ref}} = \phi_{FF_1} + \phi_{FF_2}\,\sigma_T^{\phi_{FF_3}} \tag{C-9}$$

$$\sigma_{AF} = \frac{AF}{AF_{ref}} = \phi_{AF_1} + \phi_{AF_2}\,\sigma_T^{\phi_{AF_1}} \tag{C-10}$$

$$\sigma_{L_{nac}} = \frac{L_{nac}}{L_{nac_{ref}}} = \phi_{L_1} + \phi_{L_2}\,\sigma_T^{\phi_{L_3}} \tag{C-11}$$

$$\sigma_{D_{nac}} = \frac{D_{nac}}{D_{nac_{ref}}} = \phi_{D_1} + \phi_{D_2}\,\sigma_T^{\phi_{D_3}} \tag{C-12}$$

and

$$\sigma_{W_{dry}} = \frac{W_{dry}}{W_{dry_{ref}}} = \varphi_{W_1} + \varphi_{W_2}\, \sigma_T^{\varphi_{W_3}} \qquad \text{(C-13)}$$

The factors φ can be defined by the user through the design database and typically follow from detailed, usually statistical, studies on existing engines to identify particular trends.

The dimensions and wetted areas of the engine cowlings are scaled proportionally:

$$L_{cowl_i} = \sigma_{L_{nac}}\, L_{cowl_{ref_i}} \qquad i = 1, N_{cowl} \qquad \text{(C-14)}$$

$$D_{cowl_i} = \sigma_{D_{nac}}\, D_{cowl_{ref_i}} \qquad i = 1, N_{cowl} \qquad \text{(C-15)}$$

$$S_{wet_{cowl_i}} = \sigma_{L_{nac}}\, \sigma_{D_{nac}}\, S_{wet_{cowl_{ref_i}}} \qquad i = 1, N_{cowl} \qquad \text{(C-16)}$$

Engine performance will be adjusted as follows:

$$T_{max}\,(M,h) = \sigma_T\, T_{max_{ref}}\,(M,h) \qquad \text{(C-17)}$$

$$FF\,(\Gamma,M,h) = \sigma_{FF}\, FF_{ref}\,(\Gamma,M,h) \qquad \text{(C-18)}$$

and

$$AF\,(\Gamma,M,h) = \sigma_{AF}\, AF_{ref}(\Gamma,M,h) \qquad \text{(C-19)}$$

The engine scale factors can generally be derived from statistical analysis on existing engine types, e.g. Refs. 59 and 96. A first order approximation is to assume specific fuel consumption and specific thrust constant, thus:

$$\varphi_{FF_1} = 0 \;;\; \varphi_{FF_2} = 1 \;;\; \varphi_{FF_3} = 1 \qquad \text{(C-20)}$$

$$\varphi_{AF_1} = 0 \; ; \; \varphi_{AF_2} = 1 \; ; \; \varphi_{AF_3} = 1 \qquad \text{(C-21)}$$

Assuming further that the thrust is proportional to the air mass flow and the length/diameter ratio of the nacelle remain constant, the scale factors for the nacelle dimensions become:

$$\varphi_{D_1} = 0 \; ; \; \varphi_{D_2} = 1 \; ; \; \varphi_{D_3} = 0.5 \qquad \text{(C-22)}$$

$$\varphi_{L_1} = 0 \; ; \; \varphi_{L_2} = 1 \; ; \; \varphi_{L_3} = 0.5 \qquad \text{(C-23)}$$

The scale factors for engine weight are difficult to predict as it is dependent on factors such as engine modularity. Ref. 124 suggests $\varphi_{W_1} = 0$, $\varphi_{W_2} = 1$ and $1.1 \leq \varphi_{W_3} \leq 1.2$.

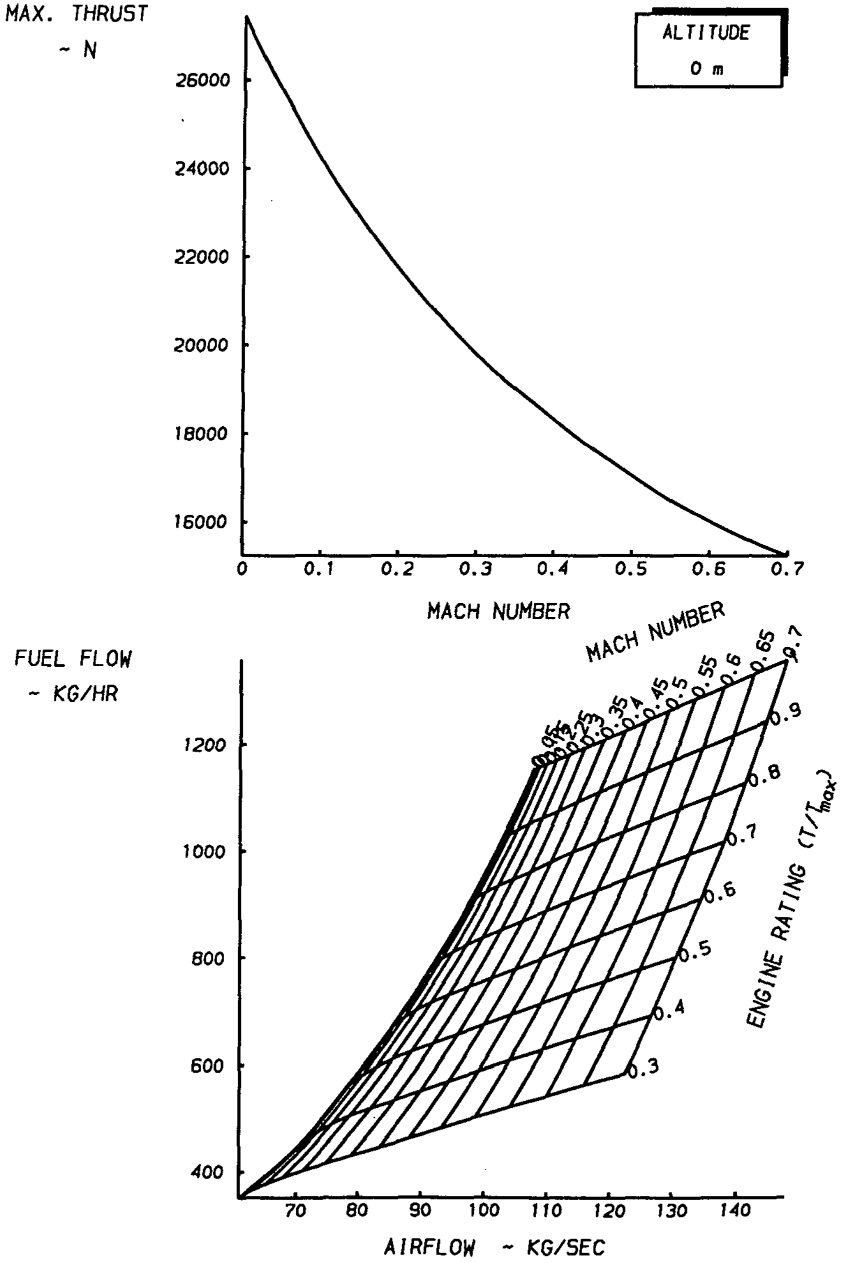


Figure C-2a: Performance representation of the ALF-502D turbofan engine (h = 0 m).

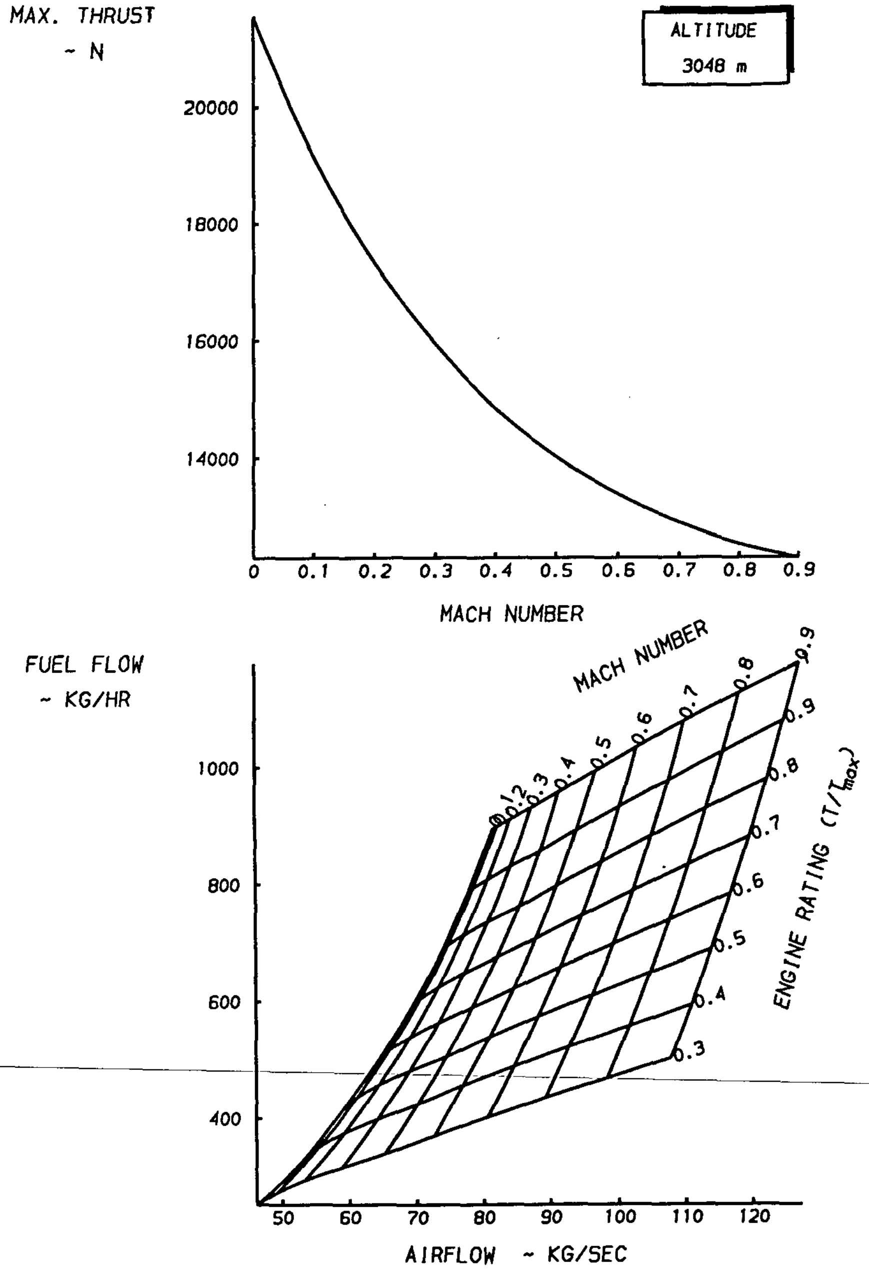


Figure C-2b: Performance representation of the ALF-502D turbofan engine (h = 3048 m).

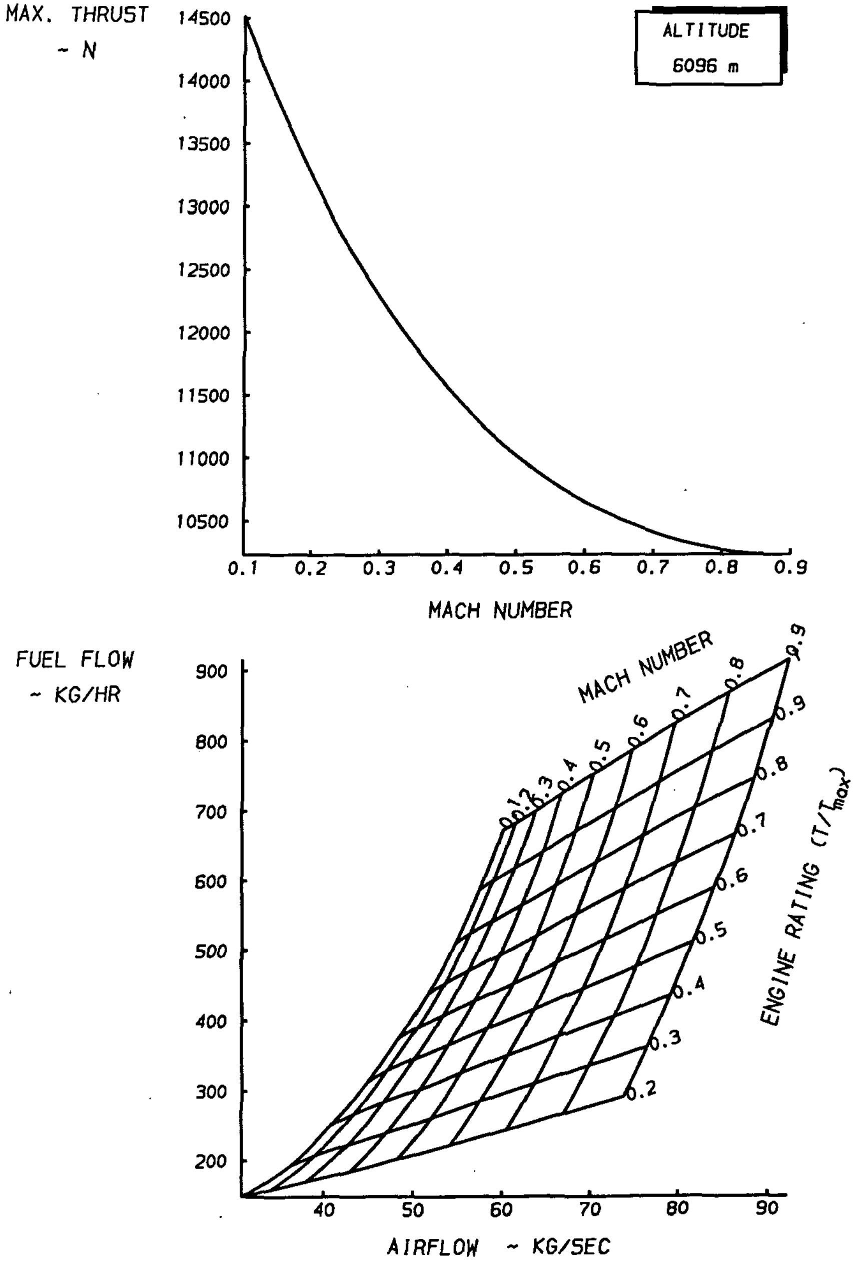


Figure C-2c: Performance representation of the ALF-502D turbofan engine (h = 6096 m).

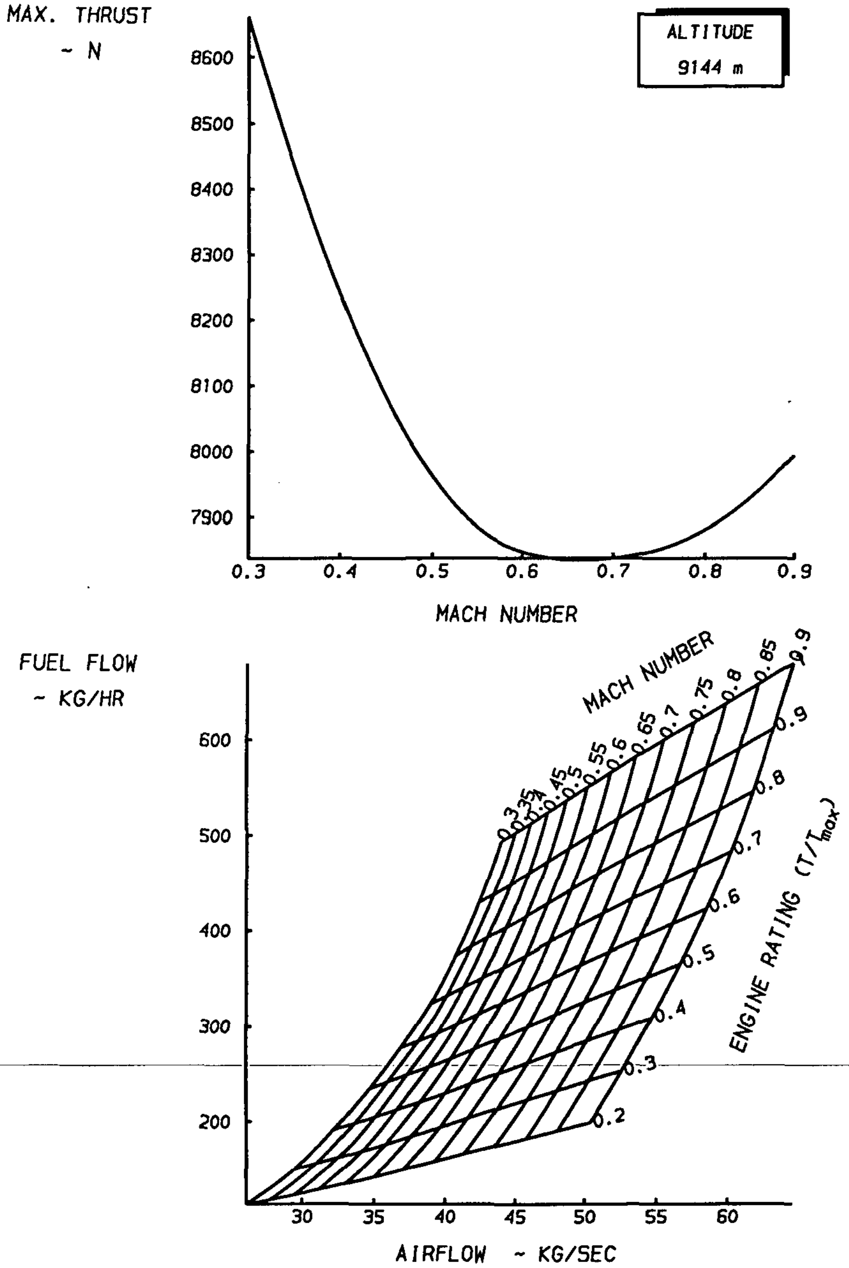


Figure C-2d: Performance representation of the ALF-502D turbofan engine (h = 9144 m).

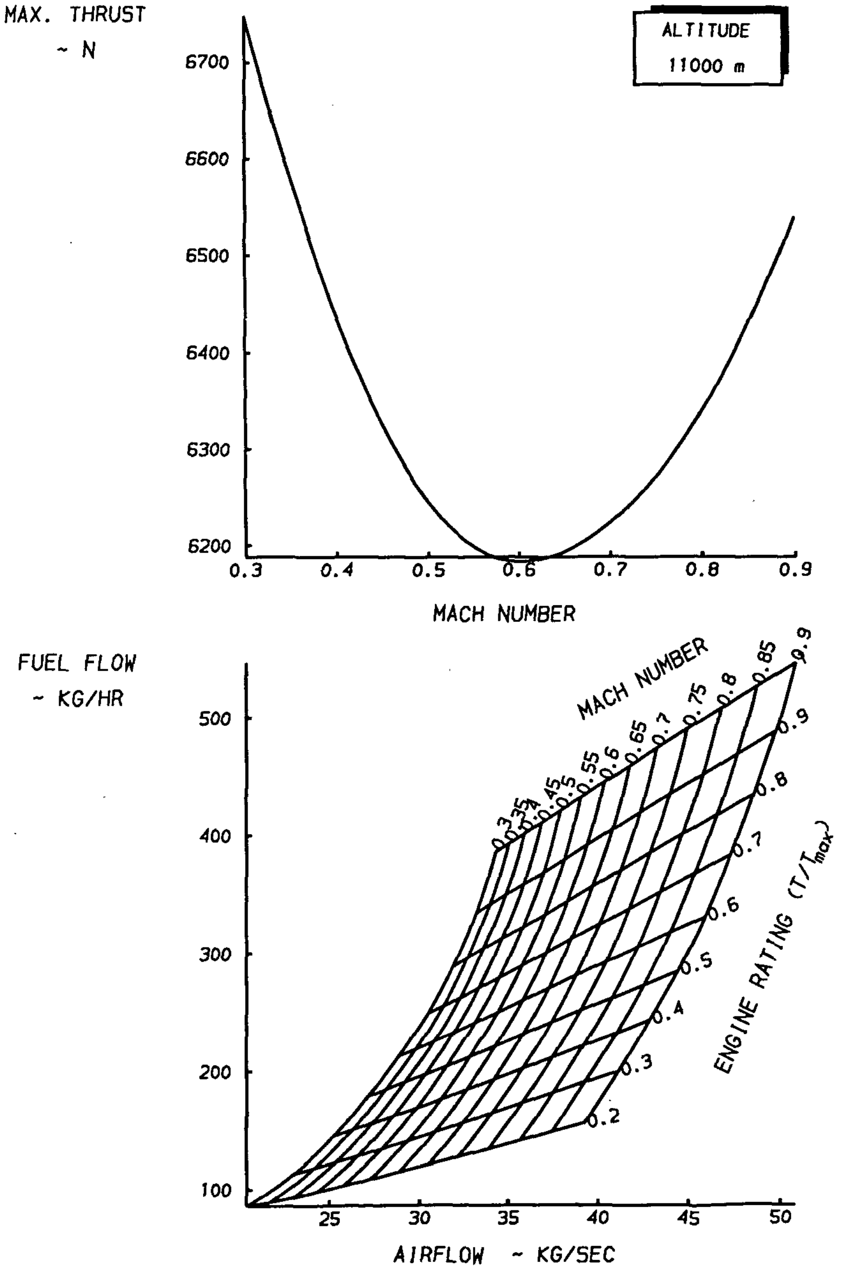


Figure C-2e: Performance representation of the ALF-502D turbofan engine (h = 11000 m).

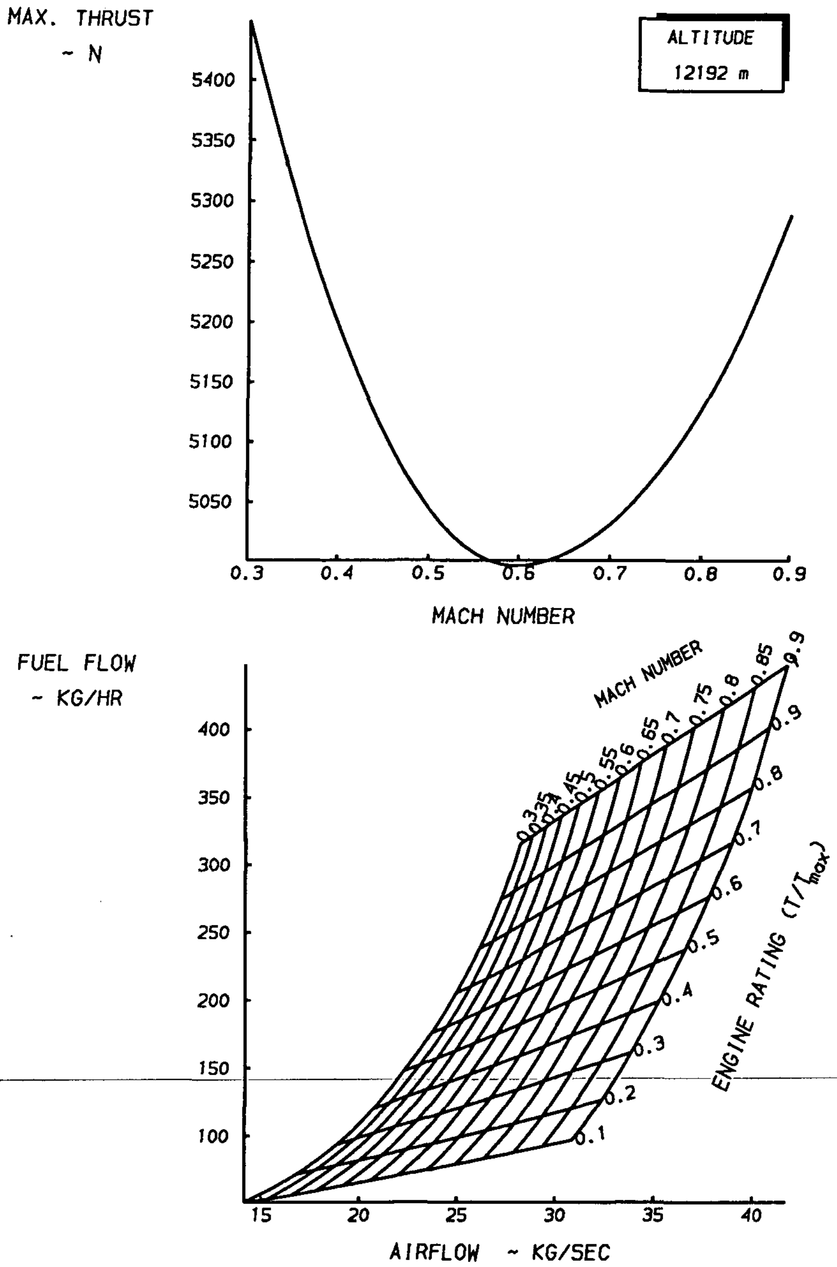


Figure C-2f: Performance representation of the ALF-502D turbofan engine (h = 12192 m).

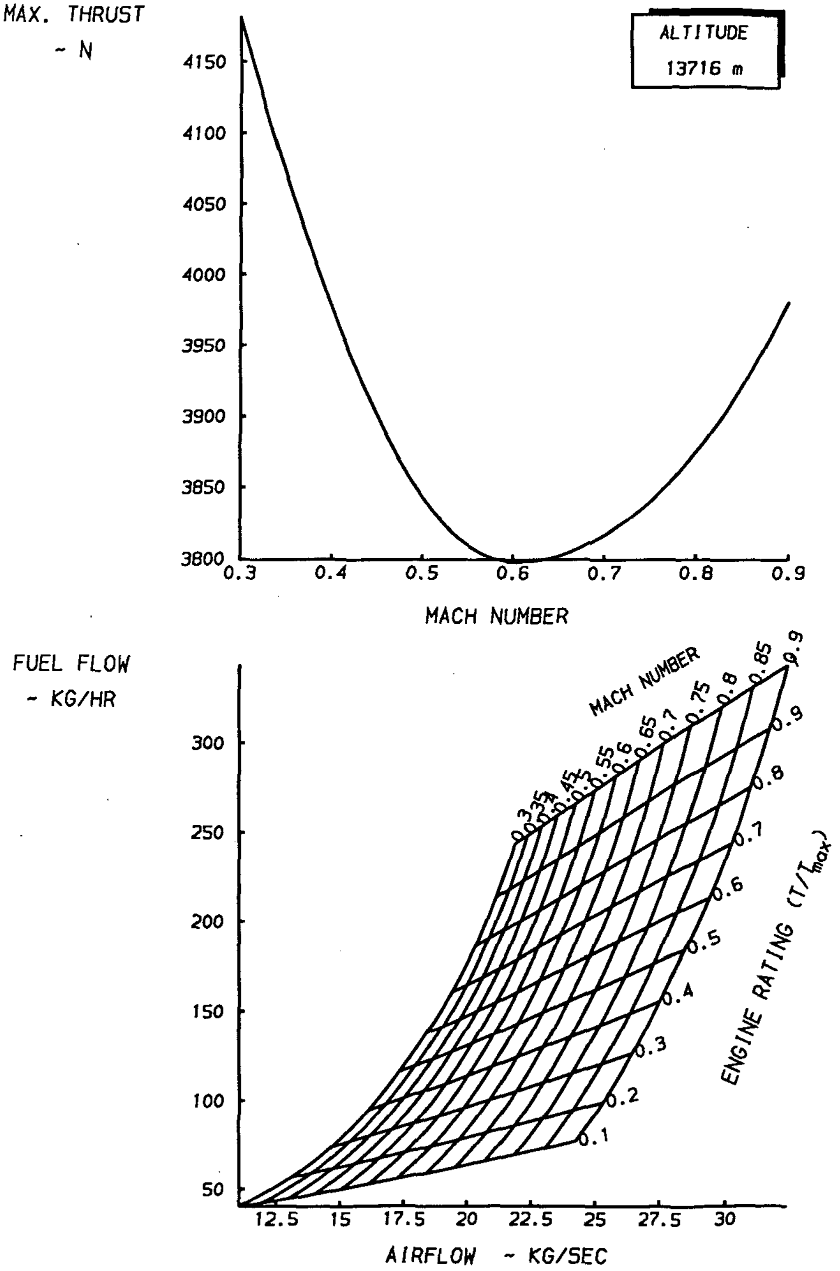


Figure C-2g: Performance representation of the ALF-502D turbofan engine (h = 13716 m).

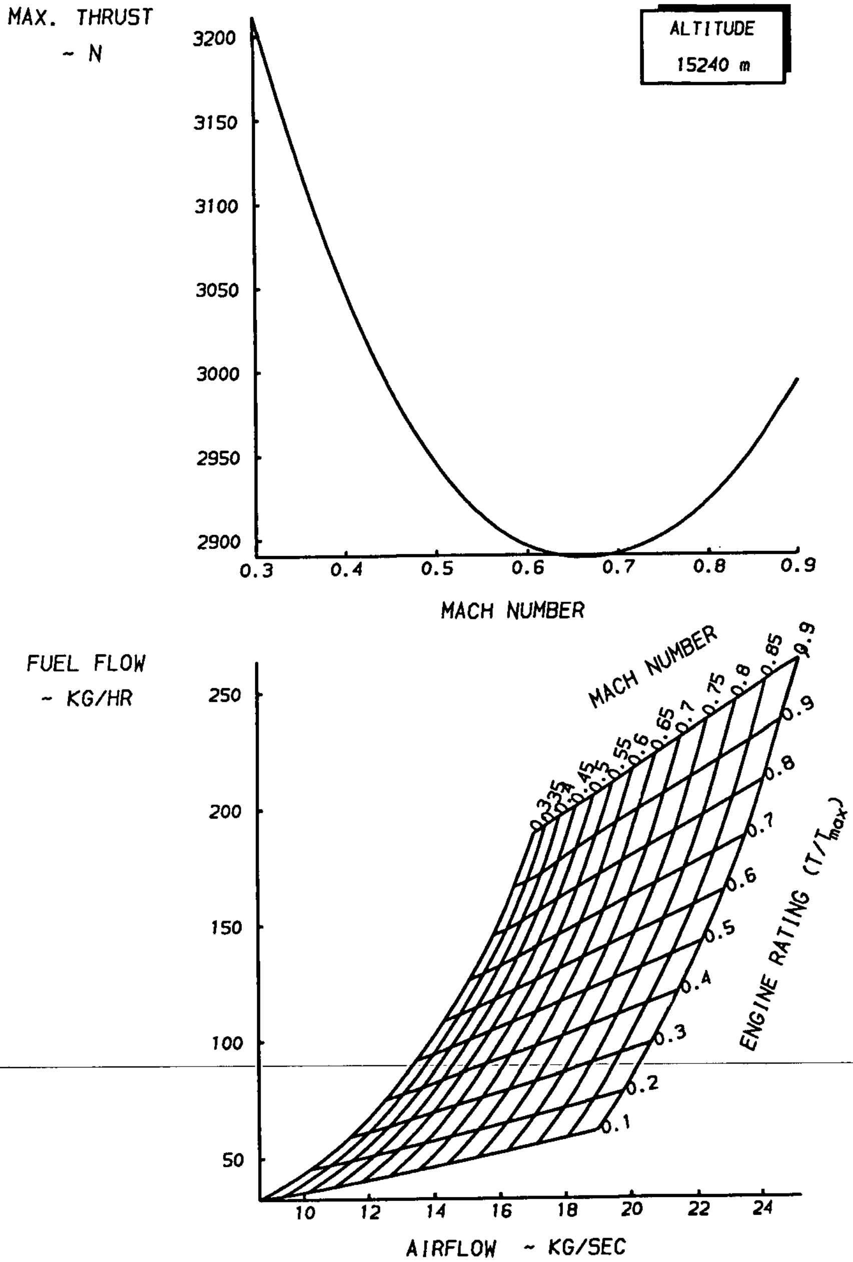


Figure C-2h: Performance representation of the ALF-502D turbofan engine (h = 15240 m).

APPENDIX D: THE AIRFOIL LIBRARY

Similar to engine data, aerodynamic data of standard airfoil sections are retained in an airfoil library. The structure and format in which this information is stored in ADAS will be discussed in this Section.

In the ADAS airfoil library, a distinction is made between airfoil geometry data and aerodynamic data which are stored in 2 separate files.

D-1: Airfoil geometry

Geometry data defines the thickness distribution of the airfoil in non-dimensional form, i.c. t/c versus x/c. The airfoil shape must be drawn with MEDUSA as a closed line with the first point at the leading edge and running clockwise. The line may contain linear and conical line segments, as shown in Figure D-1:

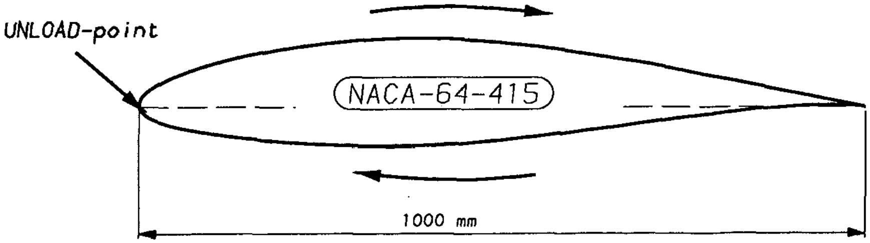


Figure D-1: Definition of an airfoil geometry with MEDUSA.

Airfoil coordinates of standard NACA-sections can be found in e.g. Ref. 1, while for other low and medium speed sections one must consult the recent literature.

After an airfoil section has been drawn with MEDUSA, the profile line is unloaded in a symbol file Y.<airfoil>. The advantage of such a type of file is that the airfoil can be loaded into any other existing drawing and the airfoil library will in general be useful to many MEDUSA-users. For this purpose, it is recommended to draw the airfoil with a chord length of 1 m, so that it can be easily scaled to a given chord length.

D-2: Airfoil aerodynamic data

For each airfoil type, a second file <airfoil> must exist which contains 2-dimensional aerodynamic data. Next to some basic information, such as the zero-lift angle of attack, aerodynamic centre and pitching moment coefficient, this file contains data tables which define the following primary aerodynamic coefficients as a functions of the local flow conditions:

$$c_l = f(\alpha, R_e) \tag{D-1}$$

$$\Delta c_l = f(\alpha, M) \tag{D-2}$$

$$c_d = f(c_l, R_e) \tag{D-3}$$

$$\Delta c_d = f(c_l, M) \tag{D-4}$$

This data is stored in the usual ADAS-format (see Figure 5.5) and can thus be graphically represented with MEDUSA, as shown in Figure D-2.

Airfoil data can for example be employed in aerodynamic analysis methods which uses 2-dimensional data to estimate the aerodynamic characteristics of lifting surfaces (see Chapter 8). The general availability of specific aerodynamic data, especially for high Reynolds numbers and Mach numbers, is limited. However, the facility to input airfoil data presents the designer with the opportunity to enhance design analysis with experimental data when available. By default, the <airfoil> and Y.<airfoil> files are placed in the UFD where ADAP will look first. The system supervisor can optionally copy these files to the airfoil library to make them available for general use.

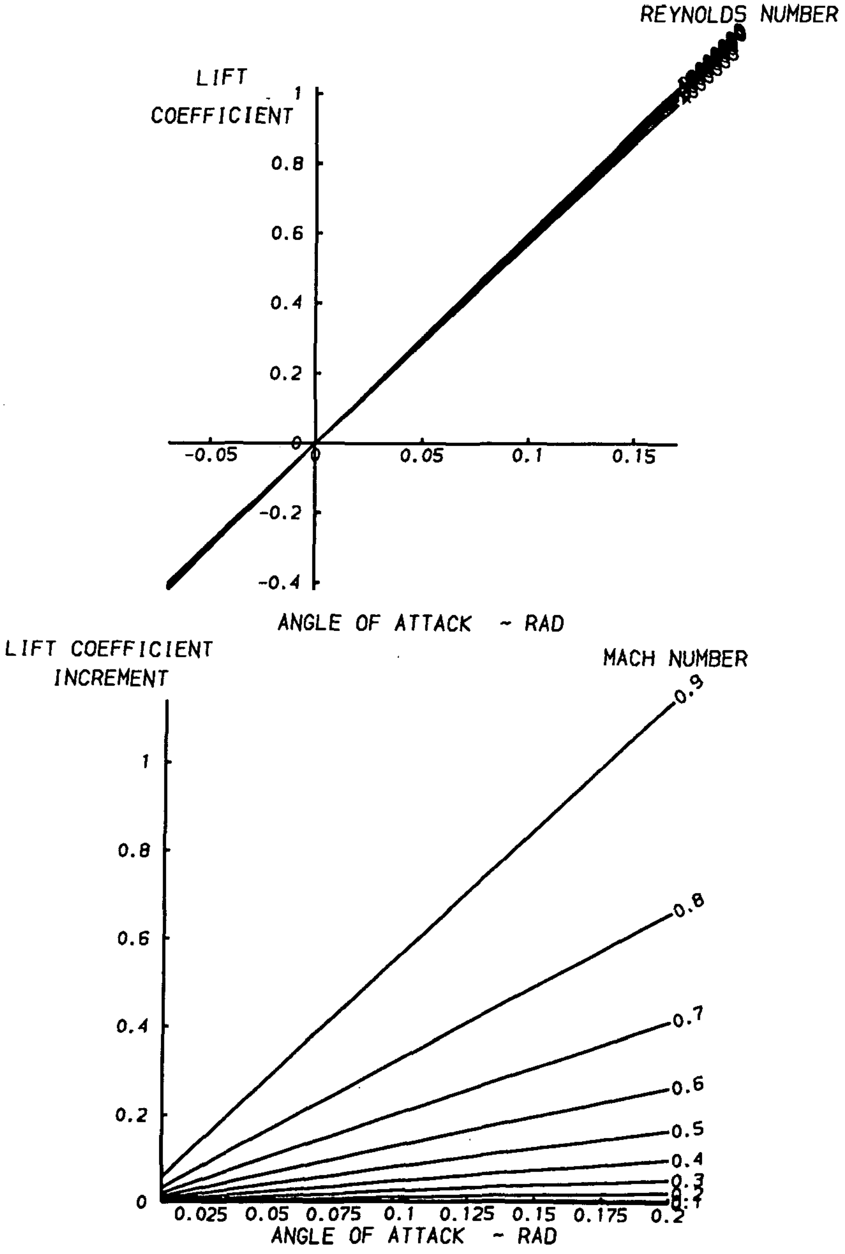


Figure D-2: Airfoil aerodynamic data representation (lift).

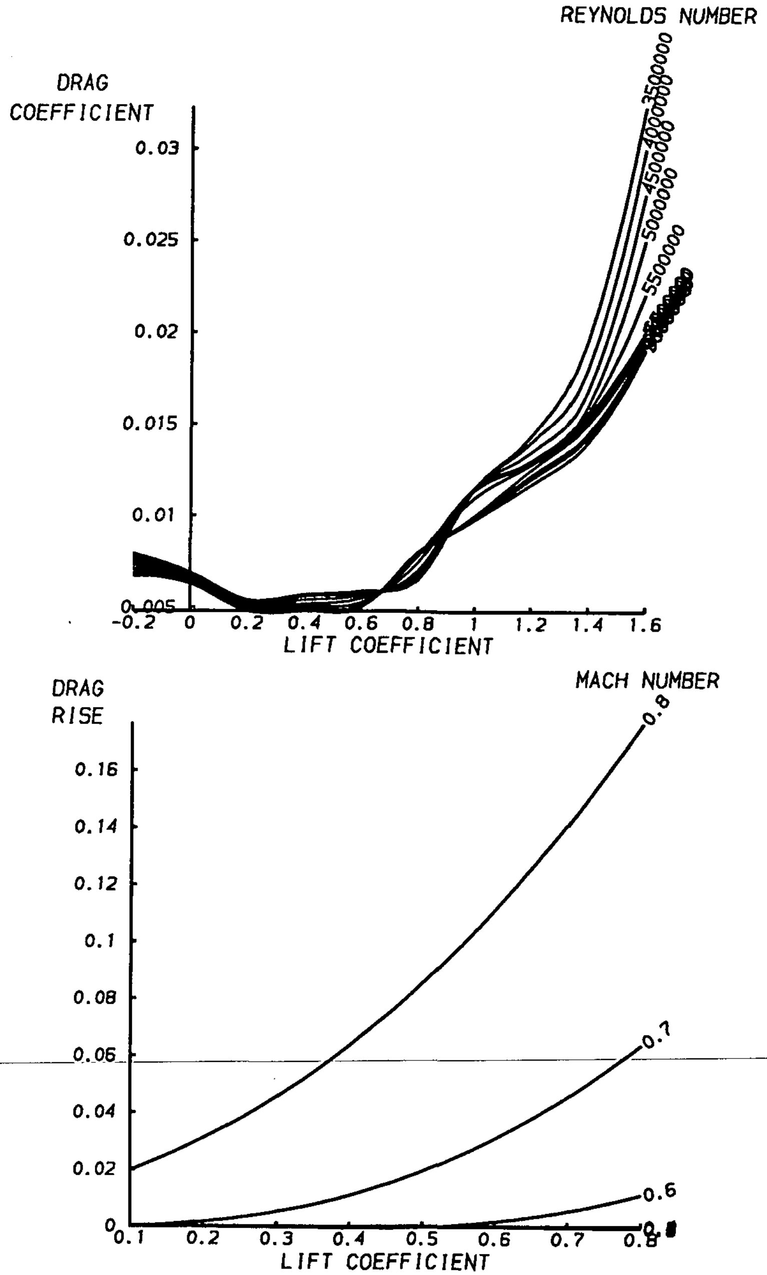


Figure D-3: Airfoil aerodynamic data representation (drag).

APPENDIX E: ADAS PROGRAM MODULES SUMMARY

This Appendix gives a list of the subprograms currently available from the program library with a brief functional description.

Module	Function
D$AIRS	Scale airfoil for a given thickness/chord ratio.
D$ATMO	Compute atmospheric properties.
D$CGEO	Compute canard geometry.
D$CGFL	Compute fuel center of gravity.
D$CGSH	Compute center of gravity travel.
D$CHIS	Draw optimization convergence history.
D$CLCC	Compute lift distribution for given lift coefficient.
D$CLMB	Compute climb performance.
D$CMUL	Draw a multi-valued curve.
D$CSGL	Draw a single-valued curve.
D$CSGM	Compute control surfaces geometry.
D$CSWT	Compute control surfaces group weight.
D$DIVM	Compute dive manoeuvre.
D$DOCT	Compute direct operating cost.
D$DSNT	Compute descent performance.
D$ENGL	Compute engines location.
D$ERAS	Erase specified MEDUSA sheet layers.
D$FURW	Compute furnishing and equipment group weight.
D$FUSA	Compute fuselage aerodynamics.
D$FUSW	Compute fuselage group weight.
D$HGEO	Compute horizontal tailplane geometry.

Module	Function
D$HLDG	Compute high-lift devices geometry.
D$HTLA	Compute horizontal tailplane aerodynamics.
D$LINE	Draw a specified line into a MEDUSA sheet.
D$LLCL	Draw a LCL line into a MEDUSA sheet.
D$LLRS	Draw a LRS line into a MEDUSA sheet.
D$LLVR	Draw a LVR line into a MEDUSA sheet.
D$LOGO	Draw ADAS logo.
D$LSCD	Compute lifting surface drag.
D$LSLO	Compute lifting surface location.
D$LSWT	Compute lifting surface wetted area.
D$MCRS	Compute flight conditions for cruise.
D$MEDP	Draw design configuration into a MEDUSA sheet.
D$MVMO	Compute maximum operational flight conditions.
D$NACA	Compute engine nacelles aerodynamics.
D$NACW	Compute nacelle group weight.
D$OIWT	Compute operational items group weight.
D$PAYW	Compute payload group weight.
D$PCOH	Select flight conditions for high-speed aerodynamics.
D$PIVI	Compute pilot's field of vision.
D$PL01	Draw payload - range performance.
D$PL02	Draw induced angle of attack distribution.
D$PL03	Draw pitching moment distribution.
D$PL04	Draw profile drag distribution.
D$PL05	Draw lift distribution.
D$PL06	Draw balanced field length.
D$PL09	Draw loading diagram.
D$PL10	Draw flight envelope.
D$PL11	Draw pilot's field of vision.
D$POLH	Generate high-speed aerodynamic polars.
D$PR01	Print lifting surfaces geometry.
D$PR02	Print weight breakdown.
D$PR03	Print fuselage geometry.
D$PR04	Print design load factors.
D$PR05	Print design operating speeds.

Module	Function
D$PR06	Print control surfaces geometry.
D$PR07	Print engines location.
D$PR08	Print fuel tanks capacity.
D$PR10	Print engine nacelles geometry.
D$PR11	Print atmospheric properties.
D$PR12	Print pylons geometry.
D$PR13	Print flightdeck windshield geometry.
D$PR14	Print fuselage compartments geometry.
D$PR15	Print cargo and freight holds geometry.
D$PR17	Print drag breakdown.
D$PR18	Print aerodynamic coefficients.
D$PR19	Print takeoff performance.
D$PR20	Print high-speed lift coefficient versus angle of attack and Mach number.
D$PR21	Print high-speed drag coefficient versus lift coefficient and Mach number.
D$PR22	Print airfoil geometry.
D$PR23	Print undercarriage geometry.
D$PR24	Print high-speed pitching moment coefficient versus lift coefficient and Mach number.
D$PR25	Print cruise performance.
D$PR26	Print passenger cabin layout.
D$PR27	Print low-speed lift coefficient versus angle of attack and flap setting.
D$PR28	Print direct operating cost.
D$PR29	Print low-speed drag coefficient versus lift coefficient and flap setting.
D$PR30	Print dive manoeuvre.
D$PR31	Print low-speed pitching moment coefficient versus lift coefficient and flap setting.
D$PR32	Print lift distribution.
D$PR33	Print climb performance.
D$PR66	Print payload - range performance.
D$PRNG	Compute payload - range performance.

Module	Function
D$PRPW	Compute propulsion group weight.
D$PYLD	Compute pylons aerodynamics.
D$PYLG	Compute pylons geometry.
D$RUBB	Scale engines for a given required takeoff thrust.
D$SPGM	Compute spoilers and speed brakes geometry.
D$STOG	Compute takeoff ground run.
D$SYSW	Compute systems group weight.
D$TALW	Compute tail group weight.
D$TANK	Compute wing fuel tank volume.
D$TIRD	Compute undercarriage tires size.
D$UCLO	Compute undercarriage location.
D$UCWT	Compute undercarriage group weight.
D$VGEO	Compute vertical tailplane geometry.
D$VTLA	Compute vertical tailplane aerodynamics.
D$WGEO	Compute wing geometry.
D$WGHT	Compute weight breakdown.
D$WNGA	Compute wing aerodynamics.
D$WNGW	Compute wing group weight.
F$AIRA	Compute airfoil enclosed area.
F$AIRC	Compute airfoil circumference.
F$BALF	Compute balanced field length.
F$CDBS	Compute fuselage base drag.
F$CDHS	Interpolate high-speed CD for given CL and M.
F$CDLS	Interpolate low-speed CD for given CL and flap setting.
F$CLHS	Interpolate high-speed lift coefficient for a given angle of attack and Mach number.
F$CLLS	Interpolate low-speed lift coefficient for a given angle of attack and flap setting.
F$CRSE	Compute equivalent range performance.
F$ERAT	Interpolate engine rating for a given thrust, Mach number and altitude.
F$FLAT	Compute flat rating.
F$FLWT	Compute flap system weight.
F$GAMA	Compute climb angle for given speed and engine rating.

Module	Function
F$GAMM	Compute maximum rate of climb.
F$GMAN	Compute load factors due to manoeuvres.
F$GUST	Compute load factors due to gust.
F$LOAD	Compute design load factors.
F$RUN1	Compute takeoff run with all engines operative.
F$RUN2	Compute takeoff run with one engine out.
F$SFCT	Compute fuel flow for given engine rating, Mach number and altitude.
F$SLND	Compute landing distance.
F$STHR	Interpolate air flow for a given engine rating, Mach number and altitude.
F$STOA	Compute takeoff airborne phase.
F$STOB	Compute brake distance after aborted takeoff.
F$STOR	Compute takeoff rotation phase.
F$THST	Interpolate thrust for given engine rating, Mach number and altitude.
F$UDEV	Compute derived gust velocity for a given altitude.
F$VMAX	Compute maximum speed.

Summary

Advances in computer/graphics technology have prompted a change in the scope of computer applications in design and engineering. The aerospace industry in particular has discerned the significance of integrated computer applications in the aircraft design and manufacturing process. Computer-Aided Engineering (CAE) has become an invaluable tool for the design and development of todays modern and complex aircraft systems.

This dissertation describes the development and application of the Aircraft Design and Analysis System (ADAS) as implemented on the central CAD-system at the Delft University of Technology (DUT). The current pilot-version of ADAS is the result of a 4-year research project at the Faculty of Aerospace Engineering, which was initiated in 1983 with the objective to develop a computer-aided system for conceptual aircraft design and to assess the practical capabilities and pitfalls.

Basically, design is an iterative process, where in successive cycles the design configuration is changed and improved until a satisfactory design solution is obtained. A design cycle in turn can logically be divided into 3 steps:

Design definition: Information is acquired to define a design configuration in sufficient detail.

Design analysis: The design is analyzed by computing selected design characteristics using suitable prediction methods.

Design evaluation: Analysis results are represented and the design is evaluated by comparing them with the design requirements and objectives.

Based on this schematization of the design process, an ADAS-architecture was conceived, consisting of 3 self-contained programs:

ADAS: Control of interactive user-system communication through a command-oriented language, primarily for data pre- and postprocessing.

ADAP: An executive program to control processing of user-supplied analysis programs. The ADAP-program is self-contained and can optionally run in a batch mode. Optionally, ADAP can perform parameter variation and/or multivariate optimization.

MEDUSA: a general-purpose drafting and modelling system to draw and represent a design configuration layout. An interface has been developed to exchange geometry information between the ADAP-program and a MEDUSA sheet file.

These programs can exchange information through a simple database system.

ADAS intends to provide the designer with a computer-assisted working environment, wherein design data and analysis methods can be manipulated in a flexible way. Only specific routine and traditionally manual tasks are automated, while aspects related to process flow, design decisions, selection and use of prediction methods, etc., are left to the designer. Specific analysis codes may be developed and can be easily interfaced with ADAS. General information, such as engine and airfoil data, can be stored in a library. Optionally, standard program modules may be stored in a program library from where they are available to any ADAS-user through a simple subprogram call.

The dissertation is subdivided into 4 parts. The first part gives a general introduction to the field of computer-aided aircraft design, with an historical overview and a assessment of the current status in computer applications in conceptual and preliminary design. Part 2 describes the Aicraft Design and Analysis System with emphasis on informatical and computer-technical aspects. Part 3 gives examples of analysis methods currently implemented in the ADAS program library. Finally, part 4 illustrates how ADAS can be employed to solve a typical design optimization problem.

Samenvatting

De rol van de computer in de huidige praktijk van het ontwerpen en fabriceren van vliegtuigen is de afgelopen decennia duidelijk van karakter veranderd. Naast de traditionele funktie, die van een snel rekenappraraat voor het oplossen van complexe numerieke berekeningen, is, vooral door de ontwikkeling van interaktief grafische technieken, de computer een belangrijk hulpmiddel voor de ontwerper geworden. De totale integratie van de computer in het totale ontwerpproces wordt in het algemeen aangeduid met Computer-Aided Engineering (CAE).

In 1983 werd aan de Faculteit der Luchtvaart- en Ruimtevaarttechniek van de Technische Universiteit Delft (TUD) een onderzoeksproject geëntameerd voor de ontwikkeling van een systeem voor computergesteund ontwerpen van vliegtuigen. Dit proefschrift geeft een beschrijving, met toepassingsvoorbeelden, van een experimenteel ontwerpsysteem welke in dit kader is ontwikkeld. Dit systeem wordt gerefereerd met ADAS: Aircraft Design and Analysis System.

De algemene opbouw van ADAS is gebaseerd op het feit dat een vliegtuigontwerp in het algemeen langs iteratieve weg tot stand komt. In iedere ontwerpronde kunnen daarbij 3 stappen worden onderscheiden:

Ontwerp defintitie: De geometrie van een ontwerp konfiguratie wordt gedefinieerd en ingevoerd;

Ontwerp analyse: De gegeven ontwerp konfiguratie wordt geanalyseerd en gespecificeerde eigenschappen worden berekend met behulp van beschikbare rekenmethoden;

Ontwerp evaluatie: De rekenresultaten van de ontwerp analyse worden vergeleken met de gewenste eigenschappen, op basis waarvan kan worden besloten de konfiguratie al dan niet te wijzigen.

Op basis hiervan is een ADAS-systeem architektuur ontwikkeld welke wordt gekenmerkt door 3 onderling onafhankelijke programma's met ieder een specifieke funktie:

ADAS: Het programma ADAS bestuurt de interaktie tussen de ontwerper en de specifieke ADAS-funkties met name invoer en verwerking van gegevens, zoals het ondervragen van gegevensbestanden, het genereren van grafieken en tabellen, enz.

ADAP: ADAP staat voor Aircraft Design and Analysis Program. Het is een niet-interaktief programma en bestuurt de eigenlijke berekeningen op basis van gespecificeerde rekenmethoden. Verder bevat ADAP opties om parameter variatie en/of numerieke optimalisatie uit te voeren.

MEDUSA: MEDUSA is een commercieel beschikbaar teken- en modellering systeem. Er is een koppeling gerealiseerd tussen ADAP en MEDUSA zodat geometrische gegevens uit een konfiguratie tekening kunnen worden afgeleid.

Bij de ontwikkeling van ADAS is getracht een zo goed mogelijk kompromis te realiseren tussen enerzijds een grote mate van flexibiliteit en toegankelijkheid, waarmee de inbreng van de ontwerper in de vorm van kennis, ervaring en kreativiteit, is gewaarborgd, en anderzijds de voordelen van computer/grafische technieken te gebruiken voor het assisteren of automatiseren van handmatige en herhalende taken en werkzaamheden. ADAS kan daarom het beste worden omschreven als computergesteund ontwerpomgeving waarbinnen de ontwerper op een efficiente wijze met ontwerpgegevens en rekenmethoden kan manipuleren. Daarnaast bevat het systeem mogelijkheden om gegevens van motoren, profielen en ook ontwerpmethodieken in een bibliotheek op te bergen of te gebruiken.

Dit proefschrift bestaat uit 4 delen. Deel 1 deel geeft, vanuit een historisch overzicht, een algemene inleiding op het gebied van computergesteund koncept-ontwerpen van vliegtuigen. Deel 2 bevat een beschrijving van het ADAS-systeem met nadruk op de computer-technische en informatica aspecten. Deel 3 geeft een aantal voorbeelden van ontwerpmethodieken zoals deze op dit moment in de programma bibliotheek zijn geimplementeerd. Als voorbeeld van een toepassing van ADAS, is in deel 4, op basis van een hypothetische ontwerpspecifikatie, een typische ontwerpprobleem geformuleerd. Dit ontwerp is vervolgens met ADAS geanalyseerd en geoptimaliseerd.

Curriculum Vitae

Cees Bil was born on November 5, 1955 in Oud-Beijerland, The Netherlands. He attended the Peter Stuyvesant College in Curaçao, Dutch Antilles, and the Zernike College secondary school in Groningen, The Netherlands. From 1975 until 1981 he studied at the Faculty of Aerospace Engineering at the Delft University of Technology. In 1979 he became student-assistent to prof. ir. E. Torenbeek under whose supervision he graduated as 'ingenieur' on the subject of analytical and numerical aircraft design optimization. In 1981 he participated in the Lockheed International Research Institute exchange program and worked at the systems engineering department of the Lockheed-Georgia Company for a period of 6 months. Following his graduation he became research-assistent at the Faculty of Aerospace Engineering of the Delft University Technology. From 1986 he is assistent-professor.