DLR-IB-AE-GO-2020-136

Loads Kernel User Guide
Version 1.01

Arne Voß

Deutsches Zentrum
DLR für Luft- und Raumfahrt

Dokumenteigenschaften

|	
Titel	Loads Kernel User Guide
Betreff	
Institut	DLR Institut für Aeroelastik
Erstellt von	Dr.-Ing. Arne Voß
Beteiligte	
Geprüft von	Dr.-Ing. Markus Ritter
Freigabe von	Prof. Dr.-Ing. Wolf-Reiner Krüger
Datum	7. October 2020
Version	1.0
Dateipfad	/work/voss_ar/Projekte/Loads Kernel/LoadsKernel_UserGuide_IB.fodt

Dokument-Historie

Version	Datum	Änderungen	Autor
1.00	07.10.2020	Version published as report DLR-IB-AE-GO-2020-136	A. Voß
1.01	11.01.2021	Typing mistake corrected in section 2.9 Stability Analysis, equation (2.69).	A. Voß

Summary
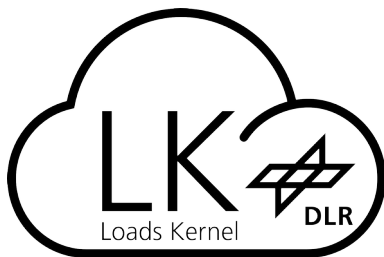
The Loads Kernel Software allows for the calculation of quasi-steady and dynamic maneuver loads, unsteady gust loads in the time and frequency domain as well as dynamic landing loads based on a generic landing gear module.

This report is a published copy of the Loads Kernel User Guide, Version 1.01 as of 11. January 2021. Due to continuous development, the most recent version is accessible on an internal wiki page (not publicly accessible).

Wiki: https://wiki.dlr.de/display/AE/Lastenrechnung%3A+Loads+Kernel

Table of Contents

Nomenclature

Abbreviations

AIC	Matrix of aerodynamic influence coefficients
AR	Aspect ratio
CFD	Computational fluid dynamics
CFL	Courant–Friedrichs–Lewy number
CFRP	Carbon fiber reinforced plastic
CS	Certification specification, control surface
DLM	Doublet lattice method
DLR	Deutsches Zentrum für Luft- und Raumfahrt
Euler	Euler equations
FE	Finite elements
FI	Failure index
FL	Flight level
FSI	Fluid structure interaction
FSM	Force summation method
GAF	Generalized aerodynamic forces
LE	Leading edge
LRA	Loads reference axis
M	Mass configuration
MAC	Modal assurance criterion, mean aerodynamic chord
MDM	Modal displacement method
MDO	Multidisciplinary optimization
MLW	Maximum landing weight
MTOW	Maximum take-off weight

NS	Navier-Stokes equations
RANS	Reynolds-Averaged-Navier-Stokes equations
RFA	Rational function approximation
TE	Trailing edge
VC / MC	Cruise speed / Mach number
VD / MD	Dive speed / Mach number
VLM	Vortex lattice method

Software

Centaur	CFD mesh generation software by CentaurSoft
Loads Kernel	Loads and aeroelastic analysis software developed in this thesis
MSC.Nastran	NASA structural analysis code, commercially distributed by MSC software
Python	High-level programming language for general-purpose programming
Scipy	Python library for scientific and technical computing
SOL103	Modal analysis sequence (MSC.Nastran)
SOL144	Static aeroelastic solution (MSC.Nastran)
SOL200	Structural optimization sequence (MSC.Nastran)
Tau	Computational fluid dynamics solver developed by DLR

Notation Conventions

$\mathbf{Q}$	Matrix
$\mathbf{q}$	Vector
q, Q	Scalars
$\dot{()}$	First derivative
$\ddot{()}$	Second derivative

Sub- and Superscripts

$()^{\mathrm{desc}}$	Descriptive superscript
$()_i$	Inertial coordinate system
$()_b$	Body-fixed coordinate system
$()_g$	g-set
$()_f$	f-set, flexible generalized coordinates
$()_h$	h-set, flexible generalized coordinates + rigid body modes

X

$()_a$	a-set	
$()_o$	o-set	
$()_k$	Center point of aerodynamic panel	
$()_l$	One quarter point of aerodynamic panel	
$()_j$	Three quarter point of aerodynamic panel	
$()_{x,y,z}$	Quantity in the direction of or about the x, y and z axis	
$()_{1,2}$	Quantity in fiber direction (1) and orthogonal to the fibers (2)	
$()^{t,c}$	Quantity for tension, compression	

Variables and Parameters

Latin:

$\mathbf{A}_{3...n+2}$	Unsteady aerodynamic terms	
AC	Aerodynamic center	
$\mathbf{AIC}$	Matrix of aerodynamic influence coefficients	
$\mathbf{B}_{jj}$	Matrix of wake downwash	
CG	Center of gravity	
CP	Center of pressure	
C_d, c_d	Drag coefficient	
C_l	Lift coefficient	
C_{my}	Pitch coefficient	
$\mathbf{D}$	Damping matrix, transformation matrix for velocities, direction cosine matrix	$Kg/s, Kg \cdot m/s$
$\mathbf{D}_{\text{angular}}$	Transformation matrix for angular rates	
E	Elastic modulus	Pa
F	Design objective	
F_z	Shear force	N
G	Shear modulus	Pa
H	Gust gradient	m
$\mathbf{K}$	Stiffness matrix	$N/m, Nm/rad$
$K_{p,i,d}$	PID-Controller coefficients	
$\mathbf{M}$	Mass matrix	$Kg, Kg \cdot m$
Ma	Mach number	
MAC	Modal assurance criterion, mean aerodynamic chord	
M_x	Bending moment	Nm

M_y	Torsional moment	Nm
N_z	Load factor	
$\mathbf{Q}_k$	Stiffness matrix of fiber layer in global coordinate system	
$\mathbf{Q}'_k$	Stiffness matrix of fiber layer in local coordinate system	
R, S	Maximum allowable stress, shear	
$\mathbf{S}_{kj}$	Aerodynamic integration matrix	
$\mathbf{T}$	Spline matrix	
U	Gust velocity	m/s
V	Onflow velocity	m/s
b_{ref}	Reference span width	m
c_p	Pressure coefficient	
c_{ref}	Reference length	m
$\mathbf{d}$	Distance between two points	m
$\mathbf{g}$	Constraints	
i	Imaginary number	
$\mathbf{lag}_{1\dots n}$	Unsteady lag states	
p, q, r	Angular rates in flight physical coordinate system	rad/s
$\mathbf{p}$	Nodal loads vector	N, Nm
$\mathbf{q}$	Onflow vector	m/s
q_∞	Dynamic pressure	Pa
s	Complex Laplace variable	
t	Time	s
t_k	Layer thickness	m
$\mathbf{u}$	Nodal displacement vector	m, rad
u, v, w	Velocities in flight physical coordinate system	m/s
$\mathbf{v}_b$	Velocities in body-fixed coordinate system	m/s
$\mathbf{w}_{j,\mathrm{cam}}$	Downwash due to camber and twist of the profiles	m/s
$\mathbf{w}_{j,\mathrm{cs}}$	Downwash due to control surface deflections	m/s
$\mathbf{w}_{j,uf}$	Downwash due to flexible structural deformation	m/s
$\mathbf{w}_{j,\dot{u}f}$	Downwash due to flexible structural motion	m/s
$\mathbf{w}_{j,\mathrm{rbm}}$	Downwash due to rigid body motion	m/s
$\mathbf{w}_j$	Downwash, induced at three quarter point	m/s
$\mathbf{x}$	Design variables	

x, y, z	Position in inertial coordinate system	m
z_k	Layer offset	m
Greek:		
Γ	Circulation strength	m^2/s
$\boldsymbol{\Phi}$	Eigenvector	
Φ, Θ, Ψ	Euler angles in inertial coordinate system	$^\circ$
α	Angle of attack	$^\circ$
$\beta_{1...n}$	Poles for rational function approximation	
δ	Local displacement	m
ϵ	Strain	-
ζ	Yaw command	$^\circ, rad$
η	Pitch command	$^\circ, rad$
$\boldsymbol{\lambda}$	Eigenvalues	
ν_{12}	Poisson's ratio	
ξ	Roll command	$^\circ, rad$
ρ	Density	Kg/m^3
σ	Stress	Pa
τ	Shear	Pa
ω	Angular rate	rad/s
$\boldsymbol{\omega}_b$	Angular rates in body-fixed coordinate system	rad/s

1 Introduction

1.1 Loads Kernel Feature List - From a User's Perspective

- Direct integration in DLR-AE loads process

  - Fewer manual steps

- Simple inputs (dictionaries and lists), always SI units

  - Less mistakes

- Human readable python code (no black box)

  - Good control and knowledge about of what is happening

  - Better physical understanding

  - Easy to modify, deviate from standard procedure

- Export of data in Python, Matlab, CPACS and Nastran format

  - Special analyses in the user's favorite environment

- Model Viewer

  - Quick visualization of simulation model

  - Identification of model shortcomings → quality control

- Loads Compare

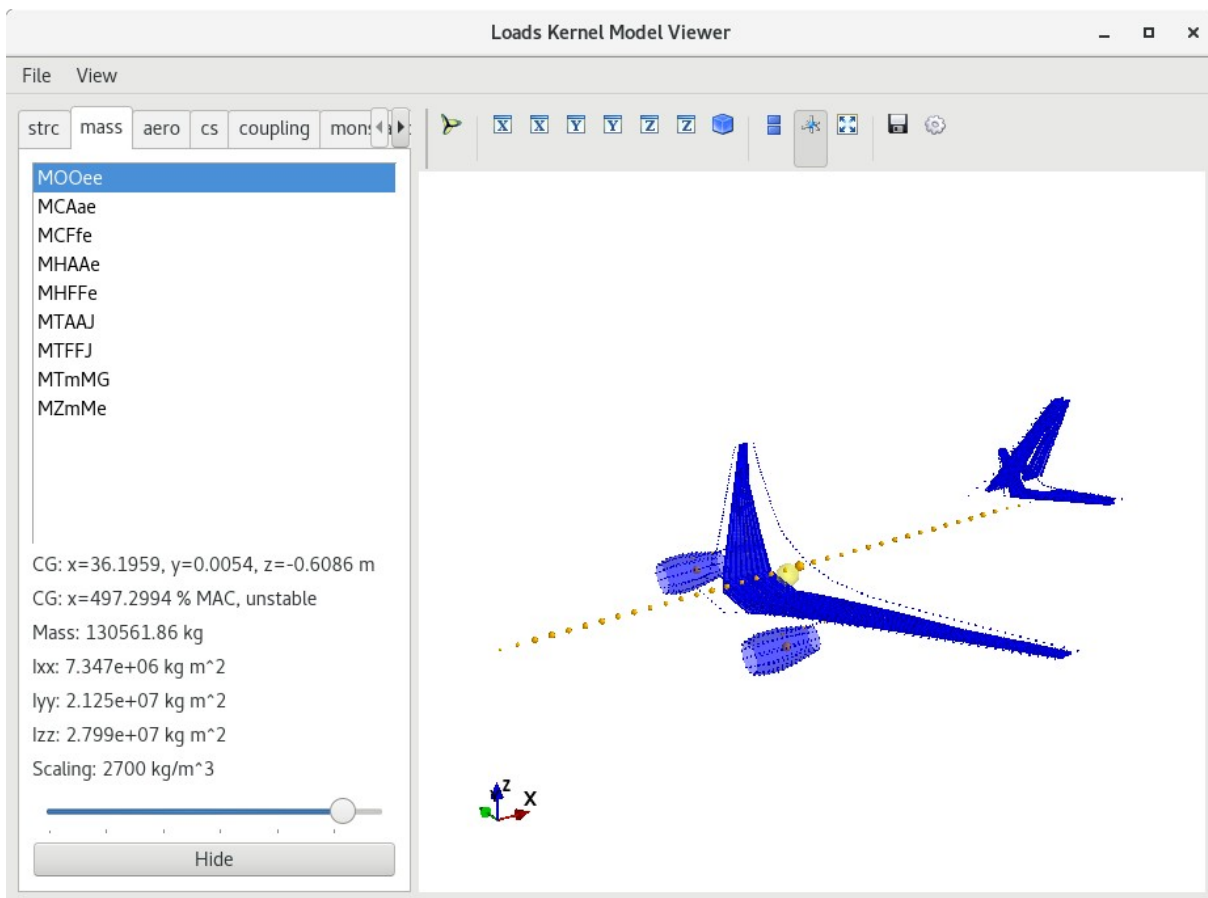 - Comparison of different sets of loads → quality control

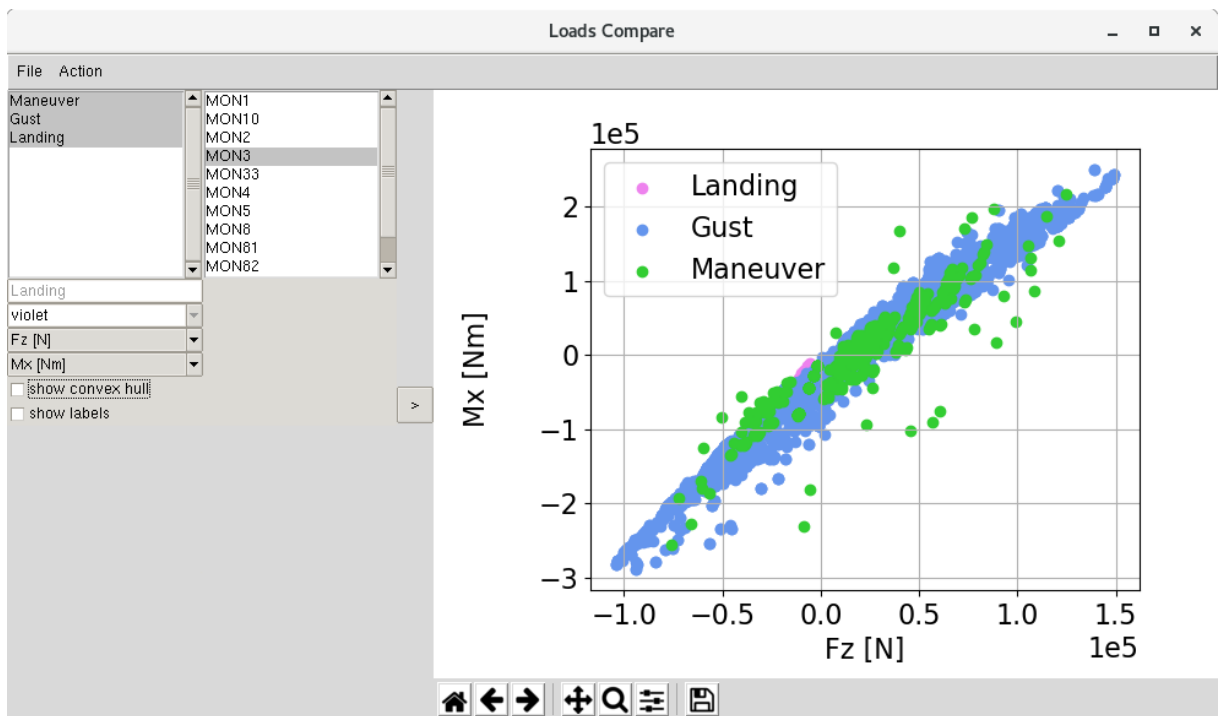*Figure 1.1: Visualization of simulation model with the Model Viewer*

Figure 1.2: Comparison of different sets of loads with Loads Compare

1.2   Loads Kernel Feature List - From a Technical Perspective

- VLM and DLM

  - Linear and nonlinear version, induced drag

- Maneuver: VLM

- Gust: VLM + DLM with rational function approximation (RFA) in time domain

- Structural flexibility

  - Modal approach

  - Eigenvalue/-vector analysis of mass and stiffness matrices

  - Matrices (Mgg, Kgg) exported from MSC.Nastran or CoFE

  - Guyan condensation (optional)

- Aero-structural coupling

  - Rigid body spline (with nearest neighbor search)

  - Surface and volume spline (radial basis functions)

- Trim conditions for different types of maneuver

  - Pitch-, roll-, yaw-maneuver, landing at constant sink rate, gliding with loss of altitude at constant speed, ...

  - EFCS for control surface scheduling

- Direct solution with non-linear trim algorithm

- Iterative solution scheme for CFD

- CFD (DLR Tau Code) for maneuver loads

  - Structural flexibility and control surfaces via mesh deformation

  - CFD surface mesh deformation by LK $\rightarrow$ volume deformation by Tau

  - Directly comparable to VLM-based solution: same trim conditions, matrices, ...

  - Benefits form LK's integration in our loads process

- Linear and non-linear equation for rigid body motion

  - Free flying, flexible aircraft in time domain

  - Flight mechanics included → good for comparisons with flight test

  - Rigid body motion captured correctly (aperiodic motion difficult in frequency domain)

  - Non-linear control possible (not possible in frequency domain)

  - Non-linear external forces, e.g. from landing gear

1.3   Dependencies

- Anaconda Python 3 with

  - Numpy, scipy, matplotlib, … (all standard modules)

  - Mayavi (optional, for visualizations in post-processing, conda install -c anaconda mayavi)

  - Pyfmi (optional, for external EFCS controller, conda install -c conda-forge pyfmi)

- Tau-Python (optional, for CFD analyses, currently only available in Python 2)

1.4   Installation & Launch

Phabricator repository: https://phabricator.ae.go.dlr.de/diffusion/LAEK/repository/

hg clone https://phabricator.ae.go.dlr.de/diffusion/LAEK/repository/ (to create a local working copy of the repository)

hg update Release_XY (to update into the most recent release)

pyenv anaconda3 (to select the python environment)

python launch.py

2 Theoretical Foundations

2.1 Coordinate Systems

In this section, a brief description of the coordinate systems and coordinate transformation is given. Only the principal coordinate systems that are important for the understanding of this work are shown. More detailed information is given in most books on flight dynamics such as by Cook [13], chapter 2 or by Brockhaus et al. [18], also chapter 2.

Most information related to mass and structure is exported from an external program, in this case MSC.Nastran, and is usually given in a global, structural coordinate system 'g'. In most cases, the origin is in front of the aircraft and axes orientation is "rear-right-up" as sketched in Figure 2.1. In addition, a body fixed coordinate system 'b' with the same orientation is placed with its origin at the center of gravity CG. That system will be used for example in Section
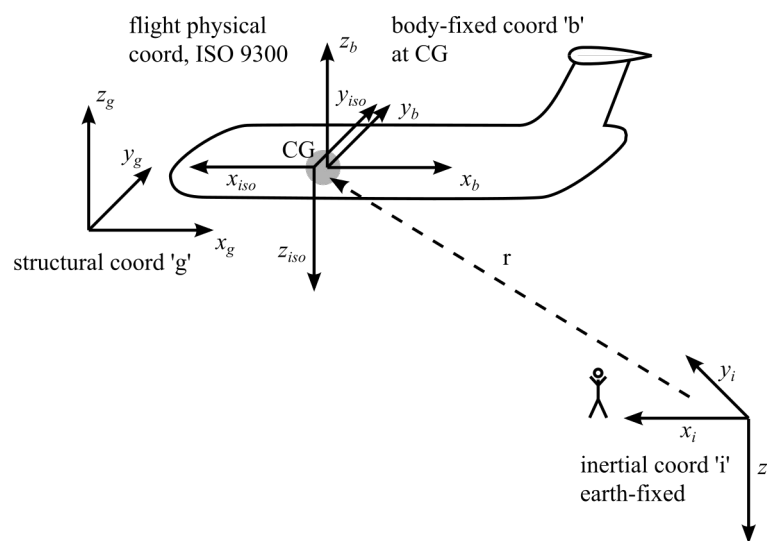

Figure 2.1: Overview of principal coordinate systems for a free-flying aircraft

2.5 for the equations of motion. The transformation between sets, such as 'g' and 'b', can be accomplished using splining methods as described in Section 2.3. The flight physical coordinate system is also placed at CG but with a "front-right-down" orientation. Finally, an earth-fixed coordinate system 'i' is used as a reference frame for the free-flying aircraft. To a certain extend and for small angles, splining methods might be used for these coordinate transformations as well. However, for the free-flying aircraft, angles can be large, and non-linear transformations need to be applied.

A transformation of velocities $[\dot{x}, \dot{y}, \dot{z}]^T$ from the earth fixed inertial coordinate system 'i' into the flight physical coordinate system according to ISO 9300 (see Brockhaus et al. [9]) is achieved by successive rotation of the Euler angles Φ, Θ and Ψ about the axis x, y, and z respectively

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{D} \cdot \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{2.1}$$

using the direction cosine matrix $\mathbf{D}$

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\Phi), & sin(\Phi) \\ 0 & -sin(\Phi) & cos(\Phi) \end{bmatrix} \cdot \begin{bmatrix} cos(\Theta) & 0 & -sin(\Theta) \\ 0 & 1 & 0 \\ sin(\Theta) & 0 & cos(\Theta) \end{bmatrix}$$
$$\cdot \begin{bmatrix} cos(\Psi) & sin(\Psi) & 0 \\ -sin(\Psi) & cos(\Psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad . \tag{2.2}$$

A transformation of angular rates $[\dot{\Phi}, \dot{\Theta}, \dot{\Psi}]^T$ from the earth fixed inertial coordinate system 'i' into the flight physical coordinate system

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \mathbf{D}_{\text{angular}} \cdot \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} \tag{2.3}$$

is achieved using matrix

$$\mathbf{D}_{\text{angular}} = \begin{bmatrix} 1 & 0 & -sin(\theta) \\ 0 & cos(\Phi) & sin(\Phi)cos(\Theta) \\ 0 & -sin(\Phi) & cos(\Phi)cos(\Theta) \end{bmatrix} \quad . \tag{2.4}$$

For a transformation in opposite direction, the inverse of $\mathbf{D}$ is given by

$$\mathbf{D}^{-1} = \mathbf{D}^T \tag{2.5}$$

and the inverse of $\mathbf{D}_{\text{angular}}$ is given by

$$\mathbf{D}_{\text{angular}}^{-1} = \begin{bmatrix} 1 & sin(\Phi)tan(\Theta) & cos(\Phi)tan(\theta) \\ 0 & cos(\Phi) & -sin(\Phi) \\ 0 & sin(\Phi)/cos(\Theta) & cos(\Phi)/cos(\Theta) \end{bmatrix} . \qquad (2.6)$$

Note that due to the trigonometric functions singularities may occur, for example if Φ or Θ approach $\pm 90°$. However, this is not the case for most flight maneuvers relevant for loads simulation. These singularities could be avoided by using quaternions instead of Euler angles for the rotations in three dimensional space.

2.2 Steady and Unsteady Aerodynamics

The classical aerodynamic approach using a steady Vortex Lattice Method (VLM) and an unsteady Doublet Lattice Method (DLM) is chosen in this work. The formulation of the VLM used and described herin follows closely the derivation given by Katz and Plotkin [26] using horse shoe vortices. The DLM is formulated as presented by Albano and Rodden [2]. It should be mentioned that the origins of compressible, unsteady aerodynamic theories date back to the early 1940s when for example Küssner published his General Airfoil Theory [34], referring again to Prandtl [43] who introduced the theory of a lifting surface based on potential accelerations instead of velocity fields in 1936. The translation of Küssners work into English language by NACA [35], shows that the development of the DLM was an international effort with contributions of scientists from multiple nations. At the time of formulation, its solution was only possible for some special cases and it took three decades until, in the late 1960s, Albano and Rodden had the computational power available for a general, numerical solution applicable to arbitrary three-dimensional wings. Therefore, Albano and Rodden didn't actually invent the DLM but were those bright minds who properly implemented the DLM for the first time. For further reading, a historical overview is given by R. Voss [57] and a very comprehensive work on the mathematical background is published by Blair [7].

An implementation of the DLM in Matlab is publicly available from Kotikalpudi [29,30]. That version is translated into the python language for performance and independence from commercial licenses. In a next step, it is slightly adapted to respect the dihedral of the wings, and the Prandtl-Glauert transformation with $\beta = \sqrt{(1 - Ma^2)}$ is applied to the VLM as

suggested by Hedman [23]. In addition, the computational process is split into more sub-functions to allow for the extraction of matrices at different stages, e.g. to obtain the circulation matrix $\boldsymbol{\Gamma}$ and the $\mathbf{AIC}$ matrices for both lift and drag.

Both the VLM and the DLM are based on a matrix of aerodynamic influence coefficients $\mathbf{AIC}$, which depends on the Mach number Ma, the reduced frequency k defined by

$$k = \frac{c_{ref}}{2V} \cdot \omega \tag{2.7}$$

and the geometry of the aircraft. The geometry is discretized using an aerodynamic panel mesh as sketched in Figure 2.2. The $\mathbf{AIC}$ matrix then relates an induced downwash $\mathbf{w}_j$ on each aerodynamic panel to a circulation strength Γ_j, which is translated to a complex pressure coefficient c_p.

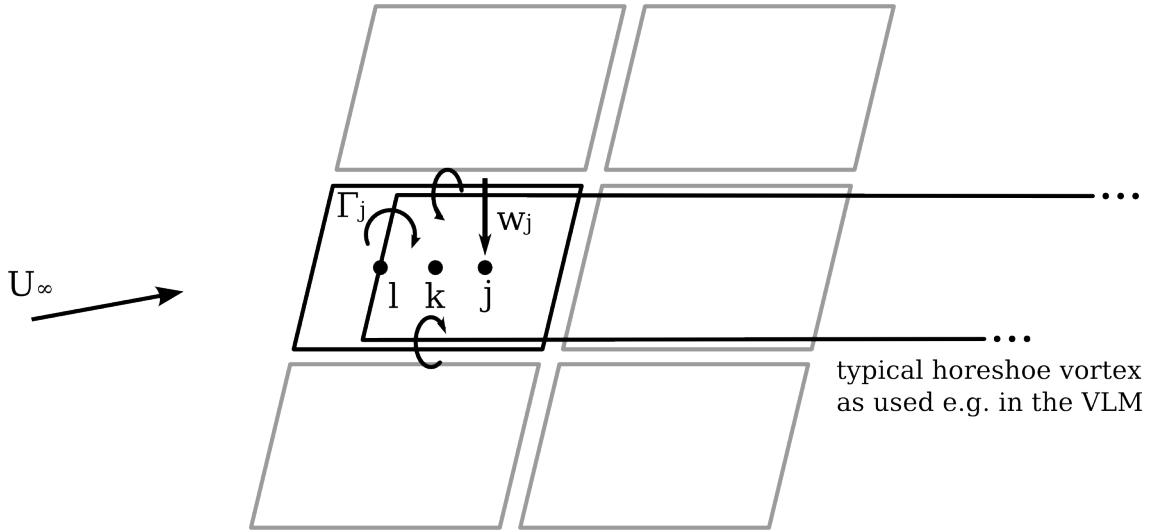$$\Delta \mathbf{c_p} = \mathbf{AIC}(Ma, k) \cdot \mathbf{w}_j \tag{2.8}$$

Figure 2.2: The geometrical mesh for aerodynamic panel methods

With $k = 0$ for the quasi static case, the solution of the DLM is equivalent to the VLM [46]. The calculation of the steady aerodynamic forces is given by

$$\mathbf{p}_k^{\text{aero,steady}} = q_\infty \cdot \mathbf{S}_{kj} \cdot \mathbf{AIC}^{\text{steady}} \cdot (\, \mathbf{w}_{j,\text{rbm}} + \mathbf{w}_{j,\text{cs}} \\ + \mathbf{w}_{j,uf} + \mathbf{w}_{j,\dot{u}f} + \mathbf{w}_{j,\text{cam}}) \tag{2.9}$$

10

q_∞ dynamic pressure

$\mathbf{S}_{kj}$ aerodynamic integration matrix

$\mathbf{AIC}$ AIC-matrix

$\mathbf{w}_{j,\mathrm{rbm}}$ downwash due to rigid body motion

$\mathbf{w}_{j,\mathrm{cs}}$ downwash due to control surface deflections

$\mathbf{w}_{j,uf}$ downwash due to flexible structural deformation

$\mathbf{w}_{j,\dot{u}f}$ downwash due to flexible structural motion

$\mathbf{w}_{j,\mathrm{cam}}$ downwash due to camber and twist of the profiles

containing several sources of aerodynamic forces. For the steady aerodynamics, the downwash due to rigid body motion $\mathbf{w}_{j,\mathrm{rbm}}$, due to control surface deflections $\mathbf{w}_{j,\mathrm{cs}}$, due to flexible structural deformation $\mathbf{w}_{j,uf}$ and due to flexible structural motion $\mathbf{w}_{j,\dot{u}f}$ is calculated from the onflow and the aircraft motions. Camber and twist of the profiles create an additional offset in downwash which is accounted for by the term $\mathbf{w}_{j,\mathrm{cam}}$. Note that the aerodynamic grid remains undeformed and the aerodynamic forces are modeled by changing the induced downwash for every panel. Using an AIC-matrix approach leads to a local pressure distribution which is integrated using matrix $\mathbf{S}_{kj}$. As the AIC-matrix is normalized with the dynamic pressure q_∞, the resulting loads need to be multiplied with q_∞ to obtain forces and moments. In this implementation, forces from the different sources given in equation (2.9) are calculated independently and superimposed

$$\mathbf{p}_k^{\mathrm{aero,steady}} = \mathbf{p}_k^{\mathrm{aero,rbm}} + \mathbf{p}_k^{\mathrm{aero,cs}} + \mathbf{p}_k^{\mathrm{aero,flex}} + \mathbf{p}_k^{\mathrm{aero,cam}} \ , \tag{2.10}$$

which is possible due to linearity.

For dynamic gust analyses, two additional components $\mathbf{p}_k^{\mathrm{aero,gust}}$ and $\mathbf{p}_k^{\mathrm{aero,unsteady}}$ are added to the steady aerodynamic forces

$$\mathbf{p}_k^{\mathrm{aero}} = \mathbf{p}_k^{\mathrm{aero,steady}} + \mathbf{p}_k^{\mathrm{aero,gust}} + \mathbf{p}_k^{\mathrm{aero,unsteady}}. \tag{2.11}$$

Term $\mathbf{p}_k^{\mathrm{aero,gust}}$ represents the aerodynamic forces due to the gust acting on the aircraft. Because the gust induced downwash is applied to every panel individually, penetration effects are taken into account. Note that this is a very generic approach and allows for arbitrary gust fields. In this case, the gust velocity

$$U = \frac{U_{ds}}{2}\left[1 - cos\left(\frac{\pi s}{H}\right)\right] \quad for \quad 0 \le s \le 2H \tag{2.12}$$

is defined by the certification specifications CS 25.341 [16] in dependence of the distance s penetrated into the gust and the design gust velocity U_{ds}. The so-called gust gradient H determines the length (parallel to the aircraft's flight path) for the gust to reach its peak velocity.

Because the gust encounter is a very short and sudden event, the certification specifications explicitly ask to account for unsteady aerodynamic characteristics. In this work, unsteady aerodynamic forces in the time domain are obtained by a rational function approximation (RFA) as suggested by Roger [47]. Similar implementations are shown by various authors. From all authors, Gupta [19] gives the most detailed description. The work by Kier and Looye [27] points out the features of the "physical RFA". Karpel and Strul [25] have a slightly different focus (minimizing the number of states of the RFA) but give an overview on possible constraints. Other publications making use of the RFA are by Goggin [18], Abel [1] and Vepa [53]. A difference of this work with respect to other authors is the approximation on panel level using physical coordinates. This leads to a large number of lag states but the implementation is more generic and leads to physical, nodal forces. This is required in order to use the force summation method, which will be explained in Section 2.5. In other works, so-called gust-modes are used that include all panels in one row. This is more difficult with highly swept flying wing configurations. In addition, these gust-modes have shown difficulties in the approximation due to a spiraling nature of the transfer function in the complex plane, see for example Figure 9 in [58]. Finding a good approximation of the gust-modes is still a field of research [58].

In the RFA, the $\mathbf{AIC}$ matrix is approximated for each reduced frequency k with the following rational function:

$$\mathbf{AIC}(k_{\mathrm{red}}) = \mathbf{A}_0 + \mathbf{A}_1 \cdot ik + \mathbf{A}_2 \cdot (ik)^2 + \sum_{n=1}^{n_{\mathrm{poles}}} \mathbf{A}_{n+2} \cdot \frac{ik}{ik + \beta_n} \ . \qquad (2.13)$$

The approximation is done by solving the equation (2.13) in a least squares sense. Matrix $\mathbf{A}_2$ is omitted during the approximation, as suggested by Kier and Looye [27]. In addition, it is necessary to separate into real and imaginary parts. This is possible with

$$\frac{ik}{ik + \beta_n} = \frac{k^2}{k^2 + \beta_n^2} + \frac{ik\beta_n}{k^2 + \beta_n^2} \ . \qquad (2.14)$$

In this work, the poles $\boldsymbol{\beta}_i$ used for the approximation are determined by

$$\boldsymbol{\beta}_i = \frac{\mathbf{k}_{\max}}{i} \tag{2.15}$$

as given by Roger [47]. A slightly different proposal is given by ZONA [62]. Both methods were tested and showed comparable results. In addition to the number of poles n_{poles}, the frequency range and frequency distribution is important. It is recommended to provide many samples close to zero with an increasing step size up to the highest reduced frequency $k_{\max}$. Typical value for the reduced frequency are $k_{\max} = 2.0$ with $n_{\text{poles}} = 6 \ldots 9$, but the actual values strongly depend on the aircraft configuration.
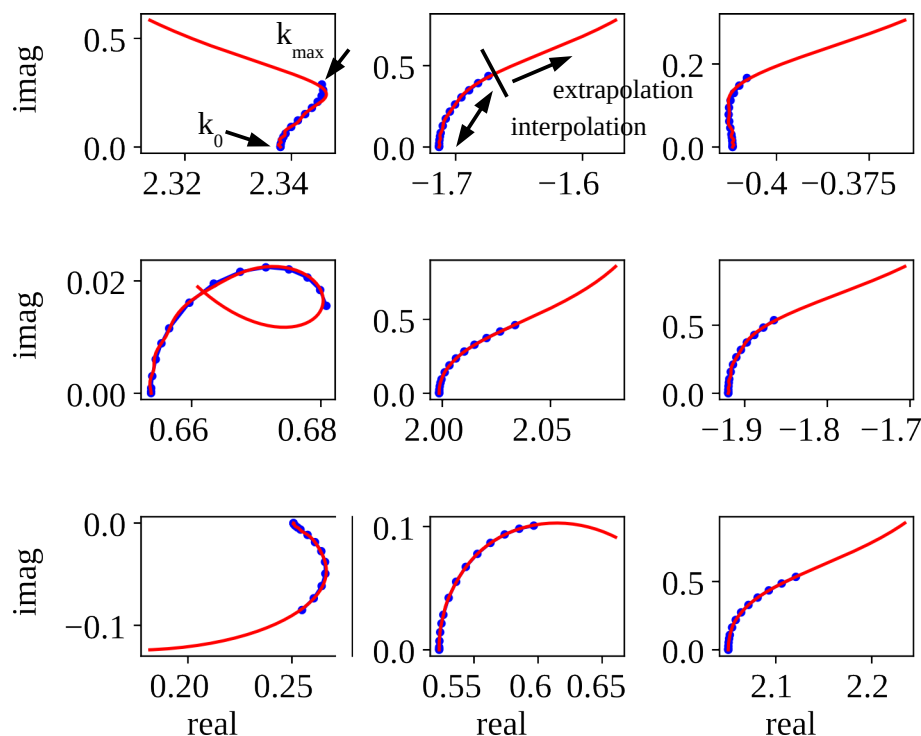

Figure 2.3: Real and imaginary parts of the first 3x3 entries of an AIC matrix (blue dots) and approximation (red line)

However, there is no general rule and the quality of the approximation has to be checked carefully, because too few poles result in a bad approximation, leading to nonphysical results. Figure 2.3 shows the real and imaginary parts of the first 3x3 entries of the **AIC** matrices and allows for a qualitative assessment of the approximation. Every blue dot marks the values of one given reduced frequency k. The red line is calculated from the approximation. In this case, the approximation looks good by visual judgment in the area covered by the given reduced frequencies. In addition, an extrapolation is shown. Here, the results are questionable in some cases. As a consequence, the given reduced frequencies should cover the complete

range of application and an extrapolation should be avoided. A quantitative assessment is possible by calculating the root mean squared error E_{RSM} of the given and the approximated matrices at the given frequencies

$$E_{\mathrm{RSM}}(k) = \sqrt{\frac{1}{n_j^2} \sum_{j=1}^{n_j^2} \left(\mathbf{AIC}(k)_{jj,\mathrm{approx}} - \mathbf{AIC}(k)_{jj,\mathrm{given}}\right)^2} \; . \tag{2.16}$$

For typical configurations, the root mean squared error E_{RSM} should drop below $10^{-2} \ldots 10^{-4}$ for all reduced frequencies.

With $s = i \cdot \omega$ and k defined in equation (2.7), ik may be written as

$$ik = i \cdot \frac{c_{\mathrm{ref}}}{2V} \cdot \omega = \left(\frac{c_{\mathrm{ref}}}{2V}\right) s \; . \tag{2.17}$$

Equation (2.13) is expanded to

$$
\begin{aligned}
\mathbf{p}_k^{\mathrm{aero}}(s) = q_\infty \cdot \mathbf{S}_{kj} \cdot w_j \cdot \Big[& \mathbf{A}_0 + \mathbf{A}_1 \cdot \left(\frac{c_{\mathrm{ref}}}{2V}\right) s + \mathbf{A}_2 \cdot \left(\left(\frac{c_{\mathrm{ref}}}{2V}\right) s\right)^2 \\
& + \sum_{n=1}^{n_{\mathrm{poles}}} \mathbf{A}_{n+2} \cdot \frac{\left(\frac{c_{\mathrm{ref}}}{2V}\right) s}{\left(\frac{c_{\mathrm{ref}}}{2V}\right) s + \beta_n} \Big]
\end{aligned}
\tag{2.18}
$$

Transformation from the Laplace domain into the time domain leads to

$$
\begin{aligned}
\mathbf{p}_k^{\mathrm{aero}}(t) = q_\infty \cdot \mathbf{S}_{kj} \cdot \Big[& \mathbf{A}_0 \mathbf{w}_j + \mathbf{A}_1 \left(\frac{c_{\mathrm{ref}}}{2V}\right) \dot{\mathbf{w}}_j + \mathbf{A}_2 \left(\frac{c_{\mathrm{ref}}}{2V}\right)^2 \ddot{\mathbf{w}}_j \\
& + \sum_{n=1}^{n_{\mathrm{poles}}} \mathbf{A}_{n+2} \cdot \mathbf{lag}_n \Big]
\end{aligned}
\tag{2.19}
$$

The lag states $\mathbf{lag}_n$ are defined as

$$\mathbf{lag}_n = \frac{s}{s + \frac{2V}{c_{\mathrm{ref}}} \beta_n} \cdot \mathbf{w}_j \tag{2.20}$$

or

$$\mathbf{lag}_n \cdot s = \mathbf{w}_j \cdot s - \left(\frac{2V}{c_{\mathrm{ref}}}\right) \boldsymbol{\beta}_n \cdot \mathbf{lag}_n \tag{2.21}$$

in the Laplace domain. Transformation into the time domain yields

$$\dot{\mathbf{lag}}_n = \dot{\mathbf{w}}_j - \left(\frac{2V}{c_{\mathrm{ref}}}\right) \cdot \boldsymbol{\beta}_n \cdot \mathbf{lag}_n \; . \tag{2.22}$$

It can be seen that the evolution of the lag states is described by a differential equation, which can be solved independently or simply appended to the system of equations which will be described in Section 2.5.

From equation (2.19) it can the seen that a rational function approximation allows for a decomposition of the aerodynamic forces into a steady term $\mathbf{A}_0$ depending on the downwash $\mathbf{w}_j$, corresponding to equation (2.10), a damping term $\mathbf{A}_1$ depending on the rate of change of the downwash $\dot{\mathbf{w}}_j$ and a term $\mathbf{A}_2$ depending on the acceleration of the downwash $\ddot{\mathbf{w}}_j$. Because $\ddot{\mathbf{w}}_j$ is difficult to calculate, it is omitted in the approximation step. The unsteady terms $\mathbf{A}_3, \mathbf{A}_4, \ldots, \mathbf{A}_{n+2}$ depend on the lag states $\mathbf{lag}_1, \mathbf{lag}_2, \ldots, \mathbf{lag}_n$. As the time simulation usually starts from an initial steady level flight, the lag states are assumed to be zero at the beginning.

Drag is neglected by most commercial software packages and has little direct influence on the loads. In addition, wing structures are typically sized by the shear force F_z, bending moment M_x or torsional moment M_y. However, the local induced drag is important to capture the roll-yaw-coupling of the aircraft. (Note that there are also other contributors to the roll-yaw-coupling, e.g. profile drag, which is not considered at this point.) Thus, the induced drag might have an indirect influence on the loads. Therefore, the VLM is extended. The standard procedure described above involves a linearization about an angle of attack $\alpha = 0°$, resulting in a lift vector orthogonal to the body frame but not orthogonal to the onflow for angles of attack $\alpha \neq 0°$. This would result in an artificial drag component depending on the angle of attack α, which is not desired. This can be avoided by selecting a slightly different formulation of the VLM

$$\mathbf{p}_k^{\text{aero,steady,nonlin}} = \mathbf{\Phi}_{lk} \cdot \rho \cdot \mathbf{\Gamma} \cdot \mathbf{w}_j \cdot (\mathbf{q} \times \mathbf{r}) \ . \tag{2.23}$$

The induced downwash $\mathbf{w}_j$ on each aerodynamic panel is multiplied with the circulation matrix $\mathbf{\Gamma}$ obtained from the VLM. The cross product of the onflow vector $\mathbf{q}$ and the panel span vector $\mathbf{r}$ at quarter chord yield a lift vector orthogonal to the onflow condition. Multiplication with the air density ρ and translation matrix $\mathbf{\Phi}_{lk}$ gives the aerodynamic forces $\mathbf{p}_k^{\text{aero,steady,nonlin}}$. The induced downwash $\mathbf{w}_j$ is calculated in the same way as before, allowing for a smooth integration into the existing code.

In a flow field, the Trefftz plane located at an infinite distance downstream of the aircraft may be used to analyze the wake in order to obtain the total induced drag C_d^{ind}. In a similar manner, the wake $\mathbf{w}_j^{\text{wake}}$ of every panel, defined by the trailing vortices of the horseshoe, may be used to calculate the local induced drag $\mathbf{c}_{dj}^{\text{ind}}$ at every panel. Formally, this can be expressed with matrix $\mathbf{B}_{jj}$ and $\mathbf{c}_{pj}$ from equation (2.8), which gives the downwash of the wake

$$\mathbf{w}_j^{\text{wake}} = -\mathbf{B}_{jj} \cdot \mathbf{c}_{pj}, \tag{2.24}$$

and the downwash leads to the induced drag

$$\mathbf{c}_{dj}^{\text{ind}} = \mathbf{w}_j^{\text{wake}} \cdot \mathbf{c}_{pj}. \tag{2.25}$$

The calculated induced drag can be compared to the theoretically lowest induced drag for planar wings based on an elliptical lift distribution. Division of theoretical and calculated induced drag yields the span efficiency value e

$$e = \frac{C_l^2}{\pi \Lambda C_d^{\text{ind}}}, \tag{2.26}$$

which should be close to $e = 1.0$.

To maintain compatibility of the formulation of the the VLM with the DLM, in this work the linearization about an angle of attack $\alpha = 0°$ is used. Also, the formulation of the induced drag is not compatible to the DLM. An application where both effects are important is shown in [55,56] at the example of a sailplane. That work was prepared using the same methods and tools but is not part of this thesis.

2.3 Aero-Structural Coupling

In a next step step, the aerodynamic forces need to be applied to the structure. Formally, the coupling can be handled using a transformation matrix $\mathbf{T}_{kf}$ which relates displacements of the structural grid $\mathbf{u}_f$ to displacements of the aerodynamic grid $\mathbf{u}_k$ with

$$\mathbf{u}_k = \mathbf{T}_{kf} \cdot \mathbf{u}_f. \tag{2.27}$$

In addition, the transposed matrix $\mathbf{T}_{kf}^T$ transfers forces and moments from the aerodynamic grid $\mathbf{p}_k$ to the structural grid $\mathbf{p}_f$ with

$$\mathbf{p}_f = \mathbf{T}_{kf}^T \cdot \mathbf{p}_k. \tag{2.28}$$

Note that the structural displacements and loads (index 'f') are not given in physical but in modal coordinates, which will be explained in more detail in Section 2.4. The size of the problem depends on the degrees of freedom of both the aerodynamic and structural grid. In this case, the structural grid has a much finer discretization with 34482 degrees of freedom. Therefore, a projection into modal coordinates in combination with a truncation of higher frequency modes implies a significant reduction.

In general, a transformation matrix $\mathbf{T}_{di}$ may be defined by various methods [54]. One commonly used approach for loads calculation is the **rigid body spline**. Each grid point of the dependent grid is mapped to exactly one point on the independent grid. The distance $\mathbf{d} = [d_x, d_y, d_z]^T$ between these two grid points is assumed as a rigid body that transfers forces and moments. In addition, forces $\mathbf{f}$ create moments $\mathbf{m}$ due to their lever arm

$$\mathbf{m} = \mathbf{d} \times \mathbf{f}. \tag{2.29}$$

In reverse, translations and rotations are directly transferred and rotations create additional translations. For one set of two points, this can be expressed by matrix $\mathbf{T}_{di}^{\star}$ with

$$\mathbf{T}_{di}^{\star} = \begin{bmatrix} 1 & & & & d_z & d_y \\ & 1 & & -d_z & & d_x \\ & & 1 & -d_y & d_x & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix}. \tag{2.30}$$

The mapping of the points may be defined manually or automatically, e.g. with a nearest neighbor search. This concept is quite versatile and can be used for many application other than the aero-structural coupling, e.g. to gather all external forces $\mathbf{p}_g^{\text{ext}}$ at the center of gravity $\mathbf{p}_b^{\text{ext}}$

$$\mathbf{p}_b^{\text{ext}} = \mathbf{T}_{bg}^T \cdot \mathbf{p}_g^{\text{ext}}. \tag{2.31}$$

Another widely used method are **radial basis functions**. A very good example is given by Neumann and Krüger [40] who show the application to large, industrial scale models. The method is best visualized by a curve running through a number of supporting points. At the supporting points, the results are exact. For the values in-between, the results are approximated. The approximation is influenced by the neighboring supporting points which are weighted according to their distance to the approximation point. Although the influence

reduces over the distance, every supporting point has an influence on all other points. This influence is calculated with the help of radial basis functions. A popular example is the Infinite Plate Spline (IPS) by Harder and Desmarais [21] which uses

$$\phi(\|d\|) = \|d\|^2 \cdot log(\|d\|) \tag{2.32}$$

as a function for the influence ϕ. Beckert and Wendland [4] have compared and investigated various radial basis functions as an alternative to equation (2.32). Some radial basis functions provide a compact support radius, limiting the influence to a local area. However, Neumann [39] showed that local radial basis functions are likely to results in crinkled surfaces, which is not beneficial. Therefore, in this work only globally supported radial basis functions will be applied. Finally, the spline can be constructed as a surface spline where only two dimensions are considered. An extension by the third dimension leads to a volume spline.

Advantages and disadvantages with respect to flying wing configurations have been discussed by Voß and Klimmek [54] and are summarized in Tables 2.1 and 2.2. The rigid body spline allows for a clear and comprehensible mapping of the aerodynamic grid to the structure. Due to the direct mapping of the rigid body spline, matrix $\mathbf{T}_{di}$ is sparse while $\mathbf{T}_{di}$ is dense when using a global surface or volume spline. To construct a surface or volume spline, a system of equations needs to be solved. This results in longer computational time in comparison to the rigid body spline. Although globally correct, a surface or volume spline may locally result in very large, opposing forces, which are not physical. This behavior has been observed when the number of structural grid points is much larger than the number of aerodynamic grid points. These large, local forces may change the magnitude of section loads significantly while the integral forces of the entire aircraft are correct. Therefore, the rigid body spline is more suitable for the transfer of forces and moments. In contrast, a surface spline is more suitable for smooth surface deformation whereas a rigid body spline usually results in bad and bumpy surface deformations. This is acceptable when using aerodynamic panel methods such as VLM or DLM, but may have a fatal impact on CFD simulations.

advantages	disadvantages
• clear and comprehensible mapping • forces and moments are conserved • forces and moments normally act on the proper structural element (e.g. intersection between wing and control surface, intersection between fuselage and wing) • use of different coordinate systems possible • fast	• possibly high nodal forces • possibly uneven/bumpy surface deformation (fatal impact on CFD simulation)

Table 2.1: Advantages and disadvantages of rigid body splines

advantages	disadvantages
• high quality surface deformation • forces are distributed evenly on the structure	• slow for large problems • splining should be conducted separately for each structural part • locally unphysical, possible problems in section forces

Table 2.2: Advantages and disadvantages of surface and volume splines

2.4 Modal Analysis, Structural Degrees of Freedom and Masses

The degrees of freedom for a grid point of a finite element model (FEM) include the six components of displacement: translation in x, y, and z direction and rotation about the x, y, and z axis. In MSC.Nastran, these degrees of freedom are organized in so-called sets [63]. The relation of the most relevant sets for this work is shown in Figure 2.4. The global set, or g-set, contains all degrees of freedom and is the top-level set. Usually, the g-set contains
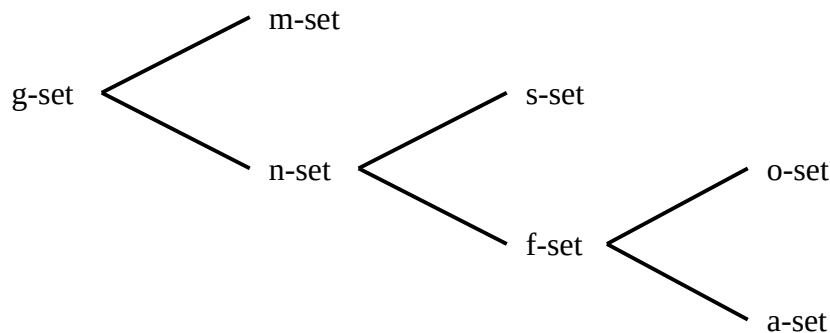

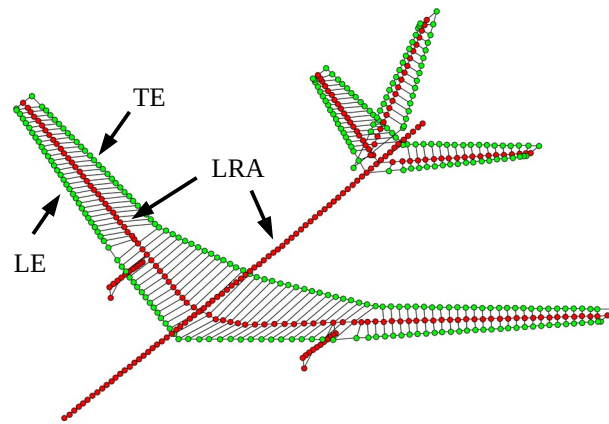*Figure 2.4: Schematic overview of structural sets*

Figure 2.5: Condensed finite element model of the FERMAT configuration [28]

linear relationships, which are for example constructed with rigid body elements and multipoint constraints. These dependent degrees of freedom are moved into the m-set. The remaining, independent degrees of freedom form the n-set. Sometimes, a structural model contains single point constraints, for example to realize a clamping. These degrees of freedom are moved into the s-set. The remaining degrees of freedom are organized in the f-set. Therefore, the f-set contains all "free" degrees of freedom, the corresponding mass and stiffness matrices $\mathbf{K}$ and $\mathbf{M}$ are no longer singular and suitable for a solution if arranged in a set of equations.

Finally, the f-set may be partitioned into the a-set and the o-set. This process can be achieved by a static condensation or Guyan reduction [20]. In aeroelastic applications, the wing and fuselage structure is often condensed to a loads reference axis (LRA). The loads reference axis is placed e.g. along the quarter chord line of the wing. A typical example is shown in Figure 2.5. Note that the points of the leading and trailing edge (LE and TE, green points) are connected with rigid body elements to the loads reference axis. The concept of a loads reference axis has several advantages. First, the model is simplified significantly, making physical interpretations easier. Second, mass estimates are often done by a different department than the structural analysis. The loads reference axis is a suitable basis for communication and data exchange, as condensed masses can be easily attached to the condensed structural grid points. Third, computational time is reduced for all following calculations. However, considering the available computational power of today, this argument is no longer relevant. Finally, local modes are avoided during the modal analysis. Such local

modes appear if for example a thin shell element vibrates at a low frequency. Using a condensation, the model is "cleaned" from such local modes.

> **Remark:** Technically, there is no such thing as a "cleaning" procedure. The spatial resolution of the structural model is reduced so much that local modes are not resolved by a sufficient number of points. Only global mode shapes can be represented.

The partitioning of the f-set into the a-set and the o-set is achieved in the following way. The equation for static deflection

$$\mathbf{K} \cdot \mathbf{u} = \mathbf{p} \tag{2.33}$$

relating forces $\mathbf{p}$ to stiffness matrix $\mathbf{K}$ times deflections $\mathbf{u}$ can be rearranged to

$$\begin{bmatrix} \mathbf{K}_{aa} & \mathbf{K}_{ao} \\ \mathbf{K}_{oa} & \mathbf{K}_{oo} \end{bmatrix} \cdot \begin{pmatrix} \mathbf{u}_a \\ \mathbf{u}_o \end{pmatrix} = \begin{pmatrix} \mathbf{p}_a \\ \mathbf{p}_o \end{pmatrix} \tag{2.34}$$

according to Guyan [20]. Here, the degrees of freedom with index 'o' are those to which no force is applied to and which can be eliminated. With $\mathbf{p}_0 = 0$, the equations are solved to

$$\left[\mathbf{K}_{aa} - \mathbf{K}_{ao} \cdot \mathbf{K}_{oo}^{-}1 \cdot \mathbf{K}_{oa} \right] \cdot (\mathbf{u}_a) = (\mathbf{p}_a). \tag{2.35}$$

From this, the reduced stiffness matrix $\mathbf{K}_{aa,\text{reduced}}$ is identified as

$$\mathbf{K}_{aa,\text{reduced}} = \mathbf{K}_{aa} - \mathbf{K}_{ao} \cdot \mathbf{K}_{oo}^{-1} \cdot \mathbf{K}_{oa}. \tag{2.36}$$

Due to the analytical solution of the problem, the Guyan reduction of the stiffness matrix is exact.

The same procedure could be applied to the mass matrix $\mathbf{M}$, although the term $\mathbf{M}_{ao}$ is usually zero as mass matrices are usually diagonal. However, multiplication with $\mathbf{M}_{ao}$ would eliminate all masses of the o-set and only the masses on the a-set would remain. This problem is solved by Guyan by combining mass and stiffness matrices

$$\begin{aligned} \mathbf{M}_{aa,\text{reduced}} =& \mathbf{M}_{aa} - \mathbf{M}_{ao} \cdot \mathbf{K}_{oo}^{-1} \cdot \mathbf{K}_{oa} \\ & - (\mathbf{K}_{oo}^{-1} \cdot \mathbf{K}_{oa})^T \cdot (\mathbf{M}_{oa} - \mathbf{M}_{oo} \cdot \mathbf{K}_{oo}^{-1} \cdot \mathbf{K}_{oa}) \end{aligned} \tag{2.37}$$

to obtain a reduced mass matrix, too. The influence of the masses on the o-set is weighted with the stiffness. Guyan states that "the eigenvalue-eigenvector problem is closely but not exactly preserved" [20]. Comparison studies show that this is true for low frequencies. The

higher the frequency, the greater the deviation. Therefore, the Guyan reduction is suitable for typical aircraft configurations.

For convenience and to significantly improve computational performance, $\mathbf{K}_{oo}^{-1} \cdot \mathbf{K}_{oa}$ is written as

$$\mathbf{K}_{oo} \cdot \mathbf{G}_{oa} = \mathbf{K}_{oa} \tag{2.38}$$

and solved for $\mathbf{G}_{oa}$. This step has to be done only once, and, in addition, the solution of a sparse linear system $\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$ is much faster than a matrix inversion.

A modal analysis comprises the characterization of the dynamic behavior of an elastic, vibrating system. The oscillating behavior of that system about an idle state is described by modal parameters such as natural frequency, mode shapes, modal mass and modal damping. The behavior of the undamped, unexcited system of mass $\mathbf{M}$ and stiffness $\mathbf{K}$ is governed by

$$\mathbf{M} \cdot \ddot{\mathbf{x}} + \mathbf{K} \cdot \mathbf{x} = 0, \tag{2.39}$$

which corresponds to

$$\omega^2 \cdot \mathbf{M} \cdot \hat{\mathbf{x}} + \mathbf{K} \cdot \hat{\mathbf{x}} = 0 \tag{2.40}$$

in the frequency domain. Damping is omitted at this step and added later in equation (2.47). The problem can be rearranged to the form of the generalized eigenvalue problem

$$-\mathbf{K} \cdot \hat{\mathbf{x}} = \omega^2 \cdot \mathbf{M} \cdot \hat{\mathbf{x}} \quad \text{or} \quad \mathbf{A} \cdot \mathbf{\Phi} = \lambda \cdot \mathbf{B} \cdot \mathbf{\Phi}, \tag{2.41}$$

with $\mathbf{\Phi}$ being the matrix of generalized eigenvectors and $\boldsymbol{\lambda}$ the generalized eigenvalues of $\mathbf{A}$ and $\mathbf{B}$. For large matrices $\mathbf{A}$ and $\mathbf{B}$, which is usually the case in structural dynamics, the system is solved iteratively for the first n_f eigenvalues. In MSC.Nastran, the calculation of the real eigenvalues typically uses the Lanczos method, in Scipy an Arnoldi method is available [36,52]. Because matrices $\mathbf{M}$ and $\mathbf{K}$ are sparse, using the sparse eigenvalue analysis capability significantly increases performance. Both methods have been tested and return numerically equivalent eigenvalues $\boldsymbol{\lambda}$ for typical aircraft configurations. The eigenvectors $\mathbf{\Phi}$ are more difficult to compare. Here, the modal assurance criterion (MAC) [3] is a suitable means for comparison. Assuming two eigenvectors ϕ_a and ϕ_b, the MAC value is calculated with

$$MAC = \frac{\left| \phi_a^T \phi_b \right|^2}{\left(\phi_a^T \phi_a \right) \left(\phi_b^T \phi_b \right)} \quad, \tag{2.42}$$

resulting in a scalar value $MAC = 0.0 \ldots 1.0$ where a value of 1.0 indicates a perfect agreement of both eigenvectors. Applied to all combinations of eigenvectors, a matrix of MAC values is constructed.

The Auto-MAC with $\mathbf{\Phi}_a = \mathbf{\Phi}_b$ is shown in Figure 2.6. As expected, all values along the diagonal are very close to 1.0. The off-diagonal terms show very small values close to 0.0. The areas of weak correlation are due to the discretization of the numerical model. Figure 2.7 shows the MAC matrix comparing the eigenvectors obtained from MSC.Nastran SOL103 and Scipy for a typical aircraft configuration. It can be seen that the eigenvectors are numerically equal.
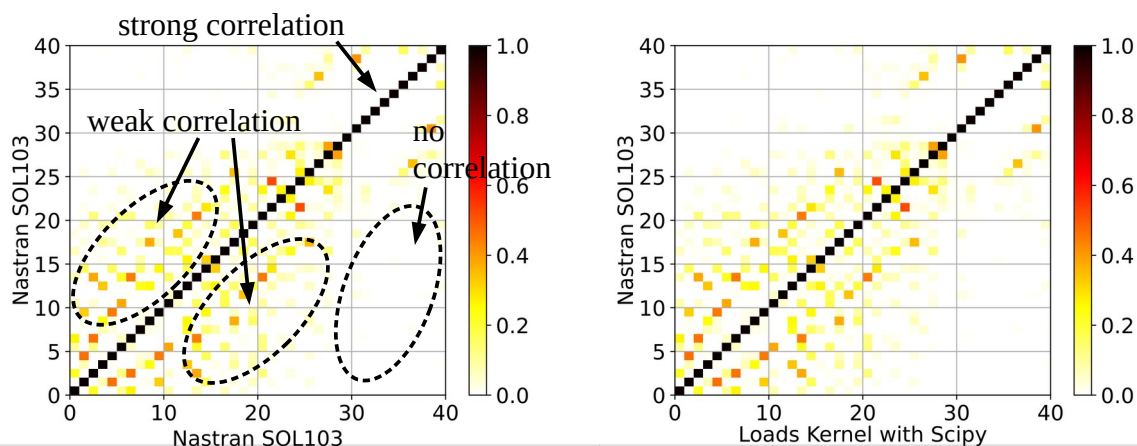

Figure 2.6: Auto MAC for the first 40 modes

Figure 2.7: MAC matrix comparing MSC.Nastran SOL103 and Scipy

Assuming the modal analysis is conducted on the reduced a-set, the corresponding displacements of the remaining degrees of freedom $\mathbf{u}_o$, $\mathbf{u}_s$ and $\mathbf{u}_m$ need to be reconstructed from the eigenvectors. The displacements $\mathbf{u}_o$ of the o-set, see equation (2.34), may be reconstructed from the a-set with

$$\mathbf{u}_o = -\mathbf{K}_{oa} \cdot \mathbf{u}_a \cdot \mathbf{K}_{oo}^{-1}. \tag{2.43}$$

The s-set displacements $\mathbf{u}_s$ are set to zero. With this information, the displacements $\mathbf{u}_n$ of the n-set can be assembled. As mentioned at the beginning, the m-set is linearly dependent on the

n-set. Therefore, the displacements $\mathbf{u}_m$ of the m-set can be derived from $\mathbf{u}_n$ by applying matrix $\mathbf{GM}$ [38]

$$\mathbf{u}_m = \mathbf{GM} \cdot \mathbf{u}_n. \tag{2.44}$$

Following that procedure, the eigenvector $\mathbf{\Phi}_{gf}$ for all degrees of freedom of the g-set is assembled.

The benefit and the physical interpretation is the following. An eigenvalue analysis of a finite element model projects the dynamic deformation into a set of flexible modes. Because the resulting mode shapes are uncoupled, a superposition of several modes is possible. With $\omega^2 = \lambda$, the eigenvalue λ defines the angular frequency of the mode. The eigenvector $\mathbf{\Phi}$ contains the corresponding deflection in physical coordinates. In addition, eigenvector $\mathbf{\Phi}$ allows for the calculation of the modal stiffness and modal mass matrices $\mathbf{K}_{ff}$ and $\mathbf{M}_{ff}$ from the original matrices $\mathbf{K}_{aa}$ and $\mathbf{M}_{aa}$ in physical coordinates:

$$\mathbf{K}_{ff} = \mathbf{\Phi}^T \mathbf{K}_{aa} \mathbf{\Phi} \tag{2.45}$$

$$\mathbf{M}_{ff} = \mathbf{\Phi}^T \mathbf{M}_{aa} \mathbf{\Phi} \tag{2.46}$$

In absence of a more rational analysis, uniform modal damping [5]

$$\mathbf{D}_{ff} = \mathbf{I} \cdot \boldsymbol{\lambda}^2 \cdot 2.0 \cdot d \tag{2.47}$$

may be assumed with typical damping values ranging from $d = 0 \ldots 3.0\%$. For simplicity, in most academic works no damping is assumed.

Note that at this step, index 'f' refers to the flexible mode set obtained from modal analysis and not the f-set including the "free" degrees of freedom. This inconsistency exists because if no reduction is made and no degrees of freedom are omitted, the f-set is equivalent to the a-set. Therefore, the a-set is favored for calculation purposes and the f-set is rarely used.

The final challenge of the Guyan reduction lies in the determination of the degrees of freedom for the a-set. The LRA concept mentioned at the beginning of this section is a very good choice for classical wing-fuselage-empennage configurations but has its limits with planar flying wings of low aspect ration. Because of the low number of ribs, only very few points would be selected for the a-set. In addition, the resolution in chord direction is questionable and not representative, especially in the fuselage region. Therefore, a new selection scheme

for the a-set is required. The use of sub section corner points is a suitable answer to the above problem. One sub segment is the area between two ribs and two spars, as visualized in Figure 2.8. The identification of the corner points of every sub section leads to a satisfactory number of points distributed over the whole aircraft. In addition, only structurally significant points are selected. The o-set then consists of the remaining degrees of freedom.
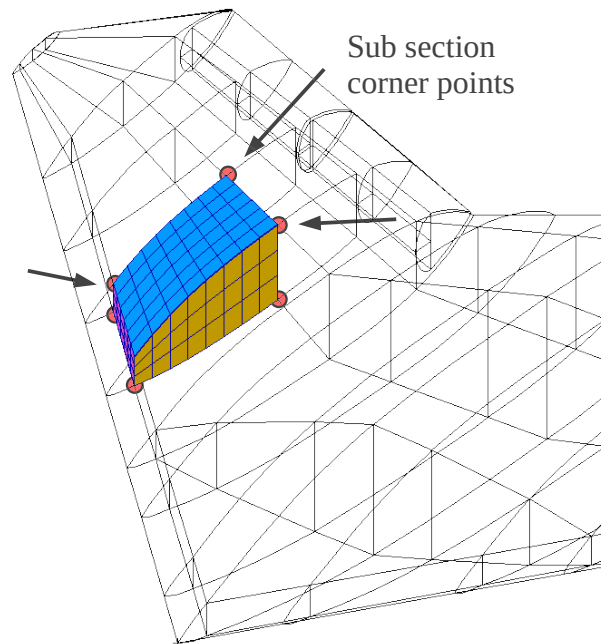

Figure 2.8: Selection of degrees of freedom of the a-set based of sub section corner points

With the assumption that low frequency modes are more important than higher frequency modes, a truncation is possible. The number of modes n_f to be kept should be selected in relation with the highest desired frequency $k_{\max}$. That frequency should be selected again in relation to the capabilities of the unsteady aerodynamics. If the system is analyzed in free-free conditions, the first six eigenvalues are characterized by $\lambda \approx 0.0$ and correspond to the rigid body motion of the system. They are discarded if only the flexible modes are required.

With the knowledge of the set definitions and a mass matrix M_{gg}, the total mass properties of the rigid body can be calculated. This is done in a two step approach. In the first step, all masses are gathered at a reference point 0 by applying Φ_{0g} and Φ_{0g}^T

$$\mathbf{M}_0 = \mathbf{\Phi}_{0g}^T \mathbf{M}_{gg} \mathbf{\Phi}_{0g} . \tag{2.48}$$

The resulting matrix M_0 has the shape 6x6 with the following entries of interest:

$$\mathbf{M}_0 = \begin{bmatrix} M_{xx} & & & & r_z M_{xx} & -r_y M_{xx} \\ & M_{yy} & & -r_z M_{yy} & & r_x M_{yy} \\ & & M_{zz} & r_y M_{zz} & -r_x M_{zz} & \\ & & & & & \\ & & & \cdots & & \end{bmatrix}. \tag{2.49}$$

The finite element method theoretically allows for different mass properties in the directions x, y and z. Practically, this is usually not the case and leads to the assumption

$$M = M_{xx} = M_{yy} = M_{zz} \tag{2.50}$$

Therefore, the total aircraft mass M can be derived from the upper left corner of M_0. The upper right corner contains a coupling between translational and rotational mass terms in the form of rM. At the center of gravity, they should become zero. With this information, these terms can be used to establish the offset r_b of the center of gravity.

$$\mathbf{r}_b = [r_x M_{yy}, r_y M_{zz}, r_z M_{xx}]^T / M \tag{2.51}$$

In the second step, the final rigid body mass matrix $\hat{\mathbf{M}}_b$ is calculated with respect to the body coordinate system 'b' located at the CG.

$$\hat{\mathbf{M}}_b = \begin{bmatrix} \mathbf{M}_b & \\ & \mathbf{I}_b \end{bmatrix} = \begin{bmatrix} M_{xx} & & & & & \\ & M_{yy} & & & & \\ & & M_{zz} & & & \\ & & & I_{xx} & I_{xy} & I_{xz} \\ & & & & I_{yy} & I_{yz} \\ & & & & & I_{zz} \end{bmatrix} \tag{2.52}$$

In the upper left corner, mass matrix $\mathbf{M}_b$ is given. In the lower right corner, the inertia matrix $\mathbf{I}_b$ is located, including moments of inertia I_{xx}, I_{yy} and I_{zz} about the the the x, y and z axis of the body coordinate system 'b', plus some additional coupling terms I_{xy}, I_{xz} and I_{yz}. All remaining entries should become zero. Matrices $\mathbf{M}_b$ and $\mathbf{I}_b$ are important to calculate the rigid body motion of the aircraft, which is presented in the next section.

2.5 Equations of Motion

The motion of the aircraft is divided into a rigid and a flexible part. For the rigid body motion, the aircraft is considered as a point mass with inertia matrices $\mathbf{M}_b$ and $\mathbf{I}_b$, where the components of the inertia tensor $\mathbf{I}_b$ are calculated with respect to the body axes 'b'. Its origin is

positioned at the center of gravity. All external forces and moments $\mathbf{p}_g^{ext}$ are gathered at the same point. The non-linear equations of motion are given by

$$\dot{\mathbf{v}}_b = \mathbf{M}_b^{-1} \cdot \mathbf{p}_b^{\text{ext, forces}} + \mathbf{v}_b \times \boldsymbol{\omega}_b + \dot{\mathbf{v}}_b^{\text{grav}} \tag{2.53}$$

and

$$\dot{\boldsymbol{\omega}}_b = \mathbf{I}_b^{-1} \cdot \left(\mathbf{p}_b^{\text{ext, moments}} - \boldsymbol{\omega}_b \times (\mathbf{I}_b \cdot \boldsymbol{\omega}_b) \right) \tag{2.54}$$

yield the translational and rotational accelerations $\dot{\boldsymbol{\omega}}_b$ and $\dot{\mathbf{v}}_b$ of the aircraft body frame. The coupling terms between translation and rotation $\mathbf{v}_b \times \boldsymbol{\omega}_b$ and $\boldsymbol{\omega}_b \times (\mathbf{I}_b \cdot \boldsymbol{\omega}_b)$ are derived by Waszak, Schmidt and Buttrill [11,59,60]. Gravitational acceleration is accounted for by $\dot{\mathbf{v}}_b^{\text{grav}}$ in equation (2.53).

In addition to the rigid body motion of the aircraft, linear structural dynamics are incorporated by

$$\mathbf{M}_{ff} \ddot{\mathbf{u}}_f + \mathbf{D}_{ff} \dot{\mathbf{u}}_f + \mathbf{K}_{ff} \mathbf{u}_f = \mathbf{p}_f^{ext} \tag{2.55}$$

Here, generalized external forces $\mathbf{p}_f^{ext}$ interact with linear elastic deflections $\mathbf{u}_f$, velocities $\dot{\mathbf{u}}_f$ and accelerations $\ddot{\mathbf{u}}_f$. The matrices $\mathbf{M}_{ff}$, $\mathbf{D}_{ff}$ and $\mathbf{K}_{ff}$ refer to the generalized mass, damping, and stiffness matrices from equations (2.45) to (2.47) from Section 2.4.

2.6 System of Equations and Quasi-Steady Solution Schemes

The calculation of aerodynamic forces and the evaluation of the equation of motion described in the previous sections are transformed into a single set of coupled equations. For the solution of this system, it is convenient to convert the equations into a first order system:

$$\begin{pmatrix} \dot{\mathbf{u}}_i \\ \ddot{\mathbf{u}}_b \\ \dot{\mathbf{u}}_f \\ \ddot{\mathbf{u}}_f \\ \dot{\mathbf{u}}_{cs} \end{pmatrix} = \mathrm{f} \begin{pmatrix} \mathbf{u}_i \\ \dot{\mathbf{u}}_b \\ \mathbf{u}_f \\ \dot{\mathbf{u}}_f \\ \mathbf{u}_{cs} \end{pmatrix} . \tag{2.56}$$

The vector $\mathbf{u}_i$ contains the aircraft position and Euler angles $(x, y, z, \Phi, \Theta, \Psi)^T$ with respect to the earth-fixed frame 'i', vector $\dot{\mathbf{u}}_i$ describes the aircraft velocities and rates $(\dot{x}, \dot{y}, \dot{z}, \dot{\Phi}, \dot{\Theta}, \dot{\Psi})^T$. The vector $\dot{\mathbf{u}}_b$ contains the aircraft velocities and rates $(u, v, w, p, q, r)^T$ in

the body-fixed frame 'b', vector $\ddot{\mathbf{u}}_b$ the aircraft translational and rotational accelerations $(\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r})^T$. Vector $\mathbf{u}_{cs}$ contains the control commands about x, y and z axis $(\xi, \eta, \zeta)^T$.

To find the trimmed state of the aircraft, trim conditions need to be defined in such a way that the system is not over- or under-determined in order to calculate one unique solution of the equations. One example of such a trim condition is a horizontal level flight at a given velocity u. This requires the roll and pitch rates $\dot{p}, \dot{q}$ to be zero while the control surface deflections ξ, η are flagged as free. Yaw is omitted due to the lack of a dedicated control about the z axis. Equation (2.56) is then solved with Powell's non-linear root-finding algorithm [17,42,51]. This procedure is validated numerically [54] against MSC.Nastran SOL144. Once the initial flight condition is found, a time simulation may be started.

The above description of the solution of the trim problem uses a direct approach based on partial derivatives to construct the jacobian matrix. Note that the structural deformation is already included by terms $\mathbf{u}_f$, $\dot{\mathbf{u}}_f$ and $\ddot{\mathbf{u}}_f$. Using a different aerodynamic method such as CFD, the calculation of partial derivatives is no longer feasible, especially if many mode shapes are considered for structural deformation. Therefore, an iterative approach is selected as sketched in Figure 2.9. Based on a CFD solution for the undeformed geometry, the structural deformation is calculated and applied to the CFD mesh. The fluid structure interaction (FSI) loop is repeated until convergence is achieved. The FSI solution is then subject to an outer loop to determine the trim condition. The solution of the trim problem is found (as before) using Powell's non-linear root-finding algorithm. The difference is that the trim problem is more non-linear. Compared to the direct trim solution using potential aerodynamics, the tolerances for convergence need to be modified as both the CFD and the FSI solutions are solved iteratively, too. The challenge is the selection of an appropriate set of convergence criteria. Convergence of the CFD solution is determined using Cauchy convergence of the lift, drag, rolling and pitching moment coefficients. Convergence of the FSI loop is determined by an inspection of the maximal relative translational deformation between the current and the last loop. The following set of parameters is selected in this work:

- Convergence of the trim solution: $1.0 \cdot 10^{-3}$

- Convergence of fluid structure interaction: $|\Delta \mathbf{u}_{g,xyz}^f| \leq 1.0 \cdot 10^{-6}$

- Convergence of the CFD solution: Cauchy convergence with $\Delta C_l \leq 1.0 \cdot 10^{-4}$, $\Delta C_d \leq 1.0 \cdot 10^{-5}$, $\Delta C_{mx} \leq 1.0 \cdot 10^{-4}$ and $\Delta C_{my} \leq 1.0 \cdot 10^{-4}$ using 30 samples
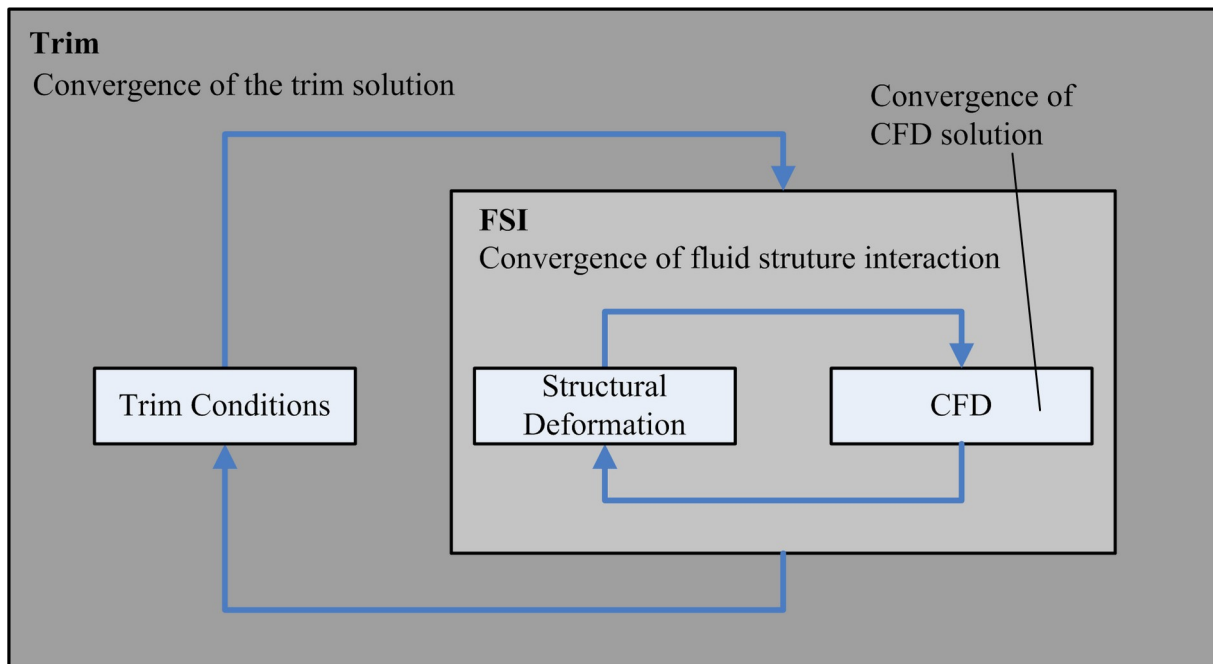

Figure 2.9: Sketch of iterative trim solution scheme

2.7 Aeroelastic Response in the Time Domain

The time simulation is performed by an integration of equation (2.56) over a period of time. In the case of unsteady aerodynamics, the lag states described in section 2.2 are added to the system. Two different integration schemes have been tested. The explicit runge-kutta method of 4th/5th order [15] and an implicit Adams-Bashforth method [10], both implemented in Scipy [50], have shown numerically equivalent results. Because of the fewer function evaluations, the Adams-Bashforth method is selected.

2.8 Aeroelastic Response in the Frequency Domain

For the frequency domain description, the rigid body motion is represented with the help of rigid body modes with index 'b'. The rigid body modes are created artificially by application of a unit translation and unit rotation about the center of gravity. To avoid numerical problems, the translation in x-direction is omitted. That modes are joined with the flexible modes with index 'f', creating the h-set.

The basic equation reads

$$\left[-\mathbf{M}_{hh} \cdot \omega^2 + \left(\mathbf{D}_{hh} - \mathbf{Q}_{hh}^2\right) \cdot i\omega + \mathbf{K}_{hh} - \mathbf{Q}_{hh}^1\right] \cdot \mathbf{u}_h = \mathbf{p}_h^{\text{excitation}} \quad , \quad (2.57)$$

where $\mathbf{Q}_{hh}^1$ and $\mathbf{Q}_{hh}^2$ are the velocity and deformation dependent aerodynamic terms. While in the time domain the downwash is calculated individually, compare equation (2.9), and using a python function, in the frequency domain the calculation of the downwash is realized based on matrices $\mathbf{D}_{jh}^1$ and $\mathbf{D}_{jh}^2$ for the velocity and deformation dependent downwash. Matrices $\mathbf{Q}_{hh}^1$ and $\mathbf{Q}_{hh}^2$ are given by

$$\mathbf{Q}_{hh}^1 = q_\infty \cdot \mathbf{\Phi}_{lh}^T \cdot \mathbf{N} \cdot \mathbf{A} \cdot \mathbf{AIC} \cdot \mathbf{D}_{jh}^1 \qquad (2.58)$$

and

$$\mathbf{Q}_{hh}^2 = q_\infty \cdot \mathbf{\Phi}_{lh}^T \cdot \mathbf{N} \cdot \mathbf{A} \cdot \mathbf{AIC} \cdot \mathbf{D}_{jh}^2 / V_{TAS} \qquad (2.59)$$

with the panel areas represented in matrix $\mathbf{A}$ and the normal vectors in $\mathbf{N}$. Matrix $\mathbf{\Phi}_{lh}^T$ projects the nodal aerodynamic forces from physical into generalized coordinates.

With

$$\omega = 2\pi \cdot f \qquad (2.60)$$

the transfer function

$$\mathbf{TF} = \left[-\mathbf{M}_{hh} \cdot \omega^2 + \left(\mathbf{D}_{hh} - \mathbf{Q}_{hh}^2\right) \cdot i\omega + \mathbf{K}_{hh} - \mathbf{Q}_{hh}^1\right]^{-1} \qquad (2.61)$$

is constructed for every frequency sample. Because the $\mathbf{AIC}$ matrices are only calculated for selected reduced frequencies k, this step requires a linear interpolation of matrices $\mathbf{Q}_{hh}^1$ and $\mathbf{Q}_{hh}^2$. Extrapolation to lower or higher reduced frequencies is usually necessary but should be done with care and for a limited range only.

The gust excitation requires a Fourier transformation of the induced gust velocities $\mathbf{w}_j^{\text{gust}}$ for every aerodynamic panel. The nodal and generalized aerodynamic gust forces $\mathbf{p}_k^{\text{gust}}$ and $\mathbf{p}_h^{\text{gust}}$ are given by

$$\mathbf{p}_k = q_\infty \cdot \mathbf{\Phi}_{lk}^T \cdot \mathbf{N} \cdot \mathbf{A} \cdot \mathbf{AIC} \cdot \mathbf{w}_j \qquad (2.62)$$

and

$$\mathbf{p}_h = q_\infty \cdot \mathbf{\Phi}_{lh}^T \cdot \mathbf{N} \cdot \mathbf{A} \cdot \mathbf{AIC} \cdot \mathbf{w}_j \, . \qquad (2.63)$$

The Fourier transformation usually results in positive and negative frequencies but the $\mathbf{AIC}$ matrices are only defined for positive frequencies. Therefor, the above operations can only be performed for positive frequencies and the results need to be mirrored using the complex conjugate. Note that this procedure is very error-prone and differs for odd and even numbers of frequency samples.

The aeroelastic responses of the aircraft due to gust is calculated in terms of generalized deformation $\mathbf{u}_h$ by simple matrix multiplication

$$\mathbf{u}_h = \mathbf{TF} \cdot \mathbf{p}_h^{\mathrm{gust}}, \tag{2.64}$$

repeated for every sample frequency. The generalized velocity and acceleration is given by

$$\dot{\mathbf{u}}_\mathbf{h} = \mathbf{u}_h \cdot i\omega \tag{2.65}$$

and

$$\ddot{\mathbf{u}}_\mathbf{h} = \mathbf{u}_h \cdot (i\omega)^2. \tag{2.66}$$

The frequency domain results are then transferred into the time domain using the inverse Fourier transformation.

For application of the force summation method, see section 3.1, the distributed nodal aerodynamic forces in physical coordinates $\mathbf{p}_k^{\mathrm{aero}}$ need to be reconstructed based on equation (2.62). This is the most time-consuming part of the frequency domain analysis as it requires an interpolation of the $\mathbf{AIC}$ matrix for every frequency sample.

The highest frequency is given by $f_{\max} = 1/\Delta t$ and the frequency resolution by $\Delta f = f_{\max}/n_{\mathrm{samples}}$. The frequency $f_{\max}$ should be sufficiently high to resolve at least all structural modes but not too high to avoid excessive extrapolation of the $\mathbf{AIC}$ matrices, which depend on the reduced frequency k. The frequency resolution Δf should be sufficiently small to resolve rigid body motion with low frequency content. With $t_{\mathrm{final}} = \Delta t \cdot n_{\mathrm{samples}}$, this typically results in much long simulation times compared to the time span of the gust encounter itself.

2.9 Stability Analysis

The stability analysis differs slightly from the frequency domain description presented in the previous section. The basic principle is to set the right hand side to zero and to determine the

eigenvalues of that system at increasing speeds. There are different solution schemes for this task, three of them are considered in this work.

The **K-Method** is probably the most simple but the fastest method. The formulation follows closely the description given by Schwochow [48], section 7.7.1, page 110. Similar to equation (2.57), the basic equation reads

$$\left[-\mathbf{M}_{hh} \cdot \omega^2 + \mathbf{K}_{hh} \cdot (1 + i\gamma) - q_\infty \cdot \mathbf{Q}_{hh}\right] \cdot \mathbf{u}_h = 0 \,, \tag{2.67}$$

where the velocity and deformation dependent contributions to the induced downwash given by matrices $\mathbf{D}_{jh}^1$ and $\mathbf{D}_{jh}^2$ are combined to one single term

$$\mathbf{Q}_{hh} = \mathbf{\Phi}_{lh}^T \cdot \mathbf{N} \cdot \mathbf{A} \cdot \mathbf{AIC} \cdot \left(\mathbf{D}_{hj}^1 + \mathbf{D}_{hj}^2 \cdot \frac{2ik}{c_\text{ref}}\right). \tag{2.68}$$

Division of equation (2.67) by ω^2 and rearrangement of the aerodynamic term $\mathbf{Q}_{hh}$ leads to

$$\left[\left(-\mathbf{M}_{hh} - \frac{\rho}{2} \cdot \frac{(c_\text{ref}/2)^2}{k^2} \cdot \mathbf{Q}_{hh}\right) + \mathbf{K}_{hh} \cdot \left(\frac{1 + i\gamma}{\omega^2}\right)\right] \cdot \mathbf{u}_h = 0 \tag{2.69}$$

with the eigenvalue

$$\lambda = \frac{1 + i\gamma}{\omega^2}. \tag{2.70}$$

The eigenvalues are calculated for the first frequency sample and sorted from low to high frequencies. To track the modes of the following samples, the MAC value is used to compare the eigenvector of the current solution with the previous one.

In a last step, the frequency f, the damping g and the speed V can be calculated from the eigenvalues with

$$f = \frac{1}{2\pi\sqrt{\lambda^\text{Re}}} \,, \tag{2.71}$$

$$g = \frac{\lambda^\text{Im}}{\lambda^\text{Re}} \,, \tag{2.72}$$

and

$$V = \frac{c_\text{ref}/2}{k\sqrt{\lambda^\text{Re}}}. \tag{2.73}$$

The **KE-Method** used in Nastran is similar to the K-Method. The formulation given by Rodden [45] places the eigenvalue with the mass term, resulting in

$$\left[\left(\frac{k^2}{(c_{\text{ref}}/2)^2} \cdot \mathbf{M}_{hh} + \frac{\rho}{2} \cdot \mathbf{Q}_{hh} \right) \cdot \left(\frac{-V^2}{1 + i\gamma} \right) + \mathbf{K}_{hh} \right] \cdot \mathbf{u}_h = 0 \qquad (2.74)$$

with the eigenvalue

$$\lambda = \frac{-V^2}{1 + i\gamma} \, . \qquad (2.75)$$

The eigenvalues are calculated for the first and second frequency sample and sorted from low to high frequencies. To track the modes of the following samples, the next expected eigenvalue is extrapolated based on the two previous eigenvalues.

The frequency f, the damping g and the speed V is given by

$$f = \frac{k}{\pi c_{\text{ref}}} \, , \qquad (2.76)$$

$$g = -\frac{\lambda^{\text{Im}}}{\lambda^{\text{Re}}} \, , \qquad (2.77)$$

and

$$V = \sqrt{\frac{\lambda^{\text{Re}^2} + \lambda^{\text{Im}^2}}{\lambda^{\text{Re}}}} \, . \qquad (2.78)$$

The formulations of the K and the KE-Method are very similar and can be converted from one to the other. The two main distinctions are the definition of the eigenvalue and the mode tracking technique. Both methods don't account for structural damping and are only valid at the point of flutter, where the damping is zero. Another disadvantage is that the sample points are defined by the reduced frequency and not by flight speed, which requires many samples and gives the flutter curves a distinct look, see section Error: Reference source not found, and gives no information for low flight speeds. Using a cubic inter- and extrapolation of the aerodynamic matrices leads to smooth flutter curves.

A more physical representation of the damping is given by the **PK-Method** proposed by Hassig [22]. This leads to an eigenvalue problem of second order, which is converted into an eigenvalue problem of first order but of double size. The result are complex conjugate eigenvalue pairs, which are more difficult to interpret physically, especially with respect to the rigid body modes. Also, the sample points are defined by the flight speed, which gives nicer flutter curves, but requires an iterative procedure to match the reduced frequency with the

flight speed for every mode and at every sample point. In this implementation, the iteration scheme as proposed by Rodden [45] is used. Because at low flight speeds the reduced frequency may become very large, only a linear inter- and extrapolation of the aerodynamic matrices is feasible.

The state-space notation of equation (2.57) is

$$\mathbf{A} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{M}_{hh}^{-1} \cdot \left(\mathbf{K}_{hh} - \frac{\rho}{2} V^2 \cdot \mathbf{Q}_{hh}^1 \right) & -\mathbf{M}_{hh}^{-1} \cdot \left(\mathbf{D}_{hh} - \frac{\rho}{2} V \cdot \mathbf{Q}_{hh}^2 \right) \end{bmatrix}. \tag{2.79}$$

Matrices $\mathbf{Q}_{hh}^1$ and $\mathbf{Q}_{hh}^2$ are defined similarly to equations (2.58) and (2.59), but without the dynamic pressure and flight speed as

$$\mathbf{Q}_{hh}^1 = \mathbf{\Phi}_{lh}^T \cdot \mathbf{N} \cdot \mathbf{A} \cdot \mathbf{AIC} \cdot \mathbf{D}_{jh}^1 \tag{2.80}$$

and

$$\mathbf{Q}_{hh}^2 = \mathbf{\Phi}_{lh}^T \cdot \mathbf{N} \cdot \mathbf{A} \cdot \mathbf{AIC} \cdot \mathbf{D}_{jh}^2. \tag{2.81}$$

The eigenvalues of matrix $\mathbf{A}$ are determined for the first flight speed with a reduced frequency $k = 0.0$. The oscillatory eigenvalues appear as complex conjugate pairs, where only those with positive imaginary part are of physical interest. The selection criterion is that the imaginary part must be $\lambda^{Im} \geq 1e^{-6}$. Note that at low flight speeds, the rigid body modes might be non-oscillatory due to missing structural stiffness and very small aerodynamic terms. This results in eigenvalues of very low imaginary part but large real part. To enable a good mode tracking, it is recommended to chose the first flight speed sufficiently high, such that all rigid body modes of interest do oscillate at that point. Although modes might not be shown in the flutter curves due to bad eigenvalue identification, their influence and interaction with other oscillatory modes is still captured correctly.

From this starting point, the reduced frequency k needs to be matched with the flight speed V for each mode of frequency f. Convergence of that iterative procedure is determined when the difference Δk between step j and step $j-1$ is $\Delta k \leq 1e^{-4}$.

The frequency f and damping g is given by

$$f = \frac{\lambda^{Im}}{2\pi} \tag{2.82}$$

and

34

$$g = 2 \cdot \frac{\lambda^{Re}}{\lambda^{Im}} \,. \tag{2.83}$$

Due to the formulation of the damping coefficient g, negative values indicate a stable system. Also, note that there is a factor of 2.0 between the structural damping d and the damping coefficient g. For example, a modal damping of $d = 2.0\%$ corresponds to $g = -0.04$.

2.10 Landing Impact

The landing gear of an aircraft has to fulfill several purposes. An overview is given by Krüger et al. [33]. State of the art aircraft landing simulations are normally carried out using multi body simulation techniques, e.g. as described by Krüger [32] and Cumnuantip [14], and include a detailed model of the landing gear and tires. However, the aircraft is often assumed as a rigid body, neglecting the dynamic response of the aircraft's flexible structure. There are several possibilities to address this shortcoming. One approach is to incorporate the aircraft's structural properties in the multi body simulation environment. This can be achieved by a modal representation of the aircraft as Lemmens [37] demonstrates for a business jet. Castrichini et al. [12] even include unsteady aerodynamics for the calculation of both gust and ground loads. An alternative to the modal representation is the discretisation of the elastic structure by means of rigid bodies, which are connected by rotational springs to account for wing bending and rotational stiffness, as shown by Krüger [31]. A different approach by Jaques and Garrigues [24] uses a dynamic, transient finite elements (FE) analysis. Special nonlinear elements, joints and hinges are added to the FE code to describe the behavior of the landing gear.

For aircraft loads analyses, the primary task is to analyze the effect of the landing impact on the aircraft structure and to include the resulting loads in the sizing process. Therefore, the selected approach in this work includes the landing gear directly within the transient, dynamic loads simulation. For a sizing procedure in the preliminary design phase, some simplifications may be made while maintaining the key elements, which are explained in the following. The emphasis lies on the absorption of the vertical kinetic energy occurring during the landing impact. Therefore, the landing gear module considers the vertical components of all accelerations, velocities, translations and forces. The following equations are derived from [14,32].
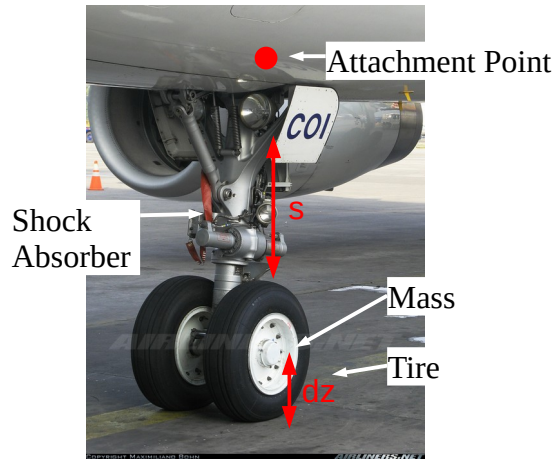
*Figure 2.10: Key feature of a typical nose
landing gear, photograph from [8]*

A typical landing gear of a large aircraft is shown in Figure 2.10. One key feature is a hydropneumatic air and oil shock absorber. The gas spring force F_f

$$F_f = F_0 \left(\frac{1-s}{s_m} \right)^{-n \cdot c_k} \tag{2.84}$$

is calculated based on a pre-stress force F_0, a stroke length s, a maximal stroke s_m and a polytropic coefficient n with $1 \leq n \leq \kappa$. The damping force F_d

$$F_d = sign(\dot{s}) \cdot d \cdot \dot{s}^2 \tag{2.85}$$

depends on the stroke velocity $\dot{s}$ and damping coefficient d.

For the tires, a linear behavior is assumed, forces act in z-direction and only when the tire makes contact with ground. Its deflection d_z is determined by subtracting its rolling radius r from its nominal radius r_{nom}. This leads to

$$F_{\mathrm{tire}} = c \cdot d_z + d \cdot \dot{d}_z \tag{2.86}$$

with a tire stiffness c and damping coefficient d. In addition, the tire may have a mass m_{tire}, causing a force F_{acc}

$$F_{\mathrm{acc}} = m_{\mathrm{tire}} \cdot \ddot{d}_z \tag{2.87}$$

In a next step, the landing gear model is incorporated into the time simulation. The positions, velocities and accelerations of the landing gear attachment point, indicated in Figure 2.10, are extracted at every time step and fed into the landing gear module. The landing gear reaction forces are then applied as external forces $\mathbf{p}_g^{\mathrm{ext}}$ on the aircraft. As the landing gear model is

evaluated "on-the-fly", the interaction between aircraft and landing gear is captured. In this way, the landing gear forces are counteracted by the aircraft's inertia, leading to a balanced set of loads.

3 Loads and Aeroelastic Analysis

3.1 Loads Recovery and Identification of Dimensioning Loads

The resulting nodal loads acting on the aircraft structure may be calculated by two different methods, the mode displacement method [6] and the force summation method [44]. The convergence of the mode displacement method (MDM), given by

$$\mathbf{p}_g = \mathbf{K}_g \cdot \mathbf{u}_g = \mathbf{K}_g \cdot \mathbf{\Phi}_{fg} \cdot \mathbf{u}_f \,, \tag{3.1}$$

strongly depends on the number of modes considered for the modal deformation vector $\mathbf{u}_f$. The more modes are used, the more precise is the result. Using all modes, both methods should lead to identical results.

With the force summation method (FSM), given by

$$\mathbf{p}_g = \mathbf{p}_g^{ext} + \mathbf{p}_g^{iner} \,, \tag{3.2}$$

the calculation is done using physical coordinates and the sum of inertia and external forces leads to the loads that are carried by the structure. Because of the more precise results, the force summation method is selected. Note that this only possible because the approximation of the unsteady aerodynamics (RFA) is performed in physical coordinates, see Section 2.2, and is thereby harmonized with the FSM. For most other applications, such as stability, flutter or aeroservoelasticity, there is no need to recover nodal forces and the approximation in generalized coordinates is sufficient.

In a typical loads analysis campaign for aircraft certification, several thousands of load cases need to be computed. This is because structural strength has to be demonstrated for all maneuvers at all flight points. Thus, the maneuver cases need to be combined with all mass configurations, flight speeds, altitudes, Mach numbers, etc. The difficulty is to determine the

load cases that yield the dimensioning loads in advance. To reduce the amount of guesswork and uncertainties already during the preliminary design, emphasis is put on a comprehensive loads process including a large number of load cases to ensure a thorough design.

From the nodal loads $\mathbf{p}_g$ obtained from equation (3.2), so-called interesting quantities are calculated. Interesting quantities usually include section forces and moments at various stations (e.g. along the wing) and attachment loads (e.g. from control surfaces, payload, landing gear, etc.). These quantities are calculated at monitoring stations. The calculation of section loads at the monitoring stations $\mathbf{p}_{\text{mon}}$ involves an integration of the nodal loads $\mathbf{p}_g$, which can be expressed as a matrix multiplication.

$$\mathbf{p}_{\text{mon}} = \mathbf{\Phi}_{g,\text{mon}} \cdot \mathbf{p}_g \tag{3.3}$$

The section loads can be plotted as one-dimensional envelopes that show the section loads at one aircraft component, e.g. along the wing. The minimum and maximum values determine the highest loads, compare Wright and Cooper [61], chapter 18.9.1, Figure 18.17. With this procedure, only one quantity can be examined. A more sophisticated approach can be realized with two-dimensional loads envelopes that show a combination of two quantities, for a wing typically the shear force F_z, bending moment M_x or torsional moment M_y. These plots, sometimes referred to as SMT-Plots, are very important for the dimensioning of an aircraft structure. Typical examples are shown in Figure 3.1. Every dot corresponds to one load case. In a next step, the convex hull is drawn to identify not only the minimum and maximum values of one quantity but also the minimum and maximum of two combined quantities. This approach is very useful for maneuver loads, but not directly applicable to gust or landing loads.
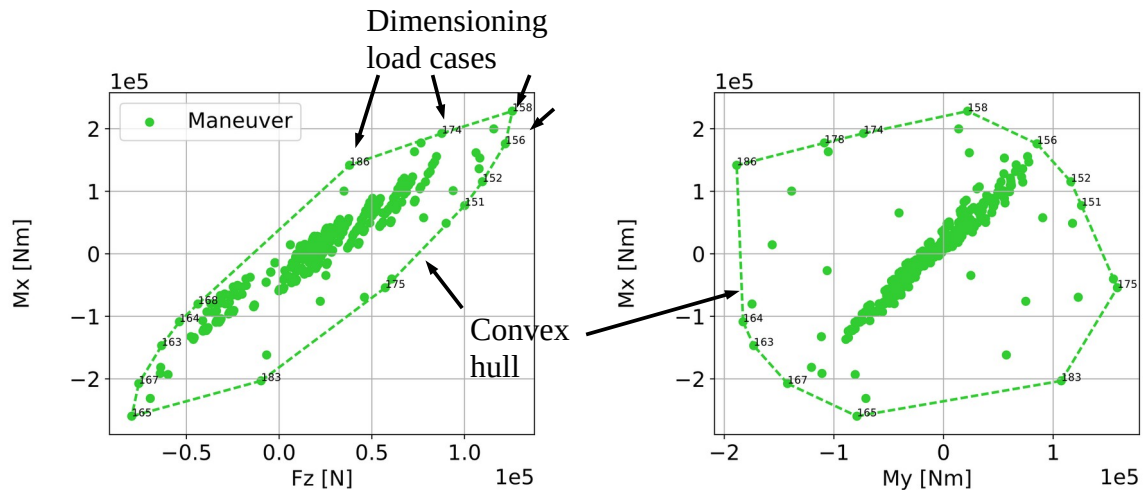
Figure 3.1: Identification of dimensioning load cases using loads envelopes and the convex hull

In contrast to maneuver loads, 1-cos gust or landing loads are the result of a time simulation. Therefore, they can no longer be represented by one single dot as shown previously in Figure 3.1. The approach selected in this work involves an extraction of several snapshots from the time simulation. In this way, dynamic loads are transferred into quasi-static loads and are usable for a dimensioning process. The snapshots are selected by identifying the minimum and maximum values at every monitoring station for every interesting quantity. An example is shown in Figure 3.2. Every dot corresponds to one snapshot, which has been identified from the time simulation, and one gust simulation produces several dots. A closer look reveals that the right upper corner of the convex hull is formed by three snapshots from gust load case number 46 at the time steps t=0.185s, 0.190s and 0.195s.
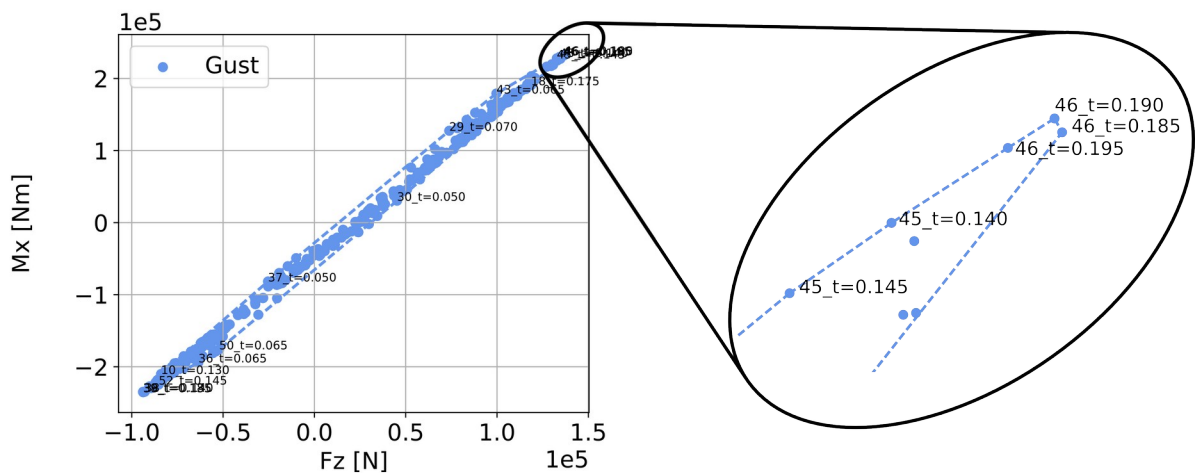

Figure 3.2: Identification of dimensioning dynamic load cases using time snapshots

One can also see that the three selected snapshots are very close to each other. In general, the number of dimensioning load cases could be reduced further by using a more conservative shape than the convex hull.

3.2   Assembly of the Methods for Flight Loads Analysis

The methods and theories presented in the Sections 2.1 through 3.1 are prepared and tested in simple scripts, first using academic examples, then with more complex aircraft configurations. Whenever possible, comparisons are made to other software. The individual scripts are integrated into functions and modules to be assembled to the Loads Kernel software package. Therefore, the development can be considered as a major contribution to this thesis and is inseparably connected to this work. From a technical perspective, the following points are improved with respect to the standard solution by MSC.Nastran:

- VLM and DLM Code, translated to python
  - Linear and nonlinear version, induced drag
- Aero-structural coupling (not new, but required e.g. for CFD surface mesh deformation)
  - Rigid body spline (with nearest neighbor search)
  - Surface and volume spline (radial basis functions)
- Trim conditions for different types of maneuver
  - Pitch-, roll-, yaw-maneuver, landing at constant sink rate, gliding with loss of altitude at constant speed, ...
  - EFCS for control surface scheduling
- Linear and non-linear equation for rigid body motion
  - Free flying, flexible aircraft in time domain
  - Flight mechanics included → good for comparisons with flight test
  - Rigid body motion captured correctly (aperiodic motion difficult in frequency domain)

- ○ Non-linear control possible (not possible in frequency domain)

- ○ Non-linear external forces, e.g. from landing gear

- CFD (DLR Tau Code) for maneuver loads

  - ○ Structural flexibility and control surfaces via mesh deformation

  - ○ CFD surface mesh deformation by LK → volume deformation by Tau

  - ○ Directly comparable to VLM-based solution: same trim conditions, matrices, ...

  - ○ Benefits form LK's integration in our loads process

The Loads Kernel is split into three processing steps:

- pre

- main

- post

These three steps need to be performed sequentially, but can be performed individually. The process flow is sketched in Figures 3.3 and 3.4. The data is stored to the file system after every step. This is an advantage if for example only the load case definitions are changed while the model itself remains unchanged. In that case, only the main- and post-processing have to be repeated. This is also beneficial if the main-processing is to be run on a different computer or a high performance cluster. The resulting model data from the pre-processing can be copied to any location, where the main-processing is started. The resulting response may be copied back to the local file system for the post-processing step. Indeed, the main-processing is parallelized in the sense that $n - 1$ load cases are calculated in parallel where n is the number of local CPUs. This is achieved by a worker / listener concept with a pool of $n - 1$ workers and one listener which collects the results. The worker and the listener communicate via a queuing system available in the Python multiprocessing module.

During the pre-processing, all required model data is read, processed and assembled to one model. Input to the pre-processing are the mass and stiffness matrices, the FE geometry and the aerodynamic panel mesh. The pre-processing is designed in such a way that it directly interfaces with the data and files prepared by ModGen. The mass and stiffness matrices are

exported from MSC.Nastran in the op4 file format, which is achieved with a DMAP alter. Technically, MSC.Nastran is only used in the role of a pre-processor and the use of any other FE processor is thinkable. From the mass matrix, the center of gravity, the mass and the corresponding moments of inertia are calculated as described in Section 2.4. Optionally, a Guyan reduction is applied. Next, a modal analysis is conducted to determine the eigenvalues and the eigenvector. Finally, the generalized mass and stiffness matrices are calculated. This procedure is repeated for all mass cases. For the definition of the aerodynamic panel mesh, CAERO4, CAERO7 or CQUAD4 cards are parsed. CAERO4 are used in MSC.Nastran while CAERO7 are used in ZAERO. With the aerodynamic panel mesh, the VLM and the DLM routines are started, according to the requirement of steady or unsteady aerodynamics. In the case of unsteady aerodynamics, the rational function approximation as described in Section 2.2 is performed automatically. This procedure is repeated for all Mach numbers defined in the operation parameters by the user. In addition, the AESURF and AELIST cards defining the control surfaces are parsed. With the help of a rigid body spline, deflection matrices are constructed that allow the calculation of the induced downwash on a control surface due to a given rotation angle about a hinge line. The aero-structural coupling depends on the FE geometry and the aerodynamic panel mesh. As described in Section 2.3, either a rigid body spline or a surface or volume spline based on radial basis functions may be used. The resulting coupling matrix is stored in the model as a sparse matrix to save memory and computational time during the main-processing. In a next step, the parameters of the international standard atmosphere (ISA) are calculated with respect to the altitudes requested in the operation parameters. Finally, a matrix for the integration of section loads is built based on the monitoring stations defined by the user. The resulting model data is stored to disk.

Remark: Because the ISA documents are not publicly available, the US standard atmosphere [64], which is equivalent up to 32 km altitude, may be taken as a convenient alternative reference.

The run time of the pre-processing can be very rapid for small, condensed models and rather lengthy if large FE models are involved or unsteady AIC matrices need to be calculated for many reduced frequencies and Mach numbers. As mentioned at the beginning, it is beneficial that the pre-processing has to be done only once if no changes are made to the model, which is an advantage with respect to commercial software.

The main-processing depends on the model data prepared in the pre-processing. In addition, a load case definition is needed as input. The load case definition contains parameters for every load case that shall be calculated and typically comprises parameters such as the type of maneuver, the mass configuration, flight speed, altitude, load factor, rates, accelerations, sub case identification number etc. In addition to the sub case identification number, the user may assign a descriptive string, preferably according to the nomenclature defined by Chiozzotto [41]. With this input, the first step of the main-processing is to establish a set of so-called trim conditions. The aircraft states are set as free or fixed according to the selected maneuver and in such a way that the system is neither over- nor under-determined. Next, the solution of this system is computed, yielding a trimmed aircraft as described in Section 2.5. For gust and landing loads, a time simulation is started, using the trimmed aircraft as an initial solution. Nodal loads are recovered from the aircraft states using the force summation method (FSM). In a next step, these nodal loads are integrated to section loads. In case of a time simulation, the dynamic section loads need to be transferred into single snapshots as described in Section 3.1. In addition, the nodal loads are translated into a global coordinate system. The results of the main-processing are stored to disk in a response file.

The calculation time for one maneuver load case is below one second for all models that have ever been used with the Loads Kernel. Therefore, even several hundreds of load cases are calculated within some minutes. Time domain simulations are a little more time consuming as they usually involve structural dynamics and unsteady aerodynamics. In most cases, the corresponding states change rapidly during the time simulation, resulting in very small time steps of the integration scheme, selected in Section 2.5. This is desirable in order to achieve good results but comes at the cost of many function evaluations. Changes of the maximum allowable time step size were tested and finally increased slightly to $\Delta t_{\max} = 5 \cdot 10^{-4} s$. The relative tolerance is set to $tol_r = 10^{-6}$. Using these parameters, no degradation of the results could be observed.

The post-processing is dedicated to the evaluation, visualization and export of the results. The dimensioning load cases are identified as described in Section 3.1. The automated plotting allows for a fast control of the results and a quick detection of erroneous data by the user. For maximum compatibility, currently four different formats are supported for the export of the corresponding nodal loads:

- MSC.Nastran format using FORCE and MOMENT cards

- Internal, hierarchical format using the pickle module in Python

- DLR CPACS format using the Tixi XML interface library [49]

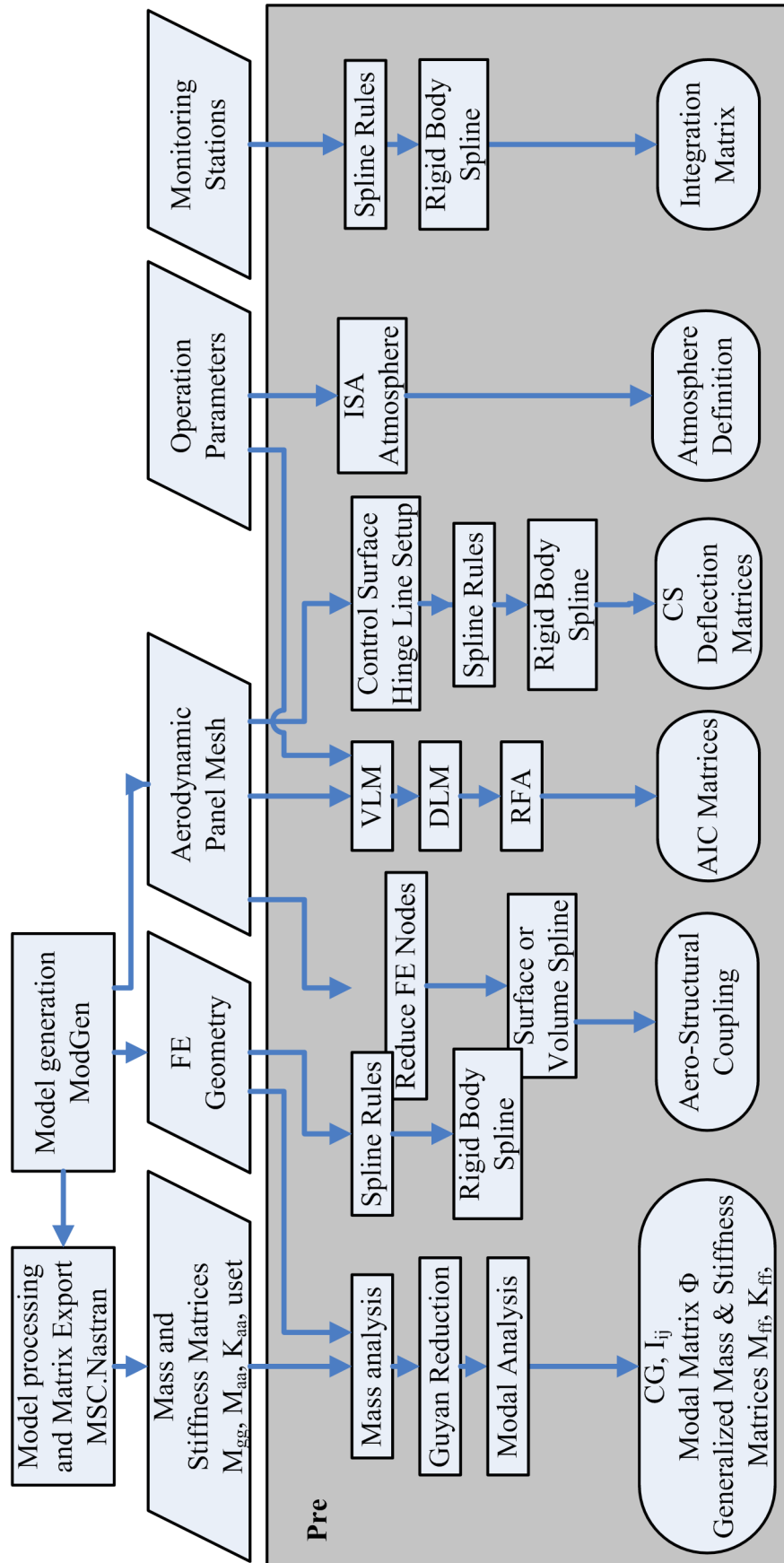- Matlab format using the scipy.io module in Python

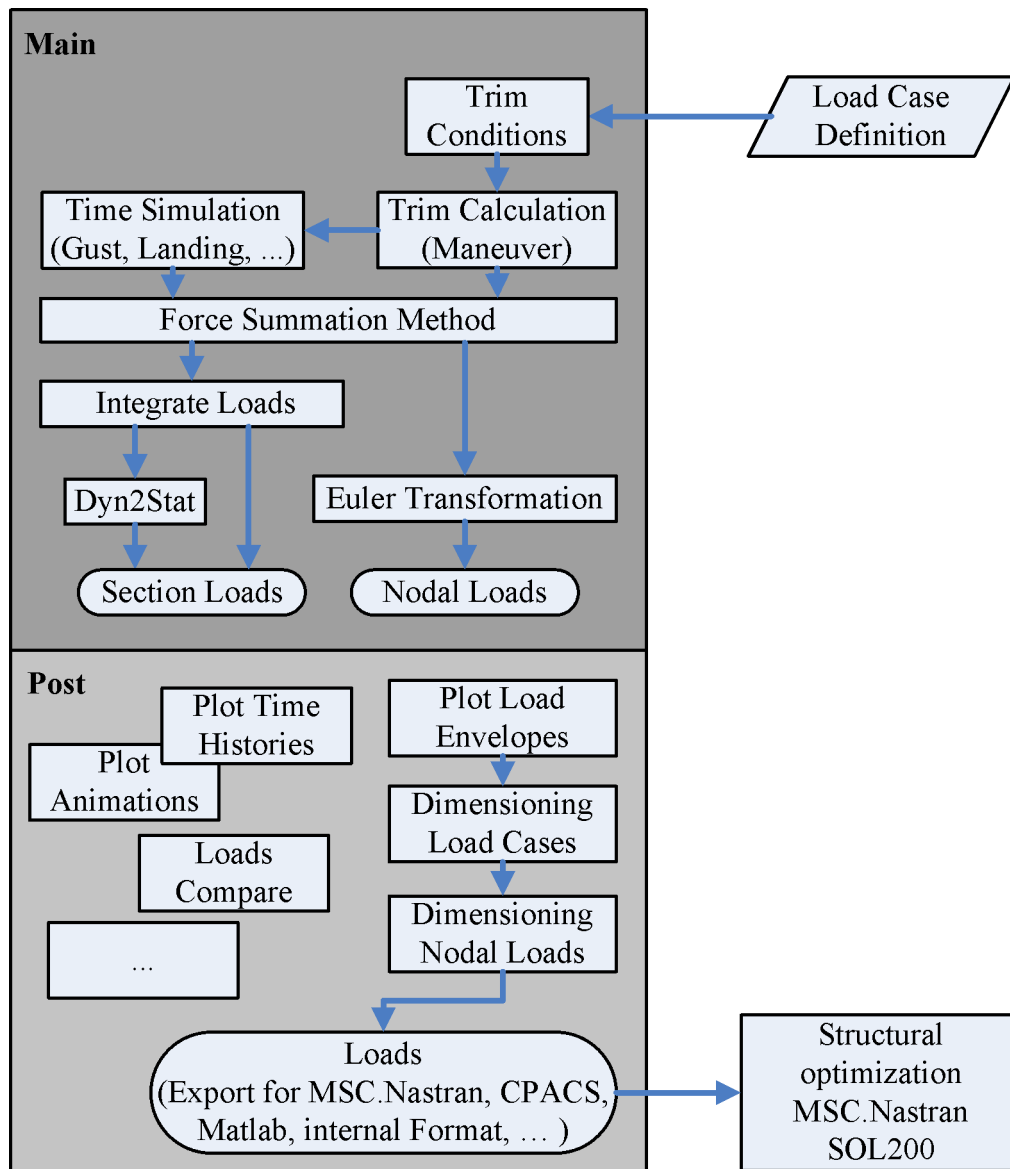*Figure 3.3: Schematic process flow of the Loads Kernel pre-processing*

Figure 3.4: Schematic process flow of the Loads Kernel main and post-processing

4 Input Definitions

4.1 Job Control

Because of the number of input parameters and possible changes, please see ./doc/jcl_documentation.py for a detailed description of all inputs and simulation parameters.

4.2 Examples

Two aircraft with different simulation set-ups are available:

- Discus2c sailplane

- Allegra configuration with forward-swept wings and T-tail

They can be found in the examples repository: https://phabricator.ae.go.dlr.de/diffusion/117/

4.3 Tips for Aeroelastic Modeling

This section gives hints for typical mistakes to help the user and to create less problems during the transfer of the aeroelastic model from Nastran to Loads Kernel.

- Nastran models created by ModGen are usually parsed without complications.

- Nastran models from other sources are possible and have been integrated successfully, but note that all Nastran input cards must be in "8-character-notation".

- The FE model should be "clean" in the sens that there are no unconnected GRID points.

- The aerodynamic panel model should be defined from left to right, such that the normal vectors point in the same direction for both left and right wing. Loads Kernel

should handle arbitrary panel orientations, however, not all combinations have been tested yet. In addition, other codes might have difficulties as well, leading to unnecessary mistakes.

- Only one control surface should be defined per AESURF card, even though two control surfaces are possible in Nastran.

Bibliography

[1]   Abel, I., "An Analytical Technique for Predicting the Characteristics of a Flexible Wing Equipped With as Active Flutter-Suppression System and Comparison With Wind-Tunnel Data," Langley Research Center, Hampton, Virginia, Technical Paper NASA Technical Paper 1367, Feb. 1979.

[2]   Albano, E., and Rodden, W. P., "A Doublet Lattice Method For Calculating Lift Distributions on Oscillation Surfaces in Subsonic Flows," in *AIAA 6th Aerospace Sciences Meeting*, New York, 1968.

[3]   Allemang, R. J., "The Modal Assurance Criterion – Twenty Years of Use and Abuse," *Sound and Vibration*, pp. 14–21, Aug. 2003.

[4]   Beckert, A., and Wendland, H., "Multivariate interpolation for fluid-structure-interaction problems using radial basis functions," *Aerospace Science and Technology*, vol. 5, no. 2, pp. 125–134, Feb. 2001, https://doi.org/10.1016/S1270-9638(00)01087-7.

[5]   Bianchi, J. P., Balmes, E., Des Roches, G. V., and Bobillot, A., "Using modal damping for full model transient analysis. Application to pantograph/catenary vibration," presented at the ISMA, 2010, https://hal.archives-ouvertes.fr/hal-00589953/.

[6]   Bisplinghoff, R. L., and Ashley, H., *Principles of Aeroelasticity*. Dover Publications, 2002, https://books.google.de/books?id=yVa8AQAAQBAJ.

[7]   Blair, M., "A Compilation of the Mathematics Leading to The Doublet Lattice Method," Airforce Wright Laboratory, Ohio, Technical Report WL-TR-92-3028, 1992.

[8]   Bohn, M., "Aviation Photo #1829344: Airbus A320-233 - LAN Airlines," *Airliners.net*, 09-Jun-2010. [Online]. Available: http://www.airliners.net/photo/LAN-Airlines/Airbus-A320-233/1829344. [Accessed: 03-Mar-2017].

[9]   Brockhaus, R., Alles, W., and Luckner, R., *Flugregelung*. Berlin: Springer Berlin, 2010, http://dx.doi.org/10.1007/978-3-642-01443-7.

[10]   Brown, P. N., Byrne, G. D., and Hindmarsh, A. C., "VODE: A Variable-Coefficient ODE Solver," *SIAM Journal on Scientific and Statistical Computing*, vol. 10, no. 5, pp. 1038–1051, Sep. 1989, https://doi.org/10.1137/0910062.

[11] Buttrill, C., Zeiler, T., and Arbuckle, P., "Nonlinear simulation of a flexible aircraft in maneuvering flight," in *Flight Simulation Technologies Conference, Guidance, Navigation, and Control and Co-located Conferences*, 1987, https://doi.org/10.2514/6.1987-2501.

[12] Castrichini, A., Cooper, J. E., Benoit, T., and Lemmens, Y., "Gust and Ground Loads Integration for Aircraft Landing Loads Prediction," presented at the 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Grapevine, Texas, 2017, http://arc.aiaa.org/doi/abs/10.2514/6.2017-0636.

[13] Cook, M. V., *Flight Dynamics Principles*, Second Edition. Amsterdam; Boston: Elsevier Ltd, 2007, http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=203095.

[14] Cumnuantip, S., "Landing gear positioning and structural mass optimization for a large blended wing body aircraft," Dissertation, TU München, Köln, 2014.

[15] Dormand, J. R., and Prince, P. J., "A family of embedded Runge-Kutta formulae," *Journal of Computational and Applied Mathematics*, vol. 6, no. 1, pp. 19–26, Mar. 1980, https://doi.org/10.1016/0771-050X(80)90013-3.

[16] European Aviation Safety Agency, ed., *Certification Specifications for Large Aeroplanes CS-25*, Amendment 16. 2015.

[17] Garbow, B. S., Hillstrom, K. E., and More, J. J., "minpack/hybrd.html," *Documentation for MINPACK subroutine HYBRD*, 1980. [Online]. Available: https://www.math.utah.edu/software/minpack/minpack/hybrd.html. [Accessed: 20-Apr-2017].

[18] Goggin, P., "A general gust and maneuver load analysis method to account for the effects of active control saturation and nonlinear aerodynamics," 1992, https://doi.org/10.2514/6.1992-2126.

[19] Gupta, K. K., Brenner, M. J., and Voelker, L. S., "Developement of an Integrated Aeroservoelastic Analysis Program and Correlation With Test Data," Dryden Flight Research Facility, Edwards, California, Technical Paper NASA Technical Paper 3120, 1991.

[20] Guyan, R., "Reduction of Stiffness and Mass Matrices," *AIAA Journal*, vol. 3, no. 2, p. 280, 1964.

[21] Harder, R. L., and Desmarais, R. N., "Interpolation using surface splines.," *Journal of Aircraft*, vol. 9, no. 2, pp. 189–191, Feb. 1972, https://doi.org/10.2514/3.44330.

[22] Hassig, H. J., "An approximate true damping solution of the flutter equation by determinant iteration.," *Journal of Aircraft*, vol. 8, no. 11, pp. 885–889, Nov. 1971, https://doi.org/10.2514/3.44311.

[23] Hedman, S. G., "Vortex Lattice Method for Calculation of Quasi Steady State Loadings on Thin Elastic Wings in Subsonic Flow," FFA Flygtekniska Försöksanstalten, Stockholm, Sweden, FFA Report 105, 1966.

[24] Jacques, V., and Garrigues, E., "Ground Dynamic Loads Calculations Strategies for Civilian and Military Aircrafts," presented at the International Forum for Aeroelasticity and Structural Dynamics, Bristol, 2013.

[25] Karpel, M., and Strul, E., "Minimum-state unsteady aerodynamic approximations with flexible constraints," *Journal of Aircraft*, vol. 33, no. 6, pp. 1190–1196, Nov. 1996, https://doi.org/10.2514/3.47074.

[26] Katz, J., and Plotkin, A., *Low-speed aerodynamics: from wing theory to panel methods*. New York: McGraw-Hill, 1991.

[27] Kier, T., and Looye, G., "Unifying Manoeuver and Gust Loads Analysis Models," in *International Forum for Aeroelasticity and Structural Dynamics*, 2009.

[28] Klimmek, T., "Parametric Set-Up of a Structural Model for FERMAT Configuration for Aeroelastic and Loads Analysis," *Journal of Aeroelasticity and Structural Dynamics*, no. 2, pp. 31–49, May 2014, https://doi.org/10.3293/asdj.2014.27.

[29] Kotikalpudi, A., "Body Freedom Flutter (BFF) Doublet Lattice Method (DLM)," *University of Minnesota Digital Conservancy*, 09-Sep-2014. [Online]. Available: http://hdl.handle.net/11299/165566. [Accessed: 12-Feb-2016].

[30] Kotikalpudi, A., Pfifer, H., and Balas, G. J., "Unsteady Aerodynamics Modeling for a Flexible Unmanned Air Vehicle," in *AIAA Atmospheric Flight Mechanics Conference*, Dallas, Texas, 2015, https://doi.org/10.2514/6.2015-2854.

[31] Krüger, W. R., "A multi-body approach for modelling manoeuvring aeroelastic aircraft during preliminary design," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 222, no. 6, pp. 887–894, Jun. 2008, https://doi.org/10.1243/09544100JAERO264.

[32] Krüger, W. R., "Integrated design process for the development of semi-active landing gears for transport aircraft," Dissertation, Universität Stuttgart, Stuttgart, Deutschland, 2000, http://elib.uni-stuttgart.de/handle/11682/3675.

[33] Krüger, W. R., Besselink, I., Cowling, D., Doan, D. B., Kortüm, W., and Krabacher, W., "Aircraft Landing Gear Dynamics: Simulation and Control," *Vehicle System Dynamics*, vol. 28, no. 2–3, pp. 119–158, Aug. 1997, https://doi.org/10.1080/00423119708969352.

[34] Küssner, H. G., "Allgemeine Tragflächentheorie," *Luftfahrtforschung*, vol. 17, no. 11/12, pp. 370–78, Dec. 1940, http://ntrs.nasa.gov/search.jsp?R=19930094437.

[35] Küssner, H. G., "General Airfoil Theory," National Advisory Committee for Aeronautics, Washington, Technical Memorandum NACA-TM-979, Jun. 1941, http://ntrs.nasa.gov/search.jsp?R=19930094437.

[36] Lehoucq, R., Maschoff, K., Sorensen, D., and Yang, C., "ARPACK - Arnoldi Package," *ARPACK SOFTWARE*. [Online]. Available: http://www.caam.rice.edu/software/ARPACK/. [Accessed: 21-Aug-2017].

[37] Lemmens, Y., Verhoogen, J., Naets, F., Olbrechts, T., Vandepitte, D., and Desmet, W., "Modelling of Aerodynamic Loads on Flexible Bodies for Ground Loads Analysis," presented at the International Forum for Aeroelasticity and Structural Dynamics, Bristol, 2013.

[38] National Aeronautics and Space Administration, *The Nastran Programmer's Manual*, NASA SP-223(01). Washington, D.C.: COSMIC, 1972.

[39] Neumann, J., "Identifikation radialer Basisfunktionen zur räumlichen Strömungs-Struktur-Kopplung unter Berücksichtigung des Interpolations- und des Lasttransformationsfehlers," Institute of Aeroelasticity, Göttingen, Germany, Internal Report DLR IB 232-2008 J 01, 2008.

[40] Neumann, J., and Krüger, W. R., "Coupling Strategies for Large Industrial Models," in *Computational Flight Testing*, vol. 123, N. Kroll, R. Radespiel, J. W. Burg, and K. Sørensen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 207–222.

[41] Pinho Chiozzotto, G., "Kriterien fuer die Erstellung eines Lastenkatalogs," Institute of Aeroelasticity, iLOADs MS1.2, Feb. 2014.

[42] Powell, M. J., "A hybrid method for nonlinear equations," *Numerical methods for nonlinear algebraic equations*, vol. 7, pp. 87–114, 1970, http://scholar.google.com/scholar?cluster=280996480504727445&hl=en&oi=scholarr.

[43] Prandtl, L., "Beitrag zur Theorie der tragenden Fläche," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 16, no. 6, pp. 360–361, 1936, https://doi.org/10.1002/zamm.19360160613.

[44] Reschke, "Integrated Flight Loads Modelling and Analysis for Flexible Transport Aircraft," Dissertation, Universität Stuttgart, Oberpfaffenhofen, Germany, 2006.

[45] Rodden, W., Harder, R., and Bellinger, D., "Aeroelastic Addition to NASTRAN," NASA Langley Research Center, Los Angeles, California, Contractor Report NASA Contractor Report 3094, Mar. 1979.

[46] Rodden, W. P., Giesing, J. P., and Kálmán, T. P., "New Developments and Application of the Subsonic Doublet-Lattice Method for Nonplanar Configurations," *AGARD-CP-80-PT-2 - Symposium on Unsteady Aerodynamics for Aeroelastic Analyses of Interfering Surfaces*, no. Part 2, Jan. 1971.

[47] Roger, K. L., "Airplane Math Modeling Methods For Active Control Design," in *AGARD-CP-228*, 1977.

[48] Schwochow, J., "Die aeroelastische Stabilitätsanalyse - Ein praxisnaher Ansatz Intervalltheoretischen Betrachtung von Modellierungsunsicherheiten am Flugzeug zur," Dissertation, Universität Kassel, Kassel, 2012.

[49] Siggel, M., and Schreiber, A., *tixi: Fast and simple XML interface library*. DLR Simulation and Software Technology, 2017, https://github.com/DLR-SC/tixi.

[50] The Scipy community, "scipy.integrate.ode," *SciPy v0.18.0 Reference Guide*. [Online]. Available: http://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.ode.html#scipy.integrate.ode. [Accessed: 08-Aug-2016].

[51] The Scipy community, "scipy.optimize.fsolve," *SciPy v0.17.0 Reference Guide*. [Online]. Available: http://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.fsolve.html. [Accessed: 24-Feb-2016].

[52] The Scipy community, "scipy.sparse.linalg.eigs," *SciPy v0.19.1 Reference Guide*. [Online]. Available: https://docs.scipy.org/doc/scipy-0.19.1/reference/generated/scipy.sparse.linalg.eigs.html. [Accessed: 21-Aug-2017].

[53] Vepa, R., "Finite State Modelling of Aeroelastic Systems," Stanford University, California, Ph.D. Dissertation NASA CR-2779, Feb. 1977.

[54] Voß, A., and Klimmek, T., "Maneuver Loads Calculation with Enhanced Aerodynamics for a UCAV Configuration," in *AIAA AVIATION Forum*, Washington, D.C., 2016, https://doi.org/10.2514/6.2016-3838.

[55] Voß, A., and Ohme, P., "Dynamic maneuver loads calculations for a sailplane and comparison with flight test," *CEAS Aeronautical Journal*, vol. 9, no. 3, pp. 445–460, Apr. 2018, https://doi.org/10.1007/s13272-018-0300-9.

[56] Voß, A., Pinho Chiozzotto, G., and Ohme, P., "Dynamic Maneuver Loads Calculation for a Sailplane and Comparison with Flight Test," presented at the IFASD 2017 - 17th International Forum on Aeroelasticity and Structural Dynamics, Como, Italy, 2017, https://elib.dlr.de/113152/.

[57] Voss, R., "The Legacy of Camillo Possio to Unsteady Aerodynamics," in *System Modeling and Optimization*, 2005, pp. 1–14, https://doi.org/10.1007/0-387-33006-2_1.

[58] Wang, Z., and Chen, P. C., "Accurate Rational Function Approximation for Time- Domain Gust Analysis," presented at the 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Grapevine, Texas, 2017.

[59] Waszak, M., Buttrill, C., and Schmidt, D., "Modeling and Model Simplification of Aeroelastic Vehicles: An Overview," NASA Langley Research Center, NASA Technical Memorandum 107691, Sep. 1992.

[60] Waszak, M. R., and Schmidt, D. K., "Flight dynamics of aeroelastic vehicles," *Journal of Aircraft*, vol. 25, no. 6, pp. 563–571, Jun. 1988, https://doi.org/10.2514/3.45623.

[61] Wright, J. R., and Cooper, J. E., *Introduction to aircraft aeroelasticity and loads*. Chichester, England ; Hoboken, NJ: John Wiley, 2007.

[62] ZONA Technology Inc., *ZAERO Theoretical Manual*, vol. Version 9.0. Scottsdale, Arizona, 2014.

[63] *MSC.Nastran 2005 Quick Reference Guide*. MSC Software Corporation, 2004.

[64] "US. Standard Atmosphere 1976," National Aeronautics and Space Administration, Washington, D.C., Technical Report NASA-TM-X-74335, NOAA-S/T-76-1562, Oct. 1976.