TU Delft

Faculty of Aerospace Engineering

MEMORANDUM M-865

MSC/NASTRAN Static Aeroelastic Analysis Using the MDO FE-Model

G.J. Rollema

February 1999

Delft University of Technology
Faculty of Aerospace Engineering
Delft, The Netherlands

CONTENTS

LIST OF TABLES

LIST OF APPENDICES

NOMENCLATURE

AEG	AEroelastic model Generator	
AOA	Angle of Attack	
BAe	British Aerospace Ltd	
c	Chord	[m]
CC	Case Control : Part of MSC/NASTRAN code	
Cd	Drag Coefficient	[-]
Cl	Lift Coefficient	[-]
cp	Pressure Coefficient	[-]
Cm	Moment Coefficient	[-]
CFD	Computer Fluid Dynamics	
DEF	DEFinition of general arrangement of aircraft	
DLM	Double-Lattice Method	
DMI	Direct Matrix Input (MSC/NASTRAN entry card)	
FE	Finite Element	
FEG	Finite Element Generator	
FEG-comp	Finite Element Generator Composites Version	
FEGPP	Finite Element Generator Post Processor	
ICAD	Interactive Computer Aided Design [Ref.5]	
M	Mach Number	[-]
mac	Mean Aerodynamic Chord	[m]
MDO	Multi-disciplinary Design Optimisation	
MCP	Manufacturing Cost Prediction	
MMG	Multi Model Generator	
M.Sc.	Master of Science	
MTOW	Maximum Take Off Weight	
NASTPP	Nastran Post Processor	
MSC/NASTRAN	Finite Element Solver [Ref. 4]	
PROPPP	PROPerty card Post Processor	
q	Dynamic Pressure	[N/m^2]
SOL	SOLution : MSC/NASTRAN solution method identification	
SPC	Single Point Constraint : Part of MSC/NASTRAN code	
SSG	Surface Shape Generator	
SLLS	Structural Loads, Layouts and Sizes	
SLLS -comp	Structural Loads, Layouts and Sizes Composites Version	
T	Temperature	[K]
TDMB	Technical Data base Manager and Browser	
TOSCA	TDMB Optimisation Solution Control Agent	
X	Position in Reference Axis System	[m]
Y	Position in Reference Axis System	[m]
Z	Position in Reference Axis System	[m]

ABSTRACT

This M.Sc. thesis built upon the "Multi-Disciplinary Design, Analysis and Optimisation of Aerospace Vehicles Project (MDO-project)". The main objectives of the thesis were to use MSC/NASTRAN to perform static aeroelastic analysis on a FE-model of an aircraft and to create facilities within the MDO-software capable of handling the results from theses analysis.

The FE-model used for the static aeroelastic analysis was generated by the MDO-software. This FE-model incorporated a simple aerodynamic model representing wing and tailplane. The MDO-software does not generate the fuselage aerodynamic model so this was added manually. The results from the static aeroelastic analysis consisted of aerodynamic pressures, forces, and moments acting on the aerodynamic models during specific flight conditions. These results are written to a special output file by MSC/NASTRAN. The MDO software was supplemented so that these results could be read from the output file, processed and put into the common data base. The data base could then be used to derive lift distributions over the wing and fuselage. During the project a lot of FE-models have been evaluated to proof the validity of aerodynamic model used. This process let to a final static aeroelastic model. Validation of the final static aeroelastic model was performed by comparing the results from the MSC/NASTRAN static aeroelastic analysis with results from CFD calculations.

The modified MDO-software is now capable of reading and processing the results from the MSC/NASTRAN static aeroelastic analysis. The MDO-software can be used to generate the FE-model needed for these analysis, but the fuselage still has to be modelled manually.
The comparison of the results from the CFD calculations and the MSC/NASTRAN static aeroelastic analysis show that the lift distributions are not matching. The twist and camber distributions in the aerodynamic model of the wing have a big influence on the MSC/NASTRAN static aeroelastic results. The project ended before detailed analysis of the influence of twist and camber on the MSC/NASTRAN results could be performed. It is expected that careful modelling of the twist and camber (matching that of the CFD-model) will give a reasonable match with the CFD results. This work will have to be carried out before MSC/NASTRAN static aeroelastic analysis generates the expected results. The modified MDO-software then will give the engineer a powerful tool to quickly generate loads on the structure of the aircraft for various flight conditions and compare the results with other data.

1. INTRODUCTION

In the final year of study at Delft University of Technology (Faculty Aerospace Engineering, Structural Design Department C1), a six month M.Sc. thesis project has to be conducted. This report is the result of the work performed during this period and concludes the thesis.

My M.Sc. thesis project is a co-operation between Delft University of Technology and British Aerospace, and is an extension of the Multi-disciplinary Design and Optimisation Project (MDO-project). One of the results from the MDO-project was a large software package capable of generating Finite Element models of aircraft.. My M.Sc. thesis will focus on MSC/NASTRAN static aeroelastic analyses using the FE-model generated by the MDO-Software..

To be able to perform MSC/NASTRAN static aeroelastic analysis, the FE-model, as generated by the MDO-software, is supplemented with an aerodynamic model representing the fuselage. The MDO-project software is supplemented and adjusted to give it the capability to read and process the specific results from the static aeroelastic analyses. Validation of the static aeroelastic FE-model is achieved by comparing the results from the analyses with results from CFD calculations.

This report continues as follows : Chapter Two describes the subject of the thesis project. Chapter Three is an introduction into the MDO-project. Chapter Four describes how MSC/NASTRAN performs static aeroelastic analysis. In chapter Five the initial results are discussed. Chapter Six describes the amendments and supplements made to the MDO-Software. In Chapter Seven the modelling of the fuselage is discussed. Chapter Eight compares the results of MSC/NASTRAN with CFD-data. Conclusions and recommendations follow in Chapter 9.

2. THESIS SUBJECT

Delft University of Technology together with British Aerospace and 13 other European organisations, companies and universities were involved in the project "Multi-Disciplinary Design, Analysis and Optimisation of Aerospace Vehicles" or MDO-project [Ref. 1]. The MDO-project concluded in January 1998. One of the deliverables of the MDO-Project was a large software package which can generate Finite Element models of an aircraft.

The MDO-project software is capable of handling and processing the results of various MSC/NASTRAN analyses (static, flutter, optimisation) using the FE-model of an aircraft. So far the MDO-software does not incorporate facilities supporting MSC/NASTRAN static aeroelastic analysis. Adding these facilities to the MDO-software is seen as very desirable by both British Aerospace and Delft University. This M.Sc. thesis is an extension of the MDO-project and will focus on the static aeroelastic analyses.

The subject chosen for this M.Sc. thesis:
Comparison of Lift distributions generated by MSC/NASTRAN (Static Aeroelastic Analysis) with results from the BAe Flutter and loads Suite and CFD data. The MSC/NASTRAN FE-model will be modified based on the comparison results.

The MDO-software will be amended and supplemented so that it can generated the necessary plots and data needed for the comparison. Validation of the static aeroelastic model will be achieved by comparing the results from the MSC/NASTRAN static aeroelastic analysis with results from other programs like the BAe Flutter and Load Suite, and with CFD data. The static aeroelastic model will give the engineer a powerful tool to generate loads on the structure of the aircraft for various flight conditions and compare the results with other data.

The first half of the M.Sc. project will be conducted at the University in Delft and the other half at BAe Airbus in Woodford UK. This allows the University and BAe direct input in the project work and gives the student valuable work experience in industry. Results of this thesis work will eventually be made available to all the partners in the MDO-project.

The project objectives and details are written up in a project proposal as shown in Appendix-A.

3. MDO-PROJECT

3.0 Introduction

The MDO-project is the basis of this M.Sc. thesis project. This makes understanding of the working of the MDO-project crucial. As part of this M.Sc. thesis an in depth study into the working of the MDO-project has been completed. The results of this study were reported in Reference 5. This chapter contains parts of this report because it is felt that without it the reader would miss valuable information needed for a better understanding of the rest of this report.
First the MDO-project is introduced, with a summation of the project themes. Then the MDO-project programs are discussed.

3.1 MDO-Project Introduction [Ref. 5]

3.1.1 Introduction

Optimisation has become an integral part of aircraft design. Aircraft manufacturers need to offer aeroplanes with ever improved performance and ever reduced costs and time scales for delivery. These continuously increasing demands are squeezing the preliminary design and development programs for new aircraft projects, both on time and cost scales, whilst demanding right-first-time innovative solutions.
Tools that rapidly produce consistent, high quality designs would enable several concepts to be considered during preliminary design to a level of detail previously reserved for the final design stage. If these tools were integrated with manufacturing and cost engineering methods then it would enable multi-disciplinary optimisation to be exploited in choosing between alternative product concepts, taking into account potential reductions in both design and production costs and lead times as well as performance issues [Ref.3].
The techniques developed within the Multi-disciplinary Design Optimisation-discipline have to date only been applied to problems of academic size. Industrial size problems requiring not only inclusion of multiple disciplines, but also participation of several co-operate partners are being addressed in the "MDO-project" (Multi-disciplinary Design, Analysis and Optimisation of Aerospace Vehicles). This is a collaboration between fifteen European aerospace companies, research establishments and universities: British Aerospace, Aerospatiale, DASA, Fokker, Saab, CASA,. Alenia Aermacchi, HAI, NLR, DRA, ONERA and the Universities of Delft and Cranfield. The project is managed by British Aerospace and funded by the European Commission under the Brite-Euram initiative [Ref.1].

3.1.2 Project Themes

For the MDO-project a number of research themes have been identified and adopted as the foci of the project. These themes cover the following aeronautical science challenges [Ref. 1]:

- Derivation of aerodynamic sensitivities, for example the sensitivity of drag to surface shape, required to enable integration of aerodynamic methods into an optimisation process.
- Integration of aeroelastic based loads derivation methods into the analysis and optimisation process, including the effect of the control system on the loads in the aircraft structure.

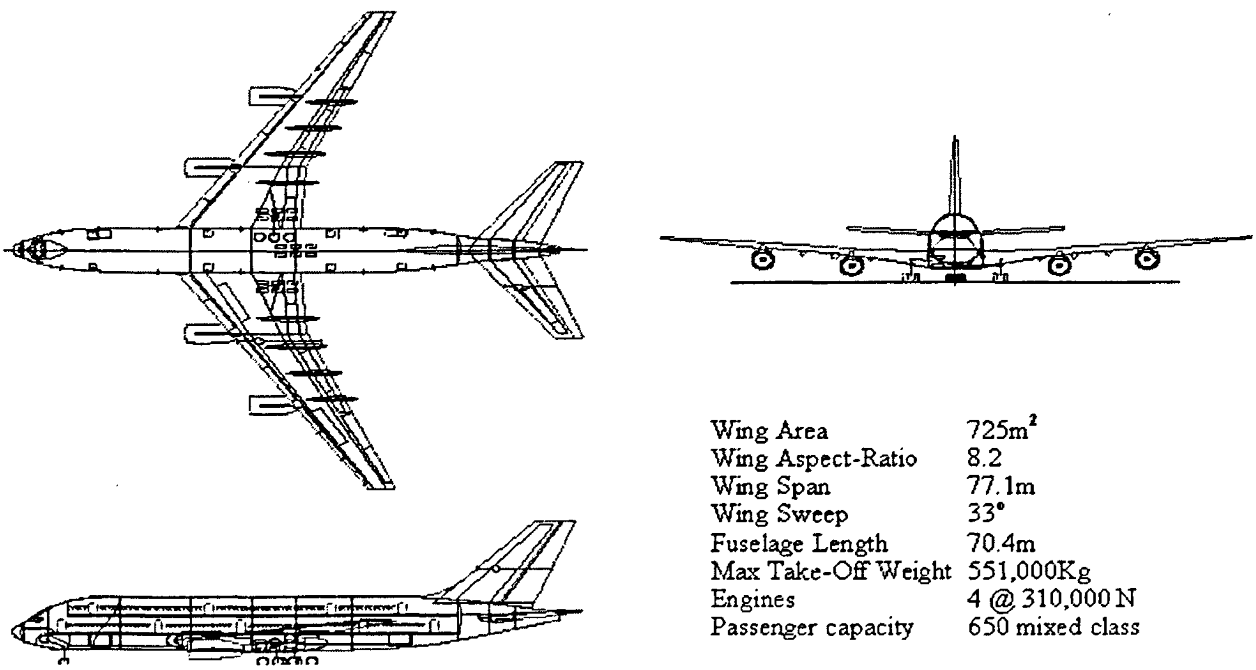
- The application of structural mass/sizing optimisation tools, in the higher level processes of surface shape/size optimisation, with structural topology related design variables, while introducing realistic modelling based producibility and manufacturing costs considerations.
- The formulation of design variables, constraints and cost functions in defining a well posed optimisation problem that captures real aircraft design issues.

And the project covers the following computational challenges:

- Control of the exchange of data, design, analysis and sensitivity data, between the various analysis methods, the optimiser and the user.
- Formulation and implementation of computationally cheap algorithms to suitably determine sensitivity of the cost-function with respect to the design variables.
- Control of the sequence of numerically intensive analysis calculations, and the distributed processing of these calculations.
- End-user visualisation, interpretations and control of the MDO-process.

3.1.3 Reference Aircraft

The MDO-project maintains a strong focus on the validation and practical exploration of the research work by the adoption of the common reference aircraft MDO-design problem. This reference aircraft design problem is the basis for all the research work, providing a datum for validation of the methodologies through the critical comparison and appraisal of results between partners, and also providing a basis for critical appraisal of the value of these results in an aircraft design program context. The reference aircraft is representative of a next generation 500-600 seat commercial aircraft (see Appendix-N) [Ref. 2]. Figure 3.1 shows the reference aircraft.



Wing Area	725m^2
Wing Aspect-Ratio	8.2
Wing Span	77.1m
Wing Sweep	33°
Fuselage Length	70.4m
Max Take-Off Weight	551,000Kg
Engines	4 @ 310,000 N
Passenger capacity	650 mixed class

***Figure 3.1** : Reference Aircraft*

3.2 MDO-Programs [Ref. 5]

One of the deliverables of the MDO-project is a comprehensive software package that consists of the following components [Ref. 2]:

- *MMG* : Multi Model Generator.
- *TDMB* : Technical Data Modeler and Browser.
- *TOSCA* : TDMB Optimisation Solution Control Agent.
- *Utilities* : Pre and post-processing programs to support MMG.
- *Documents* : Each component comes with documents written in html-code which can be read using any internet browser software.

The components of the MDO-project will be briefly discussed in the next paragraphs. For a more in depth discussion the reader is referred to Reference 5.

3.2.1 MMG : Multi Model Generator

The MMG co-ordinates the execution of a number of aircraft design, definition and model generation programs to create the multi-discipline analysis models required for assessing the overall aircraft performance. The sequence of operations of modules within the MMG is as follows:

1. *DEF* : DEFinition of general arrangement of Aircraft
2. *SSG* : Surface Shape Generation of the wing
3. *SLLS :* Structural Layout Loads and Sizing of the wing
4. *FEG :* Finite Element Generator of the wing torsion box
5. *AEG* : Aeroelastic Model Generation of the whole aircraft
6. *MCP* : Manufacturing Cost Prediction

Each of the modules has a specific task to perform and some of the modules have a number of operating options. The modules cannot be executed before its previous module has been executed.The data which every module needs is read from and written to a common data base. The MMG is operated with one command after which the MMG controls the execution of the different modules.

DEF- module

The DEF-module (DEFinition of general arrangement of aircraft) generates the detailed definition of the geometry of the wing, fin and tail plane given a top level specification of the aircraft in parameters such as wing area, wing aspect ratio etc. The module uses the minimum specification needed to specify the aircraft and derives additional values that are important in defining the detailed geometry of the aircraft general arrangement. Key wing planform parameters determined are for example : co-ordinates defining the leading and trailing edges. Further the module calculates wing position on the aircraft. The module reads data from the common database : mdo-data.tdm and writes its results back to that database.

SSG-module

The SSG-module (Surface Shape Generator) creates the wing surface shape that becomes the basis for geometric definition of all simulation models. The module uses reference wing sections and maps these to the wing planform subject to an overall spanwise wing-thickness distribution and specific dihedral. The module creates a set of straight-line. surfaces that together define the complete wing (including centre section and rounded wing tips).

The module reads data from the common database: mdo-data.tdm, but also needs fuselage geometry data from a separate data file : fuselage.dat to be able to calculate the fuselage-wing intersection. The module writes the results back to the database.

SLLS-module

The SLLS-module (Structural Lay-out Loads & Sizes) calculates the loads on the wing. It estimates the allowable stress levels at each wing rib station and makes a preliminary estimate of the required structural sizes.

The SLLS calculates loads and sizes which are sensitive to wing design variables such as aspect ratio, wing area, sweepangle. The wing platform is divided into a large number of streamwise strips. The aerodynamic and inertia loads on each of these strips are calculated separately. After adding both contributions, the resulting shear forces and pitching moments are integrated along the span and interpolated to rib positions. The preliminary sizing routine estimates allowable stress levels based on buckling criteria. Fixed stress levels are used to represent material and fatigue criteria. The applied loading intensities are calculated from the loads acting in a wingbox cross-section and a simplified I-beam representation of the wing box. The required thickness follows from the ratios of applied loading intensities to the allowable stress levels. To include aeroelastic effects, the twist distribution of the wing is updated with the estimated deformation of the wing. The effect of the modified twist distribution on the spanwise lift distribution is evaluated in a number of iterative loops. The module reads data from the common database : mdo-data.tdm and writes its results back to that database.

FEG-module

The FEG-module (Finite Element Generation) creates a finite element model of the wing torsion box and calculates the forces acting on the model reference axis. The model and the forces are then written to an output file in MSC/NASTRAN format. This file also contains the definition of the structural sizing variables and the allowable stresses.

The module computes the nodal co-ordinates of all rib-stringer intersection points on the upper and lower skins. The torsion box model geometry is completely defined by these points. The nodes are connected using membrane elements (CQUAD4) for the skins, rod elements (RODS) for the stringers, ribwebs and sparweb stiffeners. The loads are converted from TDMB to forces acting on special load introduction grids. These grids are connected to the corners of each rib with RBE3 elements. The module generates DESVAR cards to define 156 structural design variables. These variables can be optimised in a later stage. The module reads data from the common database : mdo-data.tdm and writes its results back to that database. In addition to that the module writes the FE-model to feg.out.

AEG-module
The AEG-module (Aeroelastic Model Generation) generates a Finite Element representation of the mass and stiffness of fuselage, fin, tail plane and engines. The module then connects this with the detailed wing structure FE-model as generated by de FEG-module. Aerodynamic panelling for the wing, engines, fin and tail are generated completing the aeroelastic model.

MCP-module
The MCP module (Manufacturing Cost and Producibility) generates design information for the main structural components of the wingbox to complete to complete an preliminary feature-based manufacturing cost and produceability assessment. This module is currently implemented in FEG-comp and will only be called if FEG-comp is called.

3.2.2. TDMB Technical Data Modeller & Browser

The TDMB has been developed to support the integration of technical computing tools. The TDMB caters for the interactive description of generic product data relating to aircraft assemblies, parts or abstract definitions and also for the management of case data. TDMB supports both interactive browsing and programmatic access to the data via both Fortran and C subroutine libraries enabling the range of technical computing tools to communicate through a common database. Components of the TDMB :

- ***Browser*** : This is the primary tool for browsing and editing technical data models and associated data. Via the browser data can be viewed, edited, added and saved.
- ***Database*** : This is a set of files on the file system that contains the data which can be browsed through with the browser. The data structure is object orientated comprising a tree structured network of information nodes. Each node has a name and description. Individual nodes can be grouped together and form a higher level generic data structure which can then be grouped to nested data structures. The internal structure of the database can be considered as a filing system and is used at a high level to partition data and to facilitate and control access. The structure of the data base is independent of the TDMB utility allowing multiple independent databases.
- ***Kernel*** : Software that reads and writes data from and to the database
- ***Graphics*** : The graphical display is essential for effectively understanding complex data. The graphical display utility within the TDMB allows plotting of data in 2D and 3D and is closely connected with the data in the database. Numbers, titles etc. can be directly copied out of the database.
- ***API*** : Application Programming Interface. This is a collection of routines, called from a user written fortran of C-program allowing Kernel to be used for direct interrogation and modification of the database.

The TDMB takes care of the translation of data from analysis variables to design variables and vice versa. It also provides the access to the data for the user in any stage of the design process.

3.2.3 TOSCA-TDMB Optimisation Solution Control Agent

The TDMB Optimisation Solution Control Agent-TOSCA is an utility that co-ordinates the solution of a design optimisation problem. The design variables, product model, and performance criteria are all defined in a Technical Data Model, which is accessed and managed using the TDMB.
TOSCA acts like an assistant or Agent to the user. It executes as a background process controlling the execution of product analysis tools to determine overall product performance, and controlling the execution of optimisers to recommend new designs. In this way, having specified a design problem and solution strategy, the design engineer is freed from the mundane issues of job control and file management. He can focus his time and attention on the more challenging issues of the engineering validity of the product design ideas he is developing, and the next concepts that he needs to investigate.
TOSCA examines the status of data relating to the definition and solution of a design problem and it determines what needs to be done next, for example, controlling calculations to determine product performance or asking an optimiser to recommend new designs. In this way, the tool acts as an "agent" or "assistant" to the design engineer - progressing the routine work and releasing the engineer to focus his intellect and time on more creative and challenging aspects!

3.2.4 MDO Utility Programs

The following utilities are a mix of analysis and pre- and post-processing operations that support the operation of the MMG (beside the programs that were already mentioned as part of the MMG) with specific analysis tools and integration of simulation results back into the TDMB Integrated Product Model.

- AERO2TDMB- example aerodynamic analysis to TDMB results post-processor. The program for the aerodynamic analysis is not part of the MDO-project and results are given here as example only.
- TRIM - analysis of aerodynamic trim and balance. This module needs to run after AERO2TDMB.
- FEGPP - MSC/NASTRAN model pre-processor to match property cards with design variables.
- PROPPP - MSC/NASTRAN model pre-processor, including property card rationalisation.
- FULLMODEL - MSC/NASTRAN model pre-processor, to reflect half-model creating a complete aircraft FE model.
- NASTPP - MSC/NASTRAN to TDMB results post-processor.

For this M.Sc. project the only the NASTRAN related modules are of interest and discussed further.

FEGPP-module
The FEGPP utility will update the property cards in the MSC/NASTRAN Bulk Data Deck feg.out file produced by the FEG-module to be consistent with either the initial or final design variable values of the optimisation model.
The wing box structure FE-model created by the FEG-module features structural members that are sized at each rib station giving a continuously varying thickness spanwise, and fixed thickness chordwise at each station.

The optimisation model defines design variables that group rib-bays together spanwise and splits them chordwise and then defines structural member sizes for the elements associated with each design variable. The two sets of member sizes are therefore not consistent.
The FEGPP utility will operate in one of two modes according to whether the final design variable values resulting from an optimisation run have been included in the TDMB database or not :

- Initial Values Only - if the optimisation results are not available, then FEGPP will read the initial design variable values specified in the feg.out file, and according to the design variable-property relationship cards, will update all structural member properties so that the initial structural model and optimisation model are fully self-consistent. The file ifeg.out will be created.
- Initial & Final Values - if NASTPP has been run and the optimisation results are available then FEGPP will create to new MSC/NASTRAN bulk data decks :-
  1. ifeg.out - property cards updated according to the Initial Value of design variables
  2. ffeg.out - property cards updated according to Final Optimised Value of design variables, and final design variable values also substituted in on the design variable cards.

In each case the design variable property relationship cards will be used to determine the appropriate values for property cards.

PROPPP-module

The PROPPP utility will rationalise and update the property cards in the NASTRAN Bulk Data Deck, feg.out file produced by the FEG-module to be consistent with either the initial or final design variable values of the optimisation model. The rationalisation is particularly important in reducing optimisation run times.
The wing box structure FE-model created by the FEG-module features structural members that are sized at each rib station giving a continuously varying thickness spanwise, and fixed thickness chordwise at each station. The model features one property card per element - although this anticipates support of models with continuously tailored chordwise and spanwise member sizes it is inefficient for the current model standard. This is the difference with FEGPP. The rest the PROPPP-module works in the same way as the FEGPP-module

FULLMODEL-module

This module reads the half model FE-model generated by either FEG, FEGPP or NASTPP and creates the other half of the FE-model so that a full aircraft FE model is created. This makes the model suitable for simultaneous symmetric and anti-symmetric analysis and optimisation.

NASTPP-module

The NASTPP utility will post-process the main MSC/NASTRAN output file and extract the convergence history and final design variable values back into the TDMB integrated product model. The FEGPP or PROPPP-utility can then be used to update the NASTRAN bulk data deck with the optimised design details for further FE-analysis.
The NASTPP utility will read the main MSC/NASTRAN output files for a number of different solution procedures and extract key design information and put it into the TDMB database.
The NASTPP utility can easily be set up to run in a background/batch mode. The utility creates reduced version of the main NASTRAN output files (extension .sum), containing just the key data that it has read and interpreted. If the main output files have been deleted then the utility can still be run on these reduced files - for example if it is required to merge the results into a different TDMB database.

3.3 Conclusions

The MDO-project software package gives the user the opportunity to run complex analysis on aircraft models in an early stage of the design. The software was not written is such a way that any aircraft can be analysed. Detailed data about the conceptual design is necessary for the programs to run (detailed fuselage data, payloads, fuel loads, weight estimations). The software as such is not a design program which designs the aircraft, but it is a program to help the designers to see what happens to the overall aircraft if a certain parameter (like wing area) is changed. The program also allows optimisations to different parameters (weight, cost) and is able to feed back the optimisation information into the design.
For the M.Sc. thesis the MDO-programs will be used to generate the FE-models, retrieve the analyses results from MSC/NASTRAN output files, and publish the result in graphical format.

4. Static Aeroelastic Analysis With MSC/NASTRAN

4.0 Introduction

MSC/NASTRAN (Version 68 and higher) can perform a number of Aeroelastic Analyses : Static Aeroelastic Analysis, Modal Flutter Analysis, Modal Dynamic Aeroelastic Response Analysis and Aeroelastic Optimisation. For this thesis project the focus will be on Static Aeroelastic Analysis.
Static Aeroelastic Analysis deal with the interaction of aerodynamic and structural forces on a flexible structure (like a wing) that results in a redistribution of the aerodynamic loading as a function of airspeed. The aerodynamic load redistributions consequently redistributes the internal structural loads and stresses. The structural load distribution on an elastic vehicle in trimmed flight is determined by solving the equations for static equilibrium. This solution process leads to aerodynamic stability derivatives (like lift and moment coefficients due to rotation of control surfaces), and trim variables (like angle of attack and control surface settings). The process also produces aerodynamic and structural loads, structural deflections and element stresses. MSC/NASTRAN has a special solution sequence for static aeroelastic response calculations : SOL144 [Ref. 6]. This solution sequence uses Doublet-Lattice Method (DML) aerodynamic theory for analysis at subsonic speeds, and the ZONA51 aerodynamic theory for analyses at supersonic speeds.
This chapter discusses the aerodynamic theories, the structural and aerodynamic models, and the necessary entry cards used by MSC/NASTRAN for static aeroelastic analyses.

4.1 Aerodynamic Theories

MSC/NASTRAN can use a number of different aerodynamic theories for the Aeroelastic analyses [Ref.6]:

- Doublet-Lattice subsonic lifting surface theory (DLM)
- Subsonic wing-body interference theory (DLM with slender bodies)
- ZONA51 Supersonic lifting surface theory
- Mach Box method (supersonic)
- Strip Theory (sub- and supersonic)
- Piston Theory (high supersonic)

The usage of a specific theory depends on user preference and the Mach number at which the calculations have to be performed. For subsonic cases the DLM is the better as the Strip Theory is older and less accurate for swept wings. Hereafter only the DLM and DLM slender bodies theories will be discussed.

4.1.1 Doublet-Lattice Subsonic Lifting Surface Theory

The Doublet-Lattice Subsonic Lifting Surface Theory or Doublet-Lattice method (DLM) can be used for interfering, lifting surfaces in a subsonic flow. The theory was presented by Albano and Rodden [Ref. 7], Giesling, Kalman and Rodden [Ref. 8] and Rodden, Giesling and Kalman [Ref. 9] and is only briefly reviewed here.

The theoretical basis of the DLM is lineairised aerodynamic potential theory. The undisturbed flow is uniform and is either steady of varying harmonically. All lifting surfaces (panels) are assumed to lie nearly parallel to the flow. The DLM is an extension of the steady Vortex-Lattice method to unsteady flow.

Each of the interfering surfaces (macro panels) is divided into small trapezoidal lifting elements (sub panels) such that the sub panels are arranged in strips parallel tot the free stream with the surface edges, fold lines and hinge lines lying on the subpanel boundaries.

The unknown lifting pressures are assumed to be concentrated uniformly across the one-quarter chord line of the sub panels. There is one control point per sub panel, centred spanwise on the three-quarter chord line of the box, and the surface normalwash boundary condition is satisfied at each of these points [Ref. 6].

Any number of arbitrary shaped interfering surfaces can be analysed, providing that each is idealised as one or more trapezoidal planes Figure 5.7 gives an example of subpanel division of the wing and tail plane of the reference aircraft.

4.1.2 Subsonic Wing-Body Interference Theory

The Method of Images, along with the Slender Body Theory, has been added to the DLM by Giesling, Kalman and Rodden [Ref. 10]. The DLM is used to represent the configuration of interfering lifting surfaces, while Slender Body Theory is used to represent the lifting characteristics of a body (fuselage, nacelle, external store etc.).

The primary wing body interference is approximated by a system of images of the DLM trailing vortices and doublets within a cylindrical interference body that circumscribes each slender body. The secondary wing-body interference that results from the DLM bound vortices and doublets is accounted for by a line of doublets on the longitudinal axis of each slender body. The boundary conditions of no flow though the lifting surface or through the body lead to the equations for the lifting pressures on the surfaces and for the longitudinal loading on the bodies in terms of down-wash on the wing body combination. An extensive review of the mathematical background of this theory is presented in [Ref.6].

4.2 Modelling

For Aeroelastic Analyses MSC/NASTRAN needs two models: The Structural Model and the Aerodynamic Model. The "communication" between the two models is provided by an automated interpolation procedure.

Structural Model

The Structural Model can be build with any structural Finite Element (except axi-symmetric elements). Stiffness, mass and damping matrices, required by the analysis routines are generated by MSC/NASTRAN from user input of geometry, structural, inertial, and damping data.

Aerodynamic Model
The Aerodynamic Model is built with specific aerodynamic Finite Elements. These elements consist of strips, panels (or boxes), or segments of bodies that are combined to idealise the vehicle for the computation of aerodynamic forces. The elements are defined by their geometry. Their motions are defined by degrees of freedom at the aerodynamic gridpoints. The geometry of the panels is often dictated by the aerodynamic theory used. DLM assumes trapezoidal boxes with their edges parallel to the free stream velocity. Matrices of aerodynamic influence coefficients are computed from the data describing the geometry of the aerodynamic elements. For the DLM the grid points of the aerodynamic model are physically located at the centres of the panels or bodies (other theories use other points on the panel). The aerodynamic grid points do not have to be the same as the structural grid points. The aerodynamic grid points are generated automatically by MSC/NASTRAN from the aerodynamic input data.

Interpolation
The "communication" between the models is provided by an automated interpolation procedure. This allows independent selection of grid points for the structural model and the aerodynamic model in a manner best suited to the particular aerodynamic theory. The interpolation procedure (called splining) relates the aerodynamic to the structural degrees of freedom.
The structural degrees of freedom have been chosen in MSC/NASTRAN as the independent degrees of freedom, the aerodynamic degrees of freedom are dependent. A transformation matrix is derived that relates the dependent degrees of freedom to the independent ones. The transformation matrix is used to transform the structural grid point deflections to aerodynamic grid point deflections. The transpose of this transformation matrix transfers the aerodynamic forces and moments to the structural grid points.
There are three splining methods available

1. ***Linear splines*** : A generalisation of an infinite beam which allows torsional and bending degrees of freedom. The linear spline is a "beam" passing through the known deflections with twist.
2. ***Surface splines*** : Solutions for infinite uniform plates. This is an mathematical tool to find a surface function for all points when the deflection is known for a discrete set of points.
3. ***User defined splines*** : explicit user defined interpolation.

Several splines, including combinations of the three types, may be used in one model

4.3 Nastran Cards for Static Aeroelastic Analysis

The aerodynamic model needed for aeroelastic analysis is modelled using a special set of MSC/NASTRAN entry cards. Whether a specific card is used or not depends on the aerodynamic theory used and on the extent of the FE-model. The following cards are available and/or required for static aeroelastic analysis with MSC/NASTRAN [Ref. 11]

- *CAERO1 :* This card defines an aerodynamic macro panel for use in the DLM. On the card the geometry and position of the macro panel is defined. Also the number of spanwise and chordwise subpanels must be defined. MSC/NASTRAN then generates these subpanels automatically. Any bodies interfering with this macro panel must be defined on a PAERO1 card referred to on this card.

- ***CAERO2 :*** Defines aerodynamic slender body and interference elements for DLM with slender body theory. The length of the body and the position of the nose is defined on the card. Further the number of body and interference panel divisions has to be given for evenly divided bodies. If a user defined division of the body has to be used, then one has to refer to AEFACT cards. One also has to refer to the PAERO2 card.
- ***PAERO1*** : Defines bodies associated with CAERO1 entries.
- ***PAERO2:*** Defines the cross-sectional properties of the aerodynamic bodies. On this card the halfwidths of the body and interference tube and the theta division of the interference tube are defined by referring to AEFACT cards.
- ***AEROS*** : Aerodynamic parameters for steady aerodynamics. The reference chord, span and surface area are defined here. Note: if only half an airplane is modelled, the full span has to be defined but only half the wing area.
- ***AEFACT*** : Specifies a list of real numbers for the aerodynamic model required by the CAEROi and PAEROi cards. This could be half widths, theta divisions, fuselage divisions.
- ***AESURF*** : Specifies an aerodynamic control surface. On this card the number of an AELIST card has to be specified on which the numbers of the subpanels belonging to the control surface are defined.
- ***AELIST :*** Defines aerodynamic (sub) panels associated with a control surface on the AESURF card (static aeroelasticity)
- ***AESTAT*** : This card specifies rigid body motions to be used as trim variables. A number of set labels has to be specified each with a different degree of freedom motion. Example ANGLEA defines the angle of attack of the aircraft. This is a rotational degree of freedom of the y-axis.
- ***AELINK*** : Defines the relationship between or among the AESTAT and AESURF entries. The angle of attack entry on the AESTAT can be related to the position of the elevator on the AESURF
- ***DIVERG*** : Specifies static aeroelastic divergence analysis
- ***PARAM***: Provides scalar values used in performing solutions
- ***SPLINE1*** : Defines a surface spline for interpolating out of plane motion.
- ***SPLINE2***: Defines a beam spline for interpolating panels and bodies for aeroelastic problems. The CAEROi card is defined as is the SETi card. Further the beam stiffness parameters can be set.
- ***SET1*** : Selects structural grid points to be used in the splining of aerodynamics. The nodes on the SET1 cards are connected to the SPLINEi.
- ***SET2*** : Alternative grid selection for splining in terms of aerodynamic macro elements.
- ***SUPPORT*** : Defines determinate reaction degrees of freedom in a free body analysis. The support point on this card is the point on which the plane is balanced for the flight condition.
- ***TRIM*** : Specifies a trim flight condition. Mach number, dynamic pressure and the constrained values for all but one trim variable have to be specified on this card.

5. INITIAL STATIC AEROELASTIC ANALYSES

5.0 Introduction

The non-optimised FE-model of the reference aircraft generated by the MMG will be used to perform the initial Static Aeroelastic Analyses with MSC/NASTRAN. As this model is not yet suitable for static aeroelastic calculations, the model will first be amended and adjusted. The results from the initial calculations will be used to asses the FE-models suitability. Any problems with the model first have to be solved before more detailed analysis can be performed.

5.1 The MDO-FE-model

The MMG generates a FE-model of the reference aircraft (see Appendix-N in the form of MSC/NASTRAN Bulk Data Deck entry cards The FE-model represents half the reference aircraft as symmetry is assumed in the x-z plane of the aircraft. The model consists of the wing structure (torsion box), a stick model for fuselage, tail and fin and the aerodynamic panels for wing, tail, fin and engines. Using the TDMB, variable mass (fuelload and payload) can be added into the model as point masses. In this original form the FE-model is suitable for static, flutter, and optimisation analysis with MSC/NASTRAN (see also chapter 3).

5.1.1 FE-Model Generation

The MDO-FE-model is generated by the MMG. The MMG co-ordinates the execution of a number of aircraft design, definition and model generation programs/modules. Within the MMG, the FEG module creates the Finite Element model of the wing torsion box. This FE-model is then written to a file in MSC/NASTRAN format. The FEG-module also produces the MSC/NASTRAN cards with the definition of the structural sizing variables and the allowable stresses needed for optimisation analysis. The allowable stresses and structural sizes are determined by the SLLS-module. The AEG-module creates a Finite Element stickmodel representation of the mass and stiffness of the fuselage, tailplane and fin in the form of beams and point masses. This stickmodel is connected to the FE-model generated by the FEG-module. Aerodynamic panelling for the wing, engines, tail and fin complete the MDO-FE-Model [Ref. 5].

5.1.2 FE-Model Components

The *structural* FE-model is generated by the FEG-module and partly by the AEG-module and consists of :

- ***Wing half :*** The starboard wing torsion box structure of the reference aircraft is modelled with rod and plate elements. The skin panels are modelled with QUAD4 and TRIA3 elements working as membranes. The stringers, spar cabs and formed rib cabs are represented with ROD elements. The sparwebs are made of ROD and SHEAR elements and the formed rib webs are represented by SHEAR elements. The number of elements is considerable as there are 69 wing ribs and the wing half span is 38.55 m.

- ***Stick model :*** The fuselage, tail and fin are modelled as sticks with CBEAM elements. The properties of the beam elements represent the bending and torsional stiffness of the fuselage, tail and fin structures. The stick model is connected to the wing reference axis via a RBE2 card constraining any motion or rotation between the connecting points.
- ***Wing reference axis :*** The wing reference axis is not physically modelled. It consists of a number of grid points lying along the approximate wing flexural axis. The grid points are positioned in-between the wing ribs and are connected to the wing structure with rigid RBE3 elements. The rigid connection provides the reference axis grid point with a weighted average displacement of the displacement of the grid points to which it is rigidly connected.
- ***Moments and forces*** : The moments and forces used by the MMG to size the initial wing structure are present in the model as FORCE and MOMENT entries. The forces and moments work on the wing reference axis grid points and can be used to perform static strength analysis.
- ***Optimisation cards*** : The FEG module generates DESVAR cards on which 156 design variables are defined. DVPREL1 cards link the values of the variables to element thickness. The structure is divided into 4 chordwise zones and 15 spanwise zones. In each zone the thickness of the wing skin and stringer elements are held in a constant ratio to the associated design variable value. The allowable stress levels are read by FEG from TDMB and written out for each element using DCONSTR cards. A DRESP1 card is used to define the object function in the optimisation : structural weight. With these cards MSC/NASTRAN optimisation solution (SOL 200) can be performed to optimise the objective (in this case the weight of the structure). For this thesis project, the optimisation routine will not be used.
- ***Material properties*** : MAT1 cards are used to specify the material properties. At this moment the complete structure is of the same material. It is however possible to give certain parts of the structure (like the upper and lower wing skin) different material properties.

The *aerodynamic* FE-model is generated by the AEG-module and consists of :

- ***Aerodynamic Panelling*** : The surfaces of the wing, ailerons, tail, fin and engines are modelled with aerodynamic panels on CAERO1 cards. The aerodynamic elements are connected to the structure using beam splines via SPLINE2 cards. The structural grid points to which the spline connects the aerodynamic elements are defined on the SET1 cards.
- ***Trim conditions*** : The trim flight condition is entered on the TRIM card. On this card Mach number, dynamic pressure and the values of the degrees of freedom for the model and the control surfaces are defined. AESTAT cards define the rigid body motions to be used as trim variables.
- ***Mass model :*** The mass of the wing, flaptracks, undercarriage, fuselage, tail, fin, engines, fuelload and payload is modelled as point masses at specific grid points using CONM2 cards. The mass model is particularly important for flutter analysis.

The complete MSC/NASTRAN Bulk Data Deck for the reference aircraft is a very large file : approximately 2.5 megabytes and is thus not shown.

In figures 5.1, 5.2 and 5.3 the FE-model as generated by the MMG is shown.

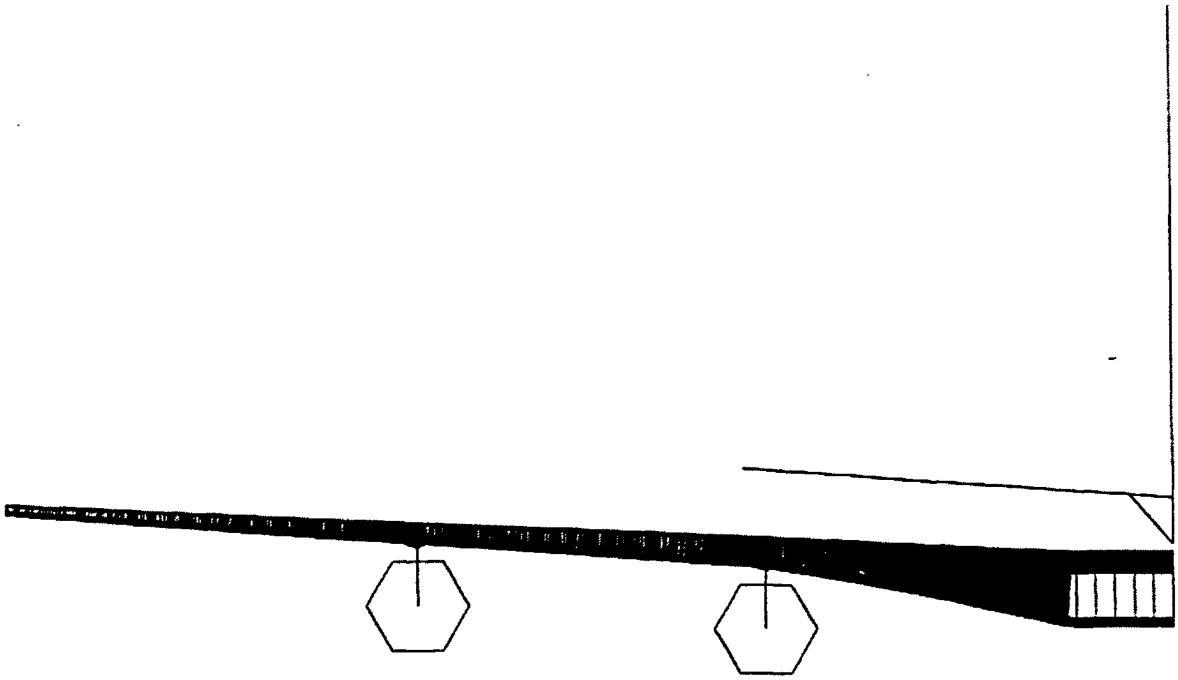


***Figure 5.1** : FE-model Front View*

In figure 5.1 is a front view of the FE-Model. The wing torsion box structure is clearly visible as a dens structure (only the ROD elements are shown here). The aerodynamic panels of the engine nacelles are visible. The stick model of the tail and fin are shown as two lines.

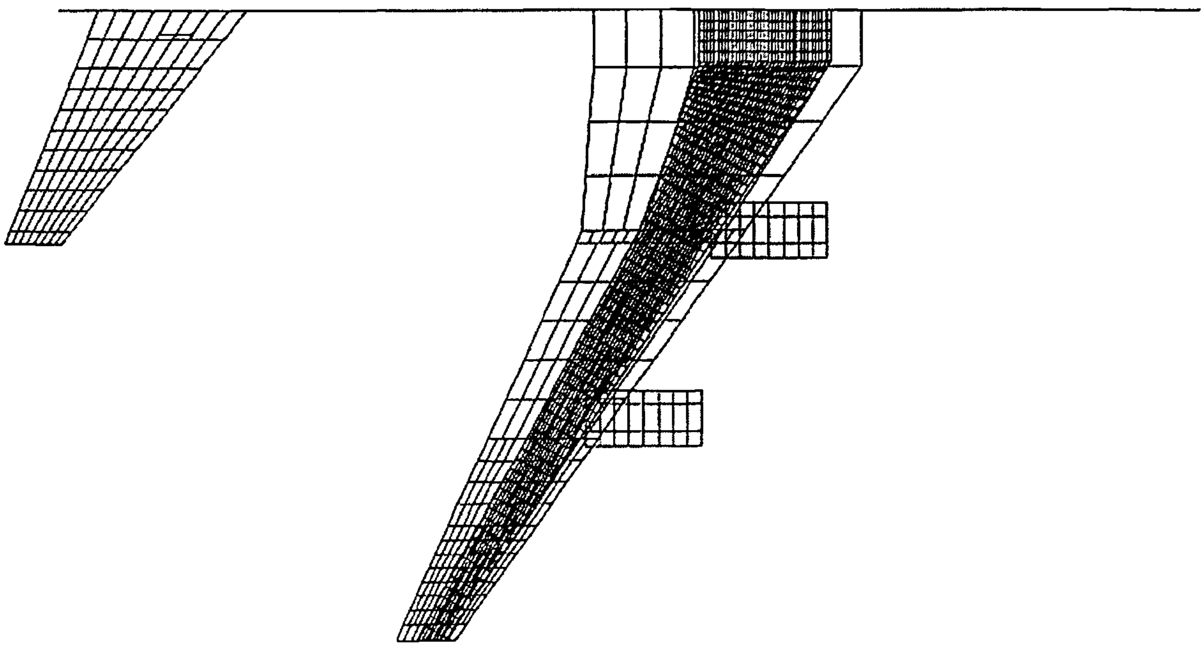


***Figure 5.2** : FE-model Top View*

In figure 5.2 is a top view of the FE-Model. The wing torsion box is visible as ribs and stringers are presented by ROD elements. The stick model of the fuselage is visible as a continues line. The aerodynamic panels of the wing, tailplane and engine nacelles are shown.

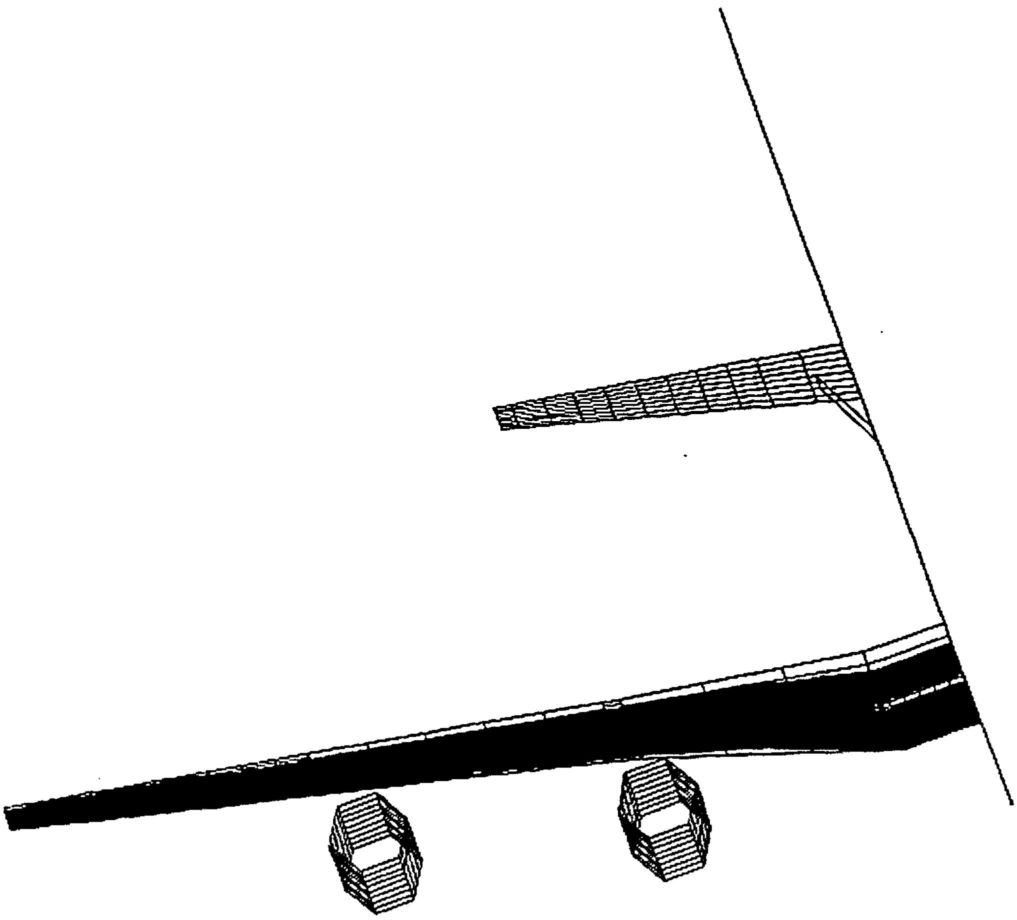


Figure 5.3 : FE-model Viewed from Elevated Point

In figure 5.3 the FE model is shown from an enchained and elevated view point, looking from the front of the aircraft towards the back

5.2 FE-Model Amendments for Static Aeroelastic Analysis

The MDO-FE-model can be used to perform various analysis using MSC/NASTRAN (static, modal, flutter, and optimisation). The FE-model as generated with the MMG comes without Case Control/Executive Control deck because each type of analysis needs its own specific load and constraints cards. These decks are provided separately for static (SOL 100), symmetric and anti symmetric modal (SOL 103), symmetric and anti-symmetric flutter (SOL 145), and optimisation analysis (SOL 200). Within the MDO-software there is no Case Control/Executive Control deck available for static aeroelastic analysis (SOL 144). In order to run SOL 144, a Case Control/Executive Control deck (CC-deck) has been made. This deck is put before the FE-model as generated by the MMG. Also a separate Single Point Constraint deck (SPC-deck) was made which is put at the end of the FE-model. Both decks are discussed in the next sub paragraphs.

5.2.1 CC-Deck

The Case Control/Executive Control deck for the SOL 144 method is shown in Appendix-B. This CC-deck is put before the FE-model as generated by the MMG and controls:

- ***Solution method.*** For Static Aeroelastic Analysis that is SOL 144.
- ***Output control.*** All forces, stresses and displacements will be written to an output file.
- ***Plot commands.*** The plot commands generate a .plt file which can be converted to a .ps file for plotting the structural and aerodynamic elements and deformations. In order to be able to plot the aerodynamic elements in undeformed shape, one has to run the complete deck with MSC/NASTRAN V70.0 (or an earlier version) as V70.5.0 does not plot the aerodynamic elements. The reason for this is unknown at the moment of writing, MSC has been notified of this problem.
- ***Load cases and SPC.*** There are two basic load cases : Subcase 1 which calls TRIM card 10 for the pull_up manoeuvre and Subcase 2 which calls TRIM card 20 for the push_down manoeuvre. The Single Point Constraint (SPC) card called, handles the specific symmetric or a-symmetric constraints for the model. It has been found that the NASTRAN results post-processor NASTPP (See chapter 7) can not handle more than three loadcases. The program returns a memory error if there are more than three loadcases.
- ***Parameter control.*** The parameters control: the Patran output file (.op2), calculate the mass of the aircraft at the support point 100099, and with the aunits parameter the acceleration on the TRIM card can be given in terms of g-units.

5.2.2 SPC-Deck

The Single Point Constraint deck for the SOL 144 method is shown in Appendix-C. This deck is put at the end of the FE-model and handles:

- ***Single Point Constraints (SPCs).*** The loads on the aircraft in both subcases are symmetric. The model has to be constrained so that it simulates a full aircraft. For the root of the wing structure, for example, this means no movement in the y-direction and no rotation about the x- and z-axis.
- ***TRIM cards.*** These control the specific sub case trim conditions. Mach number and dynamic pressure are specified. The degrees of freedom of the model and its control surfaces are restrained on this card.
- ***SUPPORT card.*** This card specifies reference degrees of freedom for rigid body motion. The point at the support card is the 'pivot' point of the whole aircraft. A point at the approximate CG position of the aircraft is generated and rigidly connected to the wingstructrure with RBE2 elements. This ensures that any wing twist measured is absolute zero at the wing root (in this case the wingroot is at the fuselage centre line) and that the fuselage bending is of no influence on the wing twist results.
- ***Elevator cards.*** The elevator cards are not included in the FE-model generated by the MMG. The elevator cards give the tailplane the possibility to pivot around it's stick model axis so that the tail can act as elevator. This is necessary to trim the model for the SOL 144 method. The outcome of the calculations will show an incidence angle for the elevator necessary to trim the model for the given conditions on the TRIM-card.

5.3 Initial Analysis

The non-optimised FE-model as discussed in the previous paragraphs including amendments will be used to do the initial static aeroelastic analysis. A number of different design operations will be analysed under certain flight conditions. The results from these calculations will be used as a reference for the future analyses.

5.3.1 Design Operations

The initial calculations with MSC/NASTRAN SOL 144 have been performed with the MDO-FE-model as generated by the MMG (see chapter 3) including the amendments needed to perform the static aeroelastic analysis (the CC & SPC Decks). The following design operations have been chosen for the static aeroelastic analysis:

- ***+2.5g pull-up manoeuvre.*** This is an important manoeuvre for static strength and relates to the 2.5 g pull_up manoeuvre in the case of an aborted landing at maximum weight. This induces a high combined upward bending and shear load in the inboard wing structure.
- ***-1g push-down manoeuvre.*** This manoeuvre is important for static strength and relates to a -1 g traffic avoidance push_down manoeuvre at maximum weight. This induces a high combined downward bending and shear load in the inboard wing structure.
- ***10 deg/sec roll manoeuvre.*** This manoeuvre is also important for static strength and for roll control. It relates to a pilot induced traffic avoidance manoeuvre at high payload/fuel mass and high equivalent airspeed. This induces a high combined bending and torque load in the wing.

There are many more operations which can be critical for the structural integrity of the aircraft. But the operations which will be investigated represent extreme conditions, and give a good indication of what can be expected in other design operations. Also, to be able to compare the results of the FE-calculations with results from other programs (such as the BAe Flutter and Loads Suite), it is necessary that the same design operations are used. The 10 deg/sec roll manoeuvre is not yet included into the FE-model.

5.3.2 Flight Conditions

The reference aircraft is given a specific payload, fuel load, flight altitude, and speed for the calculations. These settings are edited with the TDMB and influence the FE-model which is produced by the MMG. Data for the reference aircraft is as follows:

- ***Aircraft weight*** : 535.000 kg. This is near MTOW of the aircraft.
- ***Payload*** : 89.600 kg. This is the maximum payload which can be carried.
- ***Fuel load*** :192.000 kg . This is approximately 25 tonnes below maximum fuel load.
- ***Altitude*** : 33.000 ft. This is a typical cruise altitude.
- ***Mach no*** : 0.8 . Typical cruise speed.

5.3.3 Initial Results

The initial static aeroelastic calculations are performed with the non-optimised FE-model as described in chapter 5.1 with the settings as described in chapter 5.3.1 and 5.3.2. At this moment in the project no specific static aeroelastic results (like lift forces on aerodynamic panels) can be shown. The first results are interpreted by looking at the deflection and twist of the wing reference axis. These results are shown in figure 5.4 and figure 5.5.
In figure 5.4 the deflections of the wing reference axis over wingspan for the two subcases (pull_up manoeuvre & push_down manoeuvre) are shown.

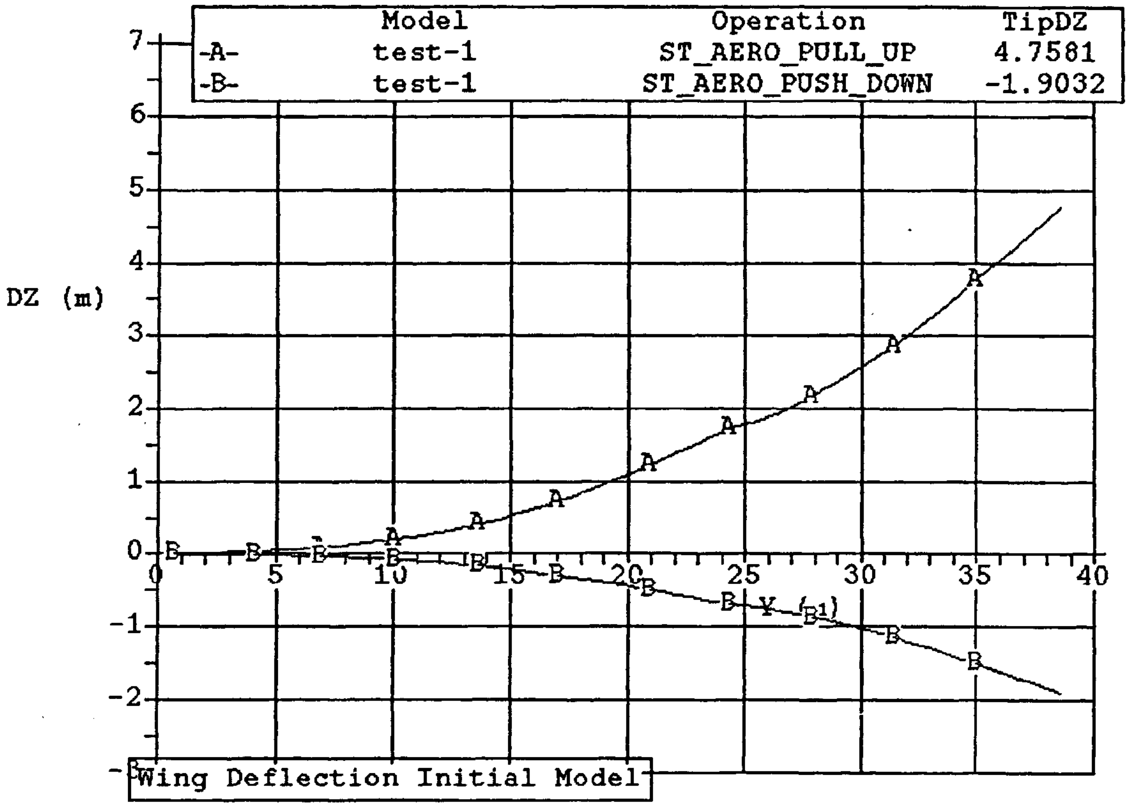


Figure 5.4 : Wing deflection

For the pull_up subcase (black line) the wing deflection is positive and for the push_down subcase (red line) it is negative as expected. From figure 5.4 it can be seen that at the approximate outboard engine position (y = 26 m) there is a dent in the wing deflection.

In figure 5.5 the twist distribution (in radians) of the wing reference axis over the wing span is shown.

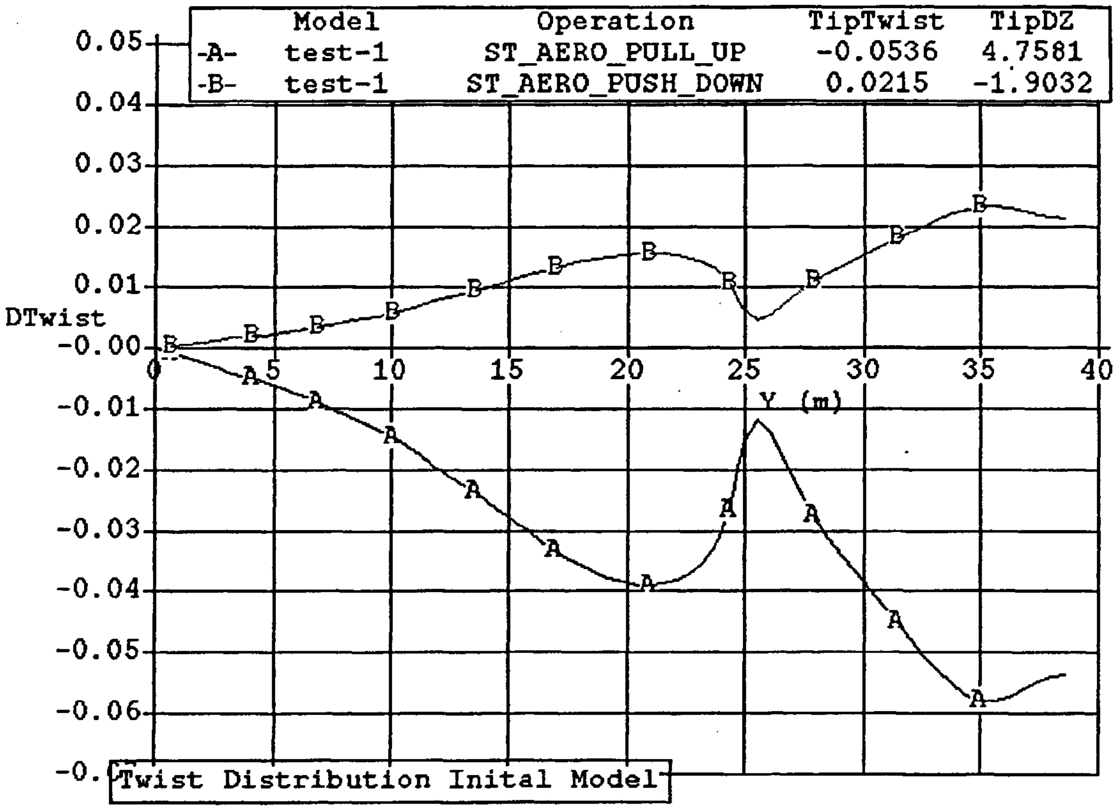


Figure 5.5 : Twist distribution initial calculations

The red line is the twist distribution for the push_down manoeuvre (-1 g), while the black line represents the pull_up manoeuvre (+2.5 g). The overall wing twist is as expected negative for the +2.5 g manoeuvre and positive for the -1 g manoeuvre, but a sudden change in wing twist is visible just before the outboard engine station (y = 26m). From the outboard engine to the wingtip, the twist distribution regains its normal pattern again (see figure 5.5)
The twist distribution as seen in figure 5.5 is very unusual. The expected twist distributions should be more or less linear from root to tip. These results require further investigation to be able to understand the phenomena seen in figure 5.5.

5.4 Problem Investigation

Before any further static aeroelastic analysis runs with the MDO-FE-model can be made, it is necessary to investigate the cause of the unusual twist distribution as seen in figure 5.5. A number of test FE-models will be made to establish the numerical effects of a number of different influences on the twist distribution.

5.4.1. Influences On Twist Distribution

A number of influences (other than the actual construction of the wing torsion box) affect the twist distribution of the wing:

- Engine lift and mass.
- Engine pylon mass.
- Fuel mass and distribution.
- Lift distribution over the wing.
- System masses.

Lift and mass of the engines and pylons cause torsional and bending moments in the wing. Fuel mass, fuel distribution, system masses, and the lift distribution are also of influence on the torsional and bending moments.

5.4.2 Test Models

The influences mentioned above each have their own specific effect on the wing twist distribution. A number of FE-models have been made to investigate whether each influence has the predicted effect, and to see which of the influences could be the cause of the unusual wing twist distribution. Data was altered via the TDMB and then the FE-models were generated by the MMG or the alterations were made directly within the MSC/NASTRAN Bulk Data Deck. The differences in each model are made with respect to the reference model. All the models fly an +2.5 g pull_up manoeuvre for this comparison. For the wing, twist downwards (resulting in a smaller angle of attack for the wing) is negative, and twist upwards is positive. The torsional moment is negative downwards. The different models are:

- ***test-1*** : *Reference aircraft.* This is the FE-model as generated by the MDO-software (version March 1998) with standard TDMB setting amended with the CC and SPC-Deck as described in chapter 3.
- ***test-2*** **:** *Outboard engine mass moved to the CG of the pylon.* As the engine mass is moved towards the flexural axis of the wing, the negative torsional moment the engine mass inflicts on the wing torsion box will decrease. This will lead to an decrease in negative wing twist.
- ***test-3*** : *Reduced outboard engine aero panel length.* The engine nacelle produces lift which is introduced into the wing torsion box as a positive torsional moment. The reduction in aero panel length (from 7.1 meter to 1.1 meter) will lead to a decrease in lift produced by the engine nacelle and thus results in an increase in negative wing twist (for the +2.5 g manoeuvre).
- ***test-4*** : *Increased number of aero panels over the wing.* This model was made to investigate the influence of the panel density over the wing on the twist distribution.
- ***test-5*** **:** *No fuel load*. Although fuel is held in the wing torsion box and its mass has almost no torsional bending moment on the wing, the fuel mass does influence the wing bending and hence the twist. The absence of fuel mass reduces the amount of total lift required, but also reduces the bending relief in the wing. The fuel load is not evenly distributed over the wing torsion box (most of the fuel is carried inboard of the outboard engine), thus the reduction in lift has a greater effect than the disappearance of the bending relief. The wing will bend less and as a result negative wing twist will decrease

- *test-7* : *No outboard engine.* The complete removal of the outboard engine will show the influence the engine as a whole (mass and lift) on the wing twist. The removal of the mass will reduce wing bending, which leads to a decrease in negative twist. The negative torsional moment the engine mass inflicts on the wing torsion box will be gone. This will lead to an decrease in negative wing twist. The removal of the lifting engine nacelle will result in an increase in negative twist. This test will show which effect will be the greater.

5.4.3 Test model Results

The test models from the previous paragraph are all analysed with Nastran V70.5. The results from the analysis are fed back into the TDMB with the NASTPP utility. TDMB provides the plots with the twist distributions. The individual results are shown in figure 5.6. Only the twist distribution (in radians) for the +2.5g pull-up manoeuvre is shown.

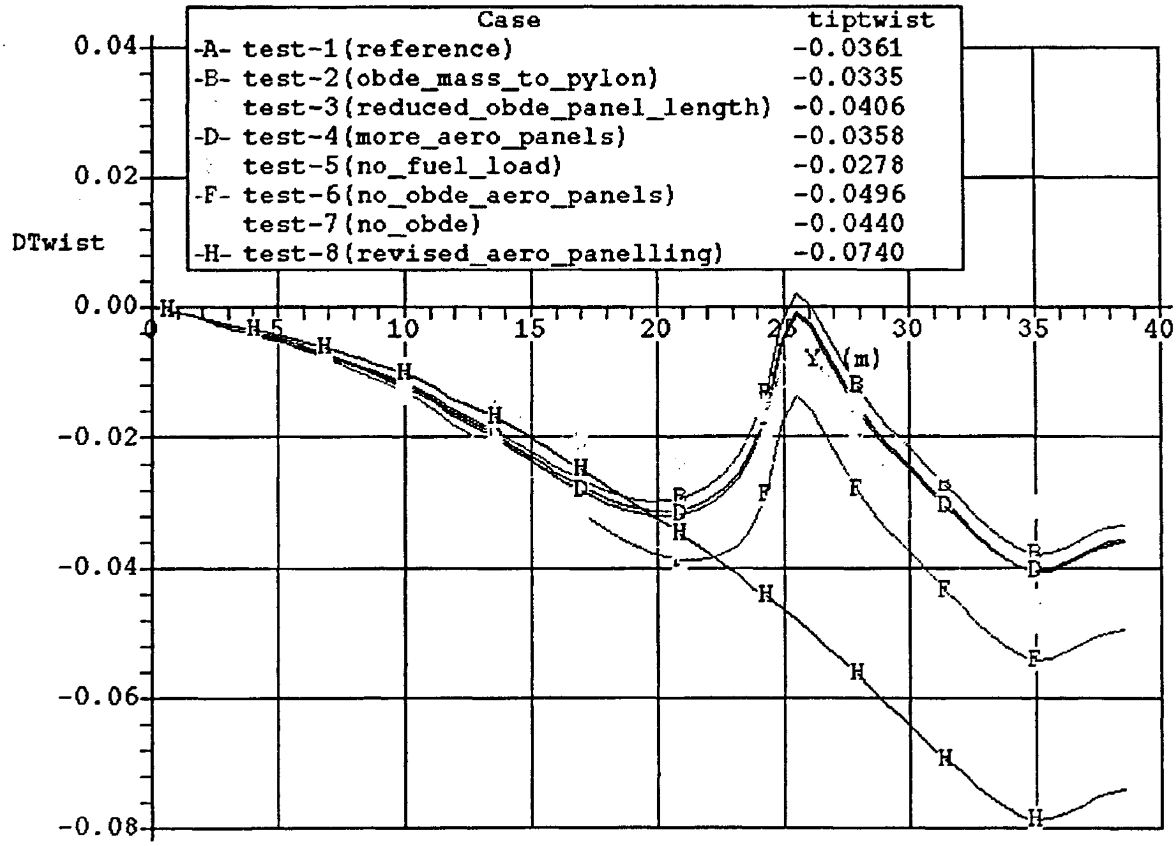


Figure 5.6 : Twist distribution different FE-models

For the wing, twist downwards (resulting in a smaller angle of attack for the wing) is negative, and twist upwards is positive. Per model the results are as follows:

- ***test-1*** : *Reference aircraft.* (Line A): The changes in all the other models are directly related to this reference model. In figure 5.4 line A is the reference aircraft, but the line is completely covered by line D.
- ***test-2*** : *Outboard engine weight moved to pylon CG.* (Line B): The engine mass moves aft towards the wing flexural axis and as predicted the negative twist in the wing is decreased. The influence of the mass move on the wing tip twist is approximately 0.2 deg.
- ***test-3*** : *Reduced outboard engine aero panel length.* (Line C): The reduced length of the aerodynamic panels results in an in creased negative wing twist.
- ***test-4*** : *Increased number of aerodynamic sub-panels over the wing.* (Line D) : Increasing the number of sub-panels over the wing results in a smoother lift distribution over the wing. It has almost no consequences for the twist distribution. Line D is almost exactly the same as line A for the reference aircraft.
- ***test-5*** : *No fuel load.* (Line E) : As predicted, the influence of the reduced lift necessary is greater than that of the increased bending moment due to the absence of fuel. The negative twist is decreased
- ***test-7*** : *No outboard engine.* (Line G): The results show an increase in negative twist. This means that the engine nacelle lift is of greater influence on the twist distribution than the mass of the engine.

The results show that the different FE-models produce the expected changes with respect to the reference aircraft. Twist increases or decreases as predicted, but the overall shape of the twist distribution remains unaffected. All the test FE-models show the unusual wing twist distribution. The conclusion at this point in this investigation is that this unusual wing twist distribution as shown in figure 5.5 and 5.6 is not related to the outboard engine's lift or mass, fuel mass or panel density over the wing.

5.4.4 Problem Definition

The results from testmodel test-4 shows that increasing the number of aerodynamic sub-panels has no effect on the wing twist distribution. To see if the macro panel distribution has any effect, another test case was made:

- **test-8** : *Revised aerodynamic macro panel distribution :* (Line H) : A simple aerodynamic macro panel (CAERO1) distribution over the wingplanform was programmed by hand.

Figure 5.6 shows the results from test-8. The twist distribution now shows an almost constant decreasing twist from root to tip without any major changes in the line. This is a twist distribution as expected from an aircraft wing. The problem thus is connected to the automatically generated aerodynamic panelling.
The results from test-8 require further investigation into the original MDO-FE-model. Within the MMG, the module responsible for the aerodynamic panelling is AEG (Aeroelastic Model Generator). The AEG-module itself will be investigated in the next paragraph. The aerodynamic elements (CAERO1) are connected to the wing's FE-structure via SPLINES. On the SPLINE card a set of nodes is defined with the SET1 card. On the SET1 card the nodes of the structure to which the SPLINE is connected are specified.

Appendix-D shows a part of the original MSC/NASTRAN Aeroelasticity Bulk Data Deck entries as generated by the MMG for the reference test case test-1. It shows the aerodynamic panel division of the wing and the cards connecting the aerodynamic panels to the wing structure.
In Appendix-D it can be seen that CAERO1 1002000 is an aerodynamic macro panel which starts at wingspan position Y1 = 3.51 m and stops at wingspan position Y4 = 13.493 m. The connection of this aerodynamic panel to the wing structure is performed via SPLINE 1002000 on which SET1 102 card is specified. On SET1 102 there are 22 rib number nodes specified (is 22 ribs). The aerodynamic forces of the panels are being introduced at the ribs nodes mentioned in the SET1 card. Panel 1002000 is approximately 10 m long and the forces on this panel go into 22 ribs (the first rib = 106 with y = 3.63 m, last rib on the card = 127 with y = 14.73 m, approximately 11 m long).
The next CAERO1 panel 1003000 starts at 13.493 meter and stops at 14.230 meter. The reason for such a short panel will be explained later. The SET1 card belonging to this panel is 103 and in this SET1 card there are 19! ribs defined (first rib = 127 with y = 14.73 m, last rib = 145 with y = 24.32 m).
The following CAERO1 panel 1004000 starts at wingspan position 14.230 m and stops at 26.215 m and is connected to the structure with SET1 104 (first rib = 145 with y = 24.32 m, last rib = 169 with y = 38.55 m which is the wing tip).
The other outboard aerodynamic panels 1005000, 1006000, 1007000 are all connected to SET1 104.
From Appendix-D it is clear that panel 1003000 is connected with too many ribs, and that panel 1004000 is connected with too little ribs and at the incorrect positions. This causes a very disturbed force introduction into the wingstructure (too much aerodynamic force is introduced into the outer wing section). This is the main reason for the unusual twist distribution. The generation of the aerodynamic panels and the connection of these panels to the wing structure will have to be changed. The next paragraph looks into the generation of the aerodynamic model.

5.5 AEG-module

The AEG-module generates a Finite Element representation of the mass and stiffness of fuselage, fin, tailplane and engines. The module then connects this with the detailed wing structure FE-model as generated by de FEG-module. Aerodynamic panelling for the wing, engines, fin and tail are generated completing the aeroelastic model [Ref. 2].
This paragraph will investigate the working of the AEG-module to identify the aerodynamic panel generation problem. It then describes the solution method developed.

5.5.1 AEG-module

The AEG-module consists of the following fortran programs:

- **main.f** : Controls the order in which programs within the AEG module are executed.
- **aeg_stick.f** : This program reads data for tail and fuselage stick models, concentrated masses on the fuselage, wing, tail and engines, from the TDMB database and outputs this information into Nastran cards.
- **aeg_modgen.f** : This program generates macro aerodynamic panels for the given wing, fin, tail and engine geometries so that a Double-Lattice Method can be used by Nastran.
- **aeg_generic_output.f** : Reads the generated macro panel distribution, connects the panels to the structure and writes the necessary Nastran cards.

The two programs within the AEG-module responsible for the aerodynamic panel distribution over the wing are **aeg_modgen.f & aeg_generic_output.f.** To locate the problem described in the previous chapter, both these programs will be investigated next.

aeg_modgen.f

This program does the following:

- Reads user input data from the TDMB. Within the TDMB (under ControlData AeroelasticModel) the user can specify the minimum and maximum allowable aspect ratio of the aerodynamic panels and the minimum number of panels in chordwise direction for the wing and for the control surfaces.
- Reads the wing, fin, tail, controlsurface and engine geometry data from TDMB.
- Generates macro aerodynamic panels for the wing, tail, fin, control surfaces and engines and puts this information into the TDMB.

The approach followed by the program [Ref. 2] :

1. Recognition of macro panels by planform changes. Any change in the planform will be used as a boundary between macro panels.
2. Co-ordination of macro panels located in each other's wake (wing and tail). If a macro panel is located in the wake of another, their spanwise macro panelling must be identical. A macro panel boundary on the tail must be extended to a macro panel boundary on the wing and vice versa.
3. Determination of the number of spanwise and chordwise sub panels per macro panel. The aspect ration and chordwise sub panel number constraints must be satisfied. If necessary the macro panels will be sub divided to satisfy constraints.
4. Co-ordination of panels located in each other's wake. This step uses the finest spanwise panelling found in the macro panels located in each other's wake and adjusts the chordwise number of panels in the other macro panels such that all panel aspect ratios are changed as little as possible. This co-ordination is performed for wing, tail, and possible control surfaces in these components.
5. Engines, modelled as hexagonal cylinders with the same cross-sectional area as the input cylinder, are not "co-ordinated" with the other components.

The wing is eventually divided into 7 macro panels. In figure 5.7 the final panel division of the wing, tail and engines is visible. The ailerons (inboard and outboard) are shown in a darker shade and each aileron has its own macro panel.

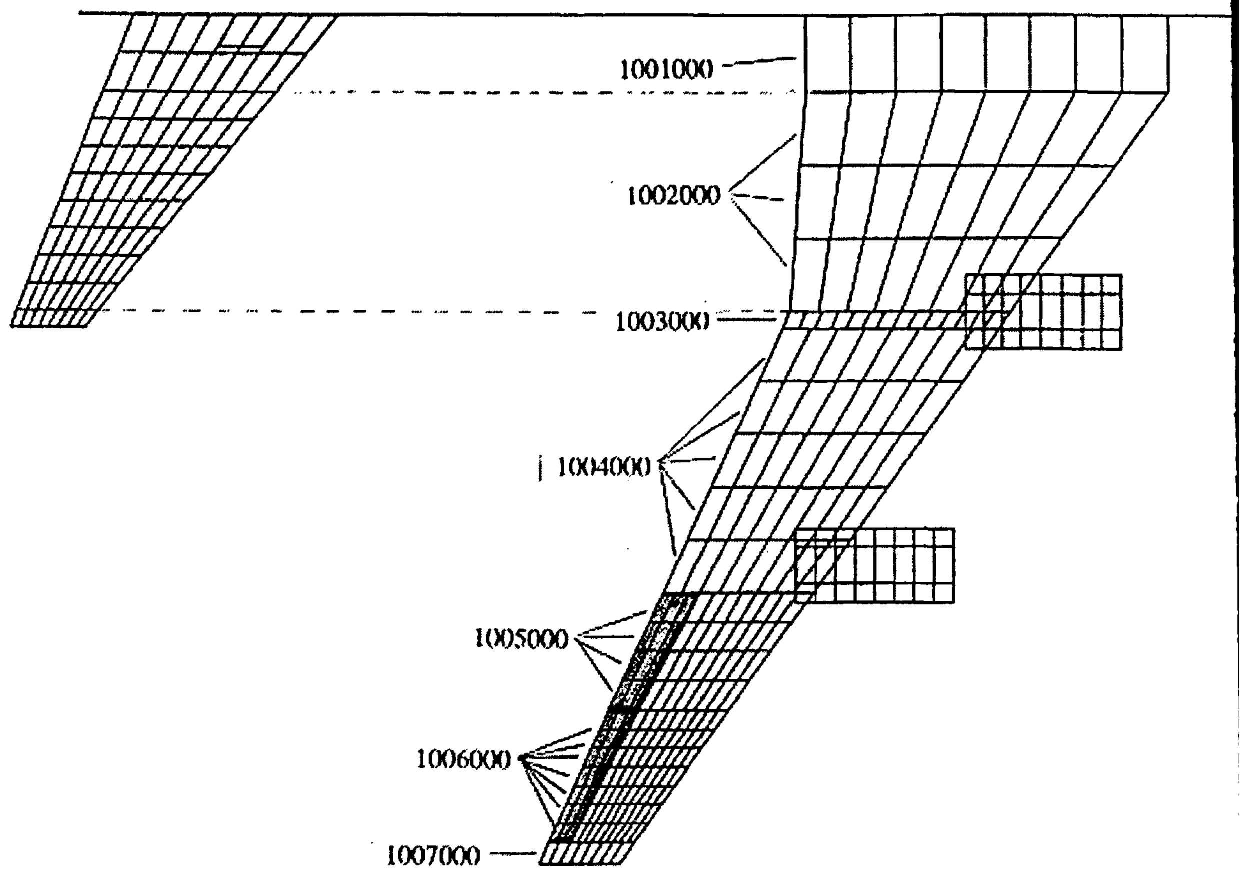


Figure 5.7 : Original panel division wing and tail plane

For the wing ,the spanwise division of the macro panels is as follows:

- Panel 1001000 : Fuselage centre line to wing root
- Panel 1002000 : Wing root to rear spar crank
- Panel 1003000 : Rear spar crank to tail tip y-position
- Panel 1004000 : Tail tip y-position to begin aileron 0
- Panel 1005000 : Begin inboard aileron (0) to end inboard aileron (0)
- Panel 1006000 : Begin outboard aileron (1) to end outboard aileron (1)
- Panel 1007000 : End outboard aileron (1) to wing tip

The aeg_modgen.f only generates the aerodynamic panelling over the surfaces. It does not create the card which connect the panels to the structure. That is done by aeg_generic_output.f . The aeg_modgen.f appears to run faultlessly.

aeg_generic_output.f

This program does the following:

- Reads the macro panel distribution and the number of chordwise and spanwise panels per macro panel as generated by aeg_modgen.f from the TDMB.
- Generates CAERO, SET, and SPLINE cards and writes this in MSC/NASTRAN format to an output file.

The approach followed is :

1. Generation of the MSC/NASTRAN cards in loops, depending on the number of surfaces defined : Wing, ailerons, engines, tail, elevators, fin, rudder.
2. Reading of the macro panel geometry from the TDMB. This includes: PanelID, in and outboard leading edge position, chord dimensions, and the number of panels in chord and spanwise direction per macro panel.
3. Definition of the CAERO1 card for this specific surface and writing it in MSC/NASTRAN format.
4. Definition of the information for the SET card and writing it in MSC/NASTRAN format.
5. Definition of the information for the SPLINE card is writing it in MSC/NASTRAN format. The loop turns back to no 1 or stops if all the panels have been defined.

Within the loop (1-5), a number of exceptions are programmed to cope with the properties of the different surfaces (the tail has no real structure for example, while the wing has a complete torsion box). So depending on the type of surface the loop will follow a different procedure to generate the necessary information. The loop will then use a general subroutine to write the information into MSC/NASTRAN cards.
Because the wing has a complete FE-model of the torsion box, the forces of the aerodynamic panels are introduced at the nodes where the ribs connect the leading edge spar and the trailing edge spar. For the other surfaces (fin and tail) the structure is represented by a stick model (generated by aeg_stick_model.f) with approximately the same mechanical properties as the real structure. There the aerodynamic forces are introduced at the nodes of this stick model.

The investigation in paragraph 5.4 found that aeg_generic_output.f does not work correctly for the wing. The aerodynamic panels are not connected to the ribs corresponding the panels position. The problem thus occurs in step 4. Within the loop a simple routine defines the number of ribs between the different wing sections at the positions the planform changes (root, crank and aileron positions). The number of ribs for each section is then used to define the SET1 cards for the wing macro panels. This is based on the assumption that the wing macro panels are only defined by planform changes. But as explained in the previous paragraph, the tail has also influence on the macro panel division of the wing. The changes the tail introduces on the macro planform division of the wing are not incorporated in the SET1 division over the wing. This is the reason why the aerodynamic panels are not connected to the corresponding ribs.

5.5.2 aeg_generic_output.f Rewrite

The loop within the aeg_generic_output.f program has been rewritten for the aerodynamic panels over the wing. The other surfaces (tail, fin, engines) are not affected by the changes. The information needed for the SET card is now dependent upon the number and the position of aerodynamic macro panels over the wing. The approach for the wing is now:

1. Macro panel geometry and positions is read from the TDMB. The programs handles one macro panel at the time. The wing consists of more than one macro panel and the next steps are run per macro panel.
2. For the specific macro panel, the CAERO1 card is written, this includes the leading edge positions of the in- and outboard side of the panel, the chord lengths and the number of spanwise and chordwise sub panels.

3. The macro panel outboard leading edge y-position is taken as reference. The macro panel width is known. The wing is presumed to start at the fuselage centreline, the next macro panel will start from the previous panel's outboard y-position.
4. The number of ribs defined for the wing is read from the TDMB. This is data from an array.
5. The rib plane y-position for the first rib of the specific panel is read from the array. The first rib of the next macro panel will be the last of the previous macro panel.
6. The rib plane y-position is compared with the outboard leading edge y-position of the macro panel. If the rib y-position is smaller, the next rib y-position is read, and the rib counter increases with one.
7. The previous point continues until the rib y-position is bigger than the outboard y-position of the macro panel. Then the previous rib is reread to establish which rib is closer to the macro panel's y-position. The rib closest to the macro panel's y-position is chosen and rib counter adjusted if needed.
8. The number of ribs and the rib numbers for this macro panel are now known.
9. SET 1 card is then written, connecting the CAERO1 card to the ribs.
10. SPLINE card is written.

In Appendix-F the rewritten sections of aeg_generic_output.f can be viewed. The whole program is not printed as this would take up a lot of pages.

5.5.3 Results from the Rewrite

In Appendix-E the corrected MSC/NASTRAN Aeroelasticity Bulk Data Deck is shown. The wing is still divided into 7 macro panels at the same locations as before. The changes were made to the SET1 cards which now refer to the ribnumbers which have approximately the same position as the CAERO1 elements. The problem card was CAERO1 panel 1003000 starting at 13.493 m and ending at 14.230 m. On the SET1 card belonging to this panel there are now only 2 ribs defined (first rib = 126 with y = 13.59 m, last rib = 127 with y = 14.17 m). The next CAERO1 panel 1004000 starts at wingspan position 14.230 m and stops at 26.215 m and is connected to the structure with SET1 104 (first rib = 127 with y = 14.17 m, last rib = 148 with y = 26.10 m).

The FE-model with the revised aerodynamic panel connection (test-8) is used to investigate a number of the same influences on wing twist as in 5.4.2. This will establish the correct working of the revised model.

- ***test-8*** : *Reference aircraft.* This model has the revised aerodynamic panel connection to the wingstructure (See 5.4.4) and is used as reference.
- ***test-8-1:*** *Outboard engine mass moved to the CG of the pylon.* As the engine mass is moved towards the flexural axis of the wing, the negative torsional moment the engine mass inflicts on the wing torsion box will decrease. This will lead to an decrease in negative wing twist.
- ***test-8-2*** : *Increased number of aero panels over the wing.* This model was made to investigate the influence of the panel density over the wing on the twist distribution.
- ***test-8-3*** : *No outboard engine.* The complete removal of the outboard engine will show the influence the engine as a whole (mass and lift) on the wing twist. The removal of the mass will reduce wing bending, which leads to a decrease in negative twist. The negative torsional moment the engine mass inflicts on the wing torsion box will be gone. This will lead to an decrease in negative wing twist. The removal of the lifting engine nacelle will result in an increase in negative twist. This test will show which effect will be the greater.

The results from the test cases are shown in figures 5.8 and 5.9.

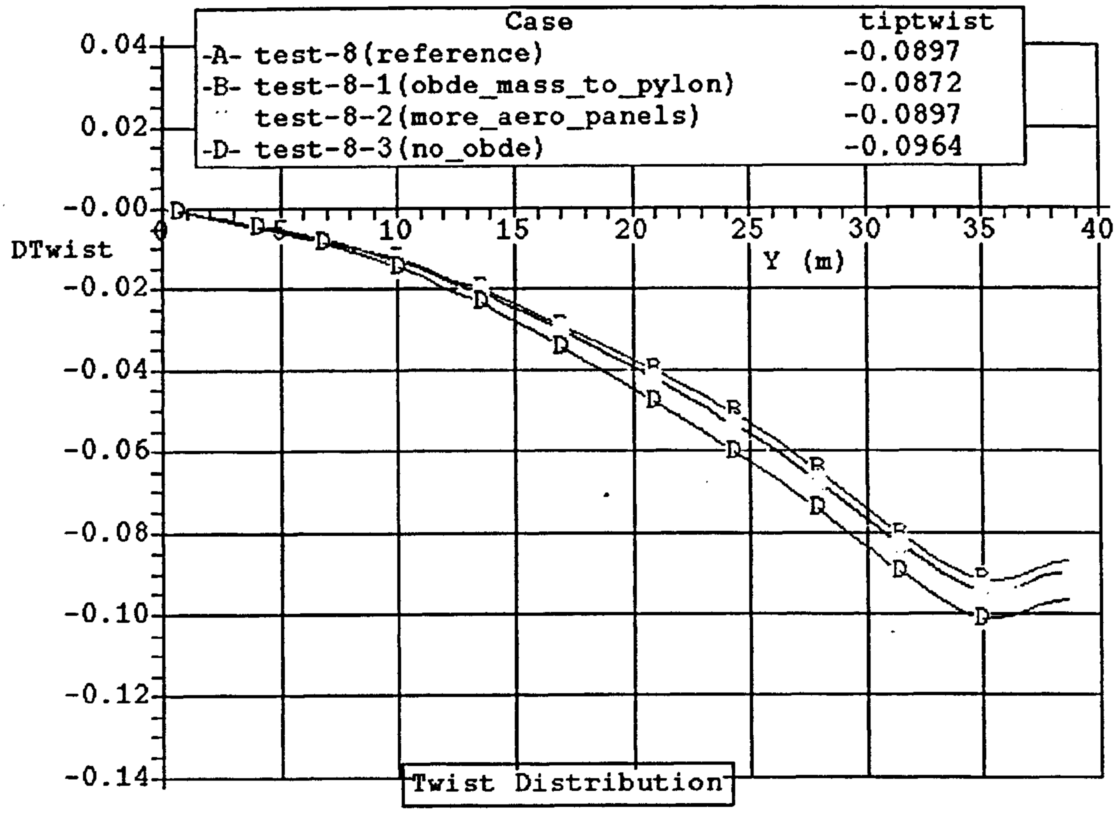


Figure 5.8 : Results Revised Aerodynamic Panelling on Twist Distributions

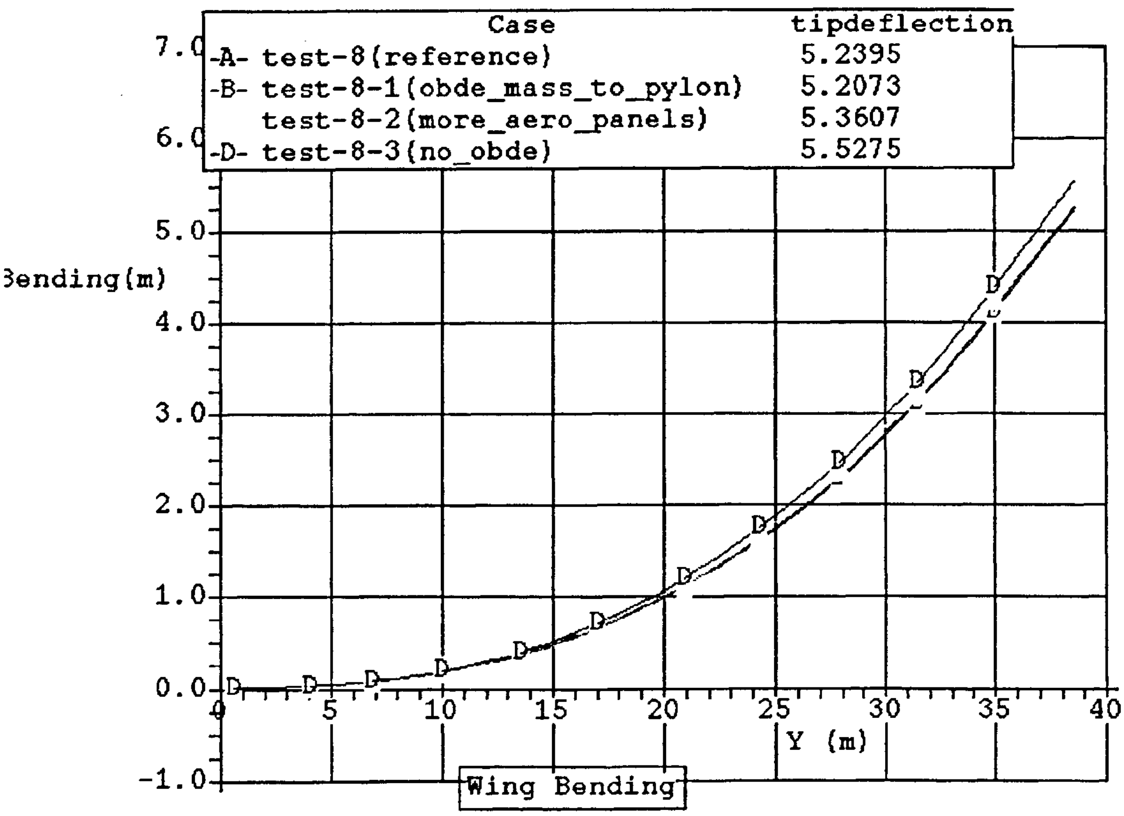


Figure 5.9 : Results with Revised Aerodynamic Panelling on Bending Distributions

From figure 5.8 it is clear that the model with the revised aerodynamic panel connection to wing structure responds as expected to the changes made in the different test models. Figure 5.9 shows the wing deflection. There is no "dent" in the slope of the wing deflection anymore and the wing tip deflects to 5.23 meter in the pull_up manoeuvre (test-8). The original case (test-1) only deflected to 4.75, see figure 5.4.
With the amendments and supplements made to NASTPP (see chapter 6) the lift forces on the aerodynamic panels can be viewed with TDMB. Figure 5.10 shows the total lift forces on the different lifting surfaces for test-8.

```
Operation          *                   *                   Analysis for each operation.
ST_AERO_PULL_UP    *                   *                   Analysis for each operation.
    WingDeflection     WingDeflection     *                    Deflected wing shape information.
    WingLiftForces     WingLiftForces     *                    Lift forces on Aerodynamic Panels.
        *                  *                  *                    *
        MacroPanels        Array              *                    Wing Macro Panels
        WingStrips         Array              *                    Aero Forces totalled over wing strip
        FuselagePanels     Array              *                    Fuselage Aerodynamic Panels
        Totals             *                  *                    Total Lift forces
            WingLift           Real          6.47446e+06           Total Lift of the Wing
            TailLift           Real          -17408.7              Total Lift of horizontal Tail
            IbdELift           Real          113588                Total inboard Engine Lift
            ObdELift           Real          146962                Total outboard Engine Lift
            AilerLift          Real          63637.6               Total aileron Lift
            FuseLift           Real          0                     Total fuselage Lift
            TotalLift          Real          6.78124e+06           Total Lift
```

***Figure 5.10 :** TDMB Results for Test-8*

From these results it can be concluded that the total lift generated by the aerodynamic panels is roughly equal to 2.5 times the (half) aircraft weight: 2.5 x 0.5 x 535000 x 9.81 = 6560437.5 N. The difference with the total lift in the figure (6781240 N) is approximately 9 tonnes and is due to the fact that the actual weight of the FE-model is slightly higher than defined in TDMB. The lift forces on the tail are relatively small. This is due to the absence of an overall negative pitching moment normally caused by the wing.

5.5.4 Remarks

- The wing rib plane y-position used to determine the position of the spanwise position of the wing rib, was chosen as reference for proof of concept reasons. The macro panels are defined is such a way that they are always lined up with the airflow, i.e. the macro panels run parallel to the x-axis of the airplane. The majority of the rib planes are for structural reasons not parallel to the x-axis of the airplane. This means that the tip of the last rib belonging to the macro panel will fall outside the macro panel's dimension and that the end of the rib will fall well inside the panel. This means that some of the aerodynamic forces and the edges of the macro panels are not introduced at the correct place. An estimate of the maximum error made is two rib pitch lengths (approximately 1.0 m) which is 2.5 percent on the wingspan of almost 40 m. The effect of this on the overall distribution of stresses etc. in the wing is estimated to be minimal.
- The aileron connection to the wingstructure has at this moment no separate routine to define the SET cards. The ailerons are connected via exciting wing SET cards. Future development is needed here, especially when there are major changes in the wing planform.
- The effect of the changes made on other calculations like flutter etc. have to be investigated, but that fall outside the scope of this thesis project.

5.6 Conclusions

The FE-model (as generated by the MMG) representing a half aircraft without fuselage aerodynamic model has to be amended with the CC & SPC deck to become suitable for initial static aeroelastic analysis with MSC/NASTRAN

Further the original AEG-module does not work correctly. The AEG-module must be changed and amended in the way described in paragraph 5.5 to correct the way in which the aerodynamic panels are connect to the wingstructure. This revised model shows correct responses to changes made in the FE-Model (see 5.5.3). The FE-model is then ready for the initial static aeroelastic analysis. The specific results from the MSC/NASTRAN static aeroelastic analysis (pressure distributions and lift forces on the aerodynamic panels) can not yet be read into and from the TDMB database. This is dealt with in the next chapter.

6 SUPPLEMENTS FOR NASTPP

6.0 Introduction

The MDO-software package does not (yet) incorporate facilities supporting static aeroelastic analysis. The results of the static aeroelastic analysis from MSC/NASTRAN (like pressure and force distributions on the aerodynamic panels) can therefore not be read into the common database for the TDMB. Before any further analyses are performed, it is necessary to make the results available in a suitable format. The utility within the MDO-software which reads data from the MSC/NASTRAN output files and writes data into the TDMB is called NASTPP. To make the results from the static aeroelastic calculations available in the TDMB, NASTPP has to be amended and supplemented with extra routines.

6.1 MSC/NASTRAN Output

MSC/NASTRAN can output the results from various analyses to an output file. The output file consists of a list with element and node numbers and their specific results (like stress, displacement etc.). For large models the output file becomes very large and finding a specific result in the file is a labour intensive job. Graphic display of the results is provided by the MSC/NASTRAN plot facility. It can make plots of (for example) all the elements in the FE-model and their displacements as result of the applied loads. But looking into the options of the plot facility, it was found that the support for plotting results of the Aeroelastic Analysis is limited [Ref. 6, page 138]. Only the geometry of the aerodynamic sub panels (including the meshing) can be plotted. Aerodynamic forces and pressures acting on the aerodynamic sub panels can not be plotted with the plot facility . This limits the use of the plot facility and a separate program is needed to make the results (for example lift distributions over wingspan) available in graphical form.

6.2 NASTPP, Approach Followed

Within the MDO-project, the NASTPP utility has been developed to post-process the main MSC/NASTRAN output file (extension.f06) and extract the convergence history and final design variable values back into the TDMB. It will read the output file for a number of different solution procedures and extract the key design information [Ref. 2] .

The utility creates a reduced version of the main NASTRAN output files (extension.sum), containing just the key data that it has read and interpreted. If the main output files have been deleted then the utility can still be run on these reduced files, for example if it is required to merge the results into a different TDMB database. The utility consists of only 1 fortran program : nastpp.f

When the NASTPP utility is invoked it prompts as follows :

```
  1     Static Analysis results [stt.f06]
  2     Symmetric Modes Analysis [sym.f06]
  3     AntiSymmetric Modes Analysis [asm.f06]
  4     Symmetric Flutter Analysis [syf.f06]
  5     AntiSymmetric Flutter Analysis [asf.f06]
 11     Optimisation Results [opt.f06]
-53     Delete current Results in TDMB Database
 -1     QUIT without saving TDMB Database
 99     Save TDMB Database and EXIT
```

The user then has to enter the integer value corresponding to the action required. The user will then be prompted to:

- Either accept the default filename or specify the name of the NASTRAN output file.
- Enter the name of the design stage that the results relate to - in general use InitialDesign of FinalDesign according to whether the results are for the initial-unoptimised or final-optimised model. NASTPP will look for the existence of this stage under Results StructResults Stages and create it if required.
- Specify the number of modes for which detailed results are to be extracted. This is just prompted for once, the first time that an option 2-5 is requested.

NASTPP will then :

1. Open the specified file and read the first line of this file.
2. Process the information in this line. If the information is of no specific value, NASTPP will go to the next line. Normally the Sorted Bulk Data Deck appears first in the output file. This part has no results in it and is thus skipped by NASTPP.
3. Process information in the next line. If the information is of specific value, NASTPP will put the processed information into the TDMB and go to the next line in the output file. After reading the Sorted Bulk Data Deck, the first results appear in the outputfile. If there are several subcases, NASTPP first looks to which subcase the result belongs so that it is written to the right subcase in the TDMB. If the subcase does not exist in TDMB, it will be created.
4. Process information in the next line. The reading and processing of information continues until the end of the file.

After NASTPP has finished reading the complete outputfile, the menu is shown again and the user can then opt to save the TDMB database and exit by entering 99.

6.3 MSC/NASTRAN Static Aeroelastic Results

The static aeroelastic solution (SOL 144) in MSC/NASTRAN produces the following results of interest [Ref.6]:

- ***Aerodynamic Forces and Pressures :*** The aerodynamic forces and pressure on each sub panel at the trimmed flight condition are listed in the output file. The forces and moments on the subpanel are calculated at the aerodynamic gridpoint of the sub element.
- ***Trim Variables*** : The results of the trim analysis for all the aerodynamic extra points are printed for each subcase.

- ***Stability Derivatives :*** Nondimanesional stability derivatives are listed for any static aeroelastic analysis that includes support degrees of freedom for as many degrees of freedom as are supported on the SUPPORT card.
- ***Stresses and deflections of the structure*** : The structure to which the aerodynamic panels are attached will be stressed and deflected by the generated aerodynamic forces. These results are written to the outputfile if the option STESS = ALL and DISPL=ALL are set in the case control deck.

The aerodynamic forces and pressures are listed per subpanel. To be able to make a graph presenting the required data it is necessary to know exactly where each subpanel is situated. This information is not directly available from the TDMB. Only the macro panel position and dimensions known in the database. But together with the known number of subpanels in spanwise and chordwise direction, the position of each subpanel can be calculated.

6.4 NASTPP for Static Aeroelastic Analysis : Approach Followed

The nastpp.f fortran program has been amended and supplemented so that the results from the static aeroelastic analysis are read from the NASTRAN.f06 output file and fed into the TDMB after processing. The adjustments/amendments in the NASTPP utility will be called "aeroelastic routine" from this point and consists of the following:

1. An option 6 has been created in the menu which prompts as the nastpp utility is invoked. This option reads the static aeroelastic analysis results from a user specified NASTRAN .f06 output file and gives the possibility to specify the design stage. If the design stage does not already exists, it will be created. Note: If the design stage does exist, the results are added to this design stage, without deleting the old results already present in this stage.
2. From the Case Control Echo and Sorted Bulk Data Deck of the NASTRAN output file.
    - The aeroelastic routine will extract the subcase information from the Case Control Echo. It will then create the subcases in the TDMB (Results StructResults Stages Operations). This is necessary to be able to make aerodynamic panel arrays for every subcase.
    - CAERO1 cards for the wing and the ailerons are read. The information extracted from the cards is : macro panelID, number of spanwise and chordwise subpanels, macro panel in- and outboard leading edge spanwise and chord position, macro panel in- and outboard chord lengths. This information is then written to an array in the TDMB (Results StructResults Stages Operations WingLiftForces MacroPanels) for each subcase (See figure 6.1). From this information the position for each subpanel (relative to the nose of the aircraft) in the macro panel is calculated and written into an array in the TDMB (..WingLiftForces MacroPanels SubPanels). Although this (sub) panel information is the same for every subcase, it is necessary to make these arrays for every subcase as the lift forces, which are put in the TDMB in a later stage, are not the same in every sub case.
    - CAERO2 cards for the fuselage are read (see chapter 7). The information extracted is : body panelID and number of body panels. From the TDMB the fuselage length, the nose and tail cone taper length is extracted. The routine then calculates the position (from the fuselage nose), the width, the surface area of each fuselage sub panel and its ID and writes this information into an array in the TDMB (..WingLiftForces FuselagePanels). Note: This routine does not work if the body is divided in sub panels with unequal lengths.

3. From the result part of the NASTRAN output file.
    - The routine will first determine to which subcase the results belong. This determines to which subcase in the TDMB the results will be written into.
    - The routine looks for the aerodynamic panels in the results and recognises the aerodynamic sub panel ID number. From the sub panel ID number the routine determines whether the sub panel it is looking at is part of the wing, ailerons, tail, engines or body (fuselage). For the different panels, the routine then acts as follows:
        - ***Wing :*** The routine reads the aerodynamic force in z-direction and writes this force to the corresponding sub panel in the array in the TDMB (..WingLiftForces MacroPanels SubPanels AeroForce). The routine then memorises the aerodynamic force and adds it to the previous force with the same spanwise location. It does so until the spanwise location of the sub panel changes over the previous one. The routine then calculates the spanwise position (eta) of the wing strip, the wing strip width, the total force on that wing strip, the stripload (N/m) of the wingstrip, the Cl of the wing strip, and the Clc/mac of the wing strip. It writes that information to an array in the TDMB (..WingLiftForces WingStrip Stripload). This enables the plot routine in the TDMB to make plots of the spanwise force and Cl distributions on the wing.
        - ***Ailerons :*** Because the ailerons are part of the wing, the routine works the same as for the wing except that the routine for the aileron reads the spanwise position of the wing strip from the TDMB (this information already exists at that stage, as the wing results are first to appear in the result file) and adds the force in z-direction of the aileron sub panel to the TotalForce and StripLoad. The routine then recalculates the Cl and Clc/mac for the wing strip where this particular aileron sub panel is located.
        - ***Tail :*** At this stage of development, the forces on the sub panels of the tail are returned to the screen as total tail force which is also written into TDMB.
        - ***Engines :*** The routine reads the aerodynamic forces on the inboard and outboard engine nacelle sub panels. The total lift of the engines is returned to the screen and written into the TDMB (under ..WingLiftForces Totals). If the option EngineLift equals 'Yes' in the ControlData StaticFEmodel section in TDMB then the routine will find the wingstrip with approximately the same spanwise position as the specific engine and add the aerodynamic force directly to the TotalForce and StripLoad of that Strip. The total engine lift force is then introduced into the wing at one specific wingstrip. This method is not satisfactory as it gives peak loads on the strips. Further development is needed here.
        - ***Body*** (fuselage) : The routine recognises the forces on the body in z- and y-direction. The force on the body sub panel in z direction are written to an array in the TDMB (WingLiftForces FuselagePanels PanelLift). The routine also calculates the Cl and loads on the specific body panel (in x and y-direction) and writes this to TDMB.
    - The total aerodynamic forces on the Wing, Ailerons,Tail, Engines and Body are returned to the screen once all the aerodynamic results from the subcase are read and calculated. These results are then written to the TDMB (..WingLiftForces Totals).

In figure 6.1 the different arrays put into TDMB by the new NASTPP routine are shown.

```
Operation          *                  *                    Analysis for each operation.
ST_AERO_PULL_UP    *                  *                    Analysis for each operation.
    WingDeflection     WingDeflection     *                    Deflected wing shape information.
    WingLiftForces     WingLiftForces     *                    Lift forces on Aerodynamic Panels.
        *                  *                  *                    *
        MacroPanels        Array              *                    Wing Macro Panels
            PanelID            *                  *                    Wing Macro Panels id number
            1002000            *                  *                    Wing Macro Panels id number
                IbdLeY             Real              3.51                 Inboard panel spanwise p
                IbdLeX             Real              20.93                Inboard panel chordwise
                IbdChord           Real              16.5                 Inboard panel chord leng
                ObdLeY             Real              13.493               Outboard panel spanwise
                ObdLeX             Real              28.053               Outboard panel chordwise
                ObdChord           Real              10.075               Outboard panel chord Len
                NSpan              Integer           6                    Number of spanwise panel
                NChord             Integer           16                   Number of chordwise pane
                SubPanels          Array             *                    Sub Panel array
            1003000            *                  *                    Wing Macro Panels id number
            1004000            *                  *                    Wing Macro Panels id number
            1005000            *                  *                    Wing Macro Panels id number
            1006000            *                  *                    Wing Macro Panels id number
            1007000            *                  *                    Wing Macro Panels id number
            2001000            *                  *                    Wing Macro Panels id number
            3001000            *                  *                    Wing Macro Panels id number
        WingStrips         Array              *                    Aero Forces totalled over wing s
        FuselagePanels     Array              *                    Fuselage Aerodynamic Panels
        Totals             *                  *                    Total Lift forces
ST_AERO_PUSH_DOWN  *                  *                    Analysis for each operation.
```

Figure 6.1 : Arrays Put into TDMB by New NASTPP.

The amendments and supplements mentioned above only work if a number of amendments are made to the following programs in the MDO-software:

- /MDO-Programs/ObjectLibrary/Index : Add the following line under "Simulation" after ":WingDeflection"
    - :WingLiftForces "Object" "Liftforces on the Aerodynamic Panels generated by Nastran SOL 144" {*}
- /MDO-Programs/ObjectLibrary : Copy the following file into this directory. This file is the Object referred to in the Index file:
    - " WingLiftForces.tom" (see appendix-G)
- /MDO-Programs/ObjectLibrary/StructResults.tom : Add the following line after ":WingDeflection"
    - :WingLiftForces "WingLiftForces" "Lift forces on Aerodynamic Panels"
- /MDO-Programs/ObjectLibrary/StaticFEModelControlData.tom : Add the following line at the end of this file:
    - :EngineLift "String" "Yes will include Engine Lift in the Static Aeroelastic Results" "No"

In Appendix-H the NASTPP routines can be viewed. The complete program is not printed as this would take up a lot of pages

6.5 Aeroelastic Plot Results from TDMB

The different arrays created with NASTPP contains the information needed to make plots of the wing force / wing load distribution over the wing span. Also the fuselage lift distribution over fuselage length can be viewed. The following figures give an impression of what is possible. These figures hold no relevant data and are shown as an example only.

Figure 6.2

This figure gives an example of the aerodynamic lift force distribution over the wing . It can be seen that for two different cases the values of wingforces differ substantially. This is due to the panel density of the wing. In the "More Panels" case there are more aerodynamic panels on the wing, hence the force per wingstrip is lower. In all the last case the engine nacelle lift force is visible as a peak load at Eta = 0.35 and Eta = 0.68.

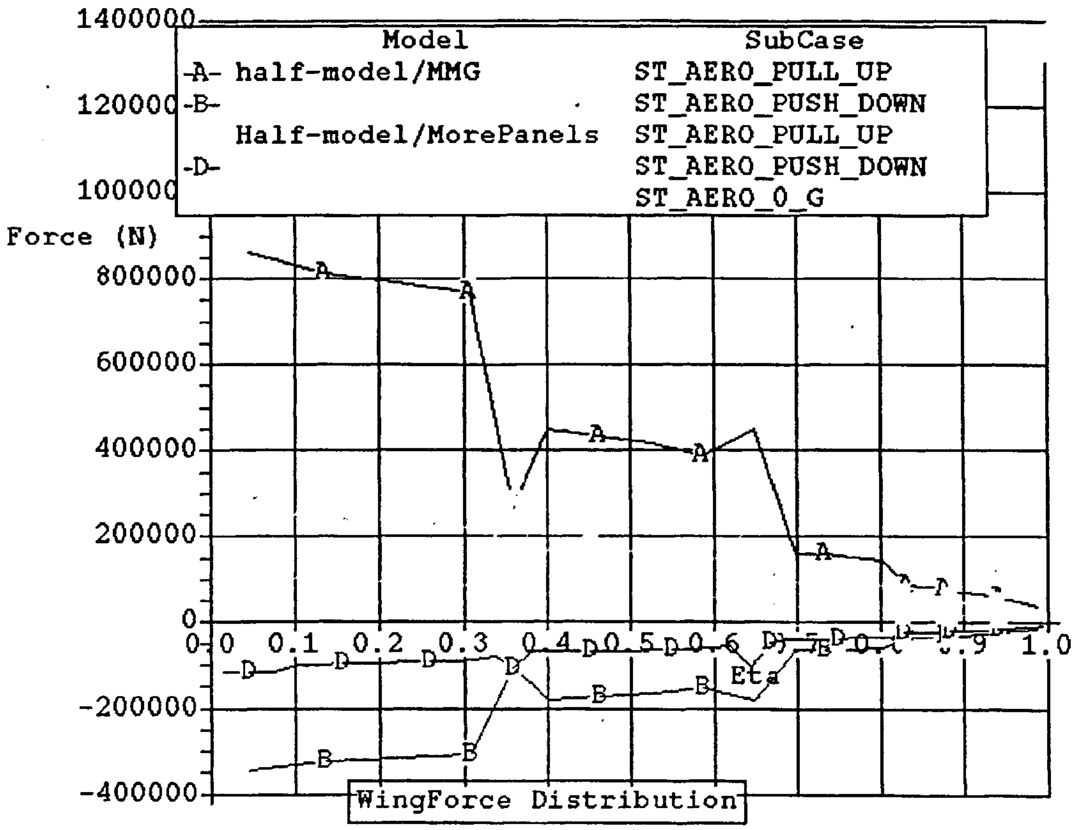


Figure 6.2 : Force Distribution over WingSpan

Figure 6.3

This figure gives an example of the load distribution (N/m) over the wing span. Here the influence of the engine nacelle lift is clearly visible. The engines cause a peak load at eta = 0.35 and at eta = 0.65. This figure can be used to compare models with different panel density as the stripload is set out in N/m.

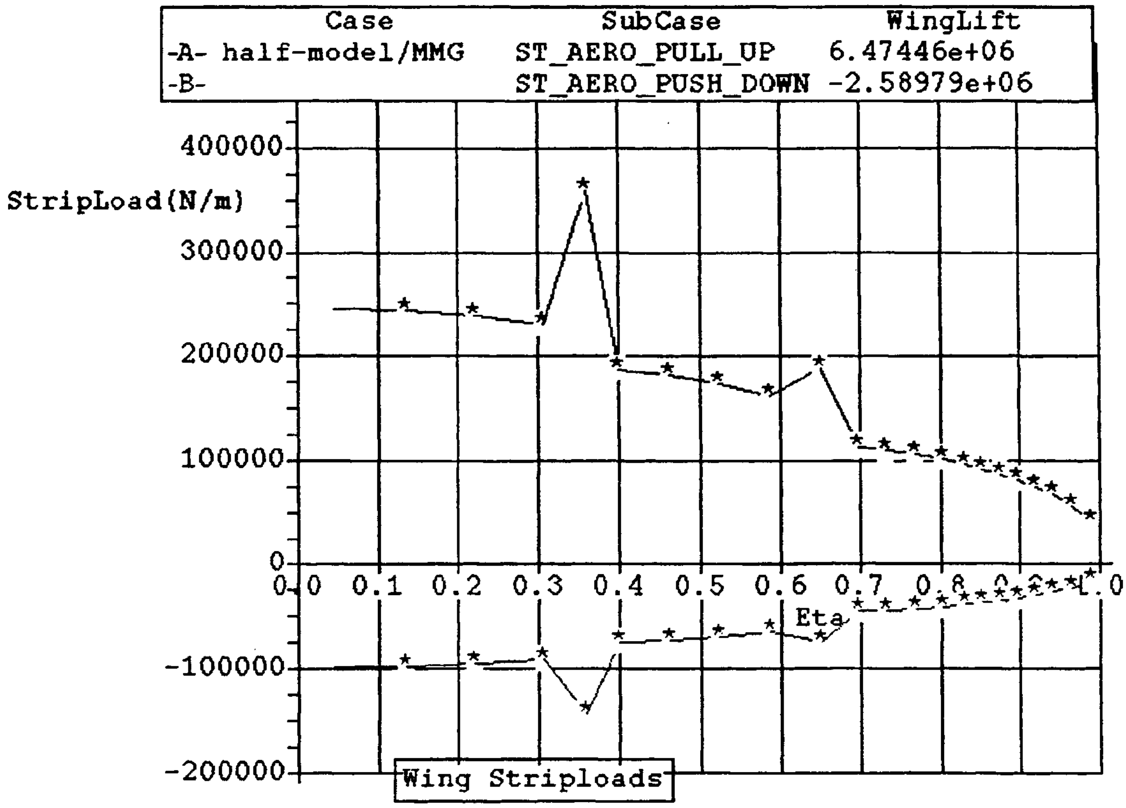


Figure 6.3 : Stripload Distribution over WingSpan

Figure 6.4

This figure gives an example of the fuselage load distribution in N/m. This figure can also contain a fuselage Cl distribution or a force (N) distribution. Clearly visible is the influence of the wing on the fuselage lift distribution.

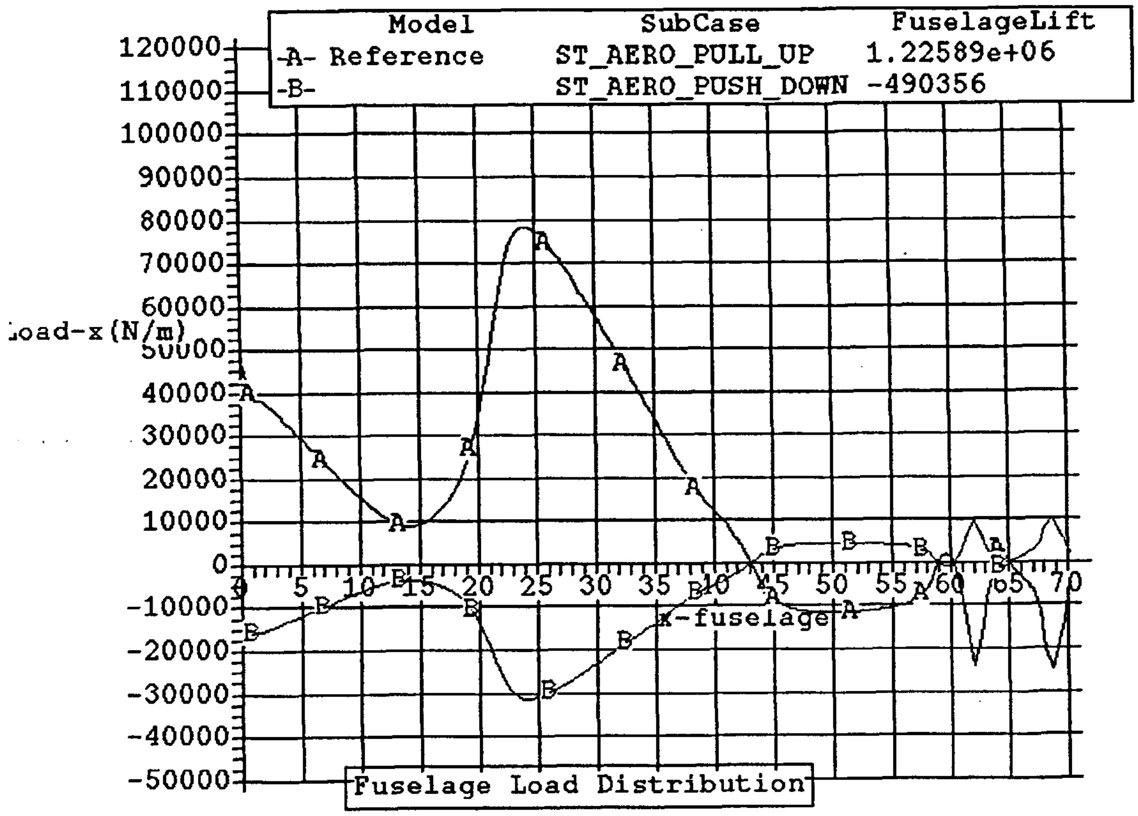


Figure 6.4 : Force Distribution over Fuselage Length

6.6 Known bugs/Remarks/Future Developments

There are a number of conditions which have to be fulfilled for this version of NASTPP to work properly with static aeroelastic results.

- This version of NASTPP only works correctly with MSC/NASTRAN V70.0 and V70.5.0. Version 70.5.1 echoes the static aeroelastic results in a different way, such that this version of NASTPP cannot read them. A NASTPP version supporting the latest MSC/NASTRAN version has to be developed in a later stage. This probably only involves changing the positions on the line where NASTPP reads the data and should thus be relatively easy. It is unknown if this version of NASTPP works correctly with earlier versions (V69, V68) of MSC/NASTRAN.

- There MUST be a subcase specified in the Case Control Deck of the model. Without subcase division NASTPP does not work correctly.
- A maximum number of only 3 subcases can be specified. More subcases will result in a memory error. The reason for this is under investigation.
- If new results are read into the TDMB database under the same Design Stage name, the old results are not deleted automatically. This MUST be done by the user (use -53 and then 99, or via TDMB) to prevent a double set of results for the same Design Stage.
- There MUST be a Case Control Deck echo and a Sorted Bulk Data Deck echo in the NASTRAN .f06 results file so that the subcase and CAERO card information can be read.
- If option 6 is invoked in the menu, wing deflection and element stress information is also read, processed and fed into the TDMB (if present in the output file). Also if another option is invoiced and there are static aeroelastic results available in the output file, these will be processed and fed into the TDMB.
- The MMG does not generate the body panelling for the fuselage (see chapter 7). The user has to model the body by hand to be able to extract the body aerodynamic forces. The body aerodynamic element reading only works if the body is divided into equal lengths by the CAERO2-card. If a user specified body division is used, the routine will not recognise the division and return an error.
- If other data is necessary (like sub panel aerodynamic pressures) NASTPP can easily be amended to read this data from the NASTRAN result file and write it into TDMB.

7. FUSELAGE AERODYNAMIC MODEL

7.0 Introduction

The aerodynamic panels for wing, tailplane and fin are automatically generated by the MMG. The MMG does not generate aerodynamic panels representing the fuselage. The lift of the fuselage is thus not accounted for in the static aeroelastic analysis. This means that the wing will have to produce more lift to balance the missing fuselage lift. This problem is partly "solved" by the aerodynamic panels of the wing centre section which are providing part of the lift normally produced by the fuselage. In reality the wing centre section is embedded in the fuselage and produces no lift. It is therefore essential to model the fuselage in the FE-model so that the lift generated by the fuselage can be calculated. The fuselage will first be modelled by hand within the FE-data deck. In a later stage the MMG will have to be amended so that the aerodynamic panels of the fuselage will be modelled automatically.

This chapter first looks at the way the fuselage is modelled after which some test models are discussed. The results of the test models are presented and conclusions drawn.

7.1 Fuselage FE-Model

Under normal conditions the fuselage of a conventional aircraft provides 10-12% of the total lift required and in the +2.5 g Pull_up manoeuvre this can increase to 15-18%. This makes it essential that fuselage body lift is accounted for within the FE-model. Within MSC/NASTRAN a body in an airflow can be modelled using the Slender Body Lift theory . Interference between lifting bodies and lifting surfaces is taken care of with the method of images (see paragraph 4.1.2).

7.1.1 : Fuselage Aerodynamic Model

Within MSC/NASTRAN the fuselage aerodynamic model has to consists of slender body elements and interference elements.

Slender Body

The slender body elements represent the fuselage body and account for the forces arising from the motion of the body through the air. The CAERO2-card specifies the geometry and divisions of the slender body (in this case the fuselage). The PAERO2 entry provides orientation and cross-section data for the slender body and interference elements.

Interference Elements

The interference elements are used to account for the interference of the slender body elements with the lifting surfaces in the same aerodynamic group (i.e. the interference between the fuselage, wing and tail). This is done by providing a surface through which the boundary condition of no flow is imposed. The interference elements provide the basis for the internal image system that cancels most of the effects of the trailing vortices from the lifting surfaces. Because of the low-dimensional basis for this approximation, the body surface has been approximated by a constant elliptical cross-section cylinder called the interference tube. This tube is divided into interference elements. [Ref.2]. All panels that intersect the body must be attached to the interference tube. Th image locations are computed from the semi-width of the interference tube for all lifting surface associated with the body. The image is only computed if it lies between the front of the firs interference element and aft of the last interference element for the associated body. Near region with substantial interference (wing-fuselage connection) shorter interference elements should b used. Longer elements can be used in regions with less interference.

7.1.2 Fuselage Bulk Data entries

The bulk data entries needed for the fuselage aerodynamic model are shown in Appendix-I Hereafter a discussion about each of the entries will follow.

- ***CAERO2*** : Defines an aerodynamic slender body and interference elements for DLM witl slender body theory. There is only one CAERO2 card necessary to model the complet fuselage. The CAERO2 card ID must be greater than any other element in the FE-model. A PEARO2 card ID is referred to. As mentioned in the previous chapter, NASTPP currentl only recognises a fuselage with evenly divided body elements. On the card the number o divisions for the body panels and for the number of interference elements is given (th interference element division does not have to be even, this can be done separately on a AEFACT card if required). The position of the nose of the fuselage in the global axe system is given (0,0,0) and the length of the fuselage (70.4 m).
- ***PAERO1*** : Defines bodies associated with CAERO1 entries. This card is already availabl in the Bulk data deck as this entry is necessary for the wing panel CAERO1 entries even i there are no bodies. On this card one has to refer to the CEARO2 entry of the interferin body.
- ***PAERO2***: Defines the cross-sectional properties of the aerodynamic bodies. The secon field defines the card ID. After that the type of motion for the body is defined. A ZY bod can move in Z and Y direction. For the interference tube, the half width of the tube is give on the fourth field (for the reference aircraft : 3.51m). The next field is a multiplier for th half width to determine the half height (4.0 m) of the interference tube. The 6th and 7t fields refer to AEFACT cards containing the body half widths of the fuselage an interference elements. If the number of elements is changed, the AEFACT cards must b changed so that the number of elements match. For the interference elements it recommended [Ref.6 page 94] to have a constant width interference tube as errors made a small (so the field is blank). Field eight contains a number for an AEFACT card containin the theta division of the interference tube. The theta point are points around the periphe of the interference tube where the interfering flow from the lifting surfaces and oth slender bodies is averaged.
- ***AEFACT*** : Specifies a list of real numbers for the aerodynamic model required by th CAEROi and PAEROi cards. This could be half widths, theta divisions, fuselage divisions
- ***SPLINE2***: Defines a beam spline for interpolating the fuselage aerodynamic panels to th structure. The CAERO2 card of the fuselage is defined as is the SET1 card. Further th beam stiffness parameters are set.
- ***SET1*** : Selects structural grid points to be used in the splining of aerodynamics. The nod on the SET1 cards are connected to the SPLINE2. The nodes of the fuselage stickmodel a used here.

7.1.3 Modelling Guide Lines

The introduction of the fuselage into the MDO-FE-model involved making a number of changes to the aerodynamic FE-model of the wing and tail.
For the fuselage-wing and fuselage-tail a number of modelling rules have to be followed [Ref. 2]:

- *Interfering wing and tail panels need to intersect with the interference tube* : The aerodynamic panels of the wing and tail root have to intersect exactly with the interference tube. As the interference tube is an ellipse, the exact spanwise location of the intersection has to be determined.
- *No aerodynamic panels are allowed inside the interference tube.* As both wing and tail have panels starting at the fuselage centre line, the panelling has to be adjusted:
    - The wing root aerodynamic panel of the wing centre sections are removed. The dimensions of the aerodynamic panels of the wing now match the "real" aircraft. To achieve physical connection between the wing and the fuselage (needed to satisfy the first modelling rule), the fuselage is lowered to the wing position such that the wing root intersects at mid position with the fuselage. This option was chosen as the "real" aircraft wing is connected to the fuselage at approximately the same spanwise position as the fuselage half width. This option affects the interference from the wing with the fuselage but it was considered to be the best option possible.
    - For the tail plane one has to consider that removing the panels of the tailplane (so that there are no panels inside the interference tube) means that a lot of tail surface area is "disappearing". The option is to move the tailplane physically upward so that the root of the tail intersects with the interference tube at a spanwise position representative for the "real" tail plane area, while maintaining the same span. The interference effects of the tailplane with the fuselage are affected by this upward move, but it is the interference effects of the wing with the fuselage which are of main importance.
- *Theta division of the fuselage needs to be equal.* : The theta point are points around the periphery of the interference tube where the interfering flow from the lifting surfaces and other slender bodies is averaged. It is advisable to have a theta division which is symmetrical in the x-z plane.
- *Intersection of wingpanels equal distance to theta points :* The distance between the place where the wing and tail panels connect to the fuselage and the theta points below and above the intersections should be equal.

In figure 7.1 the FE-Model including the fuselage aerodynamic lifting body panels is shown (top view). The fuselage body half widths defined on the AEFACT card give the fuselage its shape. Visible is that the fuselage has a constant radius at the tail plane connection. As MSC./NASTRAN can not plot the fuselage body elements, these are drawn by hand.

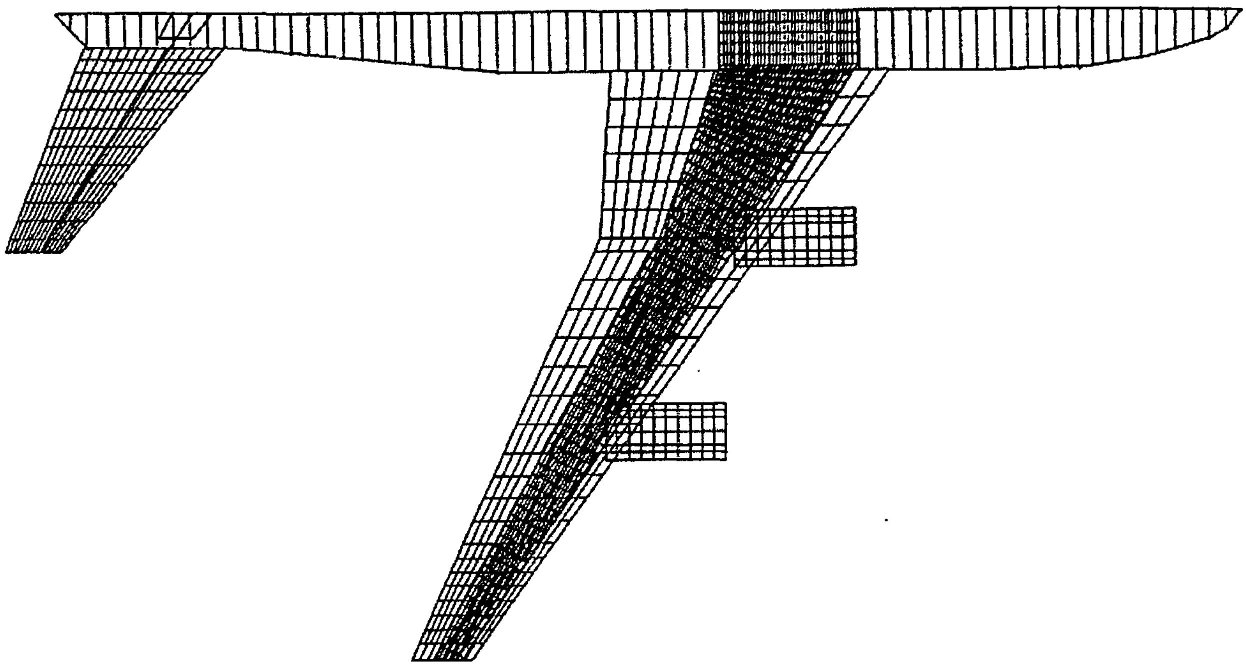


__Figure 7.1__ : The FE-Model Including Fuselage Elements

Figure 7.2 gives an idea how the wing and tail panels are attached to the fuselage and how the interference tube is defined. The ellipse represents the fuselage mid-section (height is 8.0 m, width is 7.02 m). It also represents the dimensions of the interference tube. The dimensions of the interference tube do not change along the length of the fuselage. Shown are also a number of theta points. Further it is visible that the tailplane connects to the interference tube at a 2.0 m spanwise position. Although the FE-model is only half the aircraft, the fuselage is completely modelled and constraints make sure only half the lift generated by the complete fuselage is counted.

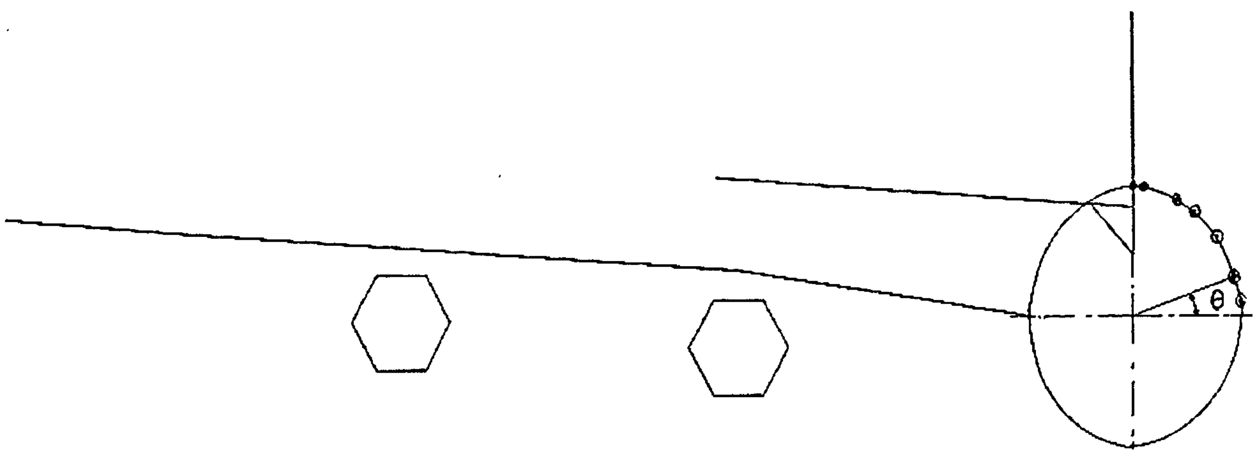


__Figure 7.2__ : Front View Fuselage and Panel Attachment.

7.2 Fuselage Test Models

A number of different FE-models have been evaluated to determine the necessary aerodynamic panel density, the necessary theta density and division, and interference element density of the fuselage model. During this evaluation it occurred that MSC/NASTRAN is sensitive to changes in the aerodynamic model. The determination of the necessary panels densities is a trial and error process as small changes in the model (more or less panels) frequently come back with inconclusive error messages in the MSC/NASTRAN output file:

received signal 10 or 33, aborting
System FATAL Error 4276 Recover error code 7777.

This error message indicates a number of possible errors for example machine overflow or underflow, matrix singularities, etc., but this error is also shown when none of the other errors are applicable.

First the number of necessary interference elements is determined. Hereafter the influence of the number of body elements is investigated. Last the influence of the number of wingpanels on the wing lift distributions is determined. All the test models are configured as follows :

- Aircraft Mass : 535.000 kg
- Payload : 89.600 kg
- Fuel load : 192.000 kg
- Mach no : 0.8
- Altitude : 33.000 ft
- +2.5 g pull_up manoeuvre
- Evenly divided fuselage body panelling and interference element division
- Wing panel density according to standard MMG settings see figure 5.4
- Tail panel density : Increased to 15 chordwise and 12 spanwise.
- 22 theta points around the periphery of the interference tube.

7.2.1 Convergence Study to Determine the Number of Interference Elements

A number of models have been evaluated to determine the necessary number of interference elements on the fuselage. An evenly divided fuselage body panelling and interference panelling is used. This is necessary for NASTPP to be able to read the results (see chapter 6) but it also means that the model can be easily adjusted/altered for quick results. In figure 7.3 the fuselage panel load (N/m) is set out against fuselage length. The table in the figure shows the model configuration (number of body panels-number of interference panels) and the total fuselage lift for the flight conditions as mentioned in 7.2.1. Five models have been examined to determine the necessary number of interference panels. The number of body elements in this study was chosen arbitrary.

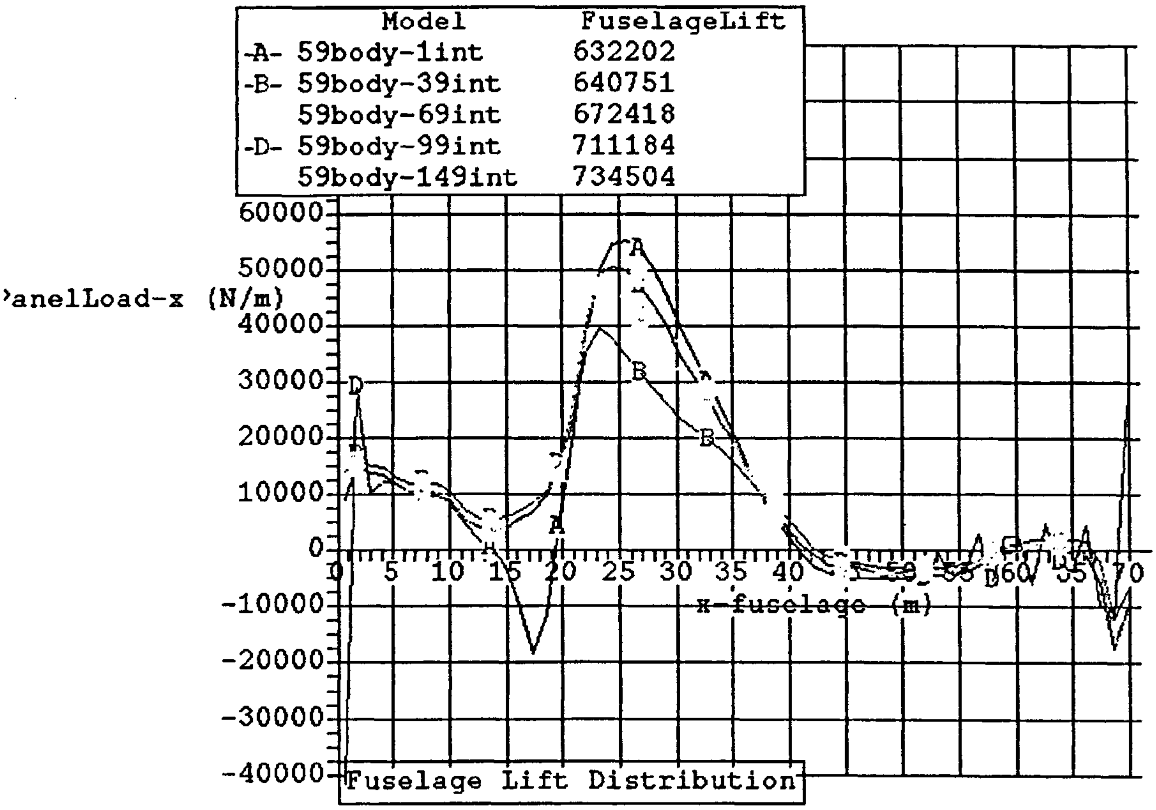


Figure 7.3 : Results Number of Interference Panels

As can be seen in figure 7.3 the model with only one interference panel already shows a remarkable close estimate of the fuselage lift compared to the other models. The model with 99 interference panels has some unexplained oscillating effects at the nose and tail. The other models are relatively close to each other. The dip at the end of the fuselage is caused by the way the fuselage is modelled at the end. The fuselage size is reduced here from 2.0 m radius to 0.0 m within 3 meters fuselage length. This is done to get the tail plane to intersect with the interference tube (see 7.1.3). This is a very rapid decrease in size and it is thought to be to much for the DLM to handle. But the effect on the overall fuselage lift distribution is negligible. From the figure it is concluded that the model with 69 interference panels is sufficient. To proof this another test is done, now with 211 body elements. Results from this are shown in figure 7.4.

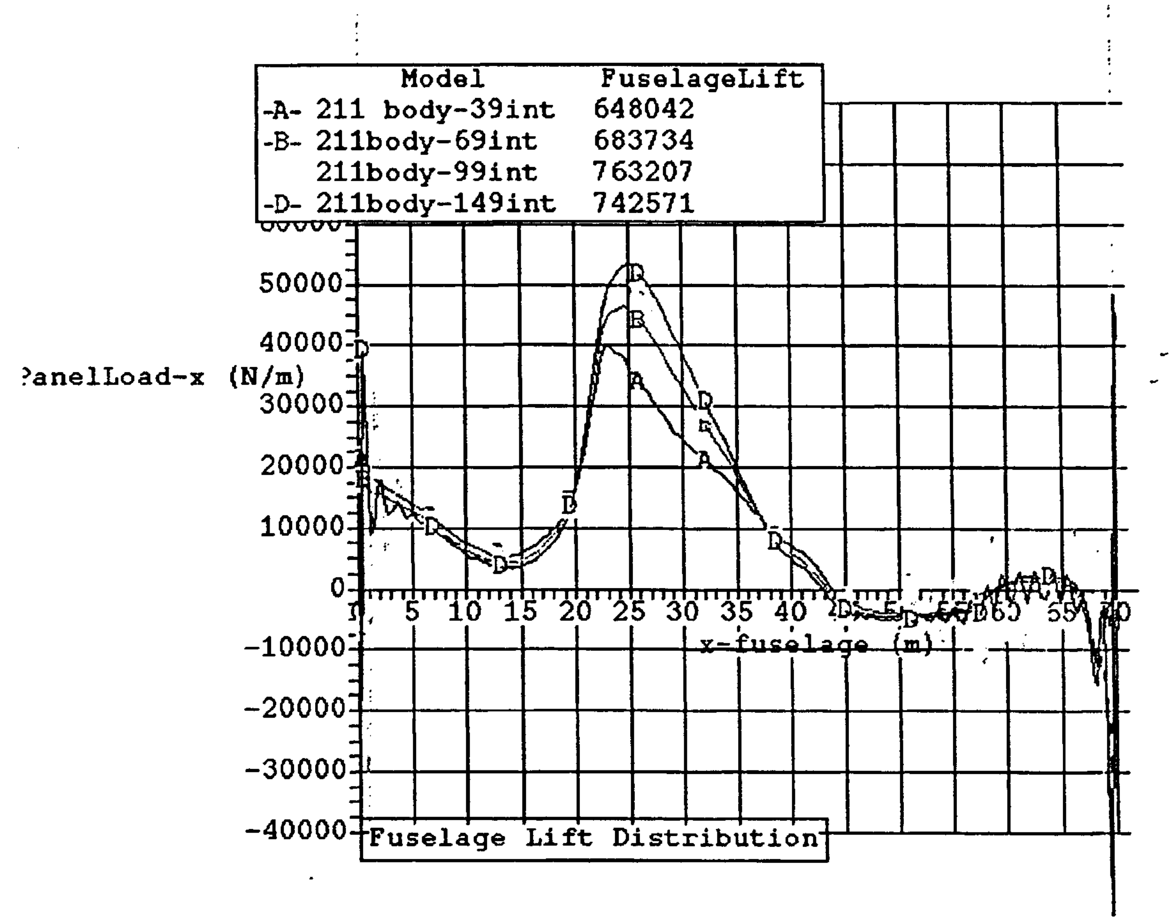


Figure 7.4 : Results Number of Interference Panels

From figure 7.4 it can be seen that the model with 99 has the same oscillating pattern as in figure 7.3 but with even larger peaks at the nose and tail end of the fuselage. The model with 149 interference panel also shows this pattern, but less severe. The total body lift for the model with 211 body and 69 interference elements (683734 N) is not much different from the model in figure 7.3 with 59 body and 69 interference elements (672718 N) and the shape of the lift distribution is very similar.

7.2.2 Convergence Study to Determine the Number of Body Panels

A number of models have been evaluated to determine the necessary number of fuselage body elements on the fuselage. The first model has body panels of 2 meters length (35 body panels) and 69 interference panels were used (see previous paragraph). The second model has 59 body elements, the third 140 and the last 211. Results are shown in figure 7.5

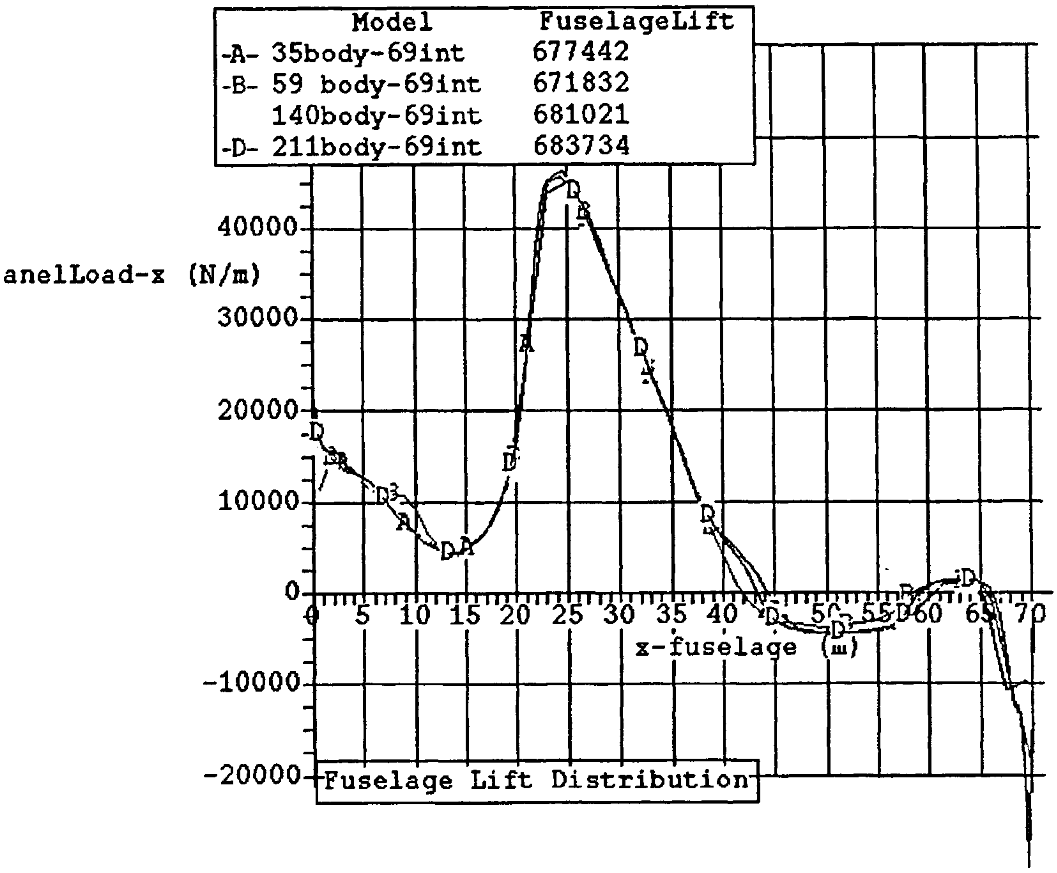


Figure 7.5 : Results Number of Body Panels

From figure 7.5 it can be seen that the influence of the number of body elements on the tota fuselage lift is negligible. The shape of the lift distribution is also relatively constant.

From the results it can be concluded that the number of interference elements is of much greate influence than the number of body elements. Per model the number of interference elements has t be determined separately. For this model 59 body and 69 interference elements are sufficient fo this model.

7.2.3 Convergence Study to Determine the Number of Wing Panels

In paragraph 5.5.3 it was concluded that increasing the number of wing panels had almost n effect on the wing lift distribution. To determine the influence of the fuselage on the wing an (visa versa) a convergence study of the wing panels in combination with the fuselage panels wil be conducted. Initially a fuselage model with 140 body elements and 69 interference elements i chosen as it is felt necessary to have more body panels if the number of wing panels is increase (increasing the number of body panels has no significant effect, see figure 7.5).In figure 7.6 th influence of the number of wing panels on the fuselage lift distribution can be seen. In figure 7. the influence of the number of wing panels on the wing lift distribution can be seen. The increas in the number of wing panels is first concentrated on the wingpanel next to the fuselage (panelII 1002000 see paragraph 5.5). The number in the model name refers to the number of chordwis panels on that macro wing panel adjacent the fuselage.

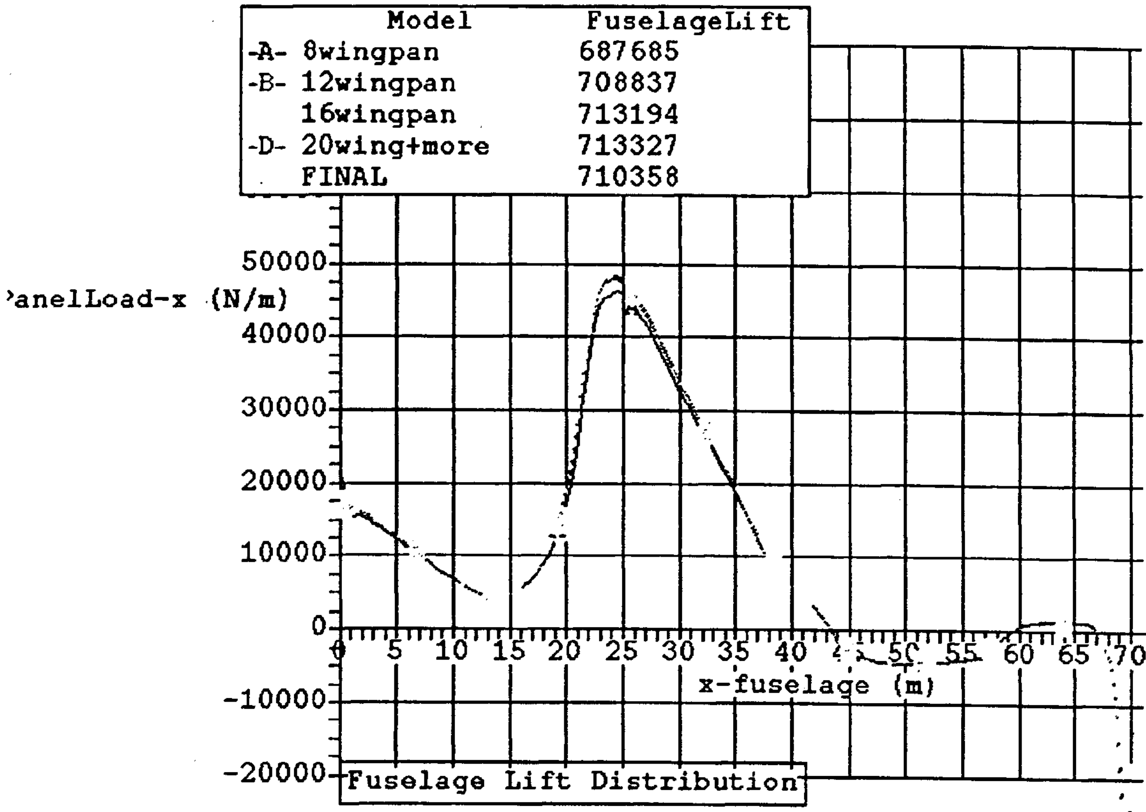


Figure 7.6 : Results Number of Wing Panels on Fuselage Lift Distribution

From figure 7.6 it can be concluded that the number of wing panels has only a marginal influence on the fuselage lift distribution and the total lift generated by the fuselage. More than 16 wing panels gives almost no increase in fuselage lift anymore

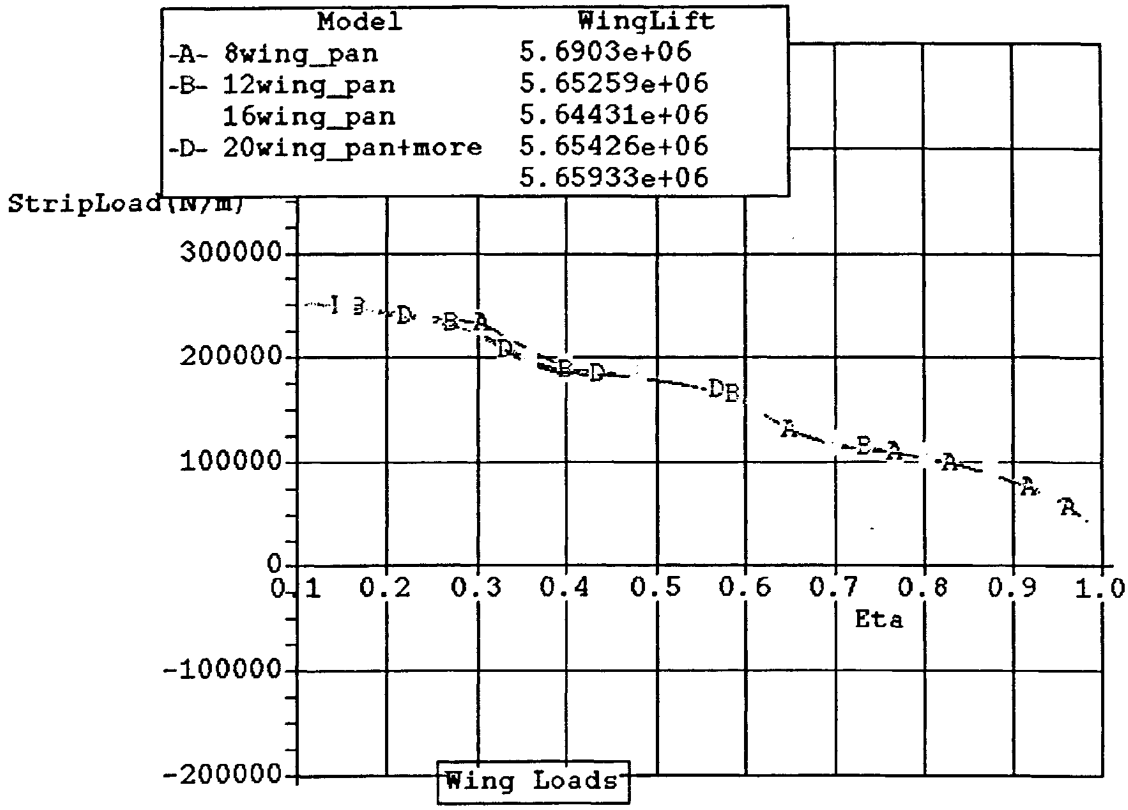


Figure 7.7 : Results Number of Wing Panels on Wing Lift Distribution

From figure 7.7 it can be concluded that the increase in number of wing panels makes the wing lift distribution smoother, but has no significant effect on the total wing lift. This is only different because the fuselage lift is changing a little with wing panel density. To have 16 panels on the wing adjacent the fuselage is sufficient. In figure 7.7 the position of the engines is visible as a dent in the line. Here the engine nacelle is influencing the wing lift.

7.3 The Final-Half-FE-Model

The convergence studies in the previous paragraphs complete the modelling of the FE-model representing half the reference aircraft. The FE-model can now be used to perform static aeroelastic analysis using MSC/NASTRAN. Results can be fed into TDMB (with NASTPP) to make the specific aeroelastic results visible in graphs. The Final-Half-FE-model consists of the following:

- The original non-optimised FE-model generated by the MMG. Unchanged are: the wing box structure, the stick model of the fuselage and fin, and the engine models. The amended AEG-module was used to ensure proper connection of the aerodynamic panels to the wing structure (see chapter 5.5)
- The CC-deck and SPC-deck needed to make the model suitable for static aeroelastic analysis (see chapter 5.3)
- The Fuselage deck needed for the aerodynamic model of the fuselage: 140 body elements and 69 interference elements (body and interference elements are evenly divided over the fuselage length) (see chapter 7.1). See also figure 7.1 which is a top view of the final-half-FE- model (the number of body elements is actually double the number drawn in figure7.1). The fuselage is lowered to ensure proper connection to the aerodynamic panels of the wing (see figure 7.2)
- Wing aerodynamic panels. Adjusted: the model now has the following panel division over the wing (see also figure 7.1 & 5.7):
  - Panel 1001000 : removed to give way for the fuselage panelling
  - Panel 1002000 : 16 sub panels chordwise, 6 spanwise
  - Panel 1003000 : 14 sub panels chordwise, 1 spanwise
  - Panel 1004000 : 12 sub panels chordwise, 7 spanwise
  - Panel 1005000 : 8 sub panels chordwise (excluding aileron which has 3 chord wise panels), 4 spanwise
  - Panel 1006000 : 9 sub panels chordwise (excluding aileron which has 3 chord wise panels), 7 spanwise
  - Panel 1007000 : 8 sub panels chordwise, 1 spanwise
- Tail aerodynamic panels. Adjusted: the tail plane is raised to ensure connection of the aerodynamic panels to the interference tube of the fuselage at a 2.0 m spanwise position. Aerodynamic panels from fuselage centreline to 2.0 m span removed to give way for fuselage panelling. The number of aerodynamic panels on the tail is increased to 15 chordwise.

A lot of the adjustments made to the original FE-model have to be made manually at this stage in the project (except for the modified AEG-module). The MMG will have to be amended so that the aerodynamic model of the fuselage is incorporated automatically into the FE-model. This however falls outside the scope of this thesis project.

The Final-Half-FE-model is used to do the static aeroelastic calculations using the flight conditions shown in chapter 5.3.2. In figure 7.8 the results of two subcases are presented (taken directly from the TDMB): the pull_up manoeuvre and the push_down manoeuvre. The numbers in the figure are in Newtons.

```
ST_AERO_PULL_  *                       *               Analysis for each operation.
    WingDeflectio   WingDeflectio   *                      Deflected wing shape informati
    WingLiftForce   WingLiftForce   *                      Lift forces on Aerodynamic Par
        *               #                   *                  *
        MacroPanels     Array               *                  Wing Macro Panels
        WingStrips      Array               *                  Aero Forces totalled over
        FuselagePanel   Array               *                  Fuselage Aerodynamic Panel
        Totals          *                   *                  Total Lift forces
            WingLift        Real                5.65933e+06    Total Lift of the Winc
            TailLift        Real                81420          Total Lift of horizont
            IbdELift        Real                130672         Total inboard Engine I
            ObdELift        Real                134815         Total outboard Engine
            AilerLift       Real                65273.3        Total aileron Lift
            FuseLift        Real                710358         Total fuselage Lift
            TotalLift       Real                6.78187e+06    Total Lift
ST_AERO_PUSH_  *                       *               Analysis for each operation.
    WingDeflectio   WingDeflectio   *                      Deflected wing shape informati
    WingLiftForce   WingLiftForce   *                      Lift forces on Aerodynamic Par
        *               #                   *                  *
        MacroPanels     Array               *                  Wing Macro Panels
        WingStrips      Array               *                  Aero Forces totalled over
        FuselagePanel   Array               *                  Fuselage Aerodynamic Panel
        Totals          *                   *                  Total Lift forces
            WingLift        Real                -2.26373e+06   Total Lift of the Winc
            TailLift        Real                -32568         Total Lift of horizont
            IbdELift        Real                -52268.9       Total inboard Engine I
            ObdELift        Real                -53925.9       Total outboard Engine
            AilerLift       Real                -26109.3       Total aileron Lift
            FuseLift        Real                -284143        Total fuselage Lift
            TotalLift       Real                -2.71275e+06   Total Lift
```

***Figure 7.8** : Final Half Model Static Aeroelastic Calculations Results.*

In the pull_up manoeuvre it can be seen that the fuselage provides 710358 N lift. This is approximately 10.5 % of the total lift generated. The lift on the tail plane is positive while a negative lift force is expected. The reason for this lies in the absence of a negative pitching moment which is normally generated by the wing. The fuselage also has a overall positive pitching moment, needed to be countered by a positive tail force (see figure 5.10, where the model without fuselage has a negative tail force). Introducing a negative pitching moment (at the aerodynamic centre of the aircraft) would not affect the total lift required or the fuselage lift generated. Only the wing would need to generate more lift. In a later stage this will be investigated. For the push_down manoeuvre a similar discussion as for the pull_up manoeuvre can be held only the results are negative.

7.4 The Full-Model

One of the MDO-software utilities is a program called FULLMODEL (see paragraph 3.2.4). This program mirrors the FE-wing model generated by the MMG in the x-z plane so that a complete FE-model of the wing is created. The stickmodel for the tail and all the aerodynamic panels are mirrored as well, making the FE-model complete. The program also adds the "missing" weight to the stickmodel of the fuselage representing the payload and fuselage mass. The model becomes then suitable for simultaneous symmetric and anti-symmetric analysis and optimisation.
Figure 7.9 is the top view of the full-FE-model (the final-half-FE-model was mirrored). Shown are the wing torsion box, the stickmodel and the aerodynamic panels of the wing and tail. The fuselage is not shown as MSC/NASTRAN cannot plot body or interference elements.

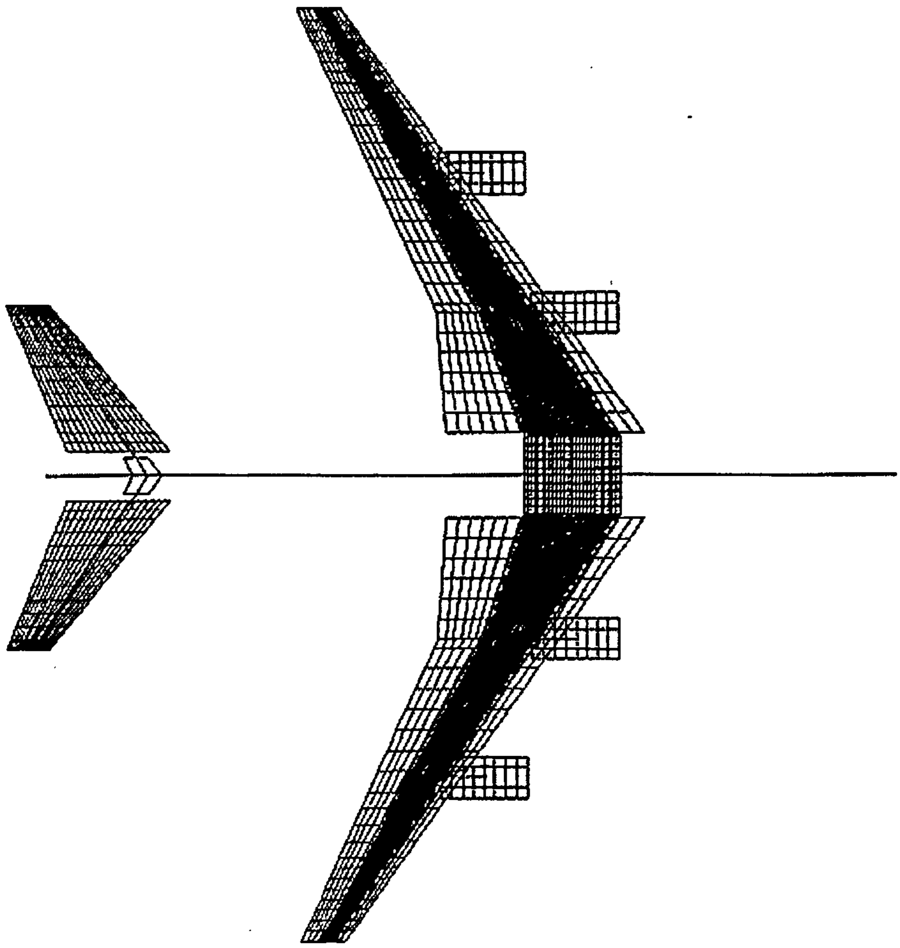


***Figure 7.9** : Top View of the Full FE-model Including Aerodynamic Panels*

The Full-FE-model will be used for checking the results of the final half-model. In figure 7.10 the wing strip loads of the half and full FE-model are shown. In figure 7.11 the fuselage strip load for the half and full model are shown.

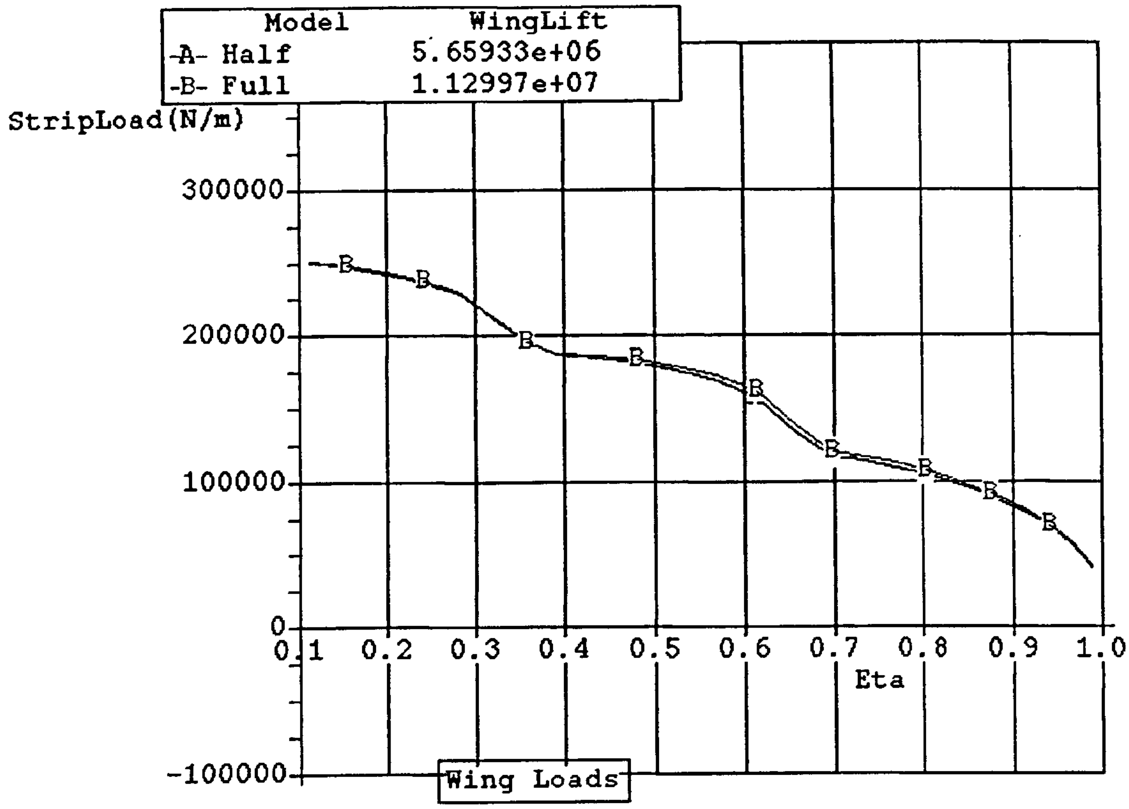


Figure 7.10 : Wing strip load Comparison for Half and Full FE-model

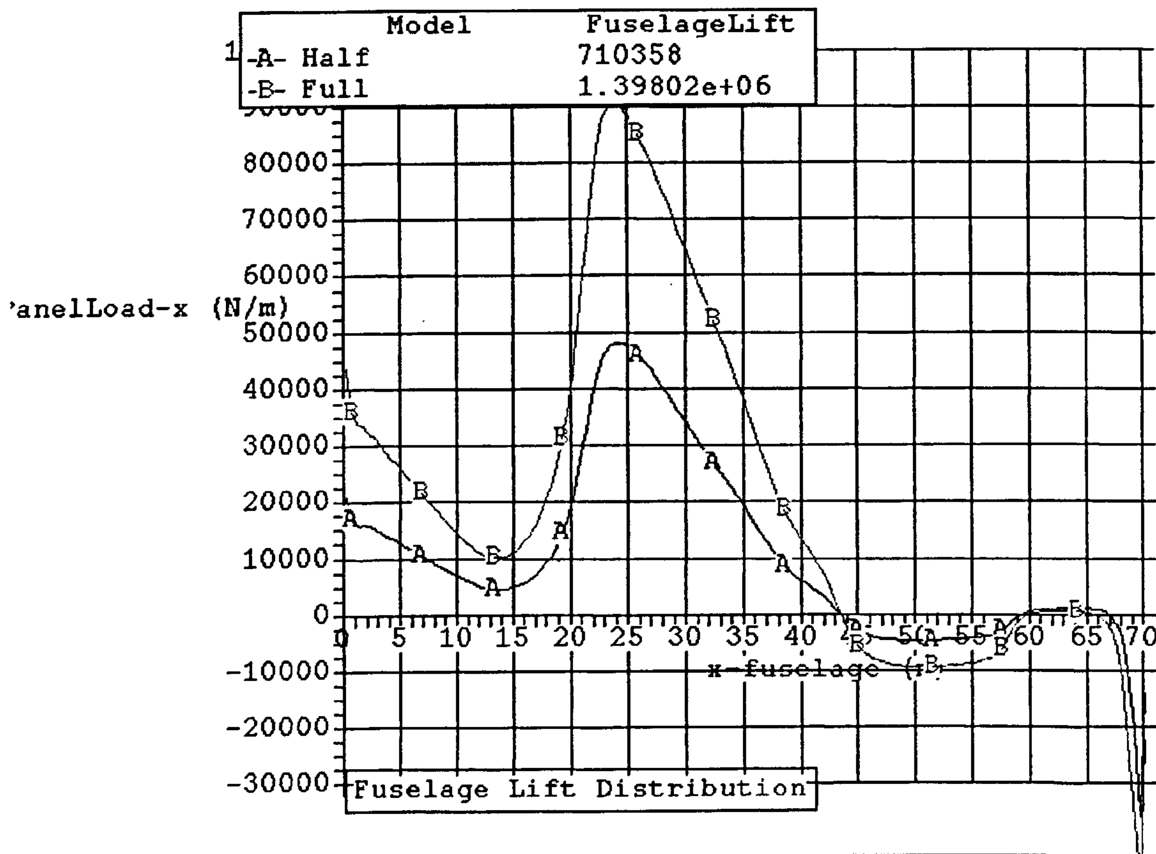


Figure 7.11 : Fuselage strip load Comparison for Half and Full FE-model

Figure 7.10 shows no significant difference between the striploads on the wing of the half and full model. The loads on the fuselage (figure 7.11) of the full model are twice the loads of the half model. This is due to the fact that with the full model the whole fuselage is modelled generating twice the lift.

From figures 7.10 & 7.11 it can be concluded that the final half FE-model is working correctly. For asymmetric calculations the full FE-model can be used.

7.4.1 Roll Manoeuvre With Full-FE-Model

The Full-FE-model is used to investigate the 10 deg/sec roll manoeuvre. For this manoeuvre the TRIM card is adjusted to give the aircraft roll speed. The ailerons are not deflected for this manoeuvre (and kept in their 0.0 position) because aileron deflection would instigate a roll acceleration. The SUPORT card is adjusted to give the aircraft the degree of freedom to roll along the longitudinal axis of the aircraft. The SUPORT point then has degrees of freedom in 1 direction (Z) and around 2 axis (X & Y). The fin of the aircraft is not modelled. This is not necessary as the degree of freedom around the top axis can just be constraint with the SPC card.

The results from MSC/NASTRAN with this model and under the trim conditions mentioned above are inconclusive. The results show unrealistic angles for the elevator needed to trim the aircraft. Various other trim conditions have been investigated, but the results remained inconclusive and not feasible.
Due to time constraints, this model is not further investigated for this project.

7.5 Conclusions

The modelling of the aerodynamic model of the fuselage in the FE-model is complicated. There are a lot of influences that affect the FE-model and the results and MSC/NASTRAN is sensitive to small changes made in the model. The convergence studies quantified the influences of wing panel density, fuselage panel density, fuselage interference panel density, interference tube theta division. The final-half-model-FE-model is the result of the convergence studies and shows satisfying results.
The results show that the static-aeroelastic simulations using the final-half-FE-model are solved reliably and consistently. The models can now be used for calculations needed for comparisons with data from CFD.

CHAPTER 8 : COMPARISON WITH CFD RESULTS

8.0 Introduction

The MSC/NASTRAN final half FE-model described in Chapter 7.3 can now be used for Stati Aeroelastic Analyses. To be able to understand and quantify the results from these analyses, comparison with results from CFD calculations will be made. The results from the BAeFlutter an Loads Suite are not be included in this comparison due to the inability of BAe to provide th necessary data in the given timespan.

The model used in the CFD calculations differs from the FE-model. As CFD calculations tak longer than FE-calculations, it is easier to adjust the FE-model for the comparison. The influenc of the adjustments on the FE-model will be investigated to determine the validity of the adjuste FE-model.

8.1. CFD-Model and Data

Computer Fluid Dynamics computations solve the mathematical equations (Euler /Navier-Stokes of the airflow around a body. The results of these calculations are the aerodynamic pressure working on the body in the airflow. From these pressures, other interesting things, for example, li and drag coefficients can be derived.

8.1.1 CFD-Model

The CFD-data used for the comparison is generated by the Aerodynamics department of Britis Aerospace Airbus Engineering, Woodford. The aircraft model used for the CFD calculation represents an early version of the future Airbus A3XX. The reference aircraft and the A3XX ar similar types of aircraft and comparison of result is possible if the FE-model is modified. Th CFD-model has the following characteristics:

- Rigid aircraft
- No Tail : Wing-Fuselage only
- No Engines on Wing

In figure 8.1 a view of the CFD model is shown. The lines in the figure show the aerodynami elements, forming the fuselage and wing of the CFD-model.

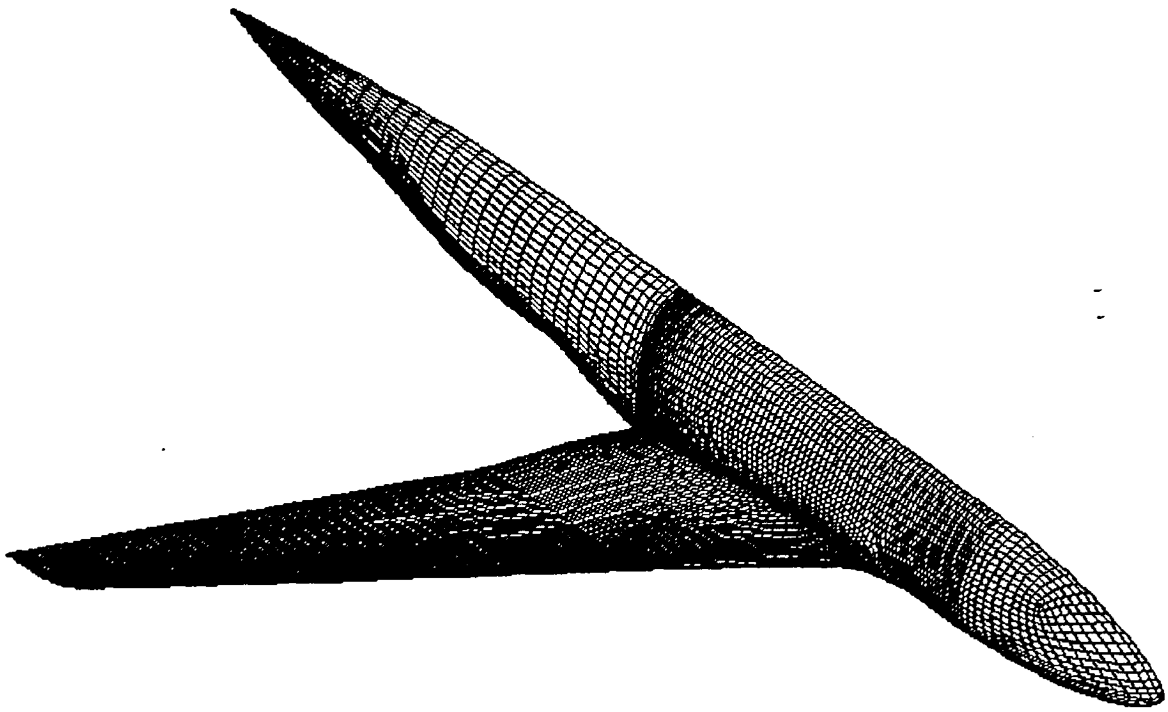


Figure 8.1 : CFD-model

The fuselage dimensions of this CFD-model are similar to the reference aircraft, so the FE-model needs no adjustments of its fuselage model:

Fuselage	Length	70.5 m
	Width	7.02 m
	Height	8.50 m

The dimensions of the wing of the CFD-model differ from the reference aircraft, although the wing planform has the same shape:

Wing	Area	800 m^2
	Span	79.0 m
	Root Chord	16.85 m
	Kink Chord	11.27 m
	tip Chord	3.97 m
	eta kink	0.334
	Sweep	33°
	Root twist	2.8°
	Tip twist	-2.25°

The FE-model can easily be adjusted using TDMB and the MMG so that the FE-model will have the same dimensions as the CFD model.

8.1.2 : CFD-Data

The model described in 8.1.1 was used in a number of CFD-calculations. The main variable has been the angle of attack. Five different angles have been calculated, starting at 1° AOA and increasing in steps of 0.5° to 3° AOA. For the comparison, only the results from 1°, 2° & 3° AOA cases will be compared with the results of the FE-model. The results from the CFD-calculations are available in text files. Appendix-J is an example of a part of one of those text files. In appendix-J the results for a fuselage section are listed : the X, Y, Z position, the calculated isentropic Mach number and the calculated pressure coefficient cp. The flight conditions are determined by the Reynolds number and Mach number. For these calculations were performed with the following numbers:

- Mach number : 0.7
- Reynolds number : 8.0 e7
- Temperature : 288 K

The results from the CFD-calculations can be made visible as contours on the surface with a program called Interactive Configuration Management System. This program is available on the computer system at BAe Airbus in Woodford. An example of one of the possible plots generated with this computer program is given in figure 8.2 . The CFD-model is shown with pressure coefficient contours on it. Each line indicates a change in pressure coefficient of 0.02. The colours in the figure are :

- Light bleu $-1.0 \leq cp < -0.6$
- Dark bleu $-0.6 \leq cp < -0.4$
- Green $-0.4 \leq cp < -0.2$
- Red $-0.2 \leq cp < 0.0$

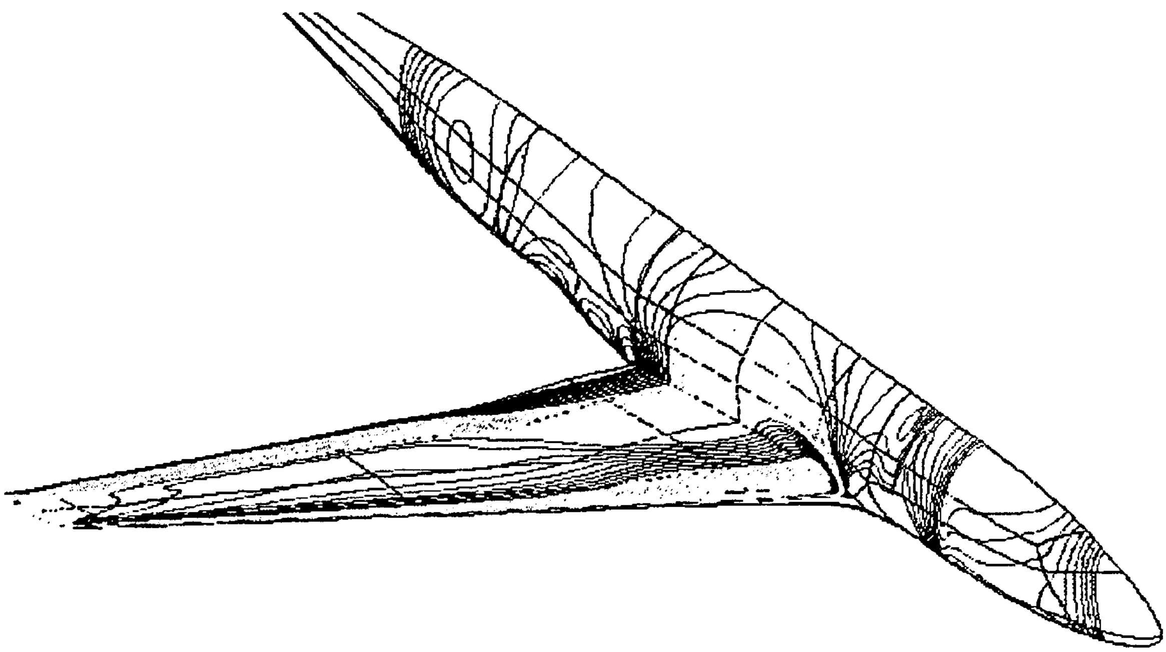


Figure 8.2 : Contours of pressure Coefficients on the CFD model (1 deg AOA)

The pressure coefficients can be used to calculate lift and drag coefficients by integrating the pressure coefficients over the surface area. A special post-process program is available at BAe Woodford to perform these integrations. In Appendix-K the results (for the fuselage under 1°AOA) from this post-process program are shown: First the information about the followed solution etc. is printed. Thereafter the results due to pressures are listed. The most important are: position x, chord (in this case fuselage width), Cl and Cd. It can be seen that there are no results for stations 20 to 37. Between these positions the wing is located and the fuselage is thus disrupted. The program cannot integrate across the disruption. The CFD-post processor generates the results in a format which can be used for comparison with FE-results. For the wing a similar listing is made by the post process program, with spanwise Cl and Cd results. As Cl distributions are readily available, the comparison will not use wing loading distributions but the Cl distributions.
In Chapter 8.5 the results of the CFD-Calculations are compared with the results from the FE-calculations.

8.1.3. CFD -NASTPP/TDMB Amendments

For the comparison it is necessary that results from both the CFD-calculations and the FE-calculations can be shown in a single graph. The CFD-results thus have to be read into the TDMB. The results from the CFD calculations are readily available in text files. This makes NASTPP (see Chapter 6) the obvious choice for reading these text files, extracting the necessary data, processing the data, and writing the processed data into TDMB. Both TDMB and NASTPP are adjusted for reading the CFD-results.

TDMB amendments

Results from the wing
Within TDMB there is already a section for aerodynamic results (Results AeroResults). In this section there is an array especially for wing section results : Result AeroResults WingSection. This array can contain all the necessary data (like eta, Chord, Cl, Cd, Cm, etc.) The CFD results from the wing will be written into this section.

Results from the fuselage
In the TDMB section mentioned above there is no space for fuselage results. A new array is created, similar to the WingSection array, in the Results AeroResults tree, called FuselageSection. It contains the same information as the WingSection array (eta, Chord, Cl, Cd, etc.). For this to work the following file has to be edited and adjusted (see also chapter 6.4):

- /MDO-Programs/ObjectLibrary/AeroResult.tom : create the FuselageSection array and add the necessary items.

NASTPP supplements
For NASTPP to be able to read the text files with the CFD-results, a new routine which reads this text file, processes the data, and writes it into TDMB, has to be added (see Chapter 6.2 for details on NASTPP). This routine can be viewed in Appendix-H.

Approach Followed for NASTPP CFD Supplements

- An option 7 has been created in the menu which prompts as the NASTPP utility is invoked. This option reads the CFD-results from the user specified CFD-results file (this can have any name). The user then has to specify if the data is from the fuselage or wing after which a design stage has to be specified.
- From the CFD results file the NASTPP reads and writes the following information (this is the same for both wing and fuselage options):
  - *Point number* : this will be the number for the section in the array WingSections or Fuselage Sections
  - *Eta* : The relative position of the section
  - *Chord* : The chord of the section (in the fuselage case, this is the width of the fuselage at this section)
  - *Cl* : The lift coefficient of the section
  - *Clccbar* : The lift coefficient times the chord divided by a reference chord
  - *Cd* : The drag coefficient
  - *Cdccbar* : The drag coefficient times the chord divided by a reference chord
  - *Cdviscous* : The viscous drag coefficient (if available) : Drag due to viscous effects.

Remarks

- The CFD results as shown in Appendix-K have to be altered so that only the numbered lines of the sections are in the file without any other information. This is because NASTPP will read the file line by line and errors might occur if there is other information in the file.
- The -53 option in the NASTPP menu does NOT erase the results in the AeroResults section of TDMB, only in the StructResults section. If on wants to delete stage results, this has to be done by hand in TDMB.
- If one wants to read more information from the CFD-files into TDMB, NASTPP can easily be adjusted to do so.

Figure 8.3 shows one of the possible figures which can be generated with the amended NASTPP and TDMB utilities. The figure shows the results from the CFD-calculations : Clc/mac distribution of the wing for the three cases :1°,2° & 3° AOA.

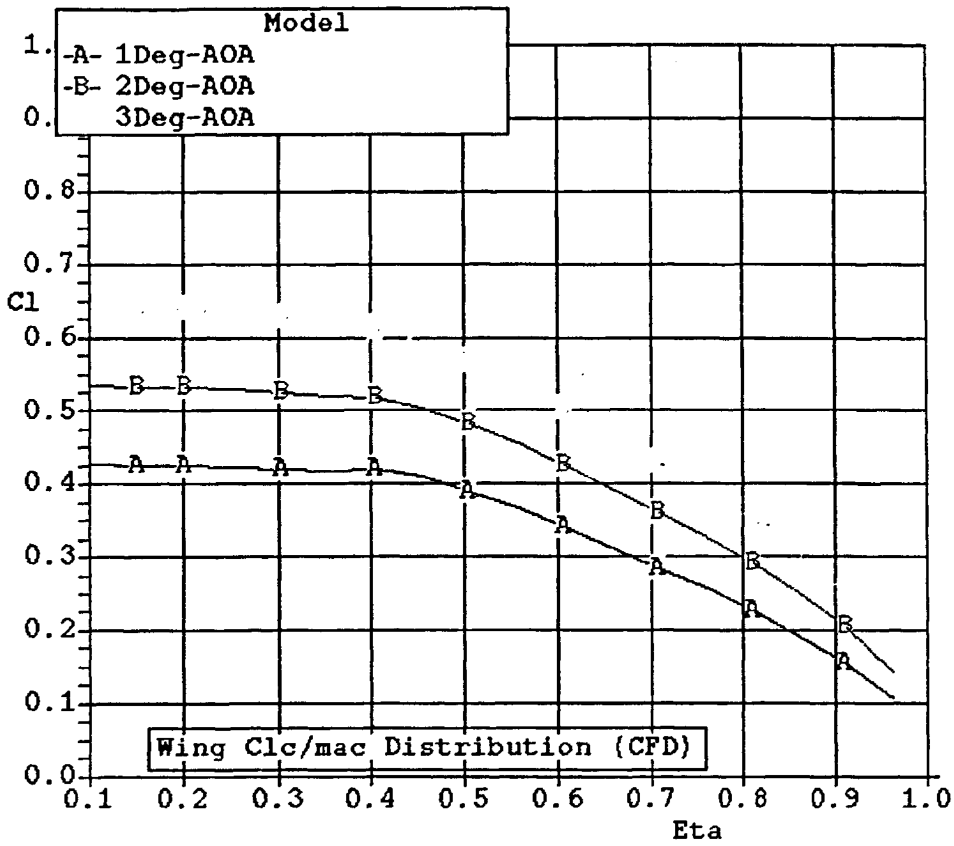


Figure 8.3 : Wing Clc/mac distribution Wing (CFD Results)

8.2. FE-model.

The FE-model which will be used for the comparison needs adjustments to match the configuration of the model used in the CFD-calculations. Changing the FE-model will influence the results and to be able to understand and quantify the influences of these adjustments, it is necessary to compare the adjusted FE-models with the non adjusted FE-model, before the comparison with the CFD data can be made. The way MSC/NASTRAN calculates the static aeroelastic results need some alteration as well.

8.2.1 FE-Model Adjustments

The model configuration used with the CFD-calculations is a rigid aircraft with only a wing and fuselage. The FE-model to be used for the comparison will have to have the same configuration: no engines, no tail plane, and a rigid structure. A number of FE-models are made to see the influence of rigidity, engines, and tail on the aeroelastic results.

These FE-test models are:

1. ***Final half model*** : This is the FE-model described in the chapter 7.3, consisting of the FE-model of half the reference aircraft with increased number of aerodynamic panels on the wing and tail, including the fuselage aerodynamic panelling. This will be used as the reference to which the changes will be compared.
2. ***Rigid wing model*** : The wing is made more rigid by increasing the value of the elasticity modules of the materials used in the FE-model. The effect on wing bending and lift distributions is investigated.
3. ***No engines model*** : The engines will be removed completely. The effect of the engine mass on the wing bending and the effect of the engine lift on the lift distributions is investigated. The engine mass is removed from the FE-model and is not replaced.
4. ***No tail model*** : The aerodynamic panels of the tail plane will be removed. The stick model of the tail, representing the mass of the tail will stay in place. The trim conditions change to balance the aircraft without tail.
5. ***Rigid, No engines, no tail model*** : The influence of all the previous models combined is investigated.

The models all fly with the conditions described in paragraph 5.3.2 for this specific comparison. Only the results for a 2.5 g pull_up manoeuvre are shown in the following figures.
In figure 8.4 the bending distribution of the wing for the different FE-models is shown. Figure 8.5 shows the lift distributions of these models and in figure 8.6 the effect of the changes on the fuselage lift distribution is shown.

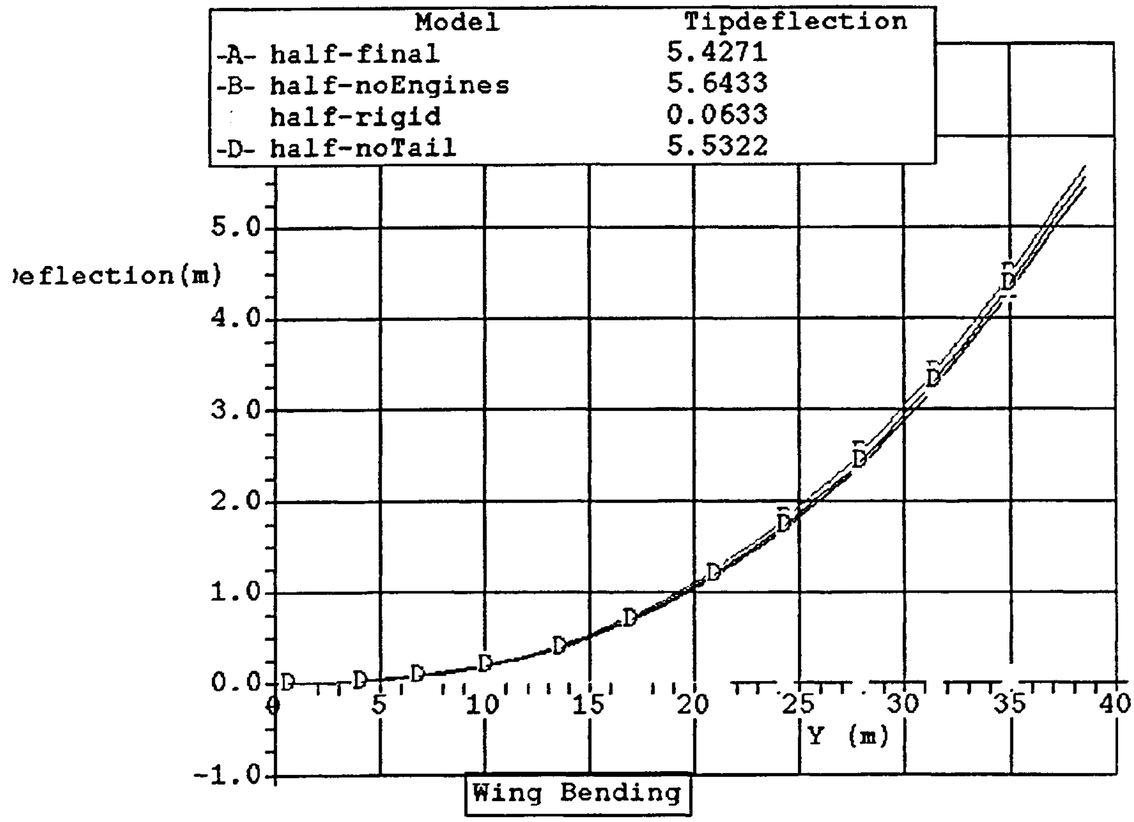


Figure 8.4 : Wing Bending of the Test Models

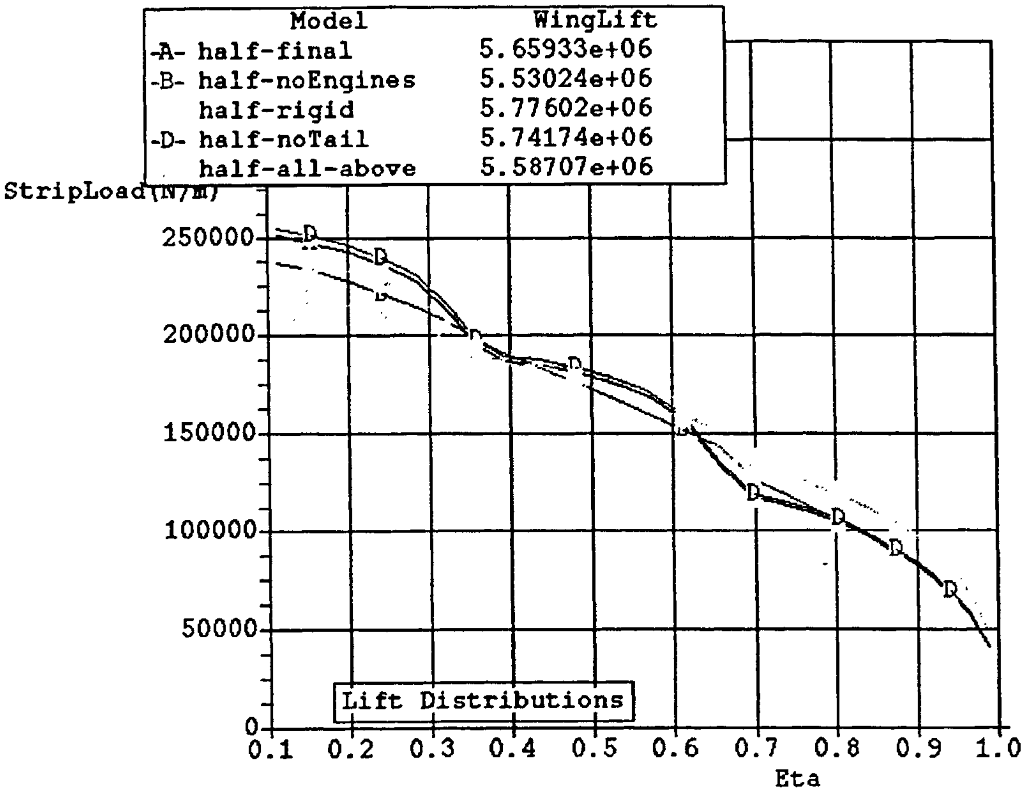


Figure 8.5 : Wing lift distribution of the Test Models

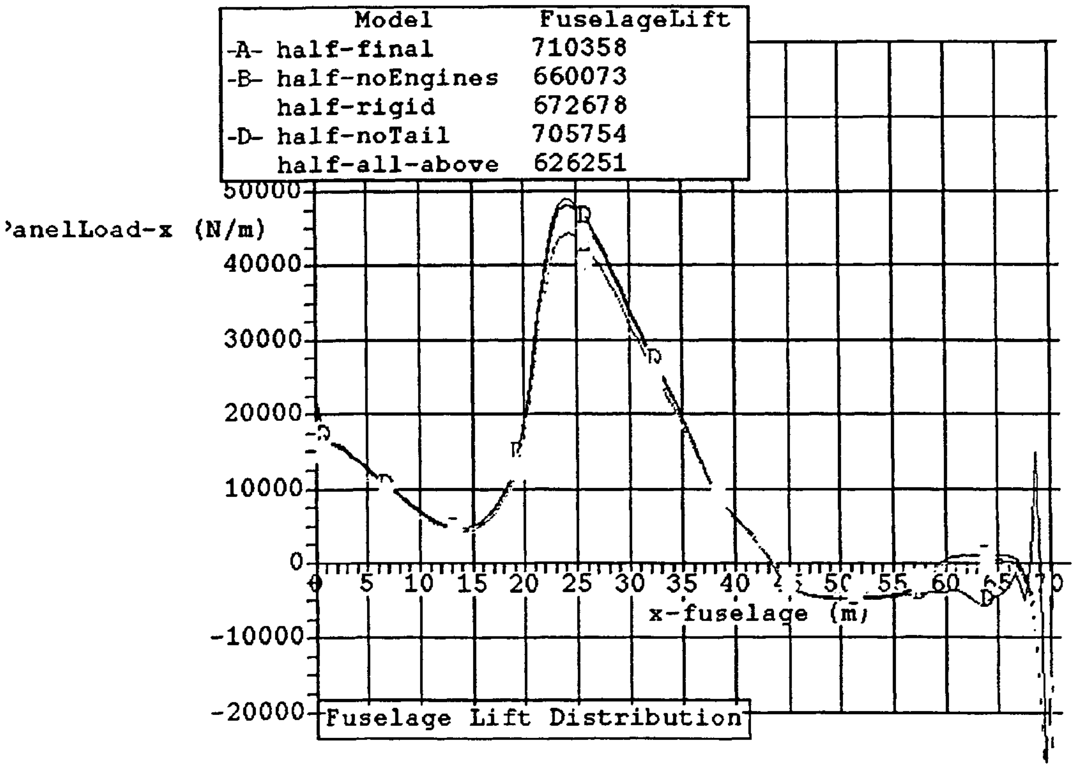


Figure 8.6 : Fuselage lift distribution of the Test Models.

For figures 8.4, 8.5 & 8.6 the following remarks can be made:

- ***Final half model*** : Reference to which the changes are compared.
- ***Rigid wing model*** : The rigidity of the wing is increased by a factor 100 to make the torsionbox of the wing and the stickmodel of the fuselage and tailplane almost inflexible. The rigid wing bends only slightly (figure 8.4). In figure 8.5 it can be seen that the outboard wing sections produce significantly more lift. If the wing bends it causes the outboard sections to twist negatively, reducing the local AOA. This then reduces the amount of lift compared to the rigid wing. The influence on the fuselage lift distribution is negligible (figure 8.6).
- ***No engines model*** : Without the relieving bending moment of the engine mass, the wing bends up more. But the total generated lift is lower because of the absence of engine mass (see figure 8.4). The lift distribution over the wing shows a smooth line (see figure 8.5) compared to the "dented" line for the other models. The engines have a significant influence on the wing load distribution. There is no real influence on fuselage lift
- ***No tail model*** : The aerodynamic panels of the tail plane are removed. The positive lift of the tail plane is replaced by extra wing lift. The model is trimmed in this case by a constant pitch rate of approximately 1.35 deg/sec. The removal of the tail has no significant effect on the wing lift distribution (figure 8.5). The nose of the fuselage is not as highly loaded without tail, which is due to the pitch rate (see figure 8.6). In this case the peaks at the end of the fuselage are still visible, confirming that the tail-fuselage combination is not the cause of these peaks (see Chapter 7.2.1)
- ***Rigid, No engines, no tail model*** : The influence of all the previous models combined is shown in figure 8.5 and 8.6 only. The increased rigidity and the removal of the engines have combined a big influence on the wing loading. At the root, wingloading decreased from 260000 N/m to 225000 N/m. The influence on the fuselage is marginal.

The adjusted FE-models respond as expected to the changes. The results are consistent and the conclusion can be drawn that the rigid FE-model without engines and aerodynamic panels on the tail can be used as basis for the comparison.

8.2.2 MSC/NASTRAN Related Adjustments

For the FE-model to be able to "fly" at a specific angle of attack, the trim condition in MSC/NASTRAN would not work correctly anymore. At 1° AOA the lift generated by the fuselage and wing would not be sufficient to support the weight of the aircraft. This would then force the trim module in MSC/NASTRAN to come up with an unrealistic trim condition. In these cases the degrees of freedom for the rigid body on the SUPORT card need to be constrained with an SPC-card. The trim module then uses the TRIM card to "set" the aircraft and the control surfaces in the desired position. Then the module calculates the aerodynamic pressures and forces, structural deflections and stresses etc. for this position. But in these cases the trim module does not calculate the stability derivatives or accelerations trim variables such as load factor.
One has to consider that the deflections of the structure in these constrained, trimmed cases are not relevant for the actual flying condition as the aircraft is not in equilibrium, but forced into a specific attitude. If for example 1° AOA does not generate enough lift to counter the weight and the structure is only deflecting due to lift generated, and not due to lift needed to counter the weight. Likewise stresses in the structure are only due to lift generated and for example there will be no effects of accelerations. For the comparison this is not a problem as the aircraft is rigid. There are no deflections of the wing due to lift and the comparison only looks at the lift generated by the wing. Stresses in the structure are not considered for this comparison.

8.2.3. FE-Model for Comparison

A separate FE-Model is now created for the comparison calculations. TDMB is used to alter the dimensions of the wing to match those of the CFD-model (see Chapter 8.1.1). Then the MMG is used to generated the FE-model, including the aerodynamic panels of the wing and tail. This FE-model is adjusted by hand: The CC and SPC deck are added (see Chapter 5.2) and the Fuselage model is added (see Chapter 7.1). The FE-model is made rigid by increasing the Elasticity modules by a factor 100 and maintaining the same Poison's ratio. The aerodynamic and structural model of the engines are removed and the tail plane aerodynamic panels are deleted. The trim cards are amended so that the desired flight conditions are met.
MSC/NASTRAN is used for the static aeroelastic calculations with this FE-model. The following cases are calculated:

- Subcase 1 : 1° AOA
- Subcase 2 : 2° AOA
- Subcase 3 : 3° AOA

All these subcases use the same FE-model and have the following TRIM card entries in common, except for the AOA (ANGLEA) entry which differs depending on the subcase.

- M = 0.7, this is the same as CFD flight condition
- q = 9195.0 N/m^2 (altitude 33000ft). The dynamic pressure is not relevant if only Cl distributions over the wing and fuselage are compared (as Cl is independent of air pressure and density). If forces are compared, the dynamic pressure is important.
- AILER0 = 0.0, No aileron deflections
- AILER1 = 0.0
- ANGLEA = 0.01745/0.0349/0.05235 depending on subcase. This is the main parameter for the model. The AOA is the angle for both fuselage and wing as there is no angle of incidence defined for the wing.

NASTPP is used to write the results from the MSC/NASTRAN output files into TDMB. In Chapter 8.3 the results of these calculations are compared with the CFD-results.

8.3 Initial Comparison Results

The results from the CFD-calculations and the FE-static aeroelastic calculations with the amended FE-model (see previous paragraph) are now available in TDMB. The database graphical module is used to produce the necessary graphs for the comparisons of the results between the CFD and FE-calculations.

8.3.1 Graphs

In the following graphs, the results of the CFD and FE calculations are shown. Figure 8.7 shows the Wing Cl distributions for both the CFD and the FE model under different angle of attacks.

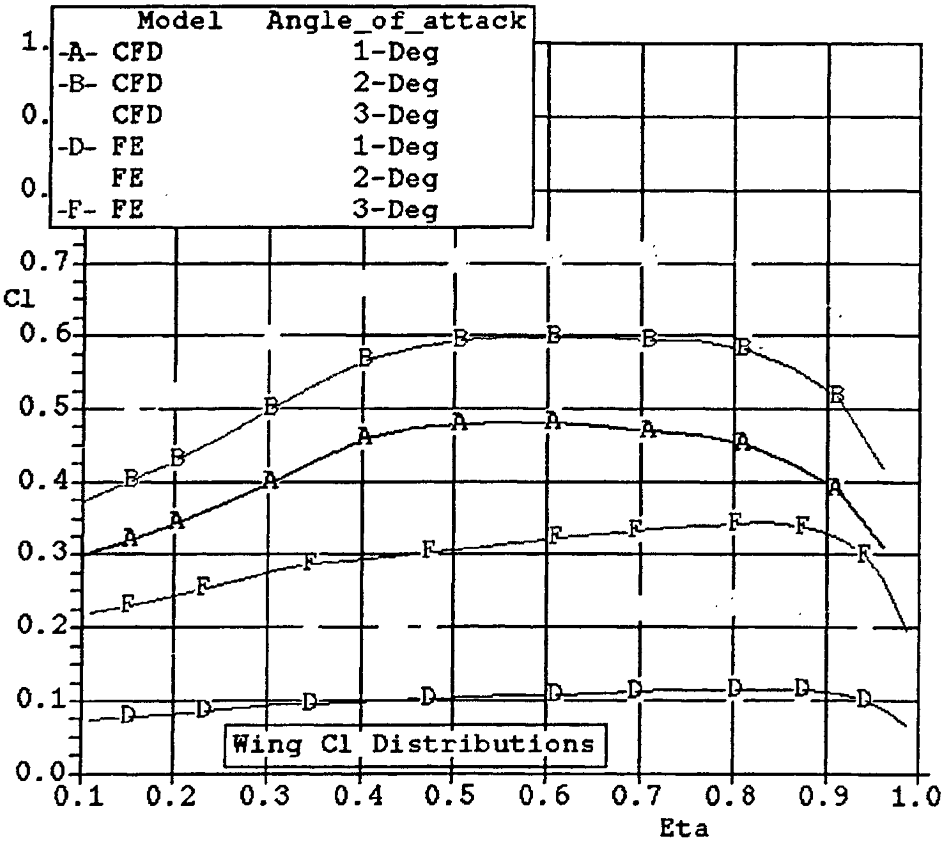


Figure 8.7 : Wing Cl Distributions

For figure 8.7 the following remarks can be made :

- Differences between the CFD and FE results are significant. The 3° FE-case has lower overall Cl than the 1° CFD-case.
- The shape of the FE-Cl-curves is a continuos inclined line until approx. 85% span, where Cl peaks. For the CFD curves there is a straight section between 50% - 80% span and a much steeper inclination from root to 40% span. The CFD curve starts to decline sooner (at 75%-80% span) towards the tip than the FE curve (starts declining at 85%-90% span)
- For both models, the differences between the AOA are similar. A one degree increase in AOA gives both models similar increases in Cl. This means that both models have similar DCl/Dalpha characteristics. This is shown in table 8.1 where for a number of spanwise positions the increase in Cl is calculated between 2° and 3° AOA. The last column shows the % difference DCl/Dalpha between the CFD and FE model. The average difference is 7.3 % (excluding eta 1.0 as this was taken at 95 % span for CFD and 98% span for FE model)

	CFD				FE				
Eta	Cl	Cl	dCl	dCl/dα	Cl	Cl	dCl	dCl/dα	Δ%
	2 deg	3deg			2 deg	3 deg			dCl/dα
0.1	0.374	0.448	0.074	4.24	0.147	0.219	0.074	4.13	2.8
0.2	0.432	0.518	0.086	4.93	0.162	0.241	0.079	4.53	8.9
0.3	0.500	0.597	0.097	5.56	0.188	0.280	0.092	5.27	5.5
0.4	0.566	0.669	0.109	6.25	0.199	0.297	0.098	5.62	11.3
0.5	0.593	0.700	0.107	6.13	0.208	0.309	0.101	5.79	5.8
0.6	0.598	0.710	0.112	6.41	0.216	0.322	0.106	6.07	5.6
0.7	0.594	0.695	0.101	5.79	0.224	0.333	0.109	6.24	-7.3
0.8	0.581	0.700	0.119	6.82	0.231	0.343	0.112	6.41	6.4
0.9	0.516	0.633	0.117	6.70	0.214	0.318	0.104	5.96	12.4
1.0	0.418	0.521	0.103	5.90	0.131	0.196	0.065	3.72	58.0

Table 8.1 : *DCl/Dalpha Comparison*

Figure 8.8 shows the Clc/mac distribution over the wing.

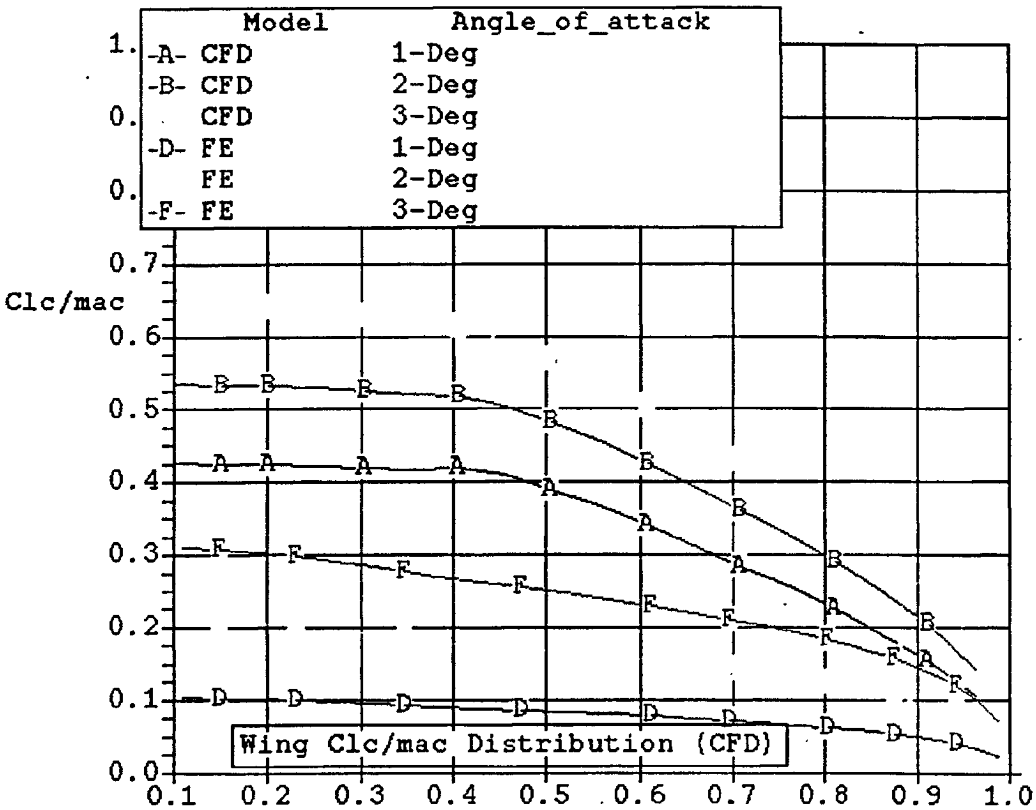


Figure 8.8 : Wing Clc/mac Distributions

For figure 8.8 the following remarks can be made :

- Differences between the CFD and FE results are significant as expected from figure 8.7. The amount of lift from the FE model is too low.
- The shape of the FE curves is an almost continuos declined line until approx. 85% span. For the CFD curves there is a straight section between 0% - 40% span and a much steeper declination from 50% span to tip. This "dent" in the curve is not visible with the FE-model.
- For both models, the differences between the AOA's are similar. A one degree increase in AOA gives both models similar increases in Clc/mac. The last column in table 8.1 proves this.

Figure 8.10 shows the Cl distribution over the fuselage.

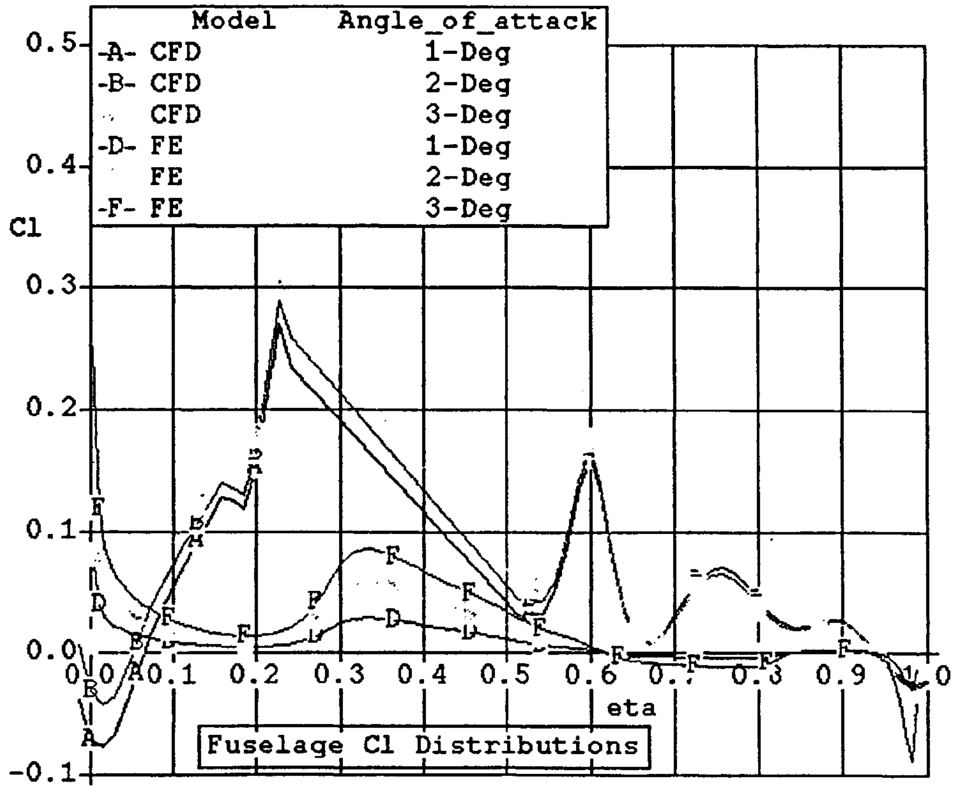


Figure 8.9 : Fuselage Cl Distributions

For figure 8.9 the following remarks can be made:

- The CFD-Cl-distribution is not smooth and shows a number of peaks (just after the wing and just before the wing). These peaks are probably due to the Wing-Fuselage fairing influences. The straight line between eta 0.23 and 0.52 is due to the fact that there are no results for these sections (see 8.1.2) and the graphical interface just draws a line between these points.
- The FE-Cl-distribution is smooth until the end of the fuselage, where the method has difficulties with the way the fuselage is modelled. The peak at 35 % length is the influence of the wing carry over lift.
- The differences between the values of the CFD and FE-models are significant. The FE-model generates less lift. This is partly due to the fact that the FE-wing does not generate the same lift as the CFD-wing, so there is a lot less fuselage lift (due to winglift) on the FE-model.
- The CFD model shows no positive peak at nose. This can be due to the shape of the cockpit which is modelled in the CFD model.
- The FE model shows a large Cl peak at the nose. The FE-fuselage model is considered to be a perfectly shaped "cigar" and the cockpit cannot be modelled.
- Both models show similar incremental differences between the different AOA's.

8.3.2 Conclusions Initial Results

The results from the initial comparison (paragraph 8.3.1) show that the FE-model does not generate the expected shape of Cl distributions. It also shows that the generated total lift is a lot less than with the CFD-model. A number of causes are responsible for the differences. The main cause is the absence of twist and camber in the aerodynamic model of the wing of the FE-model. Twist gives the root sections of the CFD model a higher local AOA so they provide more lift than the root sections of the FE model. The effects of camber are that the total lift is increased at the same AOA.
The FE -model shows a similar response to increases in angle of attack as the CFD-model (see table 8.1). The DCl/Dalpha characteristics of the FE-model are good (for the investigated AOAs). The FE-model is further adjusted in the next paragraph an attempt to better match the CFD-results.

8.4 Further FE-model Adjustments

The initial results show that the FE-model does not generate the expected shape Cl-distributions and total lift. The absence of twist and camber in the aerodynamic model of the wing can be the cause of the differences. The FE-model is adjusted (again) to incorporate both wing twist and camber to investigate whether these changes to the FE-model give the expected results.

8.4.1 MSC/NASTRAN Methods for Twist/Camber

The aerodynamic model of the FE-model has to be adjusted to incorporate wing twist and camber. As it is not possible to give each individual aerodynamic sub panel its own incidence angle by changing the attitude of the sub panel, another method is used by MSC/NASTRAN. The twist and camber is modelled in the DLM as an initial downwash distribution on the aerodynamic sub panels. With a DMI -card (Direct Matrix Input) [Ref.2] this distribution can be modelled in MSC/NASTRAN. This card builds a matrix of user specified dimensions inwhich the figures mentioned on the DMI-card are the figures of the matrix. The aerodynamic theory then uses this matrix to correct the flowfield over the aerodynamic sub panels. The initial downwash is expressed as an angle (in radians). This means that twist angles can easily be put in without conversions needed.
This method can also be used to account for differences between test data and theory, and experimental pressure data at some reference condition. The aerodynamic module can use all these matrices for correcting the calculations, and thus generating better matching results.
The modelling of the twist and camber is discussed in the sub paragraphs below.

8.4.2 Twist Model

The CFD model has a twist at the root of 3.8° and -2.25° at the tip. For ease of modelling the twist of the wing is considered to be linear from root to tip. This means that each wing strip (which has the width of one aerodynamic sub panel) will get the initial down wash according to its span position. The number of aerodynamic panels on the wing dictates the number of entries needed on the DMI card. In this case there are 26 spanwise wing strips, and the panels on each strip can be given the same initial downwash using the TRHU entry. The DMI entry cards are shown in Appendix-L.

The first DMI card is the reference card. It determines the matrix form and dimensions needed. In this case the matrix is rectangular and has 523 rows and tree columns. The rows represent the aerodynamic panels (334 wing panels, 140 body elements on the fuselage, and 49 interference elements). Each of these panels is assigned a row position in the matrix, starting with the lowest panel number first. In this case, the wing has the lowest aerodynamic panel number and the first 334 elements in the matrix refer to the wing. If the a panel number is not mentioned it is presumed to be zero. The three other DMI cards each represent a column in the matrix and the rows in the column are filled with the numbers on the DMI card.

8.4.3 Twist/Camber Model

For proof of concept the camber of the wing is assumed to be 4% at 50% chord and constant over the wing span. The chord of the wing is divided into a number of aerodynamic subpanels. Each subpanel has to be set at different angel to simulate the camber. With the assumptions of 4% camber at 50% chord a division of camber angles over chord length can be made. In figure 8.10 the wing cross-section is shown with the camber line in it. Each sub panel gets its own incidence angel due to camber. One is reminded that the sub panels do not get a physical angle of incidence, but the DLM uses the initial downwash as an incidence angle under which the airflow "hits" the subpanel.

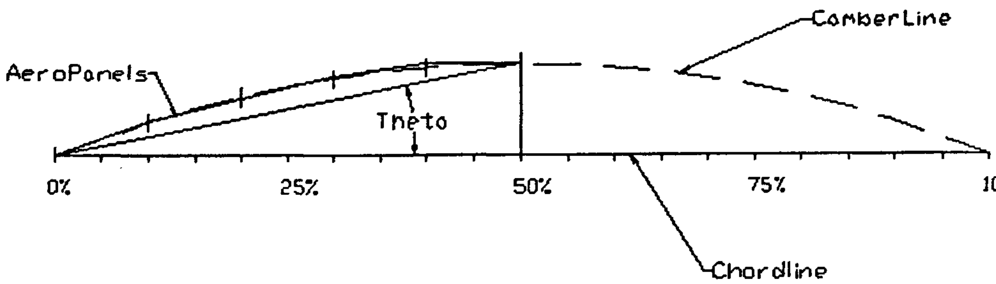


***Figure 8.10** : Chordwise Camber Determination*

From figure 8.10 the following can be derived. 4% camber means that the angle between the point where camber reaches its maximum and the leading edge is approximately 4°. assuming a camber line, one can say that the leading edge panel has approximately twice this angle. According to the figure the different chordwise positions then have the following division:

chord	incidence angle
0 %	-2 theta
25 %	- theta
50 %	0
75 %	theta
100 %	2 theta

Table 8.2 : *Camber angles*

Using table 8.2, a simple spreadsheet program is used to calculate each sub panel initial downwash now including the linear wing twist and the camber angle. These results are then put into the MSC/NASTRAN Bulk Data Deck. At this stage, this input has to be done by hand. This is quit a lot of work, as there are over 330 aerodynamic panels on the wing, each with its own incidence angle. The camber angle can thus not be simply changed at this moment in the project. This is a subject for future development of the MMG. The DMI-cards needed for twist and camber are shown in Appendix-M.

8.4.4 Result from adjustments

The effects of the introduced twist and camber on the FE-generated lift distributions is shown in figure 8.11. Only the results for the 2° & 3° AOA cases are shown. In table 8.3 these cases are compared to determine the dCl/dα figure.

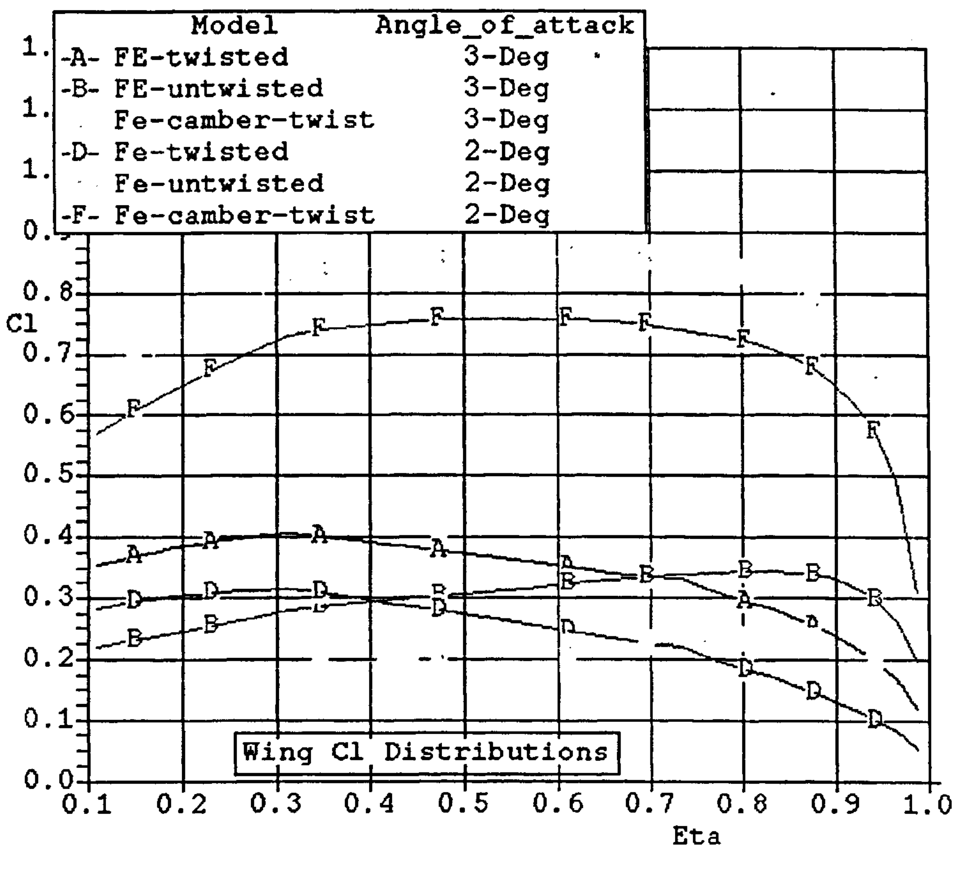


Figure 8.11 : Effects of Twist and Camber

For figure 8.11 the following remarks are made:

- The twisted FE-model (line A and D) show a higher lift at the root, and a lower lift at the tip compared to the non twisted model (line B and E). This was expected from the positive twist at the root, increasing the local AOA, and thus increasing local lift.
- The total lift of the twisted FE-model (line A and D) is still similar to the non Twisted FE-model (Line B and E)
- The FE-model with twist & camber (line C and F) show a much higher overall lift compared to the twisted and untwisted FE-models. The shape of the curve is also considerably different.

It seems that introducing twist and camber results in a much higher overall lift, giving the C distribution another shape. The effects on the dCl/dα figures is shown in table 8.3.

	FE-Twist-Camber				FE-No Twist	
Eta	Cl	Cl	ΔCl	dCl/dα	dCl/dα	Δ%
	2 deg	3 deg				dCl/dα
0.1	0.570	0.642	0.054	3.09	4.13	33.65
0.2	0.641	0.719	0.078	4.47	4.53	1.34
0.3	0.731	0.822	0.091	5.21	5.27	1.15
0.4	0.751	0.848	0.097	5.56	5.62	1.08
0.5	0.758	0.859	0.101	5.79	5.79	0.0
0.6	0.756	0.861	0.105	6.02	6.07	0.83
0.7	0.748	0.857	0.109	6.25	6.24	-0.16
0.8	0.721	0.833	0.112	6.41	6.41	0.0
0.9	0.619	0.723	0.104	5.95	5.96	0.17
1.0	0.301	0.373	0.072	4.13	3.72	-9.93

Table 8.3 : *dCl/dα FE-model with Twist/Camber*

The twist and camber have almost no effect on the dCl/dα characteristics of the FE-model. The new results will be compared to the CFD-results in the next paragraph.

8.5 Results Comparison

The adjusted FE-models described in Chapter 8.4 are calculated using the trim conditions described in Chapter 8.2.3. The results from these calculations (FE-model with twist and the FE-model with camber & twist) are compared with the CFD data and the original FE-model.
Figure 8.12 shows the Cl distribution for the CFD model, the FE model with twist, the FE model without twist, and the FE-model with camber & twist. Only the results for the 3° AOA case are shown.

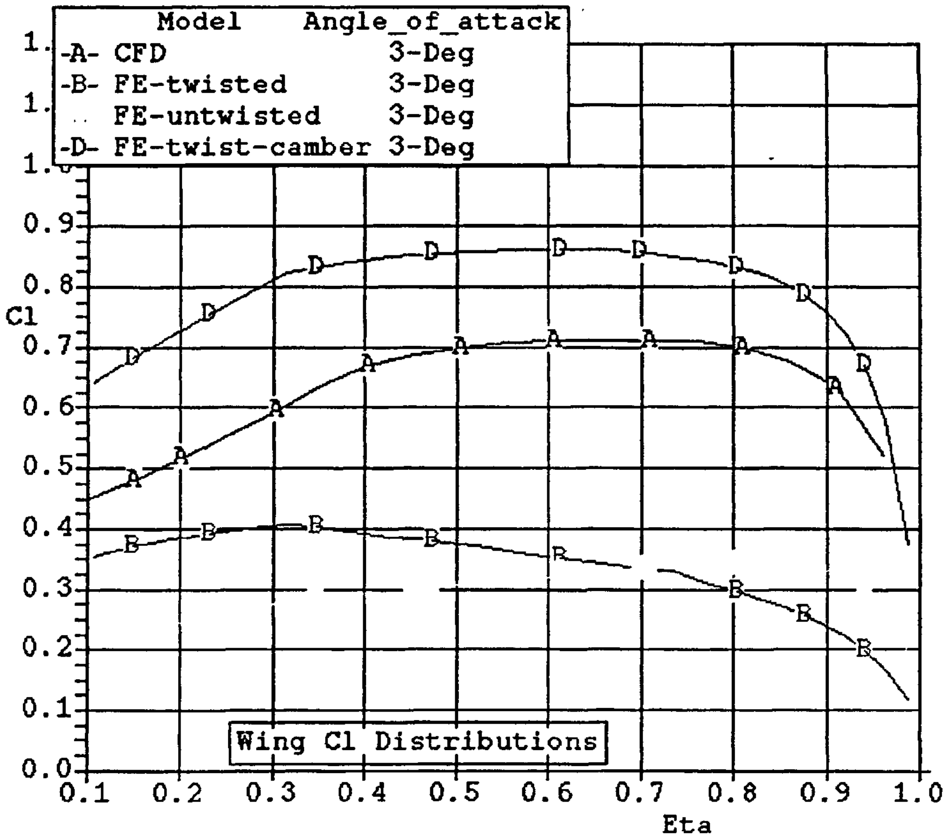


Figure 8.12 : Wing Cl Distributions Including Twist & Camber FE-model

For figure 8.12 the following remarks are made:

- The twisted FE-model (line B) shows a higher lift at the root, and a lower lift at the tip compared to the non twisted model (line C). This was expected from the positive twist at the root, increasing the local AOA, and thus increasing local lift.
- The total lift of the twisted FE-model (line B) is still considerably lower than the CFD model (Line A).
- Cl max. for the twisted model (line B) is now at 33 % span, for the CFD-model (line A) between 60 % and 80 % span
- The FE-model with twist & camber (line D) shows a much higher overall lift compared to the FE-model with twist only (line B). It is also higher than the CFD-Model (line A)

In figure 8.13 the Clc/mac distribution of the CFD-model, twisted FE & untwisted FE-models, and the FE-model with camber & twist are shown. Also only for the 3 deg AOA case.

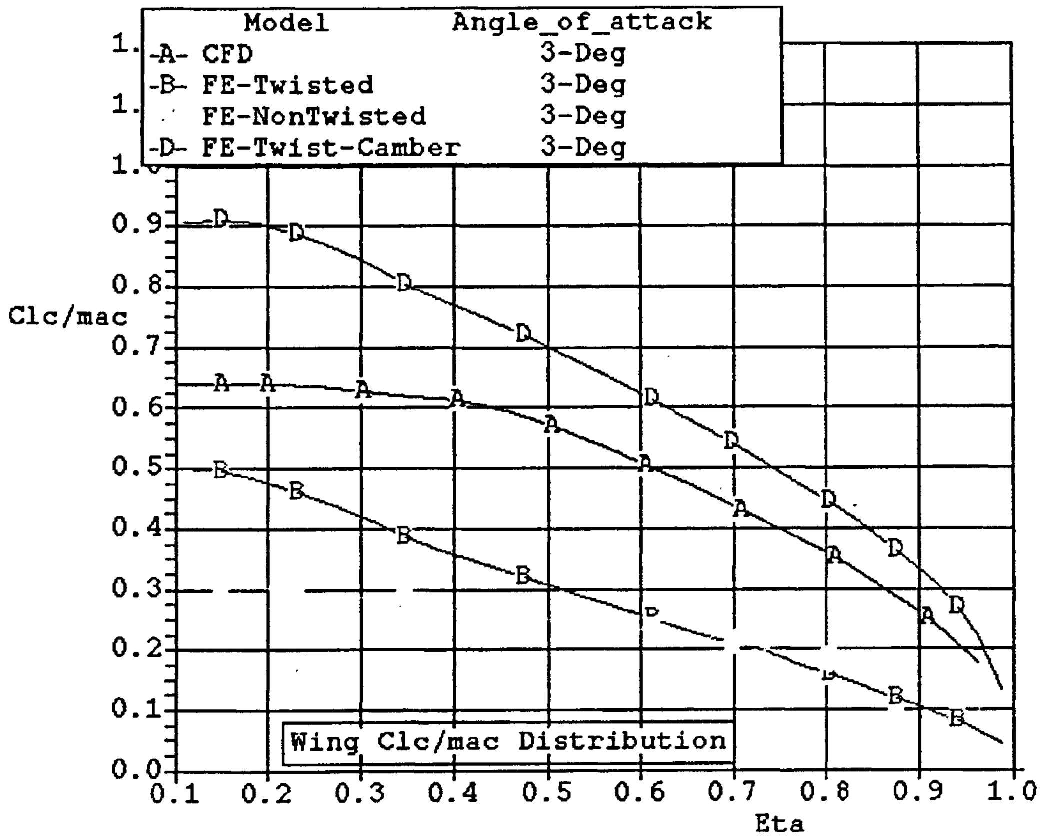


Figure 8.13 : Wing Clc/mac Distributions Included Twisted FE-model

For figure 8.13 the following remarks are made :

- The twisted FE-model (line B) shows a lot steeper line than the non twisted FE-model (line C) and the angle of the line is now comparable with the CFD model (line A).
- The shape of the curve of the twisted FE-model (line B) is not similar with the CFD model (line A) between 10 % and 40 % span.
- The twisted FE-model does not generate enough lift.
- The FE-model with camber & twist (line D) is considerably higher situated than the CFD-model (line A). The overall shape of line D compares more favourable with the CFD line (A) than the twisted FE line (B)
- The CFD-model has a more or less horizontal section between 10% and 40 % span. The FE-model with camber (line D) is only approximately horizontal from 10% to 20% span.

Figure 8.14 shows the Cl distribution of the fuselage for the CFD, twisted and non twisted FE models.

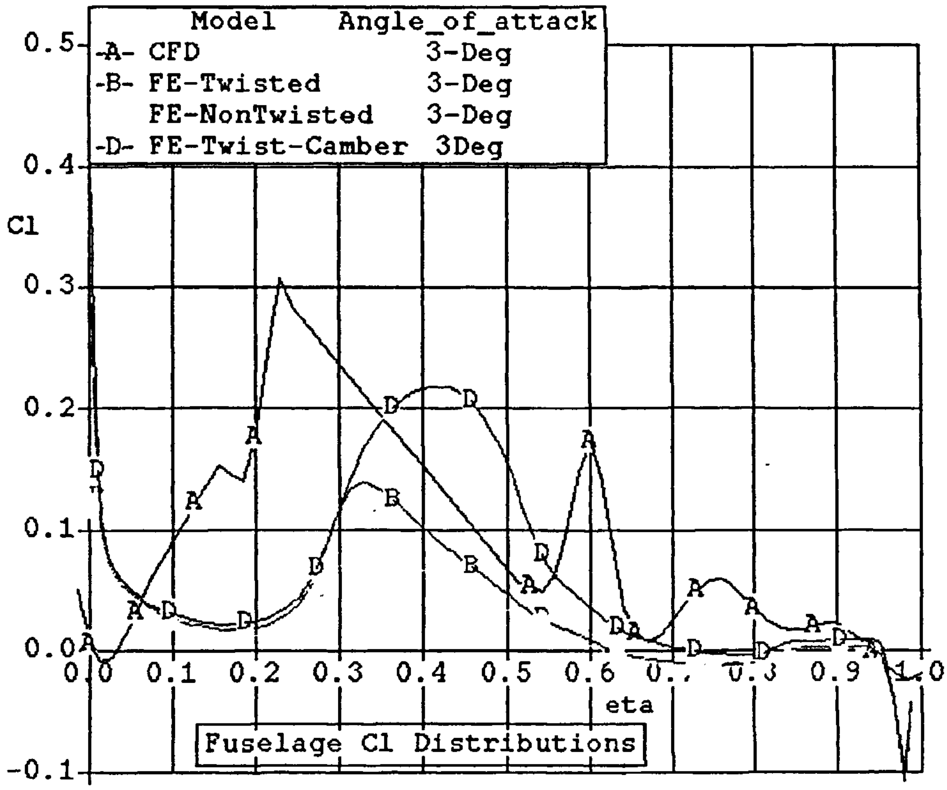


Figure 8.14 : Fuselage Cl Distributions Included Twisted FE-Model

For figure 8.14 the remarks are:

- The twisted FE-model (line B) generates considerably more lift on the fuselage than the non-twisted model (line A). This is due to the higher root lift on the twisted model.
- The twisted model with camber (line D) produces considerably more lift than the twisted model (line B). The centre of lift also is further back, due to the camber.
- The total fuselage lift of the camber & twist model still seems to be less than the CFD-model.

8.6 Conclusions

- The FE-model is suitable for Static Aeroelastic Calculations if a number of adjustments are made to the model suggested in paragraph 8.2.1 and the adjustments for twist and camber from paragraph 8.4.3. First of all the wing twist has to be modelled as initial downwashes on the aerodynamic panels. Secondly the camber of the wing has to be modelled. The lift distributions generated with twist and camber model are of comparable shape as the CFD lift distributions. This means that the introduction of twist and camber is an absolute necessity before any significant results from static aeroelastic calculations can be derived. The model tested with camber and twist show a too high overall lift at this moment. It is necessary that the twist and camber are modelled as accurate as possible as the influence of both on the distributions is substantial. For this detailed information about wing sections is necessary. At this stage in the project it is not possible to get this information on time.
- The fuselage Cl distributions of the CFD model need further investigation. As the post processor cannot integrate over the fuselage section where the wing is attached, there are simply no results available for that section. For a good comparison between the FE and CFD results this will have to be solved.
- Both the CFD and the FE model show similar increases in lift over the wing if the angle of attack is increased. From table 8.1 it can be concluded that both models have similar $dCl/d\alpha$ characteristics.
- Once the FE-model generates similar results to the CFD model, the results should be compared with a non rigid aircraft including the engines, to see the effects on the overall lift distributions.

CHAPTER 9 : CONCLUSIONS AND RECOMMENDATIONS

9.1 MDO-Project

The MDO-project software package gives the user the opportunity to run complex analysis on aircraft models in an early stage of the design. The software requires specific data (detailed fuselage data, structural data, payloads, fuel loads, weight estimations) for the programs to work. This makes it difficult to analyse multiple aircraft configurations in a short period.

The software as such can not be used as an automatic design program which designs the aircraft, but it is a program to help the designers to see what happens to the overall aircraft if certain parameters (like wing area) are changed. The program also allows optimisations to different parameters (weight, cost) and is able to feed the optimisation information back into the design.

For this M.Sc. thesis the MDO-software is used to generate FE-models with the MMG or Multi Model Generator.These models are then used for MSC/NASTRAN static aeroelastic analyses. Results from these analyses can not be used within the original of the MDO-software as this functionality was not built in.

9.2 AEG-Module

The FE-model (representing half an aircraft without fuselage aerodynamic model), as generated by the MDO-software, can not be used for static aeroelastic analysis with MSC/NASTRAN, without adding on a 'Case Control' and 'Single Point Constraint' data deck. These decks were specially made for this purpose.

The original AEG-module, which creates the aerodynamic panels of the aerodynamic model, does not work correctly. The AEG-module has to be changed and amended as outlined in paragraph 5.5. These modifications correct the way in which the aerodynamic panels are connected to the wingstructure. The FE-model, generated using the modified AEG module, shows the expected responses (see 5.5.3) to changes made to the configuration of the aircraft. The FE-model can then be used for the initial static aeroelastic analysis.

The modified AEG-module should be made available to the partners in the MDO-Project. Especially because the aerodynamic model is also used with other analyses (i.e. Flutter analyses). It is recommended that the influence of the modifications to the AEG-module on the results of these analyses is investigated.

9.3 NASTPP-Utility

The results from the MSC/NASTRAN static aeroelastic analysis (pressure distributions, lift forces and moments on the aerodynamic panels) can not be written into or read from the original TDMB database. The NASTPP utility and TDMB has to be supplemented with extra routines (see chapter 6) before the results can be shown using the TDMB graphical module. The supplements needed are substantial and took quite some time to develop.

The modified NASTPP utility is only capable of handling the results generated by earlier versions of MSC/NASTRAN. It is recommended that the NASTPP utility is developed further so that the latest MSC/NASTRAN versions can be used for the static aeroelastic analyses. The modified NASTPP utility should be made available to the partners in the MDO-Project.

9.4 Fuselage Aerodynamic Model

The modelling of the aerodynamic model of the fuselage is complicated. There are a lot of influences that affect the working of the FE-model and the results of the calculations. The MSC/NASTRAN modelling rules have to be followed carefully for the FE-model to work. A lot of modelling problems originate from the way the aerodynamic model of the wing is attached to the aerodynamic model of the fuselage. For this project, the aerodynamic model of the wing is attached at mid fuselage position (as this model works!). It is recommended that other configurations aer investigated further so that the influence of the attachment position of the wing to the fuselage can be defined..

A number of convergence studies quantify the influences of wing panel density, fuselage panel density, and fuselage interference panel density.
The convergence studies proof that the fuselage model is very sensitive to changes in the number of interference panels used. Although the modelling rules are followed, a lot of trial and error modelling is necessary to determine the necessary number of interference panels. Also if the number of aerodynamic panels in the model is changed, the user might find it necessary to have to change the number of interference panels again.
The number of body panels on the fuselage is of almost no influence on the results (if at least a reasonable number of panels is used).
The number of wing panels is of marginal influence on the fuselage lift generated, and of almost no influence on the wing lift generated. It is found that if the length of the body panels of the fuselage is more or less similar to the chord length of the root panels of the wing, the FE-model will generate feasible results.

The convergence studies determined the number of body panel, interference panels and wing panels necessary for the FE-model to generate feasible results. The final-half-FE-model incorporates all these panels and the results show that static-aeroelastic simulations using this final-half-FE-model are solved reliably and consistently. This model is used as basis for the static aeroelastic analysis with the reference aircraft.

The original version of the MMG does not generate the fuselage aerodynamic panels, and for this project the panels are created by manual input to proof the concept. It is recommended that the MMG is developed further so that the fuselage aerodynamic model can be generated automatically to user specifications by the MMG. It is expected that this future development will take considerable time as the fuselage modelling not straight forward, and a lot of modelling rules have to be taken into account.

9.5 Results Comparison

The results of the MSC/NASTRAN static aeroelastic calculations using the final-half-FE-model are validated using CFD results as reference. For this comparison, the final-half-FE-model needs to be reconfigured, as the CFD-results are based on a different aircraft configuration. The final-half-FE-model proofs to respond as expected to the changes made in the configuration. This proofs that the final-half-FE-model is suitable to use in the comparison with the CFD results.

The initial comparisons of the lift distributions over the wing span show that the FE-model does not generate enough lift, eventhough the CFD and FE-model "fly" under the same flight conditions. The shape of the lift distributions generated with the FE-model also do not match the CFD-results. The cause of this problem is the absence of twist and camber in the FE-aerodynamic model. Adding twist to the FE-model results in a change in shape of the lift distribution. The overall lift generated is almost not affected. The addition of camber to the FE-twist-model has a big influence on the overall generated lift and shape of the lift distributions over the wing span.

The shape of the lift distributions generated with the FE-twist&camber-model are comparable with the shape of the lift distributions generated by CFD. The overall lift generated by the FE-twist&camber-model is too high. This is due to the fact that, for proof of concept reasons, the twist is modelled linear from root to tip and the camber is modelled at a set percentage, nor varying in spanwise direction. This however does not match the twist and camber distributions of the CFD-model which are non linear. To be able to model the twist and camber accurately, detailed information about wing sections used in the CFD model is necessary. Due to time constraints this analysis can not be conducted during this project. It is recommended that the twist and camber distributions are modelled as accurately as possible according to the CFD-model. The results so far indicate that in that case the shape of the lift distributions and the overall generated lift will probably lie very near to the CFD results.

Both the CFD and the FE-model show similar increases in lift over the wing if the angle of attack is increased. From table 8.1 it can be concluded that both models have similar $dCl/d\alpha$ characteristics. Twist and camber seem to have no significant effect on $dCl/d\alpha$ characteristics of the FE-model).

It is recommended that the fuselage Cl-distributions of the CFD-model are investigated further. It is not possible for the post processor to integrate over the fuselage section where the wing is attached. This means that there are simply no results available for that section. For a good comparison between the FE and CFD-results (for the fuselage lift distributions) this will have to be solved. From the results generated by both the CFD-model and the FE-model, a solid conclusion can not be drawn.

It is recommended that at the moment the FE-model generates similar results to the CFD model, the results should be compared with a non rigid FE-model (with and without engines) to investigate the effects on the overall lift distributions.

9.6 Overall

The FE-model generated by the modified MDO-software can be used for MSC/NASTRAN static aeroelastic calculations if a number of adjustments and supplements are made to the model.

The modified MDO-software includes the adjusted module for generating the aerodynamic panels of the FE-model. This is an absolute necessity for the static aeroelastic analysis to work correctly.

The modified MDO-software can read the MSC/NASTRAN output file containing the results from static aeroelastic calculations. These results are processed by the modified software and written to the common data base. The data base can be used to view the results in a graphical format and compare the results with other data.

The FE-model generated by the modified MDO-software including adjustments has the potential to be used as a model standard for generating representative aerodynamic loads on wings and fuselages.

REFERENCES

1. Allwright, S.E. "*Technical Data Management For Collaborative Multi-Discipline Optimisation*", American Institute of Aeronautics an Astronautics Conference Paper, AIAA-96-416-CP.

2. Various Authors,"*MDO-Programs*", HTML-documents coming with the MDO-software package, Version March 1998.

3. Rondeau, E.A. Peck, A.F. Williams, S.E.Allwright, L.D. Shields, " *Generative Design and Optimisation of the Primary Structure for a Commercial Transport Airplane* ", American Institute of Aeronautics and Astronautics Conference Paper, AIAA-96-3135-CP.

4. MSC/NASTRAN is marketed by The MacNeal-Schwendler Corporation, 815 Colorado Boulevard, Los Angeles, California,USA.

5. Rollema, G.J.,. "*Multi-Disciplinary Design Optimisation, lr 467 Voorbereidend Afstuderen*", M.Sc. thesis project feasibility study, Delft University of Technology, The Netherlands, 1998.

6. Rodden, W.P., Johnson, E.H., " *MSC/Nastran Aeroelastic Analysis User's Guide V68*", The MacNeal-Schwendler Corporation, Los Angeles, USA, 1994.

7. Albano, E., Rodden, W.P., "*A Doublet-Lattice Method for Calculating Lift Distributions on Oscillating Surfaces in Subsonic Flows*",AIAA Journal, Volume 7, pp.279-285, p2192, 1969.

8. Giesling, J.P., Kalman, T.P., Rodden, W.P., "*Subsonic Unsteady Aerodynamics of General Configurations, Part I, Vol. I -Direct Application of the Nonplanar Doublet-Lattice Method*", Air Force Flight Dynamic Laboratory Report No. AFFDL-TR-71-5, Part I, Vol I, 1971.

9. Rodden, W.P., Giesling, J.P., Kalman, T.P.,"*Refinement of the Nonplanar Aspects of the Subsonic Doublet-Lattice Lifting Surface Method*", J.Aircraft, Volume 9, pp.69-73, 1972.

10. Giesling, J.P., Kalman, T.P., Rodden, W.P., "*Subsonic Unsteady Aerodynamics of General Configurations, Part II, Vol. I -Application of the Doublet-Lattice Method and the Method of Images to Lifting-Surface/Body Interference*", Air Force Flight Dynamic Laboratory Report No. AFFDL-TR-71-5, Part II, Vol I, 1972.

11. Reymond, M., Miller, M & Editors " *MSC/Nastran Quick Reference Guide V68*", The MacNeal-Schwendler Corporation, Los Angeles, USA, 1994.

LIST OF APPENDICES

APPENDIX-A : GRADUATION PROJECT PROPOSAL

Ir./MSc Thesis
Delft University of Technology, Faculty of Aerospace Engineering.

Student:	Mr. Gert-Jan Rollema (702193)
Department:	C1, Structural Design
At:	Delft University of Technology, Faculty of Aerospace Engineering, The Netherlands & British Aerospace Woodford, Cheshire, United Kingdom.
Supervisors:	Prof. dr. J. Arbocz (Delft University of Technology) Mr. F. van Dalen (Delft University of Technology) Mr. S. Allwright (British Aerospace) Mr. D. Rondeau (British Aerospace)
Period:	May 1998 - July 1998 : Delft University of Technology August 1998 - November 1998 : British Aerospace Woodford

Abbreviations:

MDO	Multidicsiplinary Design Optimisation (Brite-Euram project)
MMG	Multi Model Generator (part of the MDO project)
TDMB	Technical Data Modeler and Browser

PROJECT:
The project will be a co-operation between Delft University of Technology and British Aerospace. The student will conduct the first part of the project at the University and second part at British Aerospace Woodford.

SUBJECT:
Comparison of lift distributions generated with MSC/Nastran with results from the BAe Flutter and Loads suite and CFD data. Modifying the Nastran model based on the comparison results.

PROJECT OBJECTIVES:
1. To obtain an improved Nastran model standard for lift distributions over a wing generated by the MDO programs.
2. To compare the initial FE wing model with the optimised FE wing model.
3. To obtain representative loads generated by Nastran for different wing planforms.

PROJECT DETAILS:
1. Nastran will be used to generate lift distributions over the FE wing model generated by the MDO programs (this is the non-optimised FE wing model from the MMG). The following manoeuvres will be simulated : 2.5g pull up, -1g push down and if possible a 10 deg/sec roll (instigated by aileron only). The model will also be flown through continuos turbulence gust cases. Flexible as well as rigid models will be examined. An aerodynamic panel/body representation of the fuselage will be put into the Nastran model to enhance "body-lift" simulation. The results of the Nastran simulations will be put back into the TDMB database. This part of the project will be conducted at the University, the next tasks will be done at BAe Woodford, starting from the 3rd August 1998.
2. The aerodynamic loads generated with the Nastran model will be compared with the data from the BAe Flutter and Loads suite and with CFD data supplied by BAe. The results will be validated against each other and the Nastran model will be improved to achieve a better match. Based on the obtained results recommendations will be made to improve the MMG.
3. The Nastran optimisation routine will be used with the loads generated in task 1 and at the end with the loads generated with task 2. The results from the initial model and the optimised model will be compared. Task 2 & 3 will be performed simultaneously.

REMARKS
The project details mentioned above were agreed upon during a meeting between Mr. S. Allwright, Mr. D. Rondeau, Mr. F. van Dalen and the student Mr. G.J. Rollema on the 2nd of July 1998 at BAe Filton.

Gert-Jan Rollema.

APPENDIX-B : CASE CONTROL DECK FOR FE-MODEL

```
ID MDO_STATIC_AERO
$------------------NASTRAN DECK FOR MDO_STATIC_AERO_ANALYSIS---
$------------------by G.J.Rollema----------------------------
$------------------FILL IN DETAILS BELOW AS REFERENCE----------
$------------------ADJUST TRIM CARD ACCORDINGLY---------------
$
$ AC-mass : 535000 ? kg
$ Payload : 89600 ? kg
$ Fuelload : 192000 ?  kg
$ M = 0.8 ? , q = 11845 ? N/m2
$
TIME 500
SOL 144
CEND
$
TITLE=MDO_STATIC_AERO_?????
ECHO=SORT
SPC=15
DISP=ALL
STRESS=ALL
APRES=ALL
AEROF=ALL
FORCE=ALL
SUBCASE 1
LABEL = ST_AERO_PULL_UP
TRIM = 10
SUBCASE 2
LABEL = ST_AERO_PUSH_DOWN
TRIM = 20
$----------------------------PLOTTER COMMANDS------------------
$
$in order to see the aero1 elements on a plot this deck
$has to run on nastran version 70 because version 70.5
$does NOT plot the aero1 elements
$
OUTPUT(PLOT)
$
paper size 29.7 by 20.0
SET 1= ALL
SET 2= AERO1,BEAM
$---PLOT VIEW 1(FRONT)---------
view 180.,0.,0.
PTITLE=PLOT VIEW 1 (FRONT)
FIND SCALE, ORIGIN 1, SET 1
PLOT STATIC DEFORMATION 0, ORIGIN 1, SET 1
$FIND SCALE, ORIGIN 1, SET 2
$PLOT SET 2
$---PLOT VIEW 2 (SIDE)---------
view 180., 15., 20.
PTITLE=PLOT VIEW 2 (SIDE)
$FIND SCALE, ORIGIN 1, SET 1
$PLOT STATIC DEFORMATION 0, ORIGIN 1, SET 1
FIND SCALE, ORIGIN 1, SET 2
PLOT SET 2
$---PLOT VIEW 3 (TOP)---------
view 180.,90.,90.
PTITLE=PLOT VIEW 3 (TOP)
$FIND SCALE, ORIGIN 1, SET 1
$PLOT STATIC DEFORMATION 0,ORIGIN 1,SET 2
FIND SCALE, ORIGIN 1, SET 2
PLOT SET 2
$-------------------------------------------END PLOT COMMANDS----
BEGIN BULK
PARAM,POST,-1
PARAM,PATVER,3.
PARAM,PLOT,-1
PARAM,BAILOUT,-1
PARAM,AUNITS,0.10194
PARAM,GRDPNT,100099
```

APPENDIX-C: SINGLE POINT CONSTRAINT DECK FOR FE-MODEL

```
$----------------elevator------------------------------------------
$-----if number of panels is increased, these cards need to be changed
$----------------remove $ with a full model-------------------------
$----------------and make the other AERSURF card inactive------------
$
AESURF  400     ELEV    450     400
$AESURF  400     ELEV    450     400     50450  50400
$
AELIST  400     5001000 THRU    5001015 5002000 THRU    5002063 5003000
+       5003001 THRU    5003007
$AELIST  50400   5001500 THRU    5001515 5002500 THRU    5002563 5003500
$+       5003501 THRU    5003507
$       CID     RID     A1      A2      A3      B1      B2      B3
CORD2R  450             71.4097 14.2302 2.0951  71.4097 13.8805 7.0951
        76.4097 10.9812 2.0951
$CORD2R 50450           71.4097 -14.23022.0951  71.4097 -13.88057.0951
$       76.4097 -10.98122.0951
$
$----------------end elevator data-----------------------------------
$
$----------------DATA for SOL 144------------------------------------
$----------------added by G.J.Rollema--------------------------------
$----------------spc & RBE cards-------------------------------------
$
SPC1          15     246  101101    THRU  101299
SPC1          15     246  900000    THRU  912000
SPC1          15     246  914000    THRU  999999
SPC1          15     246  800000    THRU  899999
SPC1          15     246  701000
SPC1          15    1246  100099
$
$----------------TRIM card STAT_AERO_PULL_UP-------------------------
$
TRIM,10,0.8,11890.0,URDD3,2.5,AILER0,0.0,,+TR1
+TR1,AILER1,0.0,PITCH,0.0
$
$----------------TRIM card STAT_AERO_PUSH_DOWN-----------------------
$
TRIM,20,0.8,11890.0,URDD3,-1.0,AILER0,0.0,,+TR20
+TR20,AILER1,0.0,PITCH,0.0
$
$----------GRID point (approx CG position) for SOL 144----------------
$
GRID,100099,,32.40,0.0000,-1.40
$
$----------CG gridpoint connection with wing--------------------------
$
RBE2,9100001,100099,135,101101
=,*1,=,=,*1,==
=34
RBE2,9100037,100099,135,101199
RBE2,9100050,100099,135,101201
=,*1,=,=,*1,==
=34
RBE2,9100087,100099,135,101299
$
$----------------SUPORT card-----------------------------------------
$
SUPORT,100099,35
$
$
$----------------end add-on------------------------------------------
$
$
ENDDATA
```

APPENDIX-D : ORIGINAL AERODYNAMIC PANEL DATA DECK

```
$*****************************************************************************
$
$    A E R O E L A S T I C I T Y    D A T A
$
$*****************************************************************************
$    Wing              - Aeroelasticity Bulk Data
$
$             CID     PID      A1      A2      A3      B1      B2      B3
  CORD2R     1099          26.915   3.208  -3.230  26.915   3.133  -2.370
$              C1      C2      C3
           27.659   2.698  -2.408
$
$             EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
  CAERO1 1001000       1               1       8       0       0    1111 PAWin00
$              X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin00   20.930    .000  -3.262  16.500  20.930   3.510  -3.262  16.500
$             SID       1       2       3       4       5       6       7     etc
    SET1      101  101101  101199  102101  102199  103101  103199  104101 S1Win00
+S1Win00   104199  105101  105199  106101  106199
$             EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2     1001 1001000 1001000 1001007     101   1.000   1.000    1099 SPWin00
$            DTHX    DTHY
+SPWin00   -1.000  -1.000
$
$             EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
  CAERO1 1002000       1               3       8       0       0    1111 PAWin01
$              X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin01   20.930   3.510  -3.262  16.500  28.053  13.493  -1.795  10.075
$             SID       1       2       3       4       5       6       7     etc
    SET1      102  106101  106199  107101  107199  108101  108199  109101 S1Win10
+S1Win10   109199  110101  110199  111101  111199  112101  112199  113101 S1Win11
+S1Win11   113199  114101  114199  115101  115199  116101  116199  117101 S1Win12
+S1Win12   117199  118101  118199  119101  119199  120101  120199  121101 S1Win13
+S1Win13   121199  122101  122199  123101  123199  124101  124199  125101 S1Win14
+S1Win14   125199  126101  126199  127101  127199
$             EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2     1002 1002000 1002000 1002023     102   1.000   1.000    1099 SPWin01
$            DTHX    DTHY
+SPWin01   -1.000  -1.000
$
$             EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
  CAERO1 1003000       1               1      14       0       0    1111 PAWin02
$              X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin02   28.053  13.493  -1.795  10.075  28.579  14.230  -1.748   9.886
$             SID       1       2       3       4       5       6       7     etc
    SET1      103  127101  127199  128101  128199  129101  129199  130101 S1Win20
+S1Win20   130199  131101  131199  132101  132199  133101  133199  134101 S1Win21
+S1Win21   134199  135101  135199  136101  136199  137101  137199  138101 S1Win22
+S1Win22   138199  139101  139199  140101  140199  141101  141199  142101 S1Win23
+S1Win23   142199  143101  143199  144101  144199  145101  145199
$             EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2     1003 1003000 1003000 1003013     103   1.000   1.000    1099 SPWin02
$            DTHX    DTHY
+SPWin02   -1.000  -1.000
$
$             EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
  CAERO1 1004000       1               5       8       0       0    1111 PAWin03
$              X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin03   28.579  14.230  -1.748   9.886  37.131  26.215   -.983   6.812
$             SID       1       2       3       4       5       6       7     etc
    SET1      104  145101  145199  146101  146199  147101  147199  148101 S1Win30
+S1Win30   148199  149101  149199  150101  150199  151101  151199  152101 S1Win31
+S1Win31   152199  153101  153199  154101  154199  155101  155199  156101 S1Win32
+S1Win32   156199  157101  157199  158101  158199  159101  159199  160101 S1Win33
+S1Win33   160199  161101  161199  162101  162199  163101  163199  164101 S1Win34
+S1Win34   164199  165101  165199  166101  166199  167101  167199  168101 S1Win35
+S1Win35   168199  169101  169199
$             EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2     1004 1004000 1004000 1004039     104   1.000   1.000    1099 SPWin03
$            DTHX    DTHY
+SPWin03   -1.000  -1.000
```

```
$
$           EID       PID        CP     NSPAN    NCHORD     LSPAN    LCHORD      IGID
  CAERO1 1005000         1                  4         8         0         0      1111 PAWin04
$            X1        Y1        Z1       X12        X4        Y4        Z4       X43
+PAWin04  37.131    26.215     -.983     5.109    40.982    31.613     -.639     4.070
$           EID     CAERO       ID1       ID2      SETG        DZ      DTOR       CID
 SPLINE2    1005   1005000   1005000   1005031       104     1.000     1.000      1099 SPWin04
$          DTHX      DTHY
+SPWin04  -1.000    -1.000
$
$           EID       PID        CP     NSPAN    NCHORD     LSPAN    LCHORD      IGID
  CAERO1 1006000         1                  7         9         0         0      1111 PAWin05
$            X1        Y1        Z1       X12        X4        Y4        Z4       X43
+PAWin05  40.982    31.613     -.639     4.070    45.246    37.588     -.258     2.921
$           EID     CAERO       ID1       ID2      SETG        DZ      DTOR       CID
 SPLINE2    1006   1006000   1006000   1006062       104     1.000     1.000      1099 SPWin05
$          DTHX      DTHY
+SPWin05  -1.000    -1.000
$
$           EID       PID        CP     NSPAN    NCHORD     LSPAN    LCHORD      IGID
  CAERO1 1007000         1                  1         8         0         0      1111 PAWin06
$            X1        Y1        Z1       X12        X4        Y4        Z4       X43
+PAWin06  45.246    37.588     -.258     3.894    45.933    38.552     -.196     3.647
$           EID     CAERO       ID1       ID2      SETG        DZ      DTOR       CID
 SPLINE2    1007   1007000   1007000   1007007       104     1.000     1.000      1099 SPWin06
$          DTHX      DTHY
+SPWin06  -1.000    -1.000
$*******************************************************************************
```

APPENDIX-E : REWRITTEN AERODYNAMIC PANEL DATA

```
$*****************************************************************************
$
$    Wing           - Aeroelasticity Bulk Data
$
$
$           CID     PID      A1      A2      A3      B1      B2      B3
 CORD2R    1099           26.915   3.208  -3.230  26.915   3.133  -2.370
$            C1      C2      C3
         27.659   2.698  -2.408
$
$           EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
 CAERO1 1001000       1               1       8       0       0    1111 PAWin00
$            X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin00 20.930    .000  -3.262  16.500  20.930   3.510  -3.262  16.500
$           SID       1       2       3       4       5       6       7     etc
   SET1     101  101101  101199  102101  102199  103101  103199  104101 S1Win00
+S1Win00 104199  105101  105199  106101  106199  107101  107199
$           EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2   1001 1001000 1001000 1001007     101   1.000   1.000    1099 SPWin00
$          DTHX    DTHY
+SPWin00 -1.000  -1.000
$
$           EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
 CAERO1 1002000       1               4       8       0       0    1111 PAWin01
$            X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin01 20.930   3.510  -3.262  16.500  28.053  13.493  -1.795  10.075
$           SID       1       2       3       4       5       6       7     etc
   SET1     102  107101  107199  108101  108199  109101  109199  110101 S1Win10
+S1Win10 110199  111101  111199  112101  112199  113101  113199  114101 S1Win11
+S1Win11 114199  115101  115199  116101  116199  117101  117199  118101 S1Win12
+S1Win12 118199  119101  119199  120101  120199  121101  121199  122101 S1Win13
+S1Win13 122199  123101  123199  124101  124199  125101  125199  126101 S1Win14
+S1Win14 126199
$           EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2   1002 1002000 1002000 1002031     102   1.000   1.000    1099 SPWin01
$          DTHX    DTHY
+SPWin01 -1.000  -1.000
$
$           EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
 CAERO1 1003000       1               1      17       0       0    1111 PAWin02
$            X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin02 28.053  13.493  -1.795  10.075  28.579  14.230  -1.748   9.886
$           SID       1       2       3       4       5       6       7
   SET1     103  126101  126199  127101  127199
$           EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2   1003 1003000 1003000 1003016     103   1.000   1.000    1099 SPWin02
$          DTHX    DTHY
+SPWin02 -1.000  -1.000
$
$           EID     PID      CP   NSPAN  NCHORD   LSPAN  LCHORD    IGID
 CAERO1 1004000       1               8       8       0       0    1111 PAWin03
$            X1      Y1      Z1     X12      X4      Y4      Z4     X43
+PAWin03 28.579  14.230  -1.748   9.886  37.131  26.215   -.983   6.812
$           SID       1       2       3       4       5       6       7     etc
   SET1     104  127101  127199  128101  128199  129101  129199  130101 S1Win30
+S1Win30 130199  131101  131199  132101  132199  133101  133199  134101 S1Win31
+S1Win31 134199  135101  135199  136101  136199  137101  137199  138101 S1Win32
+S1Win32 138199  139101  139199  140101  140199  141101  141199  142101 S1Win33
+S1Win33 142199  143101  143199  144101  144199  145101  145199  146101 S1Win34
+S1Win34 146199  147101  147199  148101  148199
$           EID   CAERO     ID1     ID2    SETG      DZ    DTOR     CID
 SPLINE2   1004 1004000 1004000 1004063     104   1.000   1.000    1099 SPWin03
$          DTHX    DTHY
+SPWin03 -1.000  -1.000
```

```
$
$            EID      PID       CP    NSPAN   NCHORD    LSPAN   LCHORD     IGID
  CAERO1 1005000        1                 6        8        0        0     1111 PAWin04
$             X1       Y1       Z1      X12       X4       Y4       Z4      X43
+PAWin04  37.131   26.215    -.983    5.109   40.982   31.613    -.639    4.070
$            SID        1        2        3        4        5        6        7     etc
    SET1     105   148101   148199   149101   149199   150101   150199   151101 S1Win40
+S1Win40  151199   152101   152199   153101   153199   154101   154199   155101 S1Win41
+S1Win41  155199   156101   156199   157101   157199
$            EID    CAERO      ID1      ID2     SETG       DZ     DTOR      CID
 SPLINE2    1005  1005000  1005000  1005047      105    1.000    1.000     1099 SPWin04
$           DTHX     DTHY
+SPWin04  -1.000   -1.000
$
$            EID      PID       CP    NSPAN   NCHORD    LSPAN   LCHORD     IGID
  CAERO1 1006000        1                10        8        0        0     1111 PAWin05
$             X1       Y1       Z1      X12       X4       Y4       Z4      X43
+PAWin05  40.982   31.613    -.639    4.070   45.246   37.588    -.258    2.921
$            SID        1        2        3        4        5        6        7     etc
    SET1     106   157101   157199   158101   158199   159101   159199   160101 S1Win50
+S1Win50  160199   161101   161199   162101   162199   163101   163199   164101 S1Win51
+S1Win51  164199   165101   165199   166101   166199   167101   167199
$            EID    CAERO      ID1      ID2     SETG       DZ     DTOR      CID
 SPLINE2    1006  1006000  1006000  1006079      106    1.000    1.000     1099 SPWin05
$           DTHX     DTHY
+SPWin05  -1.000   -1.000
$
$            EID      PID       CP    NSPAN   NCHORD    LSPAN   LCHORD     IGID
  CAERO1 1007000        1                 2        8        0        0     1111 PAWin06
$             X1       Y1       Z1      X12       X4       Y4       Z4      X43
+PAWin06  45.246   37.588    -.258    3.894   45.933   38.552    -.196    3.647
$            SID        1        2        3        4        5        6        7
    SET1     107   167101   167199   168101   168199   169101   169199
$            EID    CAERO      ID1      ID2     SETG       DZ     DTOR      CID
 SPLINE2    1007  1007000  1007000  1007015      107    1.000    1.000     1099 SPWin06
$           DTHX     DTHY
+SPWin06  -1.000   -1.000
$****************************************************************************
$
```

APPENDIX-F : AEG_GENERIC_OUTPUT.F : AMENDED SECTIONS

In this appendix only the routines from aeg_generic.output.f which are amended and supplemented are shown.

```
      Subroutine aeg_generic_output(p_ac, chout)
c*****************************************************************************
c     reads unsteady aerodynamic panel geometry from tdmb data base
c     and writes it in form of NASTRAN cards to output channel
c     chout (from input list)
c
c     Authors/Revisions:
c
c     August 1998,      Gert-Jan Rollema (Student from Delft University)
c                       - Revision of wing SET1 card attachment
c                         The procedure works as follows:
c                          - reads macro panel geometry
c                          - finds ribs with corresponding span position
c                          - connects panel via SET1 to these ribs
c
c     November 14, 96   MS:
c                       - fixed some bugs in control surface output
c                         (in response to reports by Patrick de Visser,
c                         TU Delft)
c                       Steve Allright, BAe:
c                       - eliminated use of character*40 location
c                         for assembly of tdmb search string
c                         "... Strips ... Strip" (caused overflow when
c                         running on PC Linux)
c
c     July 16, 96       Martin Stettner
c                       DASA Military Aircraft Division
c                       Structural Dynamics Group
c*****************************************************************************
```

INFORMATION BETWEEN THESE LINES HAS BEEN INTENTIONALLY LEFT OUT

```
c----ADJUSTED RIB INFORMATION GATHERING, NEEDED FOR WING BY G.J. ROLLEMA---
c --------- (2) panel coordinates and splines
c --------  START LOOP DEPENDING ON NUMBER OF MACRO PANELS
          StrSectCounter = 1
          r = 1
          s = 1
          RibCount(0)=0
          RibCount(1)=0
          NRibs(0)=0
          location2 = 'RibDatums RibDatum'
          p_rib = tdmb_ref(p_dfsr, location2)
          pribtemp = p_rib
          jmax = tdmb_get_array_size(p_pan)
          do 50 j = 1, jmax
             Write (*,*) 'loopcounter =',StrSectCounter
c            (2a) read panel coordinates & write panel geometry card
             p_pan = tdmb_ref_next(p_pan)
             PanelID  = tdmb_get_int4  (p_pan, 'PanelID')
             IbdLe(1) = tdmb_get_double(p_pan, 'IbdLe X')
             IbdLe(2) = tdmb_get_double(p_pan, 'IbdLe Y')
             IbdLe(3) = tdmb_get_double(p_pan, 'IbdLe Z')
             ObdLe(1) = tdmb_get_double(p_pan, 'ObdLe X')
             ObdLe(2) = tdmb_get_double(p_pan, 'ObdLe Y')
             ObdLe(3) = tdmb_get_double(p_pan, 'ObdLe Z')
             IbdChord = tdmb_get_double(p_pan, 'IbdChord')
             ObdChord = tdmb_get_double(p_pan, 'ObdChord')
             NSpan    = tdmb_get_int4  (p_pan, 'NSpan')
             NChord   = tdmb_get_int4  (p_pan, 'NChord')
             if(AESURF_ind.eq.1) then
                AESURF_panels(2*j-1)  = PanelID
                AESURF_panels(2*j)    = PanelID + NSpan*NChord - 1
             endif
c            for determination of spanwise sections - exception
c            for fin (z-coordinate used, not y)
             if (i.eq.8) then
```

```
                  YOCheck = ObdLe(3)
                  YICheck = IbdLe(3)
               else
                  YOCheck = ObdLe(2)
                  YICheck = IbdLe(2)
               endif
               contB = ' PA'//component(i)
               contB = contB(1:6)//idx//CharInt(j-1)
               if (xzSwitch.eq.'$') then
                  contA = xzSwitch//contB(2:8)
               else
                  contA = '+'//contB(2:8)
               endif
               write(chout,1020) xzSwitch,PanelID, PID, CPbasic,
     *                      NSpan, NChord,
     *                      LSPAN, LCHORD, IGID, contB,
     *                      contA, (IbdLe(k),k=1,3), IbdChord,
     *                      (ObdLe(k),k=1,3), ObdChord

c           get some wing geometry information needed
            if (i.eq.1) then
               p_geoI = tdmb_ref(p_def, 'IbdEngine')
               YIbdmin =     tdmb_get_double(p_geoI, 'Y')
               p_geoO = tdmb_ref(p_def, 'ObdEngine')
               YObdmin =     tdmb_get_double(p_geoO, 'Y')
               if (WingSplineSet.eq.2) then
c                 SET2 chosen:
c                 need lower bound on z (ZMIN) to avoid
c                 inclusion of engine grid nodes in the spline
c                 (see NASTRAN aeroelastic manual for SET2 card)
c                 get position of both engines wrt. wing
                  ZIbd =  tdmb_get_double(p_geoI, 'ZDist')
                  DiameterI =  tdmb_get_double(p_geoI, 'Diameter')
                  ZIbd = - ZIbd + 0.6d0 * DiameterI
                  YIbdmin = YIbdmin - 0.6d0 * DiameterI
                  YIbdmax = YIbdmin + 1.2d0 * DiameterI
                  ZObd =        tdmb_get_double(p_geoO, 'ZDist')
                  DiameterO =  tdmb_get_double(p_geoO, 'Diameter')
                  ZObd = - ZObd + 0.6d0 * DiameterO
                  YObdmin = YObdmin - 0.6d0 * DiameterO
                  YObdmax = YObdmin + 1.2d0 * DiameterO
               else
c                 SET1 chosen:
c                 need planform information to generate separate
c                 splines for wing sections
c----------------adjusted for panelsize by g.j.rollema-------------

            YNextSection(StrSectCounter) = ObdLe(2)
c     reads rib array size from tdmb
            Write (*,*) 'p_rib =', p_rib
            s = tdmb_get_array_size(p_rib)
            Write (*,*) 'Array size', s
c     writes outboard leading edge y position of the macro panel
            Write (*,*) 'ObdLeY' ,ObdLe(2)
c     reads plane y position of the first rib of the specific panel from tdmb
            RibY = tdmb_get_double(p_rib,'Plane Point Y')
            Write (*,*) 'RibY =', RibY
c     reads next rib plane y position, and compares it with the outboard
c     leading edge y position of the macro panel. If smaller then the next
c     rib is read and ribcounter increases.
            do 15 q = 1 , s
            if (ObdLe(2).gt.RibY) then
               pribtemp = p_rib
               p_rib = tdmb_ref_next(p_rib)
               Write (*,*) 'p_rib =', p_rib
               RibCount(StrSectCounter) = RibCount(StrSectCounter)+1
               Write (*,*)'Ribcounter',RibCount(StrSectCounter)
               RibY = tdmb_get_double(p_rib,'Plane Point Y')
               Write (*,*) 'RibY =', RibY
c     if the rib plane y position is greater than the outboard y position
c     the macro panel, then the previous rib y position is read again, so
c     that the rib with the smallest distance to the macro panel is chosen
               if (RibY.gt.ObdLe(2)) then
                 write (*,*)'entering check loop'
                 RibYtemp1 = RibY
                 write (*,*)'RibYtemp',RibYtemp1
                 RibYtemp2 = tdmb_get_double(pribtemp,'Plane Point Y')
                 write (*,*)' RibYnew', RibYtemp2
                 if ((RibYtemp1-ObdLe(2)).gt.(ObdLe(2)-RibYtemp2)) then
```

```
                  write (*,*)'going back one rib'
                  RibCount(StrSectCounter) =
     *                      RibCount(StrSectCounter)-1
                  p_rib = pribtemp
               else

                  goto 15
               endif
            endif
         else
            goto 15
         endif
 15     continue
         endif
            RibCount(0)=0
            NRibs(0)=0
            Write (*,*) 'RibCount =',RibCount(StrSectCounter)
            Write (*,*) 'NRibs-1 =',NRibs(StrSectCounter-1)
            NRibs(StrSectCounter) = RibCount(StrSectCounter) +
     *                           NRibs(StrSectCounter - 1)
            Write (*,*) 'NRibs = ',NRibs(StrSectCounter)

c-----------------END adjustments for rib position reading---------70
        else
c
c           for all components but the wing: Only one spline
c           per component
            YNextSection(1) = 0.d0
            YNextSection(2) = 1.d5

        endif
c           (2b) set of structural grid points associated with this
c                panel
            ZMIN = 0.d0
            ZMAX = 0.d0
c           NOTE: SPLINE2 is a beam spline; serious problems
c                 (singular spline matrix) may occur if
c                 "attachment flexibilities" are not
c                 formulated properly (see NASTRAN Aeroelastic
c                 Analys is User's Manual, Section 3.2, and
c                 SPLINE2 bulk data description
            if (i.eq.1) then
c              - there may be more than 2 connected structural nodes
c                at the same spline axis y-location
c                - avoid spline matrix singularity by introducing
c                  translational flexibility
               DZ =    1.0d0
c              - rotational degress of freedom of structural nodes
c                connected to the skins are constrained to 0 (through
c                AUTOSPC - since the skin panel elements do not react
c                node rotation)
c                - avoid spline matrix singularity by decoupling
c                  node rotations from the splining procedure
               DTHX = -1.0d0
               DTHY = -1.0d0
               if (WingSplineSet.eq.2) then
c
c              SET2 cards chosen for wing
c
c                 for wing: check if the panel is located (partially
c                 or totally) over an engine; if that is the case, set
c                 lower bound on z-range for SET2 (see Nastran
c                 aeroelastic analysis manual)
c                 check for partial inboard overlap, total overlap,
c                 partial outboard ovelap - inboard engine
                  if (YIbdmin.lt.YOCheck
     *                  .and.YOCheck.le.YIbdmax .or.
     *              YIbdmin.ge.YICheck
     *                  .and.YOCheck.ge.YIbdmax .or.
     *              YIbdmin.lt.YICheck
     *                  .and.YICheck.le.YIbdmax) then
                    ZMIN = ZIbd
                  endif
c                 check for partial inboard overlap, total overlap,
c                 partial outboard overlap - outboard engine
                  if (YObdmin.lt.YOCheck
     *                  .and.YOCheck.le.YObdmax .or.
     *              YObdmin.ge.YICheck
     *                  .and.YOCheck.ge.YObdmax .or.
```

```
     *                 YObdmin.lt.YICheck
     *                     .and.YICheck.le.YObdmax) then
                         ZMIN = ZObd
                     endif
c                    SET2 card
                     setcounter = setcounter + 1
                     setID = minsetID + setcounter
                     write(chout,1030)      setID, PanelID,
     *                       0.0, 1.0, 0.0, 1.0, ZMAX, ZMIN
                 else
c----------minor adjustments were made for correct rib numbers-------
                     Write (*,*)' SET1 cards chosen for wing'
c
                     if (YNextSection(StrSectCounter).gt.YICheck) then
c                       generate separate splines for 4 sections
                        l = 0
                        NRibs(0)=1
                        do 20 k = NRibs(StrSectCounter - 1),
     *                           NRibs(StrSectCounter)
                           l = l+1
c                                      component
                           IntSet(l) = strucID(i) * strucFac1
c                                      rib number
     *                                 + k  * strucFac2
c                                      upper surface
     *                                 + 1  * strucFac3
c                                      FRONT spar cap
     *                                 + 1
                           l = l+1
c                                      component
                           IntSet(l) = strucID(i) * strucFac1
c                                      rib number
     *                                 + (k)  * strucFac2
c                                      upper surface
     *                                 + 1  * strucFac3
c                                      REAR spar cap
     *                                 + 99
 20                     continue
c                       number of structural nodes in spline set
                        IntSetDim = l
c                       SET1 card
                        setcounter = setcounter + 1
                        setID = minsetID + setcounter
                        call write_SET1(setID, IntSet,IntSetDim,
     *                          StrSectCounter,component(i),chout,
     *                          xzSwitch)
                        StrSectCounter = StrSectCounter + 1
                        write (*,*) 'loopcounter2 =', StrSectCounter
                        r = r + 1
                     endif
                  endif
              else
c                 for tail, fin, engine: Attachment stiffness is  needed,
c                 particularly for the tail and fin stick models!
                  DZ =   0.0d0
                  DTHX = 0.0d0
                  DTHY = 0.0d0
                  if (YOCheck.gt.
     *                  YNextSection(StrSectCounter)) then
                     if (i.eq.10.or.i.eq.11) then
c                       only 2 nodes used for engine set: the
c                       engine and pylon cg nodes, counters 5
c                       and 6
                        IntSetDim = 2
                        IntSet(1) = strucID(i) * strucFac1
c                                      node number
     *                                 + 5
                        IntSet(2) = strucID(i) * strucFac1
c                                      node number
     *                                 + 6
                     else if (i.eq.5.or.i.eq.8) then
c                       number of structural nodes determined by "strips"
c                       of component stick model
                        l = index(component(i),' ')
c                        location = component(i)
                        location= component(i)(1:l-1)//'Strips '//
     *                          component(i)(1:l-1)//'Strip'
                        p_geo = tdmb_enq(p_ac,
```

```
     *                 'Specification Assumptions PrelimStickModels '//
     *                  location)
c                      no stick model: process next component
                       if (p_geo.eq.0) then
                          write(chout,'(A)')
     *                      '$     No structural model found!'
                          goto 100
                       endif
                       IntSetDim = tdmb_get_array_size(p_geo) + 1
                       do 40 l = 1, IntSetDim
                          IntSet(l) = strucID(i) * strucFac1
c                                        node number
     *                                   + 1  * strucFac2
 40                    continue
c
c-------------------ADJUSTED SETID DETERMINATION FOR AILERON--------------
c still needs work to make it fully flexible with wing panelling
c sugestion :make separate SETs for ailerons independent from wing
c
                    else if (i.eq.2) then
                       l = index(component(i),' ')
                       location= component(i)(1:l-1)//'Strips '//
     *                           component(i)(1:l-1)//'Strip'
                       p_geo = tdmb_enq(p_ac,
     *                 'Specification Assumptions PrelimStickModels '//
     *                  location)
                       p_geo = tdmb_enq(p_ac, location)
c                      no stick model for ailerons:
c                      couple rigidly to outboard wing,
                       if (p_geo.eq.0) then
                          write(chout,'(A)')
     *                    '$     No structural model'//
     *                       ' found - coupled to outboard wing'
                          setID = minsetID + 5
                          DZ = 1.d0
                          DTHX = -1.d0
                          DTHY = -1.d0
                          goto 45
                       endif
                       IntSetDim = tdmb_get_array_size(p_geo) + 1
                       do 42 l = 1, IntSetDim
                          IntSet(l) = strucID(i) * strucFac1
c                                        node number
     *                                   + 1  * strucFac2
 42                    continue
                    else if (i.eq.3) then
                       l = index(component(i),' ')
                       location= component(i)(1:l-1)//'Strips '//
     *                           component(i)(1:l-1)//'Strip'
                       p_geo = tdmb_enq(p_ac,
     *                 'Specification Assumptions PrelimStickModels '//
     *                  location)
                       p_geo = tdmb_enq(p_ac, location)
c                      no stick model for ailerons:
c                      couple rigidly to outboard wing,
                       if (p_geo.eq.0) then
                          write(chout,'(A)')
     *                    '$     No structural model'//
     *                       ' found - coupled to outboard wing'
                          setID = minsetID + 6
                          DZ = 1.d0
                          DTHX = -1.d0
                          DTHY = -1.d0
                          goto 45
                       endif
                       IntSetDim = tdmb_get_array_size(p_geo) + 1
                       do 43 l = 1, IntSetDim
                          IntSet(l) = strucID(i) * strucFac1
c                                        node number
     *                                   + 1  * strucFac2
 43                    continue
                    endif
c------------------END ADJUSTMENTS-------------------------------
```

INFORMATION BETWEEN THESE LINES HAS BEEN INTENTIONALLY LEFT OUT

END

APPENDIX-G : FILE -WINGLIFTFORCES.TOM

```
(Define
:Description "Lift Forces On Aerodynamic Panels as Calculated with Nastran SOL 144"
:* "#" "*" '*'
:FlightConditions "*" "Flight condtions for this Subcase"
(Define
:MachNo "Real" "Mach number" "*"
:DynPress "Real" "Dynamic pressure" "*"
)
:MacroPanels "Array" "Wing Macro Panels"
(Define
:PanelID "*" "Wing Macro Panels id number"
(Define
:Name "Integer" "Wing Macro Panel identification number" "*"
:IbdLeY "Real" "Inboard panel spanwise position." "*"
:IbdLeX "Real" "Inboard panel chordwise position." "*"
:IbdChord "Real" "Inboard panel chord length" "*"
:ObdLeY "Real" "Outboard panel spanwise position." "*"
:ObdLeX "Real" "Outboard panel chordwise position." "*"
:ObdChord "Real" "Outboard panel chord Length" "*"
:NSpan "Integer" "Number of spanwise panels per macro panel." "*"
:NChord "Integer" "Number of chordwise panels per macro panel." "*"
:SubPanels "Array" "Sub Panel array"
(Define
:SubPanelID "*" "Sub panel information"
(Define
:Name "Integer" "Sub panel identification number" "*"
:SubPanX "Real" "Sub panel chordwise position" "*"
:SubPanY "Real" "Sub panel spanwise position" "*"
:SubPanWidth "Real" "Sub panel width" "*"
:LocChrd "Real" "Local Chord at this panel y position" "*"
:AeroForce "Real" "Aero dynamic force on sub panel" "*"
)
)
)
)
:WingStrips "Array" "Aero Forces totalled over wing strip position"
(Define
:Eta "*" "Spanwise Eta of wing strip"
(Define
:Name "Real" "Wing Strip Eta" "*"
:Eta "Real" "Eta of wing Strip" "*"
:TotalForce "Real" "Total Aero Force on Wing Strip" "*"
:LocChrd "Real" "Local Chord of Strip" "*"
:StripLoad "Real" "Loading per meter span for this Strip" "*"
:ClStrip "Real" "Cl of strip (spanwise)" "*"
:Clc/mac "Real" "Cl times local chord devided by mean aero chord" "*"
)
)
:FuselagePanels "Array" "Fuselage Aerodynamic Panels"
(Define
:FusePanel "*" "Fuselage aero panel"
(Define
:Name "Integer" "Fuselage Panel ID" "*"
:XPosition "Real" "Mid Panel x-position from nose fuselage" "*"
:YWidth "Real" "Fuselage panel width" "*"
:SPanel "Real" "Fuselage panel Area" "*"
:PanelLift "Real" "Fuselage panel Lift Force" "*"
:PanelCl "Real" "Fuselage panel Cl" "*"
:PanelLoad-x "Real" "Fuselage panel load (N/m) in x-direction" "*"
:PanelLoad-y "Real" "Fuselage panel load (N/m) in y-direction" "*"
)
)
:Totals "*" "Total Lift forces"
(Define
:WingLift "Real" "Total Lift of the Wing" "*"
:TailLift "Real" "Total Lift of horizontal Tail" "*"
:IbdELift "Real" "Total inboard Engine Lift" "*"
:ObdELift "Real" "Total outboard Engine Lift" "*"
:AilerLift "Real" "Total aileron Lift" "*"
:FuseLift "Real" "Total fuselage Lift" "*"
:TotalLift "Real" "Total Lift" "*"
)
)
```

APPENDIX-H : NASTPP.F SUPPLEMENTED ROUTINES

In this appendix only the routines from NASTPP.f which are amended and supplemented are shown.

```
C program NASTPP VERSION V2 FOR NASTRAN V70.0 & V70.5.0 AND CFD FILES
C ---------------------
C   Author : Steve Allwright
C     Date : 17-Nov-96
C
C   Supplemented : Gert-Jan Rollema (student from Delft University)
C          Date : August-December 98
C
C This program reads the standard NASTRAN channel 6 output file
C and writes selected data back into TDMB. The node numbering
C system adopted within the MMG is assumed, and output files
C corresponding to the example NASTRAN job control files contained
C in D205CD+O/P are assumed.
C
C The program works by detecting key character sequences/positions
C in the output file, and then reading the necessary data from the
C file and lodging it directly into TDMB. The data lines read from
C file are written out to a summary file, both so that the operation
C of the programm can be worked out, and so that reduced size files
C can be retained for testcases (Note, that the opt.f06 file in the
C data directory is one of these summary files, not the full 30MB
C Nastran output file).
C
C Program sequence is :-
C     attach to TDMB local database file
C     repeat
C        ask user to identify output type (ie. .job type)
C        ask user to identify specific output file
C        ask user to identify design stage
C        read data associated with output type and lodge
C          data into TDMB under reqd design stage.
C     end repeat
C     Save TDMB local database.
C
C Data read for each file type is :-
C
C  1) stt.job/stt.f06 : static analysis
C      reads undeformed wing geometry into "operation" array template
C      reads deflection and stress data for each subcase.
C
C  2) sym.job/sym.f06 : symmetric modes analysis
C      reads undeformed wing geometry into "mode" array templates
C      reads symmetric modal frequency data
C      reads wing deflections for each mode
C
C  3) asm.job/asm.f06 : anti-symmetric modes analysis
C      reads undeformed wing geometry into "mode" array templates
C      reads anti-symmetric modal frequency data
C      reads wing deflections for each mode
C
C  4) syf.job/syf.f06 : symmetric flutter analysis
C      reads undeformed wing geometry into "mode" array templates
C      reads symmetric modal frequency data
C      reads flutter speed/frequency/damping data.
C
C  5) asf.job/asf.f06 : anti-symmetric flutter analysis
C      reads undeformed wing geometry into "mode" array templates
C      reads anti-symmetric modal frequency data
C      reads flutter speed/frequency/damping data.
C
C 11) opt.job/opt.f06 : static strength optimisation.
C      reads undeformed wing geometry into "operation" array template
C      reads InitialDesign deflection and stress data for each subcase.
C      reads Mass, Constraints and Design Variable optimisation history.
C      reads FinalDesign deflection,stress,size data for each subcase.
C
C-------------------------------------------------------------------
C added options August-December 1998
C
C option 6) to read aerodynamic forces from sol 144 outputfile, and write
```

```
C results back into the TDMB.
C
C NOTE THIS OPTION ONLY WORKS WITH NASTRAN V70.0 OR V70.5.0 AND IF THE
C NECESSARY AMENDMENT TO THE OBJECTLIBRARY HAVE BEEN MADE.
C UNKNOWN IF IT WORKS WITH PREVIOUS NASTRAN VERSIONS.
C VERSIONS SUPPORTING LATER NASTRAN VERSIONS WILL FOLLOW.
C
C   6) femtest.f06: static aeroelastic analysis
C      1) reads aerodynamic forces on CAERO elements, and outputs totals
C         to screen
C      2) puts aerodynamic force for each wing and aileron subpanel
C         back into TDMB
C-----------------------------------------------------------------
C option 7) to read aerodynamic forces from a specific CFD outputfile,
C and write results back into the TDMB.
C
C   7) name.xxx: CFD data file
C      1) reads results generated by CFD
C      2) puts aerodynamic forces back into TDMB
C-----------------------------------------------------------------
C Notes :-
C -----
C
C Option 11 operates on an Optimisation run, extracting thickness
C and stress information for both the Zero'th and Final design
C iterations, together with design variable and objective history.
C
C Options 1-7 operate on an analysis run. This may be an analysis
C of an initial model as output by MMG, or of the final design
C model as updated/output by fegpp. For this reason, you can
C input the name of the design stage with which to associate the
C results.
C
C The testcase, read the optimisation results, and then the final
C results into FinalDesign. The user control data used for the
C testcase can be found in :-
C        Examples/BAe-Nastran/data/inputfile
C
C-----------------------------------------------------------------
```

INFORMATION BETWEEN THESE LINES HAS BEEN INTENTIONALLY LEFT OUT

```
C---This is option 6 : Read results From SOL 144 analysis-------------
      else if ( ifile.eq.6 ) then
        write (6,'(A,$)') ' Enter filename [femtest.f06] $ '
        read  (5,'(A)',end=95)file
        if ( file.eq.' ')then
          file = 'femtest.f06'
        end if
        write (6,'(3A,$)')
     +           ' Enter design stage [', stagename(1:stlen), '] $ '
        read  (5,'(A)',end=95) newname
        if ( newname(1:1).eq.'q' .or.
     +       newname(1:1).eq.'Q' .or.
     +       (newname.eq.' '.and.stagename.eq.' ' ) ) goto 95
        if ( newname.ne.' ' ) stagename = newname
        p_stage = tdmb_enq (p_ac,'Results StructResults Stages '
     +                     //stagename )
        if (p_stage.eq.0)then
          p_stage = TDMB_MAKE_NEXT ( p_ac,
     +                      'Results StructResults Stages Stage' )
          idum = TDMB_PUT_STRING ( p_stage,' Name',stagename)
        end if
        p_ops = TDMB_REF ( p_stage, 'Operations' )
        t_wing = TDMB_REF ( p_ac,'Results StructResults Stages '//
     +                    'Stage Operations Operation WingDeflection')
C--End Option 6 ---------------------------------------------------
C---This is option 7 : Read results From CFD data --------------------
      else if ( ifile.eq.7 ) then
        write (6,'(A,$)') ' Enter filename [fuse-wing.txt] $ '
        read  (5,'(A)',end=95)file
        if ( file.eq.' ')then
          file = 'fuse-wing.txt'
        end if
        write (6,'(A,$)') ' 1 = fuselage  2 = wing $ '
        read  (5,'(A)',end=95)choise
```

```
        if ( choise.eq. '1' )then
          CFD = 1
          write(*,*)'Fuselage chosen ',CFD
        else if (choise.eq. '2')then
          CFD = 2
          write(*,*)'Wing chosen ',CFD
        end if
        write (6,'(3A,$)')
     +           ' Enter design stage [', stagename(1:stlen), '] $ '
        read  (5,'(A)',end=95) newname
        if ( newname(1:1).eq.'q' .or.
     +       newname(1:1).eq.'Q' .or.
     +       (newname.eq.' '.and.stagename.eq.' ' ) ) goto 95
        if ( newname.ne.' ' ) stagename = newname
        p_stage = tdmb_enq (p_ac,'Results AeroResults '
     +                     //stagename )
        if (p_stage.eq.0)then
          p_stage = TDMB_MAKE_NEXT ( p_ac,
     +                               'Results AeroResults AeroResult' )
          idum = TDMB_PUT_STRING ( p_stage,' Name',stagename)
        end if
C--End Option 7 ----------------------------------------------------
```

INFORMATION BETWEEN THESE LINES HAS BEEN INTENTIONALLY LEFT OUT

```
C Establish which operation/subcase is active. This part has been amended
C for SOL 144 result reading/writing
      else if ( line(110:116).eq.'SUBCASE' ) then
        subcase = line(7:38)
        if ( subcase.ne.' ' .and. subcase.ne.oldcase) then
          if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
            p_oper = TDMB_ENQ ( p_ops, subcase )
            p_oper_old = TDMB_ENQ (p_ops, oldcase)
          if ( p_oper.eq.0 ) then
            p_oper = TDMB_MAKE_NEXT ( p_ops, 'Operation' )
            idum = TDMB_PUT_STRING ( p_oper,'Name',subcase)
          end if
C       --pointers set at begin of a subcase
          if (casecount.eq.1) then
            p_fcond = TDMB_REF (p_oper, 'WingLiftForces
     +                         FlightConditions')
            p_fwing = TDMB_REF ( p_oper, 'WingLiftForces MacroPanels
     +                          PanelID')
            p_fwing3 = TDMB_REF ( p_oper, 'WingLiftForces WingStrips')
            p_fwing6 = TDMB_REF ( p_oper, 'WingLiftForces WingStrips')
            p_fwing8 = TDMB_REF ( p_oper, 'WingLiftForces WingStrips')
            p_ffuse = TDMB_REF(p_oper,'WingLiftForces FuselagePanels
     +                        FusePanel')
            casecount = casecount + 1
          end if
          p_wing = TDMB_REF ( p_oper,  'WingDeflection')
          p_fuse = 0
          p_fin  = 0
          p_tail = 0
          p_ieng = 0
          p_oeng = 0
          newcounter = counter
C       --Write Total Aero Lift forces to screen and TDMB
          if (newcounter.gt.1) then
            f_totalnew = f_total +f_body
            write (*,*)'              '
            write (*,*)'SUBCASE = ',oldcase
            write (*,*)'------------------------------------------------'
            write (*,*)'Total Wing Lift [N]     ',f_wing
            write (*,*)'Total Tail Lift         ',f_tail
            write (*,*)'Inboard Engine Lift     ',f_ibde
            write (*,*)'Outboard Engine Lift    ',f_obde
            write (*,*)'Aileron Lift            ',f_ail
            write (*,*)'Body Lift (z dirction)',f_body
            write (*,*)'Body Lift (y dirction)',f_body_y
            write (*,*)'Total Lift              ',f_totalnew
            Write (*,*)'------------------------------------------------'
            idum = TDMB_PUT_DOUBLE(p_oper_old,'WingLiftForces Totals
     +                          WingLift',f_wing)
            idum = TDMB_PUT_DOUBLE(p_oper_old,'WingLiftForces Totals
     +                          TailLift',f_tail)
            idum = TDMB_PUT_DOUBLE(p_oper_old,'WingLiftForces Totals
     +                          IbdELift',f_ibde)
```

```
          idum = TDMB_PUT_DOUBLE(p_oper_old,'WingLiftForces Totals
     +                        ObdELift',f_obde)
          idum = TDMB_PUT_DOUBLE(p_oper_old,'WingLiftForces Totals
     +                        AilerLift',f_ail)
          idum = TDMB_PUT_DOUBLE(p_oper_old,'WingLiftForces Totals
     +                        FuseLift',f_body)
          idum = TDMB_PUT_DOUBLE(p_oper_old,'WingLiftForces Totals
     +                        TotalLift',f_totalnew)
          f_body_y=0
          f_body=0
          f_wing=0
          f_tail=0
          f_ibde=0
          f_obde=0
          f_total=0
          f_totalnew=0
          f_ail=0
          fspan=0
          y_subpan_old = 0
          y_pos_old = 0
          p_fwing = TDMB_REF ( p_oper, 'WingLiftForces MacroPanels
     +                       PanelID')
          p_fwing3 = TDMB_REF ( p_oper, 'WingLiftForces WingStrips')
          p_fwing6 = TDMB_REF ( p_oper, 'WingLiftForces WingStrips')
          p_fwing8 = TDMB_REF ( p_oper, 'WingLiftForces WingStrips')
          p_ffuse = TDMB_REF(p_oper,'WingLiftForces FuselagePanels
     +                       FusePanel')
          idcheck = 0
          oldcounter = newcounter
          counter = 1
        end if
        oldcase = subcase
      end if
```

INFORMATION BETWEEN THESE LINES HAS BEEN INTENTIONALLY LEFT OUT

```
C--------------------------------------------------------------------------
C-----------------------READ SOL 144 RESULTS-----------------------------
C-----------------------(by Gert-Jan Rollema)----------------------------
C--------------------------------------------------------------------------
C--This part of the program does the following:
C       1)Extract aerodynamic lift forces for wing, tail, engines ,ailerons
C         and body and writes info to screen and TDMB  (works for half and
C         full FE-models of the reference aircraft)
C       2)Write Wing lift and Aileron Lift to TDMB per subpanel (works for
C         half and full model but with full model only one wing will be
C         calculated and written to TDMB
C       3)Write Wing Lift and Aileron Lift per spanwise strip to tdmb
C------Read AeroElastic Results from File. This part has to be changed if
C       an other version than 70.0 of 70.5.0 of NASTRAN is used as with
C       version 70.5.1 the results are written in another sequence!!!!!
        else if (line(25:28).eq.' LS ') THEN
         if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
          read(line,'(75X,I2,I3,I1,I2,8X,e14.6)')id1,id2,id3,id4,fz
C---------INBOARD ENGINE
C       ---Read starboard inboard engine panels
          if ((id1.eq.10).and.(id3.eq.0)) then
            f_ibde = f_ibde + fz
C         --Write information to tdmb if required in ControlData
            engine_yes = TDMB_GET_STRING(p_ac,'ControlData
     +                     StaticFEModel EngineLift')
            If ((engine_yes.eq.'Y').or.(engine_yes.eq.'y')) then
C           --Read IdbE eta from tdmb
              IbdE_eta = TDMB_GET_DOUBLE(p_ac,'Specification
     +                     Aircraft Engines IbdEta')
C             --Find closest eta in WingStrip array. This is
C             --possible because the WingStrip array already exist
                p_fwing7 = TDMB_REF(p_fwing6,'Eta')
                etatest = TDMB_GET_DOUBLE(p_fwing7,'Eta')
                do while (((IbdE_eta-etatest).gt.0.01)
     +                .or.((etatest-IbdE_eta).gt.0.01))
                  p_fwing7 = TDMB_REF_NEXT(p_fwing7)
                  etatest = TDMB_GET_DOUBLE(p_fwing7,'Eta')
                end do
C             --Read TotalForces and StipLoad at corresponding eta
                fspan = TDMB_GET_DOUBLE(p_fwing7,'TotalForce')
                fsload = TDMB_GET_DOUBLE(p_fwing7,'StripLoad')
```

```
C             --add fz of Engine subpanel and write back to tdmb
                subwidth = fspan/fsload
                fspan = fspan + fz
                fsload = fspan/subwidth
                idum = TDMB_PUT_DOUBLE(p_fwing7,'TotalForce',fspan)
                idum = TDMB_PUT_DOUBLE(p_fwing7,'StripLoad',fsload)
             end if
C      ---Read port inboard engine panels
          else if ((id1.eq.10).and.(id3.eq.5)) then
            f_ibde = f_ibde - fz
          endif
C---------OUTBOARD ENGINE
C     ----Read starboard outboard engine panels
          if ((id1.eq.11).and.(id3.eq.0)) then
            f_obde = f_obde + fz
C         --Write information to tdmb if required in ControlData
            engine_yes = TDMB_GET_STRING(p_ac,'ControlData
     +                        StaticFEModel EngineLift')
            If ((engine_yes.eq.'Y').or.(engine_yes.eq.'y')) then
C           --Read OdbE eta from tdmb
              ObdE_eta = TDMB_GET_DOUBLE(p_ac,'Specification
     +                         Aircraft Engines ObdEta')
C             --Find closest eta in WingStrip array. This is
C             --possible because the WingStrip array already exist
                p_fwing9 = TDMB_REF(p_fwing8,'Eta')
                etatest = TDMB_GET_DOUBLE(p_fwing7,'Eta')
                do while (((ObdE_eta-etatest).gt.0.025)
     +                   .or.((etatest-ObdE_eta).gt.0.025))
                  p_fwing9 = TDMB_REF_NEXT(p_fwing9)
                  etatest = TDMB_GET_DOUBLE(p_fwing9,'Eta')
                end do
C             --Read TotalForces and StipLoad  at corresponding eta
                fspan = TDMB_GET_DOUBLE(p_fwing9,'TotalForce')
                fsload = TDMB_GET_DOUBLE(p_fwing9,'StripLoad')
C             --add fz of Engine subpanel and write back to tdmb
                subwidth = fspan/fsload
                fspan = fspan + fz
                fsload = fspan/subwidth
                idum = TDMB_PUT_DOUBLE(p_fwing9,'TotalForce',fspan)
                idum = TDMB_PUT_DOUBLE(p_fwing9,'StripLoad',fsload)
           end if
C---------Read port outboard enigne panels
          else if ((id1.eq.11).and.(id3.eq.5)) then
            f_obde = f_obde - fz
          endif
C---------TAIL
C      ---Starboard
          if ((id1.eq.5).and.((id3.eq.0).or.(id3.eq.1))) then
            f_tail = f_tail + fz
C      ---Port
          else if ((id1.eq.5).and.((id3.eq.5).or.(id3.eq.6))) then
            f_tail = f_tail - fz
          endif
C---------AILERONS
C      ---Starboard Ailerons
          if (((id1.eq.2).or.(id1.eq.3)).and.(id3.eq.0)) then
            f_ail = f_ail + fz
              if (id4.eq.0) then
C         ----Write First sub panel of the aileron macro panel to tdmb
                p_fwing = TDMB_REF_NEXT(p_fwing)
                p_fwing2 = TDMB_REF(p_fwing,'SubPanels SubPanelID')
                p_fwing2 = TDMB_REF_NEXT(p_fwing2)
                idum = TDMB_PUT_DOUBLE (p_fwing2,'AeroForce',fz)
              end if
              if (id4.gt.0)then
C         ----Write the rest of the aileron sub panels to tdmb
                p_fwing2 = TDMB_REF_NEXT (p_fwing2)
                idum = TDMB_PUT_DOUBLE (p_fwing2,'AeroForce',fz)
              end if
C         ----Write Aileron Aero Forces to WingLiftForces WingStrip

C             --Read aileron sub panel y-position and calc its eta
                mac = TDMB_GET_double(p_ac,'Definition Geometry
     +                       Wing Planform AMChord')
                y_subpan = TDMB_GET_DOUBLE(p_fwing2,'SubPanY')
                trim_dynp = TDMB_GET_DOUBLE(p_fcond,'DynPress')
                Span = TDMB_GET_DOUBLE(p_ac,'Definition Geometry
     +                               Planform Span')
                eta = y_subpan/(0.5*Span)
```

```
c              --read aileron local chord data and panel width
                 loc_chrd_ail = TDMB_GET_DOUBLE(p_fwing2,'LocChrd')
                 w_subpan = TDMB_GET_DOUBLE(p_fwing2,'SubPanWidth')
c              --write new local chrd to tdmb
c                 if (eta.gt.eta_new) then
c                 idum = TDMB_PUT_DOUBLE(p_fwing5,'LocChrd',loc_chrd_new)
c                 end if
c              --Find corresponding eta in WingStrip array. This is
c              --possible because the WingStrip array already exist
                 p_fwing5 = TDMB_REF(p_fwing3,'Eta')
                 etatest = TDMB_GET_DOUBLE(p_fwing5,'Eta')
                 do while ((eta-etatest).gt.0.01)
                   p_fwing5 = TDMB_REF_NEXT(p_fwing5)
                   etatest = TDMB_GET_DOUBLE(p_fwing5,'Eta')
                 end do
c              --Read TotalForces from WingStrip at corresponding eta
                 fspan = TDMB_GET_DOUBLE(p_fwing5,'TotalForce')
                 loc_chrd_strip  = TDMB_GET_DOUBLE(p_fwing5,'LocChrd')
                 loc_chrd_new = loc_chrd_strip+loc_chrd_ail

c              --add fz of aileron subpanel and write back to tdmb,
c                calculate Cl of wing strip
                 fspan = fspan + fz
                 Cl_strip = fspan/(loc_chrd_new*w_subpan*trim_dynp)
                 Clc_mac = ((Cl_strip * Loc_chrd_new) / mac)
                 idum = TDMB_PUT_DOUBLE(p_fwing5,'TotalForce',fspan)
                 idum = TDMB_PUT_DOUBLE(p_fwing5,'ClStrip',Cl_strip)
                 idum = TDMB_PUT_DOUBLE(p_fwing5,'Clc/mac',Clc_mac)
c                 if (eta.gt.0.96).and.( then    ,
c                 idum = TDMB_PUT_DOUBLE(p_fwing5,'LocChrd',loc_chrd_new)
c                 end if

c                 eta_new=eta


c       ---Port Ailerons
           else if(((id1.eq.2).or.(id1.eq.3)).and.(id3.eq.5))then
             f_ail = f_ail - fz
           endif
C---------WING
c       ---Starboard Wing ONLY
           if ((id1.eq.1).and.((id3.eq.0).or.(id3.eq.1))) then
             f_wing = f_wing + fz
               if ((id4.eq.0).and.(id3.eq.0)) then
c          ----Write First sub panel of the wing macro panel to tdmb
                 p_fwing = TDMB_REF_NEXT(p_fwing)
                 p_fwing2 = TDMB_REF(p_fwing,'SubPanels SubPanelID')
                 p_fwing2 = TDMB_REF_NEXT(p_fwing2)
                 idum = TDMB_PUT_DOUBLE (p_fwing2,'AeroForce',fz)
               end if
               if ((id4.gt.0).or.((id4.eq.0).and.(id3.eq.1))) then
c          ----Write Rest of the wing sub panels to tdmb
                 p_fwing2 = TDMB_REF_NEXT (p_fwing2)
                 idum = TDMB_PUT_DOUBLE (p_fwing2,'AeroForce',fz)
               end if
c          ----Write totalled aero force to WingLiftForces WingStrip
c          ----Read wing sub panel y-position and sub panel array size
               mac = TDMB_GET_double(p_ac,'Definition Geometry
     +                        Wing Planform AMChord')
               y_subpan = TDMB_GET_DOUBLE(p_fwing2,'SubPanY')
               trim_dynp = TDMB_GET_DOUBLE(p_fcond,'DynPress')
               subarrsize = TDMB_GET_ARRAY_SIZE(p_fwing2)
               y_pos_begin = TDMB_GET_DOUBLE(p_fwing,'IbdLeY')
               loc_chrd = TDMB_GET_DOUBLE(p_fwing2, 'LocChrd')
               w_subpan = TDMB_GET_DOUBLE(p_fwing2, 'SubPanWidth')
c            --add sub panel fz and calculate eta
               if ((y_subpan.eq.y_subpan_old).or.
     +             (y_subpan_old.eq.0)) then
                  fspan = fspan + fz
                  loc_chrd_old = loc_chrd
                  w_subpan_old = w_subpan
                  y_subpan_old = y_subpan
                  arrcount = arrcount + 1
                  Span = TDMB_GET_DOUBLE(p_ac,'Definition Geometry
     +                                  Planform Span')
                  eta = y_subpan/(0.5*Span)
               end if

c          --- Write Strip force Load and eta if y_subpanel changes
```

```
               if ((y_subpan.gt.y_subpan_old)
     +              .and.(arrcount.ne.subarrsize)) then
                  p_fwing4 = TDMB_MAKE_NEXT(p_fwing3,'Eta')
                  Cl_strip = fspan/(w_subpan_old*loc_chrd_old*trim_dynp)
                  Clc_mac = ((Cl_strip * Loc_chrd_old) / mac)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'Name',eta)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'Eta',eta)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'TotalForce',fspan)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'LocChrd',loc_chrd_old)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'ClStrip',Cl_strip)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'Clc/mac',Clc_mac)
C                --calculate subpanel width and Strip Load
                  if(y_pos_begin.lt.y_pos_old) then
                     y_pos = (eta*Span - y_pos_old)
                     subwidth = (y_pos - y_pos_old)
                     else
                     y_pos = (eta*Span - y_pos_begin)
                     subwidth = (y_pos - y_pos_begin)
                  end if
                  fstrip = fspan/subwidth
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'StripLoad',fstrip)
                  y_pos_old = y_pos
                  fspan= 0
                  fspan = fspan + fz
                  y_subpan_old = 0
                  arrcount = arrcount + 1
               end if
C           ---Write strip force and eta for last/only strip of macro panel
               if (arrcount.eq.subarrsize) then
                  p_fwing4 = TDMB_MAKE_NEXT(p_fwing3,'Eta')
                  Cl_strip = fspan/(w_subpan_old*loc_chrd_old*trim_dynp)
                  Clc_mac = ((Cl_strip * Loc_chrd_old) / mac)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'Name',eta)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'Eta',eta)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'TotalForce',fspan)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'LocChrd',loc_chrd)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'ClStrip',Cl_strip)
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'Clc/mac',Clc_mac)
C                --calculate subpanel width and Strip Load
                  if (y_pos_begin.lt.y_pos_old) then
                     y_pos = (eta*Span - y_pos_old)
                     subwidth = (y_pos - y_pos_old)
                     else
                     y_pos = (eta*Span - y_pos_begin)
                     subwidth = (y_pos - y_pos_begin)
                  end if
                  fstrip = fspan/subwidth
                  idum = TDMB_PUT_DOUBLE(p_fwing4,'StripLoad',fstrip)
                  y_pos_old = y_pos
                  fspan = 0
                  y_subpan_old = 0
                  arrcount = 0
               end if
C       ---Port Wing
           else if ((id1.eq.1).and.((id3.eq.5).or.(id3.eq.6))) then
             f_wing = f_wing - fz
           end if
C       -Add totals
         f_total = f_wing + f_tail + f_obde + f_ibde + f_ail
         counter = counter + 1

C---------BODY Z-direction
           else if (line(87:89).eq.'ZSB') THEN
           if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
             read(line,'(92X,e14.6)')fz
             f_body = f_body + fz
C          --Calculate strip loads and Cl for fuselage panel
             p_ffuse = TDMB_REF_NEXT(p_ffuse)
             y_fpan = TDMB_GET_DOUBLE(p_ffuse,'YWidth')
             S_fpan = TDMB_GET_DOUBLE(p_ffuse,'SPanel')
             trim_dynp = TDMB_GET_DOUBLE(p_fcond,'DynPress')
             Cl = (fz)/(trim_dynp*S_fpan)
             ff_strip_x = fz/x_fpn
             ff_strip_y = fz/y_fpan
C          --Write results to TDMB
             idum = TDMB_PUT_DOUBLE(p_ffuse,'PanelLift',fz)
             idum = TDMB_PUT_DOUBLE(p_ffuse,'PanelCl',Cl)
             idum = TDMB_PUT_DOUBLE(p_ffuse,'PanelLoad-x',ff_strip_x)
             idum = TDMB_PUT_DOUBLE(p_ffuse,'PanelLoad-y',ff_strip_y)
```

```
C---------BODY Y-direction
          else if (line(87:89).eq.'YSB') THEN
          if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
             read(line,'(92X,e14.6)')fy
             f_body_y = f_body_y + fy

C-----------------------------------------------------------------72
C-----This part of the program does the following:
C     1)Reads CAERO card information from SORTED BULK DATA DECK
C     2)Calculate each individual sub panel position
C     3)Write panel array to tdmb for each subcase

C-----Read Case Control Echo for subcase information and make
C     the subcases in StrucResults, Stages, Stage, Operations
C     This is necessary as the  CAERO1 cards are read from the
C     sorted Bulk Data Deck and at that point in reading the output file
C     there is no subcase information available in the tdmb yet
      else if ( line(26:32).eq.'SUBCASE' ) then
        if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
           read (10,'(A132)',end=20) line
           subcasewrite(subcasecounter) = line(34:65)
           p_oper = TDMB_MAKE_NEXT ( p_ops, 'Operation' )
           idum = TDMB_PUT_STRING ( p_oper,'Name',
     +                subcasewrite(subcasecounter))
           subcasecounter = subcasecounter + 1

C-----Read CAERO1 cards for aero panel information
      else if (line(30:37).eq.' CAERO1 ') THEN
        if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
          read(line,'(37X,I2,I3,I1,I2,17X,I3,5X,I3)')
     +                  id1,id2,id3,id4,n_span,n_chord
          read(line,'(37X,I8)')pan_id
C----needed for later add-ons-------------------------------------
C         if ((id1.eq.10).and.(id3.eq.0)) then
C            f_ibde = f_ibde + fz
C          else if ((id1.eq.10).and.(id3.eq.5)) then
C            f_ibde = f_ibde - fz
C
C          else if ((id1.eq.11).and.(id3.eq.0)) then
C            f_obde = f_obde + fz
C          else if ((id1.eq.11).and.(id3.eq.5)) then
C            f_obde = f_obde - fz
C
C          else if ((id1.eq.5).and.(id3.eq.0)) then
C            f_tail = f_tail + fz
C          else if ((id1.eq.5).and.(id3.eq.5)) then
C            f_tail = f_tail - fz
C
C          else if (((id1.eq.2).or.(id1.eq.3)).and.(id3.eq.0)) then
C            f_ail = f_ail + fz
C          else if(((id1.eq.2).or.(id1.eq.3)).and.(id3.eq.5))then
C           f_ail = f_ail - fz
C-----------------------------------------------------------------

C     ---If CAERO1 is wing or aileron then read geometry information
         if (((id1.eq.1).and.(id3.eq.0)).or.
     +      (((id1.eq.2).or.(id1.eq.3)).and.(id3.eq.0))) then
          n_pan = (n_span * n_chord)
          read (10,'(A132)',end=20) line
C        -----Check for empty lines/search for continuation card
             do while (line(30:31).ne.' + ')
               read (10,'(A132)',end=20) line
             end do
          read (line,'(46X,I8,24X,I8)')IbdLeY,ObdLeY
          read (line,'(62X,I8,24X,I8)')IbdChrd,ObdChrd
          read (line,'(38X,I8,24X,I8)')IbdLeX,ObdLeX
C        --Write macro panel number & geometry into tdmb for all subcases
          p_wfor = TDMB_REF (p_ops,'Operation')
          p_wfor = TDMB_REF_NEXT(p_wfor)
          do k = 1, subcasecounter
            p_wpan = TDMB_MAKE_NEXT ( p_wfor, 'WingLiftForces
     +                              MacroPanels PanelID')
C          ---Write panel information for all subcases
             idum = TDMB_PUT_INT4  ( p_wpan,'Name',pan_id)
             idum = TDMB_PUT_DOUBLE ( p_wpan,'IbdLeY',IbdLeY)
             idum = TDMB_PUT_DOUBLE ( p_wpan,'IbdLeX',IbdLeX)
             idum = TDMB_PUT_DOUBLE ( p_wpan,'IbdChord',IbdChrd)
             idum = TDMB_PUT_DOUBLE ( p_wpan,'ObdLeY',ObdLeY)
```

```
               idum = TDMB_PUT_DOUBLE ( p_wpan,'ObdLeX',ObdLeX)
               idum = TDMB_PUT_DOUBLE ( p_wpan,'ObdChord',ObdChrd)
               idum = TDMB_PUT_INT4 ( p_wpan,'NSpan',n_span)
               idum = TDMB_PUT_INT4 ( p_wpan,'NChord',n_chord)

C           ---Calculate subpanel position and write into tdmb
               subpan_id = pan_id
C           ---Calc subpan y-position & local chord at that position
               do i = 1 , (n_pan/n_chord)
                  y_subpan = (i *((ObdLeY-IbdLeY)/n_span))-
     +                 (0.5*((ObdLeY-IbdLeY)/n_span))+IbdLeY
                  localChrd = (IbdChrd+((0.5-i)*((IbdChrd-ObdChrd)/
     +                         n_span)))
                  w_subpan = (ObdLeY-IbdLeY)/n_span
C              ---Write subpanel id, calculate subpanel x-position
                  do j = 1 , (n_pan/n_span)
                    p_span =TDMB_MAKE_NEXT(p_wpan,'SubPanels
     +                                 SubpanelID')
                    idum = TDMB_PUT_INT4 (p_span,'Name',subpan_id)
                    alpha = tan((ObdLeX-IbdLeX)/(ObdLeY-IbdLeY))
                    localChrd_y = y_subpan
                    localChrd_x = atan(alpha)*(localChrd_y-IbdLeY)
                    x_subpan = (IbdLeX+localChrd_x + j *
     +                     (localChrd/n_chord) -
     +                      0.5 *(localChrd/n_chord))
C                 --write sub panel x-position
                    idum = TDMB_PUT_DOUBLE (p_span,'SubPanX',x_subpan)
C                 --write sub panel y-position
                    idum = TDMB_PUT_DOUBLE (p_span,'SubPanY',y_subpan)
C                 --write sub panel width
                    idum =TDMB_PUT_DOUBLE(p_span,'SubPanWidth',w_subpan)
C                 --write local-chord
                    idum = TDMB_PUT_DOUBLE (p_span,'locChrd',localChrd)
                    subpan_id = subpan_id +1
                  end do
               end do
               p_wfor = TDMB_REF_NEXT(p_wfor)
            end do
      end if
C-----Read CAERO2 cards for aero fuselage panel information
C     IF FUSELAGE DIMENSIONS CHANGE THIS SECTION NEEDS TO BE REVISED
      else if (line(30:37).eq.' CAERO2 ') THEN
        if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
          read(line,'(38X,I8,16X,I3)')pan_id,n_fpan
C       --Calculate panel x position, panel width y, surface S and write
C       --to TDMB for all subcases
          p_wfor = TDMB_REF (p_ops,'Operation')
          p_wfor = TDMB_REF_NEXT(p_wfor)
          f_length = TDMB_GET_DOUBLE(p_ac,'Specification
     +                          Fuselage Length')
          f_height = TDMB_GET_DOUBLE(p_ac,'Specification
     +                          Fuselage Height')
          FwdF1 = TDMB_GET_DOUBLE(p_ac,'Specification
     +                          Fuselage FwdFineness')
          FwdF = f_height* FwdF1
          AftF1 = TDMB_GET_DOUBLE(p_ac,'Specification
     +                          Fuselage AftFineness')
          AftF = f_length-(f_height*AftF1)
          Tailx = TDMB_GET_DOUBLE(p_ac,'Definition Geometry Tail
     +                           Planform RootLeX')
            do k = 1, subcasecounter
               x_fpan = -(0.5*(f_length/n_fpan))
               x_fpn = (f_length/n_fpan)
               pan_count =1
               do i = 1, n_fpan
                 x_fpan = x_fpan + (f_length/n_fpan)
C              --Determine width of the fuselage panel and the area S
                 If ((x_fpan.lt.FwdF).or.(x_fpan.eq.FwdF)) then
                  y_fpan = 2 * (((1 -(((FwdF - x_fpan)**2) /
     +                      (FwdF **2))) * 3.51 **2) **0.5)
                  S_fpan = (y_fpan * x_fpn)
                 end if
                 If ((x_fpan.gt.FwdF).and.(x_fpan.lt.AftF)) then
                  y_fpan = 7.02
                  S_fpan = y_fpan * x_fpn
                 end if
                 if ((x_fpan.gt.AftF).and.(x_fpan.lt.Tailx)) then
                  y_fpan = 7.020 - pan_count * ((7.02-4.00)/
     +                         ((Tailx-AftF)/x_fpn))
```

```
                 S_fpan = y_fpan * x_fpn
                 pan_count = pan_count + 1
                end if
                if ((x_fpan.gt.Tailx)) then
                 y_fpan = 4.00
                 S_fpan = y_fpan * x_fpn
                end if
                p_wpan = TDMB_MAKE_NEXT(p_wfor, 'WingLiftForces
     +                              FuselagePanels FusePanel')
                idum = TDMB_PUT_INT4 ( p_wpan,'Name',pan_id)
                idum = TDMB_PUT_DOUBLE ( p_wpan,'Xposition',x_fpan)
                idum = TDMB_PUT_DOUBLE ( p_wpan,'YWidth',y_fpan)
                idum = TDMB_PUT_DOUBLE ( p_wpan,'SPanel',S_fpan)
                pan_id = pan_id +1
              end do
              p_wfor = TDMB_REF_NEXT(p_wfor)
              pan_id = pan_id - n_fpan
           end do
C-----read and write TRIM card information
      else if (line(30:37).eq.' TRIM  ') THEN
        if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
          read(line,'(38X,I8,I8,I8)')trim_id,trim_mach,trim_dynp

C       --Read trim card id, mach number and dynamic pressure and write
C       --to TDMB for specific subcase
C       --Skip first trimcard as this is a flutter trim card
          if ((trim_id).gt.(1)) then
C         --pointer to subcase
            p_wfor = TDMB_REF (p_ops,'Operation')
              do j = 1 , (trimcounter)
                p_wfor = TDMB_REF_NEXT(p_wfor)
              end do
              idum = TDMB_PUT_DOUBLE(p_wfor,'WingLiftForces
     +              FlightConditions MachNo',trim_mach)
              idum = TDMB_PUT_DOUBLE(p_wfor,'WingLiftForces
     +              FlightConditions DynPress',trim_dynp)
              trimcounter = trimcounter + 1
          end if
C Still to do/Remarks
C
C 1 Engine Lift contribution: Done now so that it contributes to the
C   StripLoad at engine attachment point : Needs improvement : this gives
C   a weird picture as it comes in as peak load at the engine eta.
C   The EngineLift is a Choise in ControlData whether to include engine
C   lift like this or not
C 2 Tail plane : not incorporated yet
C-------------------------------------------------------------------
C-----------------------END SOL144 ADD ON---------------------------
C-------------------------------------------------------------------


C----------------------------------------------------------------------
C----------------------READ CFD RESULTS -------------------------------
C----------------------(by Gert-Jan Rollema)---------------------------
C----------------------------------------------------------------------
C--This part of the program does the following:
C      1)Extract aerodynamic results from a SPECIFIC file containing CFD
C        data generated by BAe Airbus Engineering Woodford
C
C------read fuselage data file if CFD = 1
       else if (CFD.eq.1) THEN
        write (*,*)'CFD data Reading'
        if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
         read(line,'(5X,e2.0,e11.0,e11.0,e10.0,e10.0,e10.0,e10.0)')
     +                  no1,no2,no3,no4,no5,no6,no7
         eta = (no2/7.0)
         t_wing = TDMB_REF ( p_stage,'FuselageSections')
         p_aerofuse = TDMB_MAKE_NEXT (t_wing,'FuselageSection')
         idum = TDMB_PUT_DOUBLE (p_aerofuse,'Name',no1)
         idum = TDMB_PUT_DOUBLE (P_aerofuse, 'Eta', eta)
         idum = TDMB_PUT_DOUBLE (p_aerofuse, 'Chord',no3)
         idum = TDMB_PUT_DOUBLE (p_aerofuse, 'Cl',no4)
         idum = TDMB_PUT_DOUBLE (p_aerofuse, 'CdViscous',no5)
         idum = TDMB_PUT_DOUBLE (p_aerofuse, 'ClCCbar',no6)
         idum = TDMB_PUT_DOUBLE (p_aerofuse, 'CdViscousCCbar',no7)

C------read wing data file if CFD = 2
       else if (CFD.eq.2) THEN
        write (*,*)'CFD data Reading'
        if ( sfile.eq.1 ) write (11,'(A)') line(1:lenstr(line))
```

```
      read(line,'(5X,e2.0,e11.0,e11.0,e10.0,e10.0,e10.0,e10.0)')
     +           no1,no2,no3,no4,no5,no6,no7
      eta = (no2/3.8)
      write (*,*)'eta',eta
      t_wing = TDMB_REF ( p_stage,'WingSections')
      p_aerofuse = TDMB_MAKE_NEXT (t_wing,'WingSection')
      no8 = no6/1.1507
      idum = TDMB_PUT_DOUBLE (p_aerofuse,'Name',no1)
      idum = TDMB_PUT_DOUBLE (P_aerofuse, 'Eta', eta)
      idum = TDMB_PUT_DOUBLE (p_aerofuse, 'Chord',no3)
      idum = TDMB_PUT_DOUBLE (p_aerofuse, 'Cl',no4)
      idum = TDMB_PUT_DOUBLE (p_aerofuse, 'CdViscous',no5)
      idum = TDMB_PUT_DOUBLE (p_aerofuse, 'ClCCbar',no8)
      idum = TDMB_PUT_DOUBLE (p_aerofuse, 'CdViscousCCbar',no7)

C-----------------------------------------------------------------------
C------------------------END CFD    ADD ON-----------------------------
C-----------------------------------------------------------------------
```

INFORMATION BETWEEN THESE LINES HAS BEEN INTENTIONALLY LEFT OUT

```
end
```

APPENDIX-I : FUSELAGE BULK DATA ENTRIES

```
$-----------------------------------------------------
$ FUSELAGE CAERO2 CARDS ADDED BY G.J. ROLLEMA
$
$--------------fuselage Aero model-----------------
$
CAERO2,99001000,9801000,,140,69,,,1111,+fus1
+fus1,0.,0.,0.,70.42
$
PAERO2,9801000,ZY,3.51,1.2108,9881000,,9882000,,+paer1
+paer1,1,69
$
$----Interference body divisions as part of fuselage length
AEFACT,9880001,0.0,0.005,0.010,0.015,0.02,0.025,0.03,+aef9880
+aef9880,0.035,0.04,0.045,0.05,0.0575,0.065,0.0725,0.080,+aef9881
+aef9881,0.0875,0.0925,0.10,0.11,0.120,0.13,0.14,0.15,+aef9882
+aef9882,0.16,0.18,0.20,0.22,0.24,0.26,0.27,0.28,+aef9884
+aef9884,0.29,0.295,0.30,0.305,0.31,0.32,0.33,0.34,+aef9885
+aef9885,0.36,0.38,0.40,0.42,0.44,0.46,0.48,0.49,+aef9886
+aef9886,0.50,0.51,0.505,0.51,0.515,0.52,0.525,0.53,+aef9887
+aef9887,0.535,0.540,0.545,0.550,0.560,0.570,0.58,0.59,+aef9888
+aef9888,0.61,0.62,0.65,0.67,0.69,0.71,0.72,0.73,+aef9890
+aef9890,0.74,0.75,0.76,0.77,0.78,0.79,0.80,0.81,+aef9891
+aef9891,0.82,0.825,0.83,0.835,0.84,0.845,0.85,0.86,+aef9892
+aef9892,0.87,0.87,0.89,0.90,0.91,0.92,0.93,0.94,+aef9893
+aef9893,0.95,0.955,0.96,0.965,0.97,0.975,0.98,0.99,+aef9894
+aef9894,1.00
$
$----Slender Body Half-Widths---------------------
AEFACT,9881000,0.0,0.94,1.320,1.60,1.83,2.03,2.19,+aef5
+aef5,2.35,2.49,2.60,2.72,2.82,2.91,2.99,3.07,+aef6
+aef6,3.14,3.20,3.25,3.30,3.35,3.38,3.42,3.45,+aef7
+aef7,3.467,3.48,3.49,3.506,3.509,3.51,3.51,3.51,+aef8
+aef8,3.51,3.51,3.51,3.51,3.51,3.51,3.51,3.51,+aef9
+aef9,3.51,3.51,3.51,3.51,3.51,3.51,3.51,3.51,+aef10
+aef10,3.51,3.51,3.51,3.51,3.51,3.51,3.51,3.51,+aef11
+aef11,3.51,3.51,3.51,3.51,3.51,3.51,3.51,3.51,+aef12
+aef12,3.51,3.51,3.51,3.51,3.51,3.51,3.51,3.51,+aef13
+aef13,3.51,3.51,3.51,3.51,3.51,3.51,3.51,3.51,+aef14
+aef14,3.51,3.51,3.51,3.51,3.51,3.51,3.51,3.51,+aef15
+aef15,3.45,3.40,3.35,3.30,3.25,3.20,3.15,3.10,+aef16
+aef16,3.05,3.00,2.95,2.90,2.85,2.80,2.75,2.70,+aef17
+aef17,2.65,2.60,2.55,2.50,2.45,2.40,2.35,2.30,+aef18
+aef18,2.25,2.20,2.15,2.10,2.05,2.00,2.00,2.00,+aef19
+aef19,2.00,2.00,2.00,2.00,2.00,2.00,2.00,2.00,+aef20
+aef20,2.00,2.00,2.00,2.00,2.00,2.00,2.00,2.00,+aef21
+aef21,2.00,2.00,1.50,0.1.00.50,0.00
$
$----THETA ARRAYS---------------------------------
AEFACT,9882000,5.,15.,29.,50.,70.,90.,110.,+aef1
+aef1,131.,151.,165.,175.,190.,210.,230.,240.,+aef2
+aef2,255.,270.,285.,300.,310.,330.,350.
$
$----SPLINES------------------------------------
SPLINE2,9891000,99001000,99001000,99001139,9990000,,1.0,,+spl2
+spl2,,-1.0
$---------------------------------------------
SET1,9990000,901000,902000,903000,904000,905000,906000,907000,+st1
+st1,908000,909000,910000,911000,912000,913000,914000,915000,+st2
+st2,916000,917000,918000,919000,920000,921000,922000,923000,+st3
+st3,924000,925000,926000,927000,928000,929000,930000,931000
```

APPENDIX J : PART OF CFD RESULT FILE

```
MB__01_1 Section  1 on surface : FUSE
  1   21    6
      X          Y            Z          CP          M_ISEN
     S/C
  -0.100000   0.000000   -0.037590   0.297394    0.582869
   0.000000
  -0.100000   0.014396   -0.035531   0.321284    0.573102
   0.132309
  -0.100000   0.023050   -0.032884   0.318092    0.574411
   0.211841
  -0.100000   0.047097   -0.021633   0.326616    0.570912
   0.432848
  -0.100000   0.070322   -0.002618   0.305903    0.579398
   0.646290
  -0.100000   0.082432    0.013023   0.288490    0.586491
   0.757589
  -0.100000   0.094721    0.034417   0.289989    0.585882
   0.870529
  -0.100000   0.105547    0.067191   0.292381    0.584909
   0.970030
  -0.100000   0.108134    0.078250   0.275276    0.591849
   0.993804
  -0.100000   0.108660    0.081487   0.271641    0.593319
   0.998641
  -0.100000   0.108660    0.081487   0.271641    0.593319
   0.998641
  -0.100000   0.108808    0.085789   0.277266    0.591044
   1.000000
  -0.100000   0.108459    0.097379   0.295988    0.583441
   0.996794
  -0.100000   0.104737    0.132390   0.283302    0.588597
   0.962583
  -0.100000   0.094280    0.173517   0.283312    0.588593
   0.866480
  -0.100000   0.092556    0.177704   0.286090    0.587466
   0.850634
  -0.100000   0.067216    0.217470   0.363286    0.555745
   0.617746
  -0.100000   0.055970    0.230155   0.381005    0.548347
   0.514387
  -0.100000   0.017682    0.252197   0.369006    0.553362
   0.162507
  -0.100000   0.012596    0.253678   0.360895    0.556740
   0.115759
  -0.100000   0.000000    0.256349   0.351737    0.560542
```

APPENDIX K : RESULTS FROM CFD POST PROCESSOR

```
**************************************************************************************************
     FFFFF     1      666      1      AAA      888
     F        11     6        11     A   A    8   8       MULTIBLOCK  EULER/NS  GRID/FLOW-FIELD  POST-PROCESSOR
     F         1     6         1     A   A    8   8       ---------------------------------------------
     FFFF      1     6666      1     AAAAA     888        Contacts:
     F         1     6   6     1     A   A    8   8          J.Benton, S.Sheard, BAe Airbus, Woodford
     F         1     6   6     1     A   A    8   8          K.Weatherill        Aerodynamics Dept, BAe Warton
     F        111     666     111    A   A     888     Version 2.8   Jul-97
**************************************************************************************************

 Flow calculation method  : RANSMB / BAe. MULTIBLOCK
 ---------------------------------------------------------------------------
 FLOW DUMP INPUT FILE DATA:
   File format              = .FLO  unformatted
   File name                =
   Program name             = ransmb
   Program version          = 5.4      Jan-98
   Program precision        = Single, with double precision grid
   Single precision         = 1.0E-15
   Double precision         = 1.0E-15
   Run start date           = 09/26/98
   Run start time           = 15:17:35
   Run title                = ATTACH Al=1.0 Recdbar 80M Fully turbulent/NS kg
   Run type                 = Navier-Stokes, turbulent
   Number of blocks         =    219
   Number of cells          =  792256
   Cycles                   =    500
   Infinity conditions:
     Mach number            =  0.70000
     Incidence (alpha)(deg) =   1.0000
     Sideslip   (beta )(deg)=  0.00000E+00
     Temperature (K)        =   288.00
     Reynolds number        =  0.80000E+08
                            = rhoref*Qref*L/muref
     Turb.intensity/Qref    =  0.00000E+00
     see Reference values below for definitions of rhoref, Qref, muref and L
   Other flow properties:
     Gamma                  =   1.4000
     Viscosity law          = SUTHERLAND
     Prandtl number         =  0.72000
     Turbulence model       = k-g low-Re Kalitzin-Gould
     Turbulent Prandtl no.  =  0.90000
   Reference values, used for Reynolds number, and
   non-dimensionalisation of turbulence intensity input:
     Reference length L (grid units)=   1.1507
     REF_FLOW flag                  = V_INF
     so reference flow (rhoref, Qref, muref) is
     infinity flow with Qref = flow speed (Qinf).
 ---------------------------------------------------------------------------

 SECTIONS : INTEGRATION OF SECTIONAL FORCES : MB
 ----------------------------------------
   Surface names                   : FUSE
   Reference chord cbar            :   1.0000
   Reference lengths for X,Y,Z moments:   1.0000       1.0000       1.0000
   Moment reference point X,Y,Z coord :  0.00000E+00  0.00000E+00  0.00000E+00
   Number of sections requested    :   72
   Number of sections integrated   :   54
   Note: only continuous sections integrated.
   CX,CY,CZ,CMX,CMY,CMZ are based on grid X,Y,Z axes,
   CL,CD are based on free-stream Xf,Yf,Zf axes.
   Due to pressure:
```

J	X	chord	CL	CD	CLc/cbar	CDc/cbar	CX	CY	CZ	CMX	CMY	CMZ
1	-0.10000	0.10881	-0.03864	-0.00067	-0.00420	-0.00007	0.00000	-0.83086	-0.03865	-0.09278	-0.00387	-0.08309
2	0.00000	0.15569	-0.07169	-0.00125	-0.01116	-0.00019	0.00000	-0.33225	-0.07170	-0.04226	0.00000	0.00000
3	0.10000	0.19216	-0.07729	-0.00135	-0.01485	-0.00026	0.00000	-0.03481	-0.07731	-0.00946	0.00773	0.00348
4	0.20000	0.22055	-0.06693	-0.00117	-0.01476	-0.00026	0.00000	0.15384	-0.06694	0.01502	0.01339	-0.03077
5	0.30000	0.24509	-0.04379	-0.00076	-0.01073	-0.00019	0.00000	0.25961	-0.04380	0.03265	0.01314	-0.07788
6	0.40000	0.26389	-0.01792	-0.00031	-0.00473	-0.00008	0.00000	0.32076	-0.01792	0.04593	0.00717	-0.12830
7	0.50000	0.28141	0.01201	0.00021	0.00338	0.00006	0.00000	0.34714	0.01201	0.05475	-0.00600	-0.17357
8	0.60000	0.29636	0.03343	0.00058	0.00991	0.00017	0.00000	0.35932	0.03343	0.06057	-0.02006	-0.21559
9	0.70000	0.30998	0.05405	0.00094	0.01675	0.00029	0.00000	0.36611	0.05405	0.06529	-0.03784	-0.25627
10	0.80000	0.32159	0.07334	0.00128	0.02358	0.00041	0.00000	0.36745	0.07335	0.06847	-0.05868	-0.29396
11	0.90000	0.33130	0.09153	0.00160	0.03032	0.00053	0.00000	0.36374	0.09155	0.07014	-0.08239	-0.32736
12	1.00000	0.33971	0.10876	0.00190	0.03695	0.00064	0.00000	0.35472	0.10878	0.07027	-0.10878	-0.35472
13	1.10000	0.34589	0.12754	0.00223	0.04412	0.00077	0.00000	0.35390	0.12756	0.07137	-0.14032	-0.38929
14	1.20000	0.34855	0.12521	0.00219	0.04364	0.00076	0.00000	0.27252	0.12523	0.05657	-0.15027	-0.32703
15	1.30000	0.34858	0.11777	0.00206	0.04105	0.00072	0.00000	0.09297	0.11779	0.02449	-0.15313	-0.12086
16	1.40000	0.34849	0.15129	0.00264	0.05272	0.00092	0.00000	-0.01426	0.15132	0.00779	-0.21184	0.01997
17	1.50000	0.34850	0.21244	0.00371	0.07403	0.00129	0.00000	-0.18037	0.21247	-0.01762	-0.31871	0.27055

18	1.60000	0.35047	0.26929	0.00470	0.09438	0.00165	0.00000	-0.09713	0.26933	0.00174	-0.43093	0.15541
19	1.70000	0.35035	0.23441	0.00409	0.08213	0.00143	0.00000	-0.13841	0.23445	-0.00298	-0.39856	0.23529
38	3.60000	0.34976	0.03045	0.00053	0.01065	0.00019	0.00000	0.15341	0.03046	0.03849	-0.10964	-0.55227
39	3.70000	0.35011	0.03260	0.00057	0.01141	0.00020	0.00000	0.16655	0.03260	0.02719	-0.12063	-0.61623
40	3.80000	0.35046	0.03281	0.00057	0.01150	0.00020	0.00000	0.15688	0.03282	0.02140	-0.12471	-0.59614
41	3.90000	0.35092	0.05061	0.00088	0.01776	0.00031	0.00000	0.15098	0.05062	0.02026	-0.19740	-0.58882
42	4.00000	0.35044	0.08717	0.00152	0.03055	0.00053	0.00000	0.09810	0.08718	0.01540	-0.34873	-0.39241
43	4.10000	0.34998	0.13506	0.00236	0.04727	0.00083	0.00000	0.04673	0.13508	0.01108	-0.55382	-0.19161
44	4.20000	0.35038	0.16221	0.00283	0.05684	0.00099	0.00000	0.00723	0.16224	0.00891	-0.68139	-0.03037
45	4.30000	0.34996	0.13070	0.00228	0.04574	0.00080	0.00000	0.02043	0.13072	0.01186	-0.56209	-0.08786
46	4.40000	0.34980	0.07071	0.00123	0.02473	0.00043	0.00000	0.08926	0.07072	0.02070	-0.31115	-0.39276
47	4.50000	0.34873	0.02865	0.00050	0.00999	0.00017	0.00000	0.15942	0.02866	0.03079	-0.12897	-0.71739
48	4.60000	0.34734	0.01257	0.00022	0.00437	0.00008	0.00000	0.21425	0.01257	0.04018	-0.05782	-0.98556
49	4.70000	0.34366	0.00728	0.00013	0.00250	0.00004	0.00000	0.26325	0.00728	0.04981	-0.03423	-1.23727
50	4.80000	0.33863	0.01505	0.00026	0.00510	0.00009	0.00000	0.29685	0.01505	0.05793	-0.07225	-1.42487
51	4.90000	0.33157	0.03007	0.00052	0.00997	0.00017	0.00000	0.31306	0.03008	0.06372	-0.14738	-1.53400
52	5.00000	0.32208	0.04594	0.00080	0.01480	0.00026	0.00000	0.30584	0.04594	0.06522	-0.22972	-1.52922
53	5.10000	0.31029	0.05930	0.00104	0.01840	0.00032	0.00000	0.27985	0.05930	0.06273	-0.30245	-1.42724
54	5.20000	0.29624	0.06802	0.00119	0.02015	0.00035	0.00000	0.23456	0.06803	0.05557	-0.35377	-1.21974
55	5.30000	0.28000	0.07080	0.00124	0.01982	0.00035	0.00000	0.16799	0.07081	0.04284	-0.37531	-0.89033
56	5.40000	0.26191	0.06661	0.00116	0.01744	0.00030	0.00000	0.07753	0.06662	0.02343	-0.35973	-0.41866
57	5.50000	0.24216	0.05556	0.00097	0.01346	0.00023	0.00000	-0.03488	0.05557	-0.00269	-0.30565	0.19186
58	5.60000	0.22257	0.04303	0.00075	0.00958	0.00017	0.00000	-0.14359	0.04304	-0.02945	-0.24100	0.80411
59	5.70000	0.20497	0.03180	0.00056	0.00652	0.00011	0.00000	-0.22655	0.03181	-0.05123	-0.18131	1.29135
60	5.80000	0.19030	0.02392	0.00042	0.00455	0.00008	0.00000	-0.26878	0.02392	-0.06362	-0.13873	1.55890
61	5.90000	0.17896	0.02014	0.00035	0.00360	0.00006	0.00000	-0.26274	0.02014	-0.06400	-0.11882	1.55019
62	6.00000	0.17117	0.02103	0.00037	0.00360	0.00006	0.00000	-0.20486	0.02103	-0.05082	-0.12621	1.22913
63	6.10000	0.16438	0.02424	0.00042	0.00399	0.00007	0.00000	-0.13906	0.02425	-0.03496	-0.14791	0.84825
64	6.20000	0.15666	0.02745	0.00048	0.00430	0.00008	0.00000	-0.09227	0.02745	-0.02344	-0.17019	0.57206
65	6.30000	0.14713	0.02745	0.00048	0.00404	0.00007	0.00000	-0.07932	0.02745	-0.02062	-0.17296	0.49974
66	6.40000	0.13517	0.01882	0.00033	0.00254	0.00004	0.00000	-0.11329	0.01883	-0.03058	-0.12049	0.72504
67	6.50000	0.11995	0.00882	0.00015	0.00106	0.00002	0.00000	-0.15010	0.00882	-0.04137	-0.05736	0.97564
68	6.60000	0.10255	-0.00347	-0.00006	-0.00036	-0.00001	0.00000	-0.18949	-0.00347	-0.05293	0.02289	1.25062
69	6.70000	0.08357	-0.01519	-0.00027	-0.00127	-0.00002	0.00000	-0.22710	-0.01519	-0.06398	0.10176	1.52157
70	6.80000	0.06263	-0.02422	-0.00042	-0.00152	-0.00003	0.00000	-0.26146	-0.02422	-0.07412	0.16471	1.77794
71	6.90000	0.03905	-0.02899	-0.00051	-0.00113	-0.00002	0.00000	-0.30484	-0.02900	-0.08698	0.20007	2.10338
72	7.00000	0.01307	-0.02592	-0.00045	-0.00034	-0.00001	0.00000	-0.49101	-0.02593	-0.14099	0.18148	3.43709

APPENDIX L : DMI ENTRIES FOR TWIST

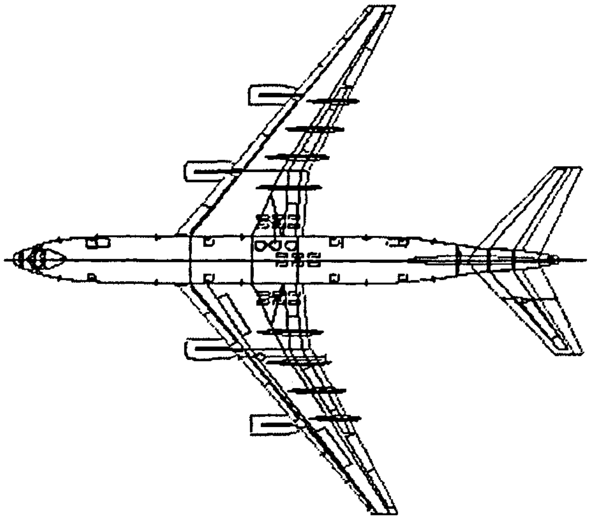
```
DMI,W2GJ,0,2,1,0,,523,3

DMI,W2GJ,1,1,0.06545,THRU,16,17,0.05725,+DMI1
+DMI1,THRU,32,33,0.04904,THRU,48,49,0.04102,+DMI2
+DMI2,THRU,64,65,0.03281,THRU,80,81,0.02461,+DMI3
+DMI3,THRU,96,97,0.01641,THRU,110,111,0.01221,+DMI4
+DMI4,THRU,122,123,0.0082,THRU,134,135,0.0041,+DMI5
+DMI5,THRU,146,147,0.0001,THRU,158,159,-0.0042,+DMI6
+DMI6,THRU,170,171,-0.0084,THRU,182,183,-0.01257,+DMI7
+DMI7,THRU,194,195,-0.0157,THRU,203,204,-0.00189,+DMI8
+DMI8,THRU,212,213,-0.0222,THRU,221,222,-0.02531,+DMI9
+DMI9,THRU,230,231,-0.02757,THRU,239,240,-0.02987,+DMI10
+DMI10,THRU,248,249,-0.03194,THRU,257,258,-0.03421,+DMI11
+DMI11,THRU,266,267,-0.03647,THRU,275,276,-0.03857,+DMI12
+DMI12,THRU,284,285,-0.0408,THRU,293,294,-0.0431,+DMI13
+DMI13,THRU,301,302,-0.0157,THRU,304,305,-0.0189,+DMI14
+DMI14,THRU,307,308,-0.0222,THRU,310,311,-0.02531,+DMI15
+DMI15,THRU,313,314,-0.02757,THRU,316,317,-0.02987,+DMI16
+DMI16,THRU,319,320,-0.03194,THRU,322,323,-0.03421,+DMI17
+DMI17,THRU,325,326,-0.03647,THRU,328,329,-0.03857,+DMI18
+DMI18,THRU,331,332,-0.0408,THRU,334

DMI,W2GJ,2,1,0.06545,THRU,16,17,0.05725,+DMI21
+DMI21,THRU,32,33,0.04904,THRU,48,49,0.04102,+DMI22
+DMI22,THRU,64,65,0.03281,THRU,80,81,0.02461,+DMI32
+DMI32,THRU,96,97,0.01641,THRU,110,111,0.01221,+DMI42
+DMI42,THRU,122,123,0.0082,THRU,134,135,0.0041,+DMI52
+DMI52,THRU,146,147,0.0001,THRU,158,159,-0.0042,+DMI62
+DMI62,THRU,170,171,-0.0084,THRU,182,183,-0.01257,+DMI72
+DMI72,THRU,194,195,-0.0157,THRU,203,204,-0.00189,+DMI82
+DMI82,THRU,212,213,-0.0222,THRU,221,222,-0.02531,+DMI92
+DMI92,THRU,230,231,-0.02757,THRU,239,240,-0.02987,+DMI102
+DMI102,THRU,248,249,-0.03194,THRU,257,258,-0.03421,+DMI112
+DMI112,THRU,266,267,-0.03647,THRU,275,276,-0.03857,+DMI122
+DMI122,THRU,284,285,-0.0408,THRU,293,294,-0.0431,+DMI132
+DMI132,THRU,301,302,-0.0157,THRU,304,305,-0.00189,+DMI142
+DMI142,THRU,307,308,-0.0222,THRU,310,311,-0.02531,+DMI152
+DMI152,THRU,313,314,-0.02757,THRU,316,317,-0.02987,+DMI162
+DMI162,THRU,319,320,-0.03194,THRU,322,323,-0.03421,+DMI172
+DMI172,THRU,325,326,-0.03647,THRU,328,329,-0.03857,+DMI182
+DMI182,THRU,331,332,-0.0408,THRU,334

DMI,W2GJ,3,1,0.06545,THRU,16,17,0.05725,+DMI31
+DMI31,THRU,32,33,0.04904,THRU,48,49,0.04102,+DMI23
+DMI23,THRU,64,65,0.03281,THRU,80,81,0.02461,+DMI33
+DMI33,THRU,96,97,0.01641,THRU,110,111,0.01221,+DMI43
+DMI43,THRU,122,123,0.0082,THRU,134,135,0.0041,+DMI53
+DMI53,THRU,146,147,0.0001,THRU,158,159,-0.0042,+DMI63
+DMI63,THRU,170,171,-0.0084,THRU,182,183,-0.01257,+DMI73
+DMI73,THRU,194,195,-0.0157,THRU,203,204,-0.00189,+DMI83
+DMI83,THRU,212,213,-0.0222,THRU,221,222,-0.02531,+DMI93
+DMI93,THRU,230,231,-0.02757,THRU,239,240,-0.02987,+DMI103
+DMI103,THRU,248,249,-0.03194,THRU,257,258,-0.03421,+DMI113
+DMI113,THRU,266,267,-0.03647,THRU,275,276,-0.03857,+DMI123
+DMI123,THRU,284,285,-0.0408,THRU,293,294,-0.0431,+DMI133
+DMI133,THRU,301,302,-0.0157,THRU,304,305,-0.00189,+DMI143
+DMI143,THRU,307,308,-0.0222,THRU,310,311,-0.02531,+DMI153
+DMI153,THRU,313,314,-0.02757,THRU,316,317,-0.02987,+DMI163
+DMI163,THRU,319,320,-0.03194,THRU,322,323,-0.03421,+DMI173
+DMI173,THRU,325,326,-0.03647,THRU,328,329,-0.03857,+DMI183
+DMI183,THRU,331,332,-0.0408,THRU,334
```

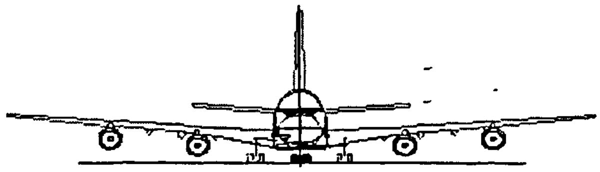
APPENDIX M : DMI ENTRIES FOR CAMBER & TWIST

```
DMI,W2GJ,0,2,1,0,,523,3

DMI,W2GJ,1,1, 0.0021, 0.0124, 0.0227, 0.0330, 0.0330,+DMI1
+DMI1 , 0.0443, 0.0536, 0.0639, 0.0639, 0.0742, 0.0845, 0.0948, 0.0948,+DMI2
+DMI2 , 0.11050 0.1153, 0.1256, 0.0024, 0.0118, 0.0212, 0.0307, 0.0307,+DMI3
+DMI3 , 0.0401, 0.0495, 0.0589, 0.0589, 0.0684, 0.0778, 0.0872, 0.0872,+DMI4
+DMI4 , 0.0966, 0.1061, 0.1155, 0.0142, 0.0209, 0.0275, 0.0341, 0.0341,+DMI5
+DMI5 , 0.0408, 0.0474, 0.0540, 0.0540, 0.0606, 0.0673, 0.0739, 0.0793,+DMI6
+DMI6 , 0.0805, 0.0872, 0.0938, 0.0268, 0.0305, 0.0342, 0.0379, 0.0379,+DMI7
+DMI7 , 0.0416, 0.0454, 0.0491, 0.0491, 0.0528, 0.0565, 0.0603, 0.0603,+DMI8
+DMI8 , 0.0640, 0.0677, 0.0714, 0.0351, 0.0366, 0.0381, 0.0396, 0.0396,+DMI9
+DMI9 , 0.0411, 0.0427, 0.0442, 0.0442, 0.0457, 0.0472, 0.0487, 0.0487,+DMI10
+DMI10, 0.0502, 0.0517, 0.0532, 0.0309, 0.0323, 0.0337, 0.0351, 0.0351,+DMI11
+DMI11, 0.0365, 0.0379, 0.0393, 0.0393, 0.0406, 0.0420, 0.0434, 0.0434,+DMI12
+DMI12, 0.0448, 0.0462, 0.0476, 0.0213, 0.0237, 0.0260, 0.0283, 0.0283,+DMI13
+DMI13, 0.0307, 0.0330, 0.0353, 0.0376, 0.0479, 0.0479, 0.0582, 0.0685,+DMI14
+DMI14, 0.0788, 0.0170, 0.0202, 0.0233, 0.0264, 0.0296, 0.0327, 0.0327,+DMI15
+DMI15, 0.0359, 0.0390, 0.0421, 0.0453, 0.0485, 0.0119, 0.0150, 0.0182,+DMI16
+DMI16, 0.0213, 0.0244, 0.0276, 0.0276, 0.0307, 0.0339, 0.0370, 0.0401,+DMI17
+DMI17, 0.0433, 0.0067, 0.0099, 0.0130, 0.0161, 0.0193, 0.0224, 0.0224,+DMI18
+DMI18, 0.0256, 0.0287, 0.0319, 0.0350, 0.0381, 0.0016, 0.0047, 0.0079,+DMI19
+DMI19, 0.0110, 0.0141, 0.0173, 0.0173, 0.0204, 0.0236, 0.0267, 0.0298,+DMI20
+DMI20, 0.0330,-0.0036,-0.0004, 0.0027, 0.0059, 0.0090, 0.0121, 0.0121,+DMI21
+DMI21, 0.0153, 0.0184, 0.0216, 0.0247, 0.0278,-0.0087,-0.0056,-0.0024,+DMI22
+DMI22, 0.0007, 0.0038, 0.0070, 0.0070, 0.0101, 0.0133, 0.0164, 0.0196,+DMI2
+DMI23, 0.0227,-0.0174,-0.0135,-0.0098,-0.0058,-0.0020, 0.0018, 0.0018,+DMI24
+DMI24, 0.0057, 0.0095, 0.0134, 0.0172,-0.0261,-0.0206,-0.0149,-0.0093,+DMI25
+DMI25,-0.0037, 0.0018, 0.0074, 0.0074, 0.0130, 0.0186,-0.0502,-0.0414,+DMI26
+DMI26,-0.0326,-0.0238,-0.0150,-0.0062,-0.0062, 0.0025, 0.0113,-0.0543,+DMI27
+DMI27,-0.0455,-0.0367,-0.0279,-0.0191,-0.0103,-0.0103, 0.0021, 0.0109,+DMI28
+DMI28,-0.0583,-0.0495,-0.0372,-0.0248,-0.0125,-0.0001,-0.0001, 0.0123,+DMI29
+DMI29, 0.0246,-0.0617,-0.0529,-0.0441,-0.0353,-0.0256,-0.0177,-0.0177,+DMI30
+DMI30,-0.0089,-0.0001,-0.0642,-0.0554,-0.0466,-0.0378,-0.0290,-0.0202,+DMI31
+DMI31,-0.0202,-0.0114,-0.0027,-0.0668,-0.0580,-0.0492,-0.0404,-0.0316,+DMI32
+DMI32,-0.0228,-0.0228,-0.0140,-0.0052,-0.0693,-0.0605,-0.0517,-0.0429,+DMI35
+DMI35,-0.0342,-0.0254,-0.0254,-0.0130,-0.0006,-0.0719,-0.0631,-0.0543,+DMI36
+DMI36,-0.0455,-0.0367,-0.0279,-0.0279,-0.0191,-0.0103,-0.0745,-0.0657,+DMI37
+DMI37,-0.0569,-0.0481,-0.0393,-0.0305,-0.0305,-0.0217, 0.0129,-0.0749,+DMI38
+DMI38,-0.0665,-0.0582,-0.0498,-0.0414,-0.0330,-0.0330,-0.0247,-0.0163,+DMI39
+DMI39,-0.0637,-0.0581,-0.0525,-0.0469,-0.0414,-0.0358,-0.0302,-0.0246,+DMI40
+DMI40, 0.0242, 0.0298, 0.0354, 0.0113, 0.0201, 0.0377, 0.0109, 0.0287,+DMI41
+DMI41, 0.0372, 0.0370, 0.0493, 0.0617, 0.0087, 0.0175, 0.0263, 0.0061,+DMI42
+DMI42, 0.0149, 0.0237, 0.0036, 0.0124, 0.0212, 0.0117, 0.0241, 0.0364,+DMI43
+DMI43,-0.0015, 0.00073 0.0161,-0.0041, 0.0047, 0.0135,-0.0079, 0.0005,+DMI44
+DMI44, 0.0089

DMI,W2GJ,2,1, 0.0021, 0.0124, 0.0227, 0.0330, 0.0330,+DM21
+DM21 , 0.0443, 0.0536, 0.0639, 0.0639, 0.0742, 0.0845, 0.0948, 0.0948,+DM22
+
+ etc. same as above
+
+DM243,-0.0015, 0.00073 0.0161,-0.0041, 0.0047, 0.0135,-0.0079, 0.0005,+DM244
+DM244, 0.0089


DMI,W2GJ,3,1, 0.0021, 0.0124, 0.0227, 0.0330, 0.0330,+D3I1
+D3I1 , 0.0443, 0.0536, 0.0639, 0.0639, 0.0742, 0.0845, 0.0948, 0.0948,+D3I2
+
+ etc. same as above
+
+D3I43,-0.0015, 0.00073 0.0161,-0.0041, 0.0047, 0.0135,-0.0079, 0.0005,+D3I44
+D3I44, 0.0089
```

APPENDIX-N: REFERENCE AIRCRAFT SPECIFICATION [REF. 2]

The reference aircraft selected for the project is representative of a 650 seat passenger aircraft, with a wing span of nearly 80m and a maximum take-off weight of about 550.000 kg.





Wing Area	725m^2
Wing Aspect-Ratio	8.2
Wing Span	77.1m
Wing Sweep	33°
Fuselage Length	70.4m
Max Take-Off Weight	551,000Kg
Engines	4 @ 310,000 N
Passenger capacity	650 mixed class

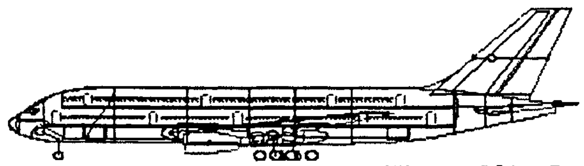


Figure N1 : *Reference Aircraf*

REFERENCE AXES AND UNITS

The reference aircraft is defined using the SI metric system of units :

Length	Metres	m
Mass	Kilograms	kg
Time	Seconds	s

Table N1 : *SI Units*

All technical data associated with the aircraft (and exchanged between partners) is defined using these units (or non-dimensionalised where appropriate). All numeric is exchanged using these or directly derived standard units (e.g. Pascal Pa) so that computer programmes accessing the data need no conditional conversion of units; however tabular, diagrammatic or graphical presentations of the data can use MPa etc. for clarity of interpretation.

A common set of reference axes is adopted for the definition of all geometric data associated with the aircraft (and exchanged between partners). All numeric data is exchanged using this axis system with lengths in metres, however local axes systems may be defined and used in diagrammatic and graphical presentation of geometry data (or of course internally within proprietary analysis methods).

x-direction	aft	0 on the aircraft nose
y-direction	starboard	0 on the plane of symmetry
z-direction	vertical	0 on the fuselage centreline

Table N2 : *Reference Axis*

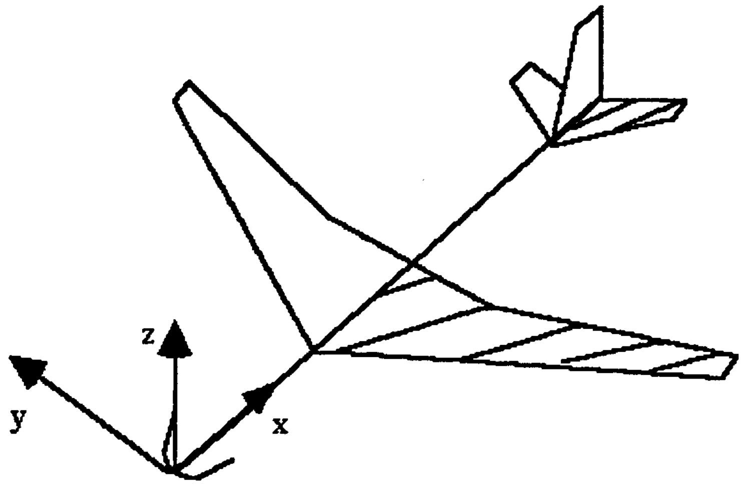


Figure N2: *Reference Axes*

GENERAL ARRANGEMENT

The following figures and tables illustrate the definition of the leading dimensional parameters defining the reference aircraft. Redundant specification of the same information is avoided in this specification, so for example wing area and aspect-ratio are included but span and geometric-mean-chord are not included.

Wing GA

The following figures and tables illustrate the definition of parameters defining the planform and vertical form of the wing :

Wing Aspect-Ratio	8.2	Ratio of wing span to geometric mean chord.
Wing Area	725m^2	Area of wing-pair including parallel centre-section
Wing QCSweep	33.0°	Quarter-Chord sweep of outboard wing
Wing TipTaper	0.22103	Ratio of wing tip chord to wing root chord
Wing CrankTaper	0.610606	Ratio of wing crank chord to wing root chord
Wing Root-Y	3.51m	Spanwise position (m) of wing root
Wing Crank-Eta	0.35	Position of crank as fraction of overall wing semi-span
Wing-AMC-Pos	0.452487	Position of wing quarter-chord aerodynamic mean chord as fraction of fuselage length

Tabel N3 : *Wing Data*

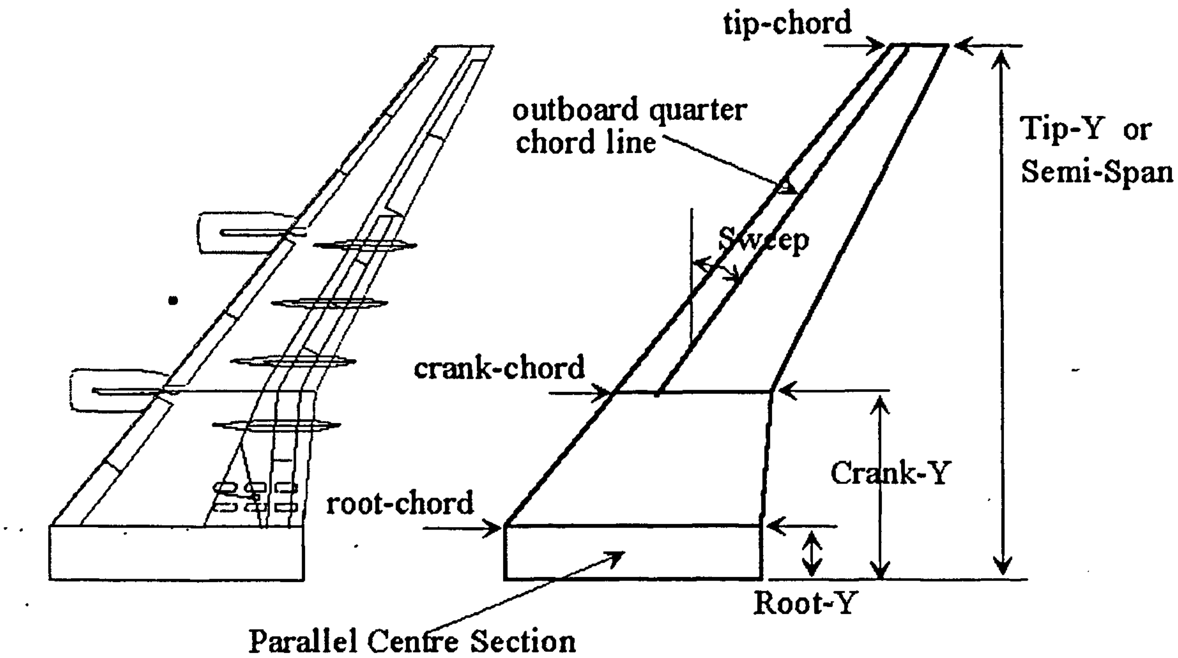


Figure N3 : *Wing Top View*

The definition of wing area and aerodynamic mean chord is detailed in section 3.1.1, together with the methodology for deriving all wing planform ordinates. The sweep of the inboard wing is determined by the straight leading edge. The wing is positioned longitudinally on the fuselage by locating the quarter chord point of the aerodynamic mean chord at the position along the fuselage defined by *Wing-AMC-Pos.*

The following figure and table show the front elevation of the aircraft, and illustrate the definition of parameters defining the wing in the vertical direction :-

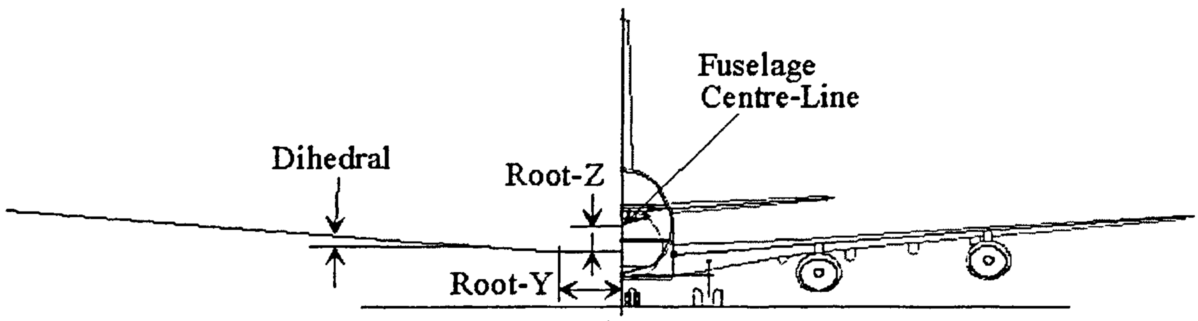


Figure N4 : *Airplane Front View*

Wing Root-Z	-2.6m	Height of the wing root leading edge point above the centre-line of the aircraft
Wing Dihedral	5°	Dihedral of wing reference plane (outboard of root)
Wing Root-Tc	0.14	Thickness/chord ratio at root
Wing Crank-Tc	0.1	Thickness/chord ratio at crank
Wing Tip-Tc	0.088	Thickness/chord ratio at tip

Table N4 : *Wing Data*

Engine GA

The following figures and tables illustrate the definition of parameters defining the engines and their location relative to the wing :

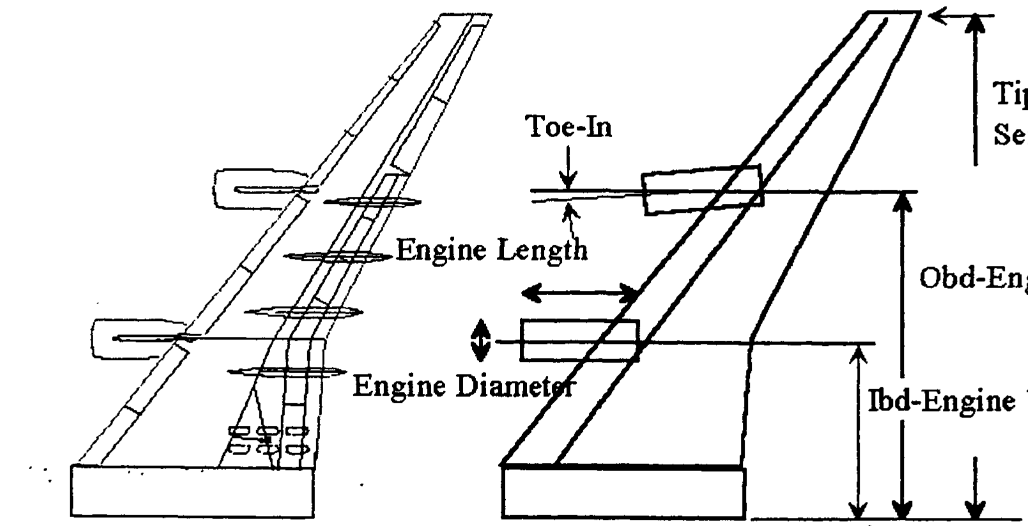


Figure N11 : *Engine Position*

Engine Diameter	3.075m	Nominal diameter of engine
Engine Length	7.15m	Nominal length of engine
Engine Toe-In	0°	Toe in of engine in planform view
Engine Etas	0.35 & 0.5575	Spanwise position of inboard engine as fraction of overall wing semi-span (eg IbdEngineY/Semispan)

Table 8: *Engine Data*

Memorandum 865