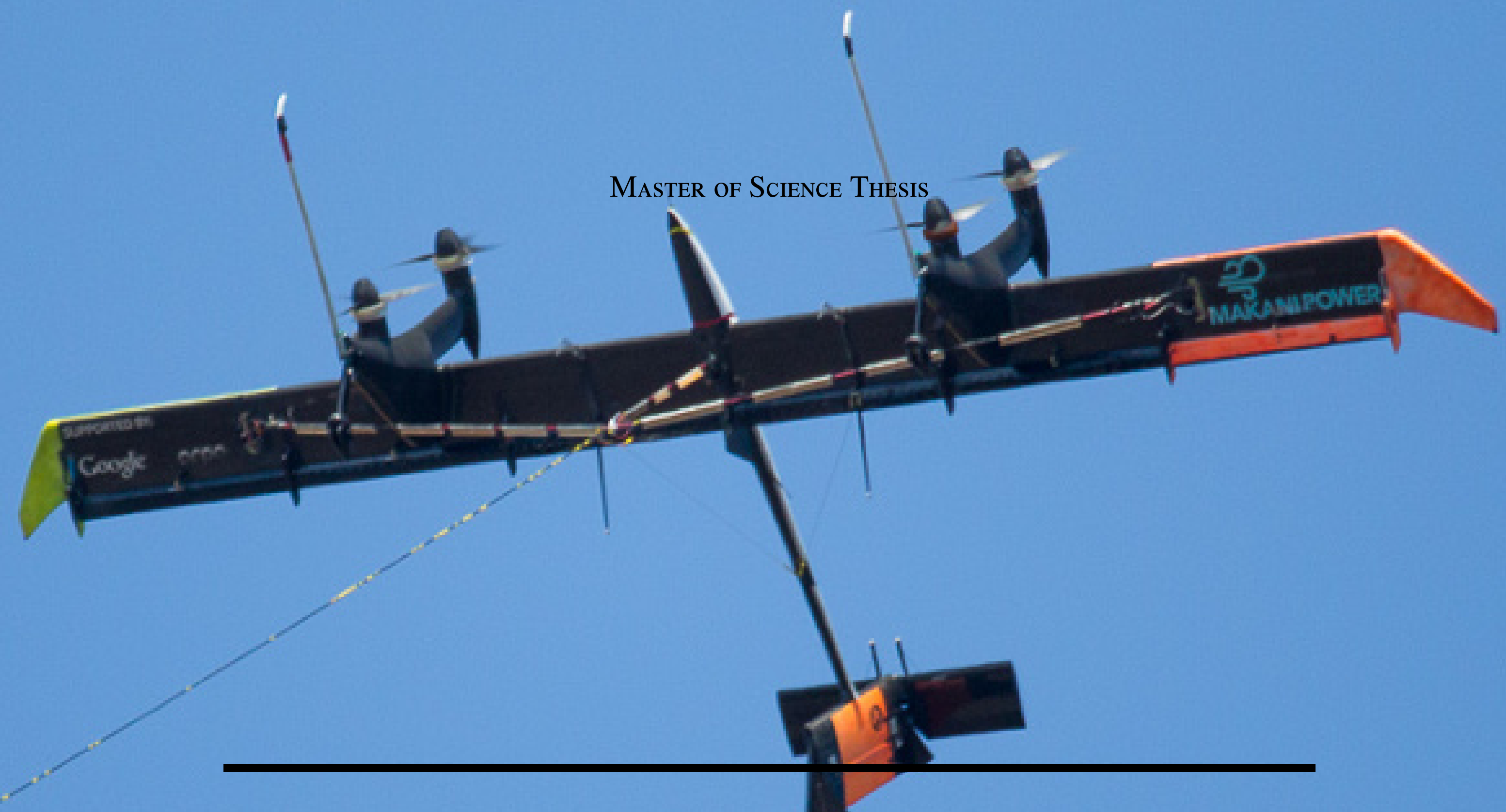
Master of Science Thesis



Aero-elastic analysis of a large airborne wind turbine

Prediction of divergence, control reversal and effectiveness, and flutter of a tethered wing

J. Wijnja B.Sc.

December 18, 2013

Faculty of Aerospace Engineering · Delft University of Technology

Aero-elastic analysis of a large airborne wind turbine

Prediction of divergence, control reversal and effectiveness, and flutter of a tethered wing

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

J. Wijnja B.Sc.

December 18, 2013

Faculty of Aerospace Engineering · Delft University of Technology

Delft University Of Technology
Department Of
Aerodynamics & Wind Energy

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Aero-elastic analysis of a large airborne wind turbine"** by **J. Wijnja B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: December 18, 2013

Head of Wind Energy section: ______________________
Prof.dr. G.J.W. van Bussel

Supervisor TU Delft: ______________________
Dr.-Ing. R. Schmehl

Supervisor TU Delft: ______________________
Dr.ir. R. De Breuker

Chief Technical Engineer Google[x] Makani Power: ______________________
D. Vander Lind B.Sc.

Supervisor Google[x] Makani Power: ______________________
Dr. K. Jensen

Abstract

The objective of this research is the simulation of aero-elastic behaviour of Makani's large airborne wind turbine. This tethered wing operates in crosswind motion, and is equipped with on-board wind turbines. The tether-bridle system attaches the energy generating system to the ground station. It is likely that the structure of this, $28m$ span, carbon fibre, high aspect ratio wing, will deform considerable under aerodynamic loads. In the worst case scenario, static and/or dynamic aero-elastic effects cause destructive failure.

The aero-elastic simulation program ASWING is used for the analysis. This program uses a fully non-linear Bernoulli-Euler beam representation for structural modelling in combination with a lifting line-representation for aerodynamic modelling. Linearised unsteady analyses are derived for the Eigenmode analysis. Since the tether-bridle system cannot be modelled in the current program version, an additional, ASWING compatible, module is written. The tether is modelled as a spring with user defined characteristics for the spring stiffness, mass and aerodynamic drag area. The bridle lines are assumed massless and perfectly rigid. The tether and bridle forces are dependent on the wing flexibility, and wing position and orientation. The tether-bridle module is verified against analytical expressions and by using MATLAB. A wind tunnel test validates the dynamic aero-elastic responses.

For the Makani wing, divergence, aileron effectiveness and reversal, and flutter behaviour is analysed. Divergence and aileron reversal are no critical modes. However, aileron effectiveness is critical. The requirements state a minimum aileron effectiveness of 75% at $95m/s$ flight speed. The program calculated this minimum aileron effectiveness at $92m/s$ flight speed. These problems can be resolved by a 10% increase in the wing's torsional stiffness or a 10% increase of lift force increment with aileron deflection. The Eigenmode results showed a critical flutter mode at flight speeds higher than $90m/s$, whereas the design flutter speed is equal to $120m/s$. This susceptibility to flutter can be resolved by (1) a 50% increase in torsional stiffness, (2) a 50% increase in in-plane-bending stiffness or (3) a $10cm$ upstream shift in center of gravity. A $50cm$ upstream shift of bridle-wing attachment location increases the flutter speed to $110m/s$.

It was found that the effects, of tether aerodynamic drag and tether weight, are negligible for the aero-elastic behaviour. Also, in the analysis for the Makani wing, the tether spring constant does not contribute to the static and dynamic aero-elastic effects. The position of the bridle-wing attachments influences the twist angles and tip deflections of the wing. These results are useful in case maximum twist angles and/or wing tip deflections are critical. For the dynamic aero-elastic behaviour the wing-bridle attachment positions can be adjusted to decrease the susceptibility to flutter.

Acknowledgement

This Master's thesis emerged as a first collaboration project between the faculty aerospace engineering of Delft University of Technology and Makani Power, which was acquired by Google[x] in the course of the project. From both parties many people inspired, advised, and helped to advance the research challenges I faced.

From the faculty of aerospace engineering I would like to gratefully thank my two supervisors Dr.-Ing. Roland Schmehl and Dr.ir. Roeland De Breuker for their continuous guidance, their patience and valuable discussions. My appreciation goes to Prof.dr. Gerard J.W. van Bussel for his involvement throughout the project and for being the head of the graduation committee. Additionally, I would like to thank Prof.dr. Leo L.M. Veldhuis, Dipl.-Ing. Stefan Bernardy and Leo Molenwijk for their help with wind tunnel preparations and especially Dipl.-Ing. Stefan Bernardy for his continuous guidance during the entire measurement campaign at the Delft University of Technology.

From Google[x] Makani Power my gratitude goes to chief technical engineer Damon Vander Lind B.Sc. for my application repetition at Makani Power and the set-up of an interesting and challenging research project. My thank goes to Dr. Kenneth Jensen for very valuable weekly meetings and his continuous help throughout the project. Also I would like to thank Schuyler McAlister for his specialized help and guidance in the production phase of the wind tunnel model. Additionally, I would like to thank Nathan Treat for his indispensable help during measurement campaign.

My special thanks goes to the entire team at Makani Power for interesting conversations at coffee or lunch breaks, very interesting football matches and above all an amazing atmosphere in which I felt at home.

My last words go to my parents, friends, family and especially my girlfriend, who supported me when I was far from home and who had to continuously compete for quality time during the Master's thesis project. When this is finished, I'll make up for the lack of quality time. I Promise!

Delft University of Technology　　　　J. Wijnja
December 12, 2013

Contents

List of Figures

List of Tables

Nomenclature

Latin Symbols

A	area	$[m^2]$
b	wing span	$[m]$
c	chord	$[m]$
c	coefficient	$[-]$
c_t	tether loading constant	$[-]$
D	aerodynamic drag force	$[N]$
d	aerodynamic drag for an element	$[N]$
d_t	tether diameter	$[m]$
E	Young's modulus of elasticity	$[N/m^2]$
E	energy	$[J]$
e	eccentricity factor	$[-]$
$\vec{f}$	local force vector vector of a beam element	$[N]$
G	shear modulus	$[N/m^2]$
g	gravitational acceleration	$[m\ s]$
GK	shear stiffness	$[N]$
$\vec{H}$	angular momentum vector	$[Nms]$
I	area moment of inertia	$[m^4]$
J	polar moment of inertia	$[m^4]$
K_θ	torsional stiffness	$[N/m]$
K_h	bending stiffness	$[N/m]$
L	aerodynamic lift force	$[N]$
L	total length	$[m]$
l	aerodynamic lift for an element	$[N]$
l	element length	$[m]$

$\vec{m}$	local moment of a beam element	$[Nm]$
m	mass	$[kg]$
P	power	$[W]$
q	dynamic flow pressure	$[N/m^2]$
q	shear flow	$[N/m]$
$\vec{R}$	Cartesian position vector	$[m]$
R	radius	$[m]$
Re	Reynolds number	$[-]$
$\vec{r}$	Local element position vector	$[m]$
r	radius	$[m]$
S	wing surface area	$[m^2]$
$\bar{\bar{T}}$	transformation tensor, aircraft axes to local beam element axes	$[-]$
$\bar{\bar{T}}_E$	transformation tensor, aircraft axes to Earth reference axes	$[-]$
t	thickness	$[m]$
t	time	$[s]$
$\vec{U}$	flight velocity vector in aircraft axes	$[m/s]$
$\mathbf{u}$	commanded variable	$[-]$
$\vec{u}$	flight velocity vector in local beam axes	$[m/s]$
$\vec{V}$	velocity vector	$[m/s]$
v	velocity magnitude	$[m/s]$
$\mathbf{v_k}$	eigenvector	$[-]$
W	gravitational weight force	$[N]$
$\mathbf{x}$	state vector	$[-]$
$\ddot{x}$	linear acceleration	$[m/s^2]$
$\dot{\mathbf{x}}$	time rate of change of state vector	$[-]$
x_θ	displacement of the center of gravity from the elastic axis	$[m]$

Greek Symbols

α	angle of attack	[deg]
β	side slip angle	[deg]
β_t	tether elevation angle	[deg]
$\vec{\Gamma}$	circulation	$[m^2/s]$
γ	strain	$[-]$
$\gamma_{w,\mu}$	correction factor for weight and weight inertia	$[-]$
δ	flap deflection	[deg]
ϵ	extensional strain	$[-]$
ε	material stiffness	$[N]$, $[Nm^2]$
η_{cs}	control surface efficiency	$[\%]$
Θ	pitch angle	[deg]
$\vec{\Theta}$	vector of Euler angles in Earth reference frame	[deg]
ϑ	bending angle about y-axis	[deg]

$\vec{\theta}$	local beam Euler angle vector	[deg]
$\ddot{\theta}$	angular acceleration	$[\deg/s^2]$
ιg	weight-inertia per unit length	$[Nm]$
κ	curvature	$[-]$
$\bar{\bar{\kappa}}$	beam curvature tensor	$[-]$
λ_k	eigenvalue	$[-]$
μg	weight per unit length	$[N/m]$
ν	kinematic viscosity	$[m^2/s]$
ρ	density	$[kg/m^2]$
σ	growth rate	$[-]$
Φ	roll angle	[deg]
φ	bending angle about x-axis	[deg]
Ψ	yaw angle	[deg]
ψ	bending angle about z-axis	[deg]
$\vec{\Omega}$	rotation rate vector relative to Earth reference frame	$[\deg/s]$
ω	angular velocity	$[\deg/s]$
$\vec{\omega}$	rotation rate vector relative to aircraft reference frame	$[\deg/s]$

Subscripts

$*$	projection
0	unloaded
α	angle of attack
δ	flap deflection
ϵ	extensional-strain
γ	shear-strain
∞	undisturbed
div	divergence
rev	reversal
acc	acceleration
$aero$	aerodynamic
Ao	zero-lift above c-axis
ap	attachment point
bf	bridled flight
bm	bench mark
b	bridle
cg	center of gravity
cp	center of pressure
c	local beam-axis c
d,f	friction drag
d,p	pressure drag
ea	elastic axis

enc	enclosed
e	engine
f	fibre
f	fuselage
IAS	indicated airspeed
ind	induced
ini	initial
i	index for beam element
J	joint
kin	kinetic
l	lift
m	moment
m	resin
n	local beam-axis n
pm	point mass
p	pylon
r	rotor
$ssic$	sphere sphere intersection circle
st	strut
s	local beam-axis s
ta	tensile axis
tba	tether bridle attachment
t	tether
uf	unbridled flight
w	wind
X	Earth-axis X
x	aircraft-axis x
Y	Earth-axis Y
y	aircraft-axis y
Z	Earth-axis Z
z	aircraft-axis z

Superscripts

$'$	per unit span
BT	bottom
R	rigid
TP	top
T	transpose

Abbreviations

ARPACK	Arnoldi Package
AWE	Airborne Wind Energy
AWT	Airborne Wind Turbine
CFD	Computational Fluid Dynamics
CLA	Classical Laminate Analysis
FEM	Finite Element Method
FSI	Fluid Structure Interaction
LLT	Low speed Low Turbulence wind tunnel
MODE	Eigenmode menu ASWING
OPER	Operational menu ASWING
ROM	Rule of Mixture
TU Delft	Delft University of Technology

Chapter 1

Introduction

This Master's thesis is a collaboration between academia and industry with the aim of developing an aero-elastic model for a large scale airborne wind turbine. From the academic side the faculty aerospace engineering of the Delft University of Technology (TU Delft) is involved and from the industry side Google[x] Makani Power is involved. Google[x] Makani Power is a pioneer in the airborne wind energy sector. It is the goal of this introductory chapter to introduce the airborne wind energy (AWE) concept and to set-up a framework for the Master's thesis. In section 1.1 the thesis relevance is given. Section 1.2 aims to briefly introduce the airborne wind energy concept and to list the key research and development players. Special attention is given for the TU Delft and Makani Power AWE concept. Many different aerodynamic and structural simulations programs are available for aero-elastic simulation. In section 1.3 a deliberate choice is made for the optimum aero-elastic simulation program for this thesis. Finally the main research question is defined in section 1.4, which also outlines the structure of the report.

1.1 Thesis relevance

The global demand for electricity has increased by 80% over the last 20 years. And is expected to rise by an additional 76% in the coming 20 years (U.S. Department of Energy; International Energy Agency). The energy reserves from oil, coal and gas are 35, 107 and 37 years, respectively (Shahriar and Erkan, 2009). The rising energy demand in combination with the depleting fossil fuels make sustainable energy sources a necessity already in the near future. With increasing scarcity of fossil reserves, prices increase and sustainable energy resources become more attractive for consumers. Whereas conventional wind turbines are reaching their structural and economic limits, a new technology is rising: airborne wind energy. Airborne wind turbines (AWTs) are a promising innovative technology in the field of sustainable energy, which could be economically advantageous with respect to other sustainable energy resources (Zillmann and Hach, 2013).

1.2 Airborne wind energy

Airborne wind energy systems harvest winds at higher altitudes, which generally contain higher energy density winds (Archer, 2013). In comparison with conventional horizontal axis wind turbines, the tower is substituted by a tether. The wing flies crosswind and mimics the highly efficient outer part of the turbine blades, see Figure 1.1. Generally bridles are attached at several locations on the wing and come together at the tether-bridle attachment to connect the kite with the tether.

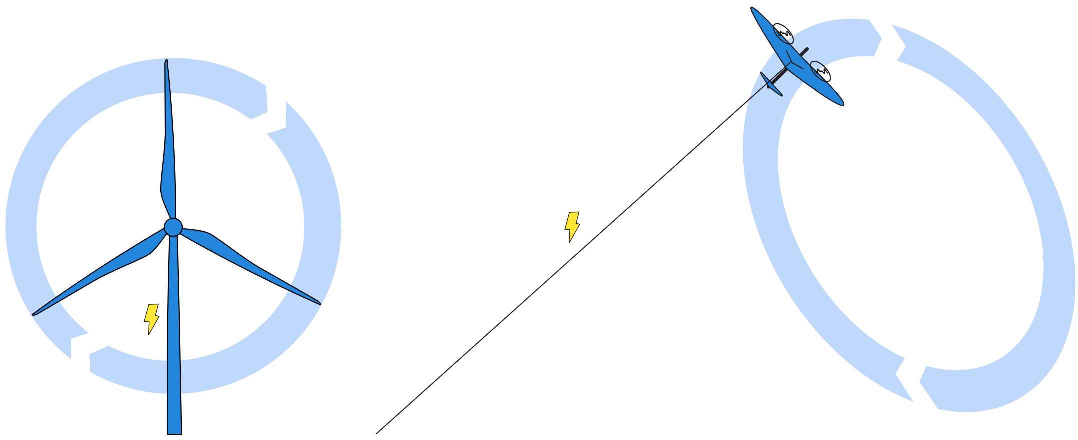


Figure 1.1: Windturbine versus airborne wind turbine (Diehl, 2013)

The AWE concept is nothing new; already in 1820s, the transportation with kite systems was explored and a kite coach was developed. This is a small vehicle powered by a kite (Pocock, 1827). Next to land transport, kite systems were applied to transport ships overseas. In the early 1900s kite research was booming and the first man lifting kites were developed. In these decades at the dawn of air transportation technology, kites were a serious competitor for airplanes. Since powered aircraft are more versatile and independent of wind, the kite systems lost this competition and the research stagnated (Breukels, 2010; Ahrens et al., 2013).

Serious interest in airborne wind energy arose again in 1980 with a publication of Loyd (1980), which describes the concept of kites for large scale wind energy production. A C-5A aircraft is simulated as a kite to demonstrate a theoretical power output of $6.7MW$. Loyd's theory was further developed by several academics, and currently principles of crosswind power generation are well understood (Argatov et al., 2009; Breukels, 2010; Williams et al., 2008).

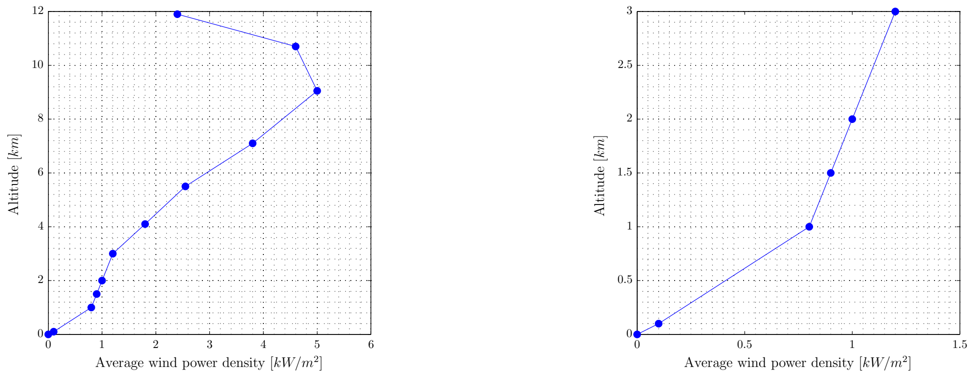
Airborne wind energy uses a flying vehicle to extract energy from air. The tether allows for higher altitudes compared to conventional wind turbines. The kinetic energy of a certain mass of air is equal to:

$$E_{\text{kin}} = \int \frac{1}{2} v_w^2 \text{d}m = \int \frac{1}{2} v_w^2 \text{d}(A v_w \rho t) = \int \frac{1}{2} A \rho v_w^3 \text{d}t \tag{1.1}$$

In this equation E_{kin} is the kinetic energy, dm the air mass, v_w the wind velocity, ρ the air density, A the area and t the time. The associated wind power is equal to the time rate of change of this kinetic energy.

$$P = \frac{\text{d}E_{kin}}{\text{d}t} = \frac{1}{2} A \rho v_w^3 \tag{1.2}$$

This equation illustrates that wind velocity is a very important parameter, because wind power is proportional to its cube. Generally, wind velocity is positively related to altitude as shown in Figures 1.2a and 1.2b. These Figures show the 20 year average wind profile for De Bild, the Netherlands. This wind profile differs from location to location, but generally, wind velocity increases with altitude, up to a certain limit.



(a) Wind velocity versus altitude up to 12km

(b) Wind velocity versus altitude up to 3km

Figure 1.2: 20 year average wind velocity versus altitude, retrieved from KNMI station De Bild, the Netherlands (Ockels et al., 2004)

1.2.1 AWE concepts

Ahrens et al. (2013) give an overview of the field of airborne wind energy. In 2000 about 3 institutions were actively involved in AWE. Over the following years significant interest in AWE arose and as of 2013 about 50 institutions are actively involved in AWE. Most of the research and development activities are concentrated in Europe and Northern America as illustrated in Figure 1.3. From the 50 AWE institutions, a classification in on-board power generation and ground-based power generation is made. These concepts will be explained next.

On-board power generation

Most on-board power generating systems apply an energy generating system, which is is tethered to the ground. One way to generate power is by attaching wind turbines to a crosswind flying kite to extract wind energy from the high relative air velocity. The energy is transmitted to the ground station by a high voltage power line. Google[x] Makani Power uses this concept, which is explained in more detail in section 1.2.3. Another technique that uses on-board power generation relies on lighter than air material to lift a rotor or another device to generate power in the medium to high altitude winds (Diehl, 2013). These are almost all 'non-crosswind' systems.

Ground-based power generation

Most ground-based power generation systems are based on a cross-wind flying kite that creates tether tension to unroll the tether from a drum to drive the generator at the ground. For continuous operation a so-called '*pumping cycle*' is used. In the reel-out phase the kite is flying at its optimum lift and drag coefficient to create maximum tether force. This high tether force drives the drum and creates a significant amount of power. In the reel-in phase the kite is de-powered to create a low tether force. Therefore, the motor reels in using a fraction of the energy that was extracted in the reel-out phase. A battery network is used to buffer the energy over the cycles.

Ground-based power generation concepts have been devised with with rigid or flexible wings, such as the TU Delft concept explained in section 1.2.2. Many flexible kite systems exist, however Ahrens et al. (2013) lists few rigid wing ground-based power generation concepts. AmpyxPower is probably in the furthest state of development. AmpyxPower deploys a 5.5m span plane with an aspect ratio of 10. This

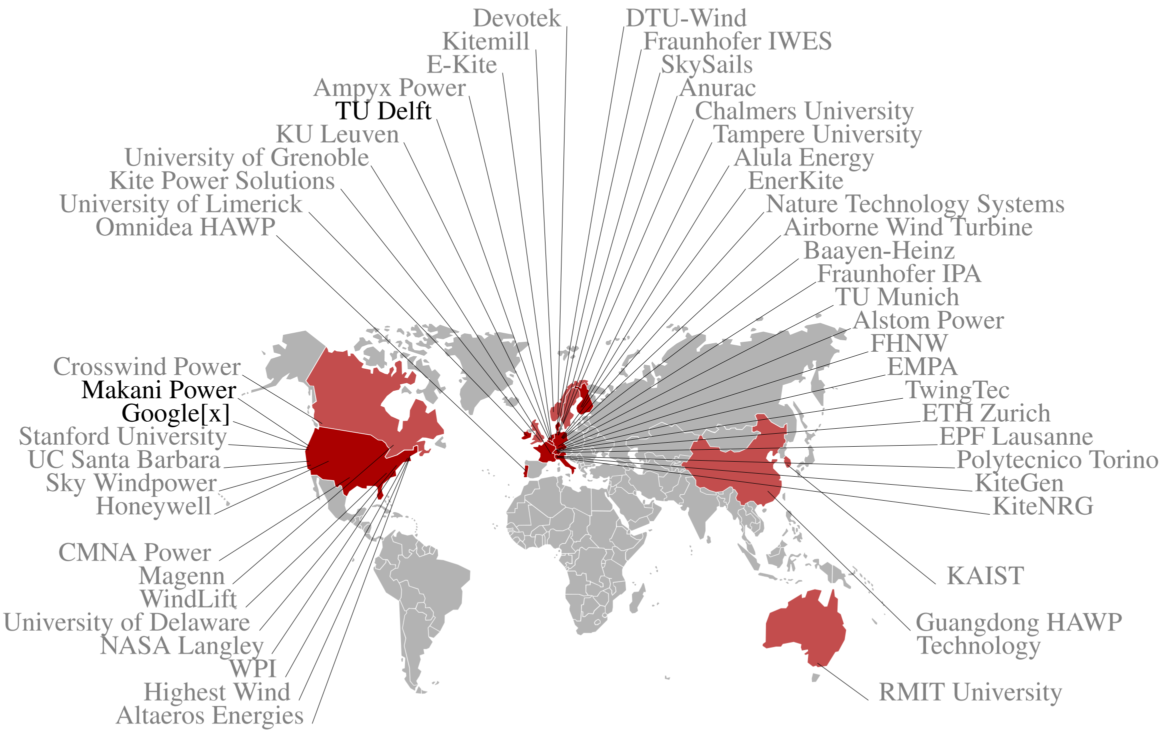


Figure 1.3: Airborne wind energy institutions worldwide (Ahrens et al., 2013)

plane could possibly feature similar aero-elastic behaviour with respect to the Makani wing. Agten (2012) and Bontekoe (2010) analyse the aerodynamics and launch and retrieval system, but no documentation is present about aero-elastic simulation programs developed by AmpyxPower.

1.2.2 TU Delft pumping cycle power system

Meijaard et al. (1999) describe the laddermill concept with rigid wings. In a subsequent paper, this concept is explored for kites and Ockels (2001) states *A laddermill is a self-supporting system that consists of an endless cable connected to a series of high-lifting wings or kites moving up in a linear fashion, combined with a series of low-lifting wings or kites going down. The cable drives an energy generator placed on the ground.* The concept is elaborated further and it is proposed, on the basis of theoretical analysis, that a single standalone laddermill could generate $50MW$ (Ockels et al., 2004). This would be equal to 10 currently available large offshore wind turbines with $120m$ rotor diameter. This potential in energy production leads to the installation of the kite lab at the faculty aerospace engineering and the set-up of a dedicated research group for kite power. In 2012, the team consists of 20-25 staff members and students.

However, the laddermill concept relies on many kites and faces many technical challenges. Hence, research is currently focused on single kite systems to establish a body of knowledge about controlled and reliable operation of kites for energy generation. Since January 2010, TU Delft AWE uses a $25m^2$, $20kW$ tech-demonstrator, which is fully instrumented and with cameras mounted at several positions. This kite is used for experimental purposes and validating theoretical results with measurement data. In June 2010, the automatic generation of the power generating kite was demonstrated over extended periods of time (van der Vlugt et al., 2013).

1.2.3 Makani Power airborne wind turbine

This section describes a brief history of Makani Power followed by an introduction into the working principle and development plan.

Company history

Makani Power was founded in 2006 by Saul Griffith, Corwin Harham and Don Montague. The first 6 years of development were supported by Google and the U.S. Department of Energy. From 2006 to 2009 the concept of a soft textile kite powering a generator on the ground was used. In 2009 a revolutionary change of concept took place; from a soft kite to a rigid wing with on-board power generation. In 2010 the first wing with on-board power generation was built in combination with autonomous control. In 2011 Makani designed a new airframe, which was the first wing to launch and land from a perch. In 2012 a full autonomous flight was performed including launching and landing. The kite took off from a perch, hovered while the tether reeled out, transitioned to a crosswind flight mode, an finally transitioned back to a hovering flight mode and landed. In 2013 Google[x] acquired Makani Power (Makani Power, 2012a).

Working principle and development plan

The working principle of Makani's current airborne wind turbine is similar to the TU Delft AWE concept in that both systems harvest high energy dense winds from high altitudes with the absence of a tower. However, the TU Delft pumping cycle system, is based on ground-based power generation and a flexible kite, whereas the Makani principle is based on on-board power generation and a rigid wing. Makani's current AWT prototype consists of a tethered wing outfitted with wind turbines as shown in Figure 1.4.

Figure 1.4: Makani AWT (Makani Power, 2012b)

This AWT and a conventional wind turbine operate on the same aerodynamic principles. A wind turbine rotates because the airflow through the blades creates local lift forces that make the blades turn. The lift forces created by the shape of the wind turbine blade can be broken into two components: one that rotates the blades and one that pushes against the tower. The shape of the AWT's wing creates lift in a similar way as shown in Figure 1.5 (Vander Lind, 2013c).

By tethering the wing to the ground it becomes a kite. The aerodynamic forces are balanced by the tether. The traction force at the tether is not used to generate power, but allows fast crosswind flight. Energy is extracted from the wind using small on-board turbines driving high-speed, direct drive generators. The electricity is transmitted to the ground via a conducting tether, where it is fed into the grid (Makani Power, 2012c).

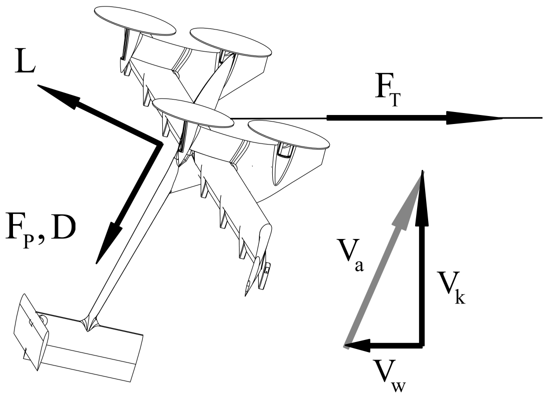


Figure 1.5: Makani's aerodynamic principle (Vander Lind, 2013c). L is the lift force, F_P the force used for power generation, D the drag force, F_T the tether force, V_a the apparent wind velocity, V_w the wind velocity and V_k the flight velocity of the wing/kite.

The on-board avionics computer guides the wing along a circular path. Due to its speed, the tip of conventional wind turbine blades is the most effective part. In some cases the last 25% of the blade is responsible for 75% of the energy generated. The Makani wing mimics the path and speed of these blade tips, capturing all of the benefits using only a fraction of the materials. At scale, the entire span of the Makani wing operates at the speed of the aerodynamically effective tip of the wind turbine.

Next to the generation of wind energy the on-board turbines at the blade serve a second purpose in the launch stage. They act as propellers to launch the AWT using energy from the grid. When reaching the target altitude, the wing is operated crosswind in a circular pattern. The turbines now act as a wind turbine and energy is created by driving a generator. To land the system, the wing is transitioned into hover mode, by using the turbines as propellers, and slowly descended to the perch.

Currently, Makani has developed 'Wing7', an $8m$ wing span prototype with $30kW$ rated power. Makani aims to scale this system to a $600kW$ system within the next two years. This carbon fibre M600 has a full rated power wind speed $v_{rated} = 11.5m/s$, will operate at altitudes $140 - 310m$, has a wing span of about $b = 28m$, a characteristic chord $\bar{c} = 1.30m$ and a lift to weight ratio of 10.

The $8m$ span Wing7 did not experience any serious aero-elastic effects. However, by increasing the span width and the aspect ratio, the susceptible for static and dynamic aero-elastic effects is increased. In the worst case scenario, this will lead to destructive failure. By including the aero-elastic analysis, the structural changes can be implemented in an early design stage.

To determine the most appropriate aero-elastic design method, several methods and programs are analysed in the next section.

1.3 Aero-elasticity simulation models

Aero-elastic behaviour of structures can be modelled by a combination of an aerodynamic model and a structural deformation model. Consider an arbitrary non-rigid body in an airflow. The airflow over the body creates certain distributed aerodynamic forces. These distributed loads deform the body. Subsequently, the aerodynamic model calculates the distributed forces over the deformed body, which in turn, again, deform the body. The coupling between the aerodynamic model and the structural deformations model is a challenging problem and generally referred to as fluid-structure interaction (FSI).

Three different FSI approaches are considered: (1) ANSYS: full Finite Element Methods (FEM)-Computational Fluid Dynamics (CFD) coupling, (2) NASTRAN: FEM coupled to a more simple aerodynamic model with respect to CFD and (3) ASWING: structural beam representation coupled to a more simple aerodynamic model with respect to CFD.

In the subsonic flow regime, NASTRAN combines the doublet-lattice subsonic lifting surface theory with subsonic wing body interference theory. For supersonic speeds the Mach Box method, Piston Theory and the ZONA51 are used to determine aerodynamic forces. NATRAN's aero-elastic module is capable of static aero-elastic analysis as well as flutter analysis (MSC, 2004).

ASWING, is especially developed for static and dynamic aircraft aero-elastic analysis. An integrated aerodynamic and structural simulation code allows for arbitrary large deformation. The structural analysis consists of non-linear Bernoulli-Euler beams for fuselage and surface structures. The aerodynamic analysis is performed with a lifting line model with wing-aligned trailing vorticity, a Prandtl-Glauert compressibility transformation and local-stall lift coefficient (Drela, 1999).

ANSYS' main advantage with respect to NASTRAN is its increased accuracy in aerodynamic modelling using a CFD method. The main disadvantage of CFD with respect to more simplified aerodynamic models is its high computational time. In this conceptual/preliminary design stage, a low computational time is more important then increased accuracy and hence CFD/ANSYS is not the preferred option. The main advantage of NASTRAN with respect to ASWING is its increased accuracy in both aerodynamic and structural modelling. Additionally the aerodynamic modelling is more versatile with its supersonic model. ASWING's main advantage with respect to NASTRAN is its lower computational time, due to a more simplified approach in both aerodynamic and structural modelling. Additionally ASWING comes with its source code, which could possibly be adjusted for tethered flight.

In summary: both NASTRAN and ASWING could be used for effective static and dynamic aero-elastic analysis. However in this early design stage a low computational time is more beneficial than high accuracy. Therefore ASWING is most beneficial for this design stage. With low computational time the design space can be explored quickly and serve as the basis for subsequent detailed design at with NASTRAN could be the preferred analysis program.

1.4 Thesis goal and structure

The M600 is a relatively large, lightweight and high aspect ratio wing. Wing such as these are notorious for their aero-elastic susceptibility. In the worst case, static and/or dynamic aero-elasticity effects cause destructive failure and hence an aero-elastic analysis is critical in the design process. Currently no aero-elastic module for rigid airborne wind turbines is available. This leads to the goal of this graduation research:

> *'Design an aero-elastic module for rigid airborne wind turbines and analyse the M600 aero-elastic design boundaries'*

In this preliminary design stage, ASWING is the most suitable aero-elastic program to simulate aero-elastic behaviour of rigid tethered wings. The program combines low computational time with reasonable accuracy. Makani's M600 will be main subject of this thesis. However it is the aim of this research to create a more generic aero-elasticity program, which is suitable for tethered flight in general. A more generic aero-elasticity program could be used in the future development of the Makani wing and other rigid airborne wind turbines, such as AmpyxPower. To accomplish the main research goal, four steps are taken:

1. To analyse the M600's static and dynamic aero-elastic behaviour, first a proper understanding of ASWING is built by simulating benchmark models and testing the output. The M600 is under constant development. Therefore the other design scripts (aerodynamic shape, structural stiffnesses, weight distribution, etc.) should be linked to ASWING.
2. Next a mechanism is developed to simulate the tether and bridle lines,
3. This tether-bridle model should be verified and validated with wind tunnel tests,
4. Finally the parameter space is explored to determine the aero-elastic design boundaries of

This report is structured to answer the research question in a consecutive manner. Each part of the report aims to finalize one the above mentioned steps.

- In part I; first a description of ASWING is given in chapter 2, next the inputs are described in chapter 3 and finally, the output is presented and verified in chapter 4. From these chapters follow ASWING's capabilities and limitations and a deliberate decision can be made on the method to implement the tether-bridle system.
- In part II; the tether-bridle module is explained in chapter 5. From this tether-bridle system the Jacobian entries follow, which are described in chapter 6. Finally in chapter 7 the tether induced aerodynamic and gravitational loads are calculated.
- In part III; the tether-bridle module is verified in chapter 8 and validated with wind tunnel data in chapter 9.
- In part IV; in chapter 10 the modified ASWING version is used to analyse the M600's static and dynamic aero-elastic behaviour.
- Finally in chapter 11 the conclusions and recommendations for future research are given.

Part I

ASWING Introduction

Chapter 2

Description ASWING 5.96

ASWING is an aero-elastic simulation code written in FORTRAN77 which applies a fully non-linear Bernoulli-Euler beam representation to model structural deformations in combination with a lifting line-representation to model the aerodynamic surface characteristics. Linearised unsteady analyses are derived for the Eigenmode analysis. It is the goal of this chapter to briefly describe ASWING 5.96 and focus on the functionalities which could be useful for the modelling of the tether-bridle system. For a complete description, see Drela (2009) and Drela (2008a). In section 2.1 equilibrium flight is described and in section 2.2 the Eigenmode computational method is introduced. Finally in section 2.3 the conclusions of this chapter are drawn.

2.1 Equilibrium flight

In this section first the main program structure is presented followed by a brief description of its main functionalities: the reference frames, the aerodynamic principles and the structural equations. A brief description of the strut definition is given as well, because, at first, the mechanical behaviour of a strut seems similar to a tether-bridle system.

2.1.1 Program structure

This section describes the relevant user choices and program units (subroutines) to determine equilibrium flight. In this state the static aero-elastic effects such as divergence and control reversal and effectiveness can be analysed. In a subsequent analysis, this equilibrium is used for flutter analysis.

After start of the program, the user defined data for an aircraft (`xxx.asw`) is loaded into the system. This includes the aircraft geometrical, structural and aerodynamic parameters. Next, the atmospheric conditions are set and nodes are distributed which finalizes the zero load state definition. The state of the aircraft at which no external forces are applied. The geometrical, structural and aerodynamic parameters can be adjusted within the program to redefine the aircraft properties. The flow chart of this program part and the subsequent parts is given in Figure 2.1. Several different menus can be entered such as the plot routine, OPER menu at which the operating points can be calculated, MODE menu at which the eigenmodes can be calculated, the BODE menu at which the frequency responses can be calculated and the EDIT menu at

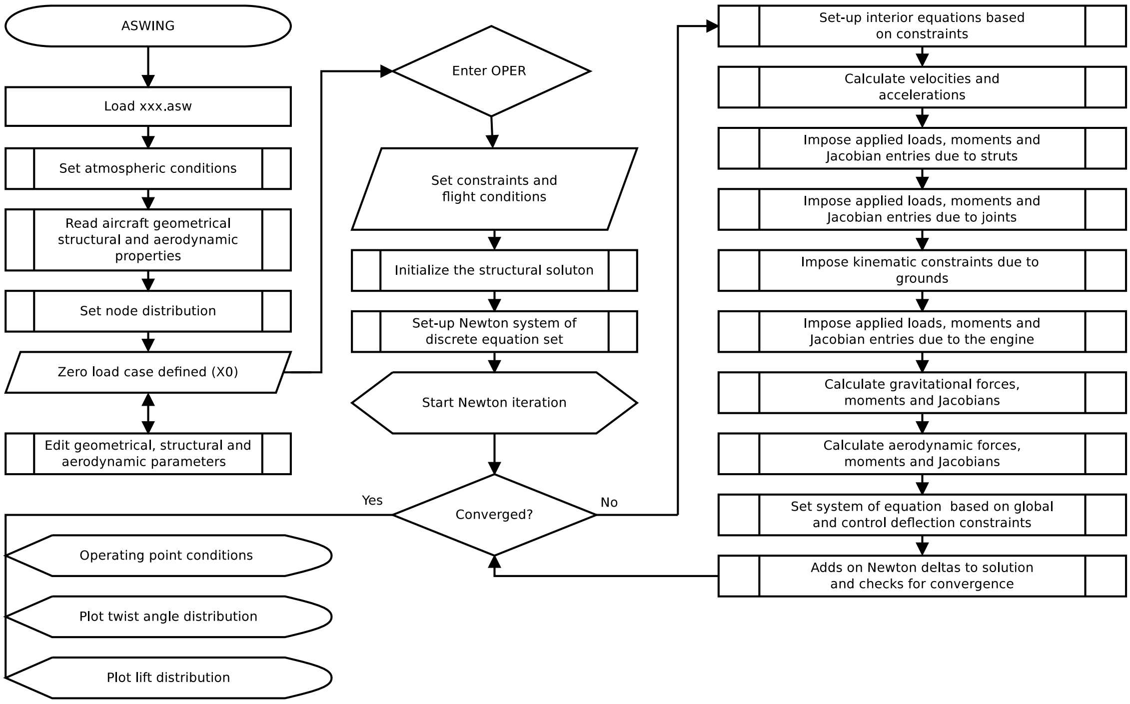


Figure 2.1: Flowchart ASWING for equilibrium solutions

which the structural distributions can be edited.

Assume the operational menu (OPER) is entered at which the constraints and flight conditions for each operating point can be defined. The structural solution is initialized and the Newton system of discrete equations in set-up followed by the Newton iterations to calculate a state (converged) solution. These Newton iterations include the calculations of velocities and accelerations, the set-up of loads and moment due to struts, joints and engines, aerodynamic loads and moments, gravitational loads and moment and impose kinematic constraints due to grounds.

In case the solution has converged the operating point conditions are given in tabular form and standard three plots emerge: (1) the twist angle distribution for all surface beams (2) the lift coefficient distribution for all surface beam and (3) the lift distribution for all surface beams. The user can request other plots for all beam elements such as the force and moment distributions, bending plots and curvature strain plots.

2.1.2 Reference frames

Different reference frames are applied at many subroutines to determine the equilibrium solution and its use will become more clear in the remaining of this Master's thesis when the tether-bridle system is added to ASWING.

Three different reference frames are used in ASWING, the inertial reference frame is denoted as $\vec{R} = \{X\ Y\ Z\}^T$, the Cartesian body reference frame $\vec{r} = \{x\ y\ z\}^T$ and a local beam-element $\vec{r}_i = \{c\ s\ n\}^T$. The reference frames are visualized in Figure 2.2.

The transformation from xyz-body axes to the Earth reference frame is via Euler angles, $\vec{\Theta} = \{\Phi\ \Theta\ \Psi\}^T$ with transformation tensor $\bar{\bar{T}}_E$ as:

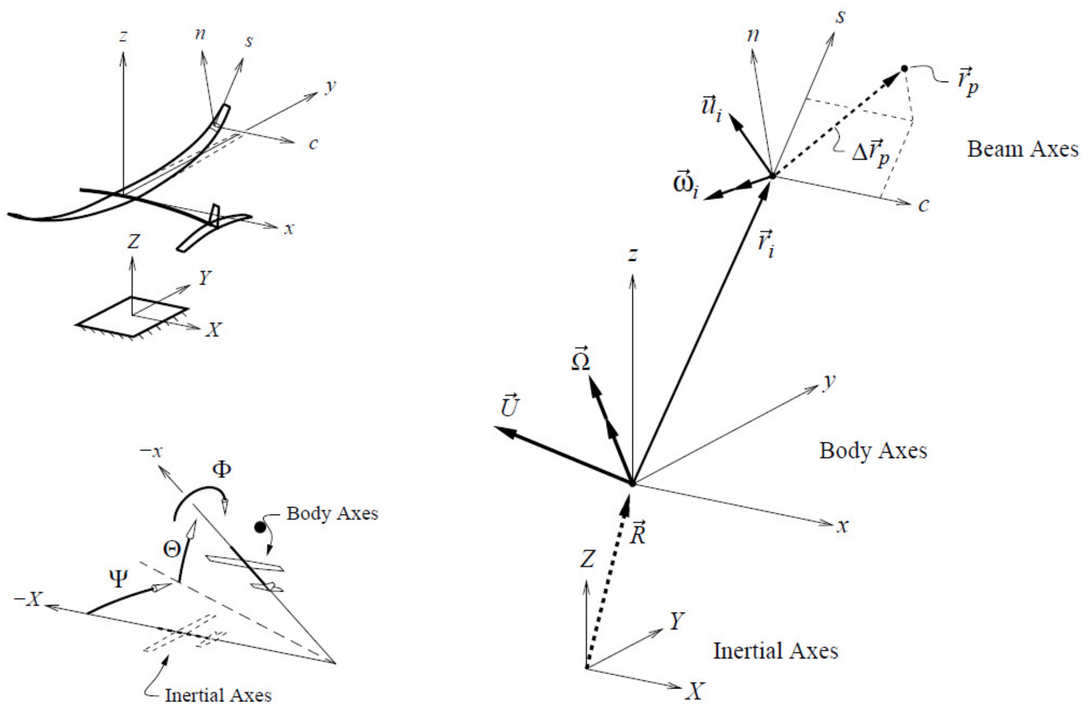


Figure 2.2: Reference frames as used in ASWING (Drela, 2008a)

$$\bar{\bar{T}}_E = \begin{bmatrix} \cos\Psi & \sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\Theta & 0 & \sin\Theta \\ 0 & 1 & 0 \\ -\sin\Theta & 0 & \cos\Theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{bmatrix} \tag{2.1}$$

$$\bar{\bar{T}}_E = \begin{bmatrix} \cos\Theta\cos\Psi & -\sin\Phi\sin\Theta\cos\Psi + \cos\Phi\sin\Psi & \cos\Phi\sin\Theta\cos\Psi + \sin\Phi\sin\Psi \\ -\cos\Theta\sin\Psi & \sin\Phi\sin\Theta\sin\Psi + \cos\Phi\cos\Psi & -\cos\Phi\sin\Theta\sin\Psi + \sin\Phi\cos\Psi \\ -\sin\Theta & -\sin\Phi\cos\Theta & \cos\Phi\cos\Theta \end{bmatrix} \tag{2.2}$$

The transformation from the xyz-airplane body axes to the local csn beam element axis is via Euler angles $\vec{\theta}_i = \{\varphi\ \vartheta\ \psi\}^T$.

$$\bar{\bar{T}} = \begin{bmatrix} \cos\vartheta & 0 & -\sin\vartheta \\ 0 & 1 & 0 \\ \sin\vartheta & 0 & \cos\vartheta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix} \tag{2.3}$$

$$\bar{\bar{T}} = \begin{bmatrix} \cos\vartheta\cos\psi & \cos\vartheta\sin\psi\cos\varphi + \sin\vartheta\sin\varphi & \cos\vartheta\sin\psi\sin\varphi - \sin\vartheta\cos\varphi \\ -\sin\psi & \cos\psi\cos\varphi & \cos\psi\sin\varphi \\ \sin\vartheta\cos\psi & \sin\vartheta\sin\psi\cos\varphi - \cos\vartheta\sin\varphi & \sin\vartheta\sin\psi\sin\varphi + \cos\vartheta\cos\varphi \end{bmatrix} \tag{2.4}$$

The beam curvature tensor $\bar{\bar{\kappa}}$ is related to the rate of change of transformation tensor $\bar{\bar{T}}$ and this tensor is singular at sweep angles $\psi = \pm 90°$. For surface beams 90° sweep angles are unlikely, but fuselage beams are generally aligned with the x axis. This problem is eliminated by switching the order of the φ and ψ rotations.

2.1.3 Aerodynamic principles

ASWING's aerodynamic model is based on a lifting line model, which employs wind-aligned trailing vorticity, a Prandtl-Glauert compressibility transformation and local-stall lift coefficients. In this section the aerodynamic principles are briefly introduced. For a more detailed description, see Drela (2009) and Drela (2008a).

The aerodynamic lift vector ($\vec{f}_{\text{lift}}$) for surface beams is determined from the Kutta-Joukowsky theorem:

$$\vec{f}_{\text{lift}} = \rho\Gamma\vec{V} \times \hat{s} \tag{2.5}$$

In this equation ρ is density, Γ is the local circulation and $\vec{V}$ the local velocity relative to the beam section at location $\vec{r}$:

$$\vec{V}(\vec{r}) = \vec{V}_\infty - \vec{\Omega} \times \vec{r} + \vec{V}_{\text{ind}}(\vec{r}) + \vec{V}_{\text{gust}}(\vec{r}) \tag{2.6}$$

In this equation $\vec{V}_\infty$ is the undisturbed flight velocity, $\vec{\Omega}$ is the rotational velocity at location $\vec{r}$, $\vec{V}_{\text{ind}}$ is the induced velocity and $\vec{V}_{\text{gust}}$ the gust velocity.

The aerodynamic drag force vector is a combination of the friction and pressure drag components:

$$\vec{f}_{\text{drag}} = \frac{1}{2}\rho\left|\vec{V}\right|\vec{V}\bar{c}c_{d,f} + \frac{1}{2}\rho\left|\vec{V}_\perp\right|\vec{V}_\perp\bar{c}c_{d,p} + 2\rho\frac{\vec{V}_\perp}{\left|\vec{V}_\perp\right|}\left(\vec{V}\cdot\hat{n}\right)^2_{c.p.}\bar{c} \tag{2.7}$$

In this equation $\bar{c}$ is the characteristic chord, $c_{d,f}$ is the friction drag coefficient, $c_{d,p}$ is the pressure drag coefficient, $\vec{V}_\perp$ is the velocity perpendicular to the wing's spanwise axis and $\hat{n}$ is the normalized n.

The aerodynamic profile moment vector is calculated as:

$$\vec{m}_{\text{lift}} = (\bar{c}/4 - \vec{x}_0)\,\hat{c} \times \vec{f}_{\text{lift}} + \frac{1}{2}\rho\left|\vec{V}_\perp\right|^2\bar{c}^2 c_m\hat{s} \tag{2.8}$$

In this equation $\vec{m}_{\text{lift}}$ is the moment due to the lift force, $\vec{x}_0$ is the chordwise location of the csn origin, c_m is the moment coefficient, and $\hat{c}$ and $\hat{s}$ are respectively the normalized c and s.

The moments due to friction forces are generally negligible (Drela, 2009).

2.1.4 Structural equations

The ASWING structural equations are based on a fully non-linear Bernouilli-Euler beam representation for all surface and fuselage structures. This section introduces the basic structural equations of this beam theory.

The total extensional strain at arbitrary location $c,\ n$ is:

$$\epsilon = \epsilon_s + c\left(\kappa_n - \kappa_{n0}\right) - n\left(\kappa_c - \kappa_{c0}\right) \tag{2.9}$$

In this equation κ is the curvature and a subscript $()_0$ denotes the unloaded beam. The relations of local forces ($\vec{F}_i$) and moments ($\vec{M}_i$) with beam strains (γ and ϵ) and curvatures are now determined with the stiffness matrix:

$$\left\{\begin{array}{c} F_c \\ F_s \\ F_n \\ \hline M_c \\ M_s \\ M_n \end{array}\right\} = \left[\begin{array}{ccc|ccc} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} & \varepsilon_{14} & \varepsilon_{15} & \varepsilon_{16} \\ \cdot & \varepsilon_{22} & \varepsilon_{23} & \varepsilon_{24} & \varepsilon_{25} & \varepsilon_{26} \\ \cdot & \cdot & \varepsilon_{33} & \varepsilon_{34} & \varepsilon_{35} & \varepsilon_{36} \\ \hline \cdot & \cdot & \cdot & \varepsilon_{44} & \varepsilon_{45} & \varepsilon_{46} \\ \cdot & \cdot & \cdot & \cdot & \varepsilon_{55} & \varepsilon_{56} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \varepsilon_{66} \end{array}\right] \left\{\begin{array}{c} \gamma_c \\ \epsilon_s \\ \gamma_n \\ \kappa_c - \kappa_{c0} \\ \kappa_s - \kappa_{s0} \\ \kappa_n - \kappa_{n0} \end{array}\right\} \tag{2.10}$$

Each entry of the stiffness matrix ε denotes a stiffness. The top 3 rows of the stiffness matrix are assumed the restricted form written in terms of shear stiffness (GK_c and GK_n), extensional stiffness (EA), elastic axis (c_{ea} and n_{ea}) and the tensile axis (c_{ta} and n_{ta}):

$$\left[\begin{array}{ccc|ccc} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} & \varepsilon_{14} & \varepsilon_{15} & \varepsilon_{16} \\ \cdot & \varepsilon_{22} & \varepsilon_{23} & \varepsilon_{24} & \varepsilon_{25} & \varepsilon_{26} \\ \cdot & \cdot & \varepsilon_{33} & \varepsilon_{34} & \varepsilon_{35} & \varepsilon_{36} \\ \hline \cdot & \cdot & \cdot & \varepsilon_{44} & \varepsilon_{45} & \varepsilon_{46} \\ \cdot & \cdot & \cdot & \cdot & \varepsilon_{55} & \varepsilon_{56} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \varepsilon_{66} \end{array}\right] \rightarrow \left[\begin{array}{ccc|ccc} GK_c & 0 & 0 & 0 & GK_c n_{ea} & 0 \\ \cdot & EA & 0 & -EAn_{ta} & 0 & EAc_{ta} \\ \cdot & \cdot & GK_n & 0 & -GK_n c_{ea} & 0 \\ \hline \cdot & \cdot & \cdot & \varepsilon_{44} & \varepsilon_{45} & \varepsilon_{46} \\ \cdot & \cdot & \cdot & \cdot & \varepsilon_{55} & \varepsilon_{56} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \varepsilon_{66} \end{array}\right] \tag{2.11}$$

The lower right quadrant of the stiffness matrix is defined as a function of the bending stiffness moment of inertia (EI_{cc} and EI_{nn}), the bending stiffness product of inertia (EI_{cn}), the bending/torsion coupling stiffness (EI_{cs} and EI_{sn}) and the torsional stiffness (GJ):

$$\left[\quad \bar{\bar{E}} \quad\right] = \left[\begin{array}{ccc} EI_{cc} & EI_{cs} & EI_{cn} \\ \cdot & GJ & EI_{sn} \\ \cdot & \cdot & EI_{nn} \end{array}\right] = \left[\begin{array}{lll} \varepsilon_{44} - \varepsilon_{22} - n_{ta}^2 & \varepsilon_{45} & \varepsilon_{46} + \varepsilon_{22} c_{ta} n_{ta} \\ \cdot & \varepsilon_{55} - \varepsilon_{11} n_{ea}^2 - \varepsilon_{33} c_{ea}^2 & \varepsilon_{56} \\ \cdot & \cdot & \varepsilon_{66} - \varepsilon_{22} c_{ta}^2 \end{array}\right] \tag{2.12}$$

With equation 2.11 and 2.12 equation 2.10 can be rewritten as:

$$\left\{\begin{array}{c} \gamma_c \\ \epsilon_s \\ \gamma_n \end{array}\right\} = \left\{\begin{array}{c} F_c/GK_c \\ F_s/EA \\ F_n/GK_n \end{array}\right\} + \left[\begin{array}{ccc} 0 & -n_{ea} & 0 \\ n_{ta} & 0 & -c_{ta} \\ 0 & c_{ea} & 0 \end{array}\right] \left[\quad \bar{\bar{E}} \quad\right] \left\{\begin{array}{c} M'_c \\ M'_s \\ M'_n \end{array}\right\} \tag{2.13}$$

$$\left\{\begin{array}{c} \kappa_c - \kappa_{c0} \\ \kappa_s - \kappa_{s0} \\ \kappa_n - \kappa_{n0} \end{array}\right\} = \left[\quad \bar{\bar{E}} \quad\right] \left\{\begin{array}{c} M'_c \\ M'_s \\ M'_n \end{array}\right\} \tag{2.14}$$

In this equation $\vec{M}'$ is the moment translated to the tension and elastic axis:

$$\left\{\begin{array}{c} M'_c \\ M'_s \\ M'_n \end{array}\right\} = \left\{\begin{array}{c} M_c \\ M_s \\ M_n \end{array}\right\} + \left[\begin{array}{ccc} 0 & n_{ta} & 0 \\ -n_{ea} & 0 & c_{ea} \\ 0 & -c_{ta} & 0 \end{array}\right] \left\{\begin{array}{c} F_c \\ F_s \\ F_n \end{array}\right\} \tag{2.15}$$

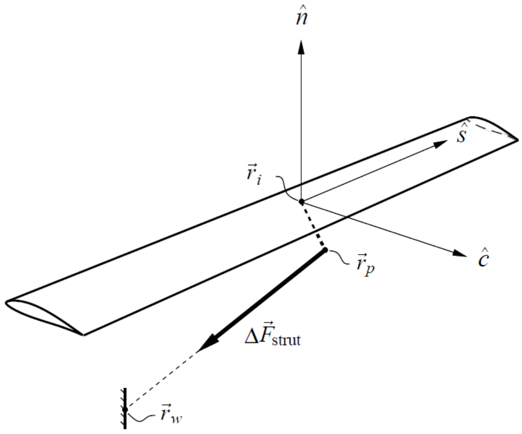


Figure 2.3: Strut as used in ASWING (Drela, 2008a)

2.1.5 Struts

The ASWING strut is assumed perfectly flexible in bending, but has a finite extensional stiffness, which allows the strut to change its length in response to extensional or compressive loads. One end is attached at the aircraft, whereas the other end is fixed in the aircraft reference frame. A representation of an ASWING strut connection is given in Figure 2.3.

The strut forces and moments are determined as:

$$\Delta\vec{F}_{st} = EA_{st}\left(\frac{\left|\vec{L}_{st}\right|}{\left|\vec{L}_{st}\right|_0} - 1\right)\frac{\left|\vec{L}_{st}\right|}{\left|\vec{L}_{st}\right|_0} \tag{2.16}$$

$$\Delta\vec{M}_{st} = \Delta\vec{r}_p \times \Delta\vec{F}_{st} \tag{2.17}$$

In these equation EA_{st} is the strut extensional stiffness, $\left|\vec{L}_{st}\right|$ is the strut length, $\left|\vec{L}_{st}\right|_0$ is the strut zero load length and $\Delta\vec{r}_p$ is the rigid pylon offset from the attachment location at the wing.

2.2 Eigenmode analysis

This section first briefly explains the ASWING Eigenmode analysis (MODE) with a flowchart and next a concise description of the equation set-up for Arnoldi iterations.

2.2.1 Program structure

ASWING's linearised unsteady natural response is based on the Fortran77 compatible Arnoldi package (ARPACK). In OPER the equilibrium solutions are determined and the trimmed aircraft is entered into the Eigenmode analysis. In this module the loads and moments due to engines are determined first. Next the Newton system of the discrete equation set are determined for each beam and operating point. Then the loads, moments and Jacobians due to aerodynamics, gravity, joints, struts and point masses are included into the Newton system as well as the kinematic constrains due to grounds. This Newton system is completed with the unsteady equations.

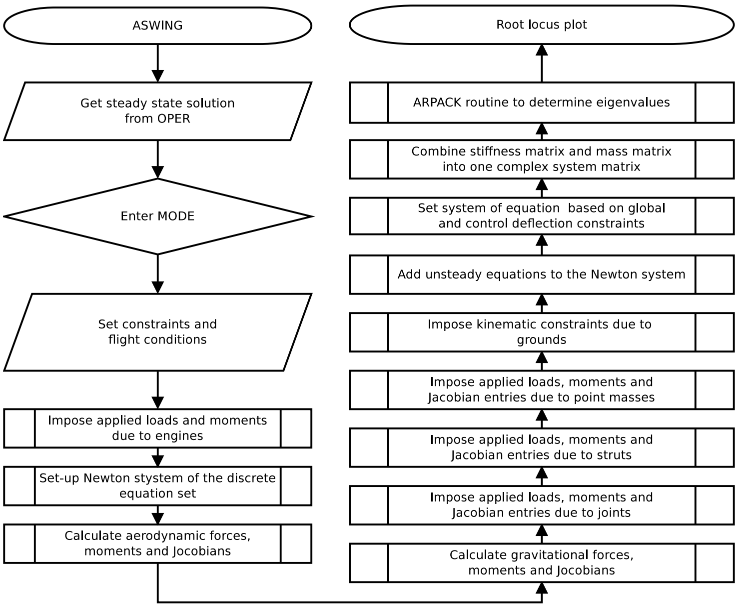


Figure 2.4: Flowchart ASWING Eigenmode analysis

The stiffness and mass matrices are combined into one complex system matrix and rewritten for ARPACK compatibility. Next the eigenvalues are calculated and ASWING plots a root locus plot: a plot with frequencies at one axis and growth rate at the other. Modes with a negative growth rate are stable, whereas a positive growth rate indicates an unstable mode. The Eigenmode analysis is visualized in a flowchart in Figure 2.4. This flowchart includes only the main calculation steps.

2.2.2 Equations set-up

Drela (2008a) gives the equation system as a function of a state vector $\mathbf{x}$, its time rate of change $\dot{\mathbf{x}}$ and commanded variables $\mathbf{u}$:

$$\mathbf{x} = \Big(\vec{r}_i \;\; \vec{\theta}_i \;\; \vec{M}_i \;\; \vec{F}_i \;\; \vec{u}_i \;\; \vec{\omega}_i \\ \Delta\vec{r}_J \;\; \Delta\vec{\theta}_J \;\; \Delta\vec{M}_J \;\; \Delta\vec{F}_J \;\; A_1 \;\; A_2 \ldots A_K \;\; \vec{R}_E \;\; \vec{\Theta} \;\; \vec{U} \;\; \vec{\Omega} \;\; \vec{a}_0 \;\; \vec{\alpha}_0 \\ \delta_{F_1} \;\; \delta_{F_2} \ldots \;\; \delta_{e_1} \;\; \delta_{e_2} \ldots \;\; \delta_{g_1} \;\; \delta_{g_2} \ldots \;\; \mathbf{e} \Big) \tag{2.18}$$

$$\dot{\mathbf{x}} = \Big(\dot{\vec{r}}_i \;\; \dot{\vec{\theta}}_i \;\; \dot{\vec{u}}_i \;\; \dot{\vec{\omega}}_i \\ \dot{A}_1 \;\; \dot{A}_2 \ldots \dot{A}_K \;\; \dot{\vec{R}}_E \;\; \dot{\vec{\Theta}} \;\; \dot{\vec{U}} \;\; \dot{\vec{\Omega}} \;\; \dot{\mathbf{e}} \Big) \tag{2.19}$$

$$\mathbf{u} = (V_c \;\; \alpha_c \;\; \beta_c \;\; \Phi_c \;\; \Theta_c \;\; \Psi_c \;\; \delta_{F_1,c} \;\; \delta_{F_2,c} \ldots) \tag{2.20}$$

The first row of the variables listed at state vector $\mathbf{x}$ are local variables, the second global variables and the third row user defined variables. In this equation set up $\vec{r}_i$ is the local csn coordinate for a beam element, $\vec{\theta}_i$ the deflection angles, $\vec{F}_i$ the forces, $\vec{M}_i$ the moments, $\vec{u}_i$ the linear velocities and $\vec{\omega}_i$ the angular velocities. $A_1 \;\; A_2 \ldots A_K$ are the vortex strengths, $\vec{R}_E$ is the Earth reference frame, $\vec{\Theta}$ are the roll, pitch and yaw angle, $\vec{U}$ the linear flight velocity, $\vec{\Omega}$ the angular flight velocity, $\vec{a}_0$ the linear accelerations and $\vec{\alpha}_0$ the angular accelerations. The flap deflections, engine settings and gust input is given by respectively δ_F, δ_e and δ_g. The error-integral vector is denoted as $\mathbf{e}$. The commanded variables $\mathbf{u}$ are user inputs and subscript $()_c$

denotes control.

The general solution is written in the residual form:

$$\mathbf{r}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}) = 0 \tag{2.21}$$

The general perturbation of the system of equations is linearised via three Jacobian matrix terms.

$$\delta\mathbf{r} = \left[\frac{\partial \mathbf{r}}{\partial \mathbf{x}}\right]\delta\mathbf{x} + \left[\frac{\partial \mathbf{r}}{\partial \dot{\mathbf{x}}}\right]\delta\dot{\mathbf{x}} + \left[\frac{\partial \mathbf{r}}{\partial \mathbf{u}}\right]\delta\mathbf{u} \tag{2.22}$$

This equation forms the foundation of Eigenmode analysis. For a converged solution with state $\mathbf{x}$ and prescribed state $\mathbf{u}$, the the sum of residuals remain zero and equation 2.22 can be rewritten as:

$$-\left[\frac{\partial \mathbf{r}}{\partial \dot{\mathbf{x}}}\right]\delta\dot{\mathbf{x}} = \left[\frac{\partial \mathbf{r}}{\partial \mathbf{x}}\right]\delta\mathbf{x} + \left[\frac{\partial \mathbf{r}}{\partial \mathbf{u}}\right]\delta\mathbf{u} \tag{2.23}$$

This equation is only valid for a converged solution in the trimmed state, which has time-invariant Jacobian matrices. The simplest case is straight and level flight with zero climb rate and zero bank angle.

Assume a perturbation solution of the form:

$$\delta\mathbf{x}(t) = \hat{\mathbf{x}}e^{\lambda t} \tag{2.24}$$

Eigenmode analyses concern the unforced case at which $\delta\mathbf{u} = 0$. Substitute equation 2.24 in equation 2.23 to get a solution of the form:

$$\left[\frac{\partial \mathbf{r}}{\partial \mathbf{x}}\right]\hat{\mathbf{x}} = -\left[\frac{\partial \mathbf{r}}{\partial \dot{\mathbf{x}}}\right]\hat{\mathbf{x}}\lambda \tag{2.25}$$

Eigenvalues (λ_k) and eigenvector ($\mathbf{v}_k$) are defined as the nontrivial solutions to the unforced perturbed system:

$$\left[\frac{\partial \mathbf{r}}{\partial \mathbf{x}}\right]\mathbf{v_k} = -\left[\frac{\partial \mathbf{r}}{\partial \dot{\mathbf{x}}}\right]\mathbf{v_k}\lambda_k \tag{2.26}$$

2.3 Conclusions

The structure of ASWING allows for adding multiple subroutines for several concentrated applied loads such as struts, point masses and engine loads. The strut module can model bracing wires, which are attached by one end at the aircraft and at the other end fixed in the aircraft reference frame. However these struts cannot be attached to each other and additionally one strut end (the tether) should be fixed in the Earth reference frame. Therefore the strut module lacks the ability to simulate a tether-bridle system.

Chapter 3

Input ASWING

The goal of this chapter is to (1) describe the parameters needed in an ASWING input file (`xxx.asw`), (2) for surface and fuselage beams, calculate the aerodynamic and structural parameters and (3) to verify the structural parameters with FEM data. In this chapter the M600 characteristics are used as sample input. The resulting ASWING input file is used in the aero-elastic analysis in chapter 10. A more detailed description of ASWING input files can be found at Drela (2008b). In section 3.1, the M600 geometry is defined, in section 3.2 the aerodynamic properties are defined, in section 3.3 the structural properties and in section 3.4 the engine properties. In section 3.5 the conclusions of this chapter are drawn.

3.1 Geometry of the M600

In ASWING, the geometry can be specified with the parameters listed in Table 3.1. The location of the distance to the c, n origin (x_0/c) is dependent on the structural properties of the wing and explained in section 3.3. The remaining parameters are described in this section.

A top-view and side-view of the M600 geometry are respectively given in Figure 3.1 and 3.2. The full wingspan, $b = 28m$, the characteristic chord, $\bar{c} = 1.40m$ and hence the wing's aspect ratio is equal to $AR = 20$. From the wing root to half-way span, the chord is constant. From the bridle-wing attachment towards the tip, the chord tapers off, because structural moment loads are significantly reduced. Flaps are attached at the rear of the wing. These flaps create extra lift when needed and de-power the wing in case of high flight speeds. The flaps at the tips control the roll motion. The rudder and horizontal stabilizer are attached to the 8 meter long fuselage for yaw and pitch stability. The rudder has an additional flap for yaw control. The entire horizontal stabilizer is hinged for pitch control.

Table 3.1: ASWING geometry inputs

ASWING keyword	Symbol	Description	Unit
`x, y, z`	$\vec{r}_0$	location with respect to s-axis of unloaded beam	$[m]$
`twist`	$\vec{\theta}_0$	twist angle of unloaded beam	[deg]
`chord`	c	wing chord	$[m]$
`Xax`	(x_0/c)	distance/chord from leading edge to s axis (c, n origin)	$[-]$

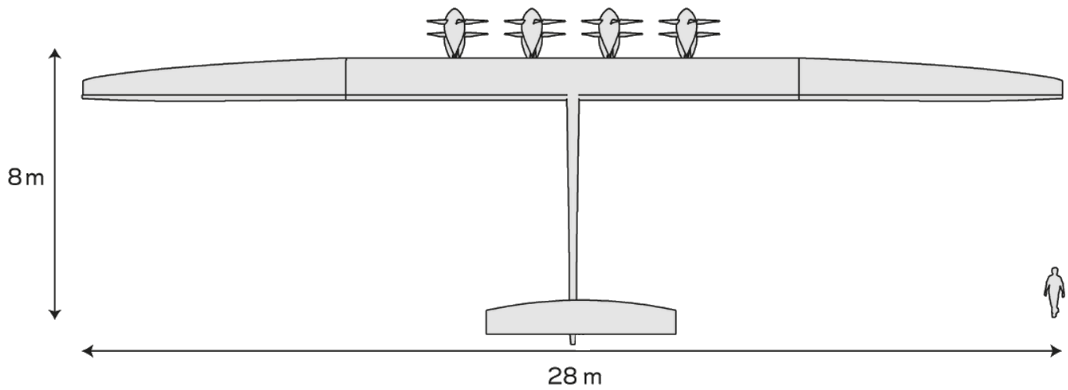


Figure 3.1: M600 geometry, top-view

The main wing is outfitted with eight motors, best shown in Figure 3.2. Each motor has a $75kW$ rated power output.

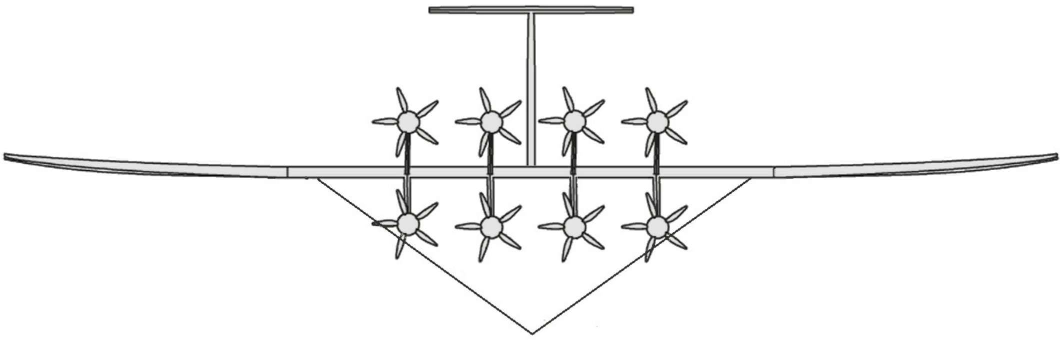


Figure 3.2: M600 geometry, side-view

3.2 Aerodynamic properties

The aerodynamic properties for the lifting surfaces; the main wing, the horizontal stabilizer, the rudder and the motor pylons are characterized with specific characteristic properties. The method for defining these properties in an `xxx.asw` input file is similar and is described for the main wing first.

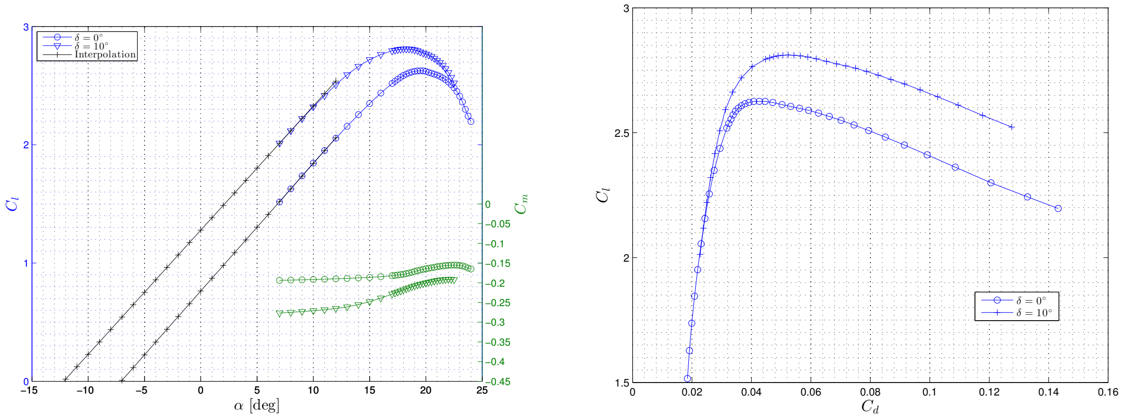
3.2.1 Aerodynamic properties of the M600 main wing

ASWING has a specific set-up for the aerodynamic properties. For each lifting surface the aerodynamic properties are given in Table 3.2.

All of the properties given in Table 3.2 are determined from the lift curve, the drag polar and the moment coefficient curve. For the main wing these aerodynamic properties are given in Figure 3.3.

Table 3.2: ASWING aerodynamic inputs

ASWING keyword	Symbol	Description	Unit
`Cdf`	$c_{d,f}$	section profile friction drag coefficient	[-]
`Cdp`	$c_{d,p}$	section profile pressure drag coefficient	[-]
`alpha`	α_{Ao}	angle of zero-lift line above c-axis	[°]
`Cm`	c_m	section pitching moment coefficient about chord/4	[-]
`CLmax`	$c_{l,\text{ max}}$	section maximum lift coefficient	[-]
`CLmin`	$c_{l,\text{ min}}$	section minimum lift coefficient	[-]
`dCLda`	c_{l_α}	section lift-curve slope	[1/rad]
`dCLFi`	c_{l_δ}	lift coefficient increment with flap deflection	[1/deg]
`dCMdFi`	c_{m_δ}	moment coefficient increment with flap deflection	[1/deg]



(a) Lift- and moment coeffecient curve M600 main wing

(b) Drag polar M600 main wing

Figure 3.3: Aerodynamic properties M600 main wing

The aerodynamic data is available for an angle-of-attack range between $7° \leq \alpha \leq 24°$, at which $c_l > 1.5$. At greater angles-of-attack, the airfoil is in a deep stall regime. This regime is outside the M600's flight envelope. However at high flight speeds, a lower lift coefficient than $c_l = 1.5$ is likely. Therefore, the lift curve is interpolated up to $c_l = 0$. For 0° and 10° flap deflection, the lift curve slope is respectively $c_{l_\alpha} = 1.97\pi$ and $c_{l_\alpha} = 1.92\pi$, which is close to the theoretical lift curve slope for a two dimensional flat plate, $c_{l_\alpha} = 2.00\pi$.

ASWING assumes a moment coefficient which is independent of angle-of-attack. From Figure 3.3a follows that the moment coefficient increases slightly with angle-of-attack. This occurs in the deep stall regime during which the flow separates from the airfoil. The angle-of-attack at design lift coefficient, $c_{l,\text{ design}} = 2.1$, is equal to $\alpha_{\text{design}} = 12.4°$. The difference in moment coefficients from $\alpha_{\text{design}} - 4° \leq \alpha \leq \alpha_{\text{design}} + 4°$ is about 2%. Hence the constant moment coefficient assumption is valid in this flight regime.
Flap deflection shifts the lift- and moment coefficient curves. This effect is determined in the linear part of the lift curve.

$$c_{l_\delta} = \frac{\Delta c_l}{\Delta \delta}, \qquad c_{m_\delta} = \frac{\Delta c_m}{\Delta \delta} \tag{3.1}$$

The total drag is determined as the sum of profile, friction and induced drag. According to Drela (2009) the sum of the profile and friction drag is the drag polar lower limit, which follows from Figure 3.3b.

3.2.2 Aerodynamic properties of the M600 surface areas excluding the main wing

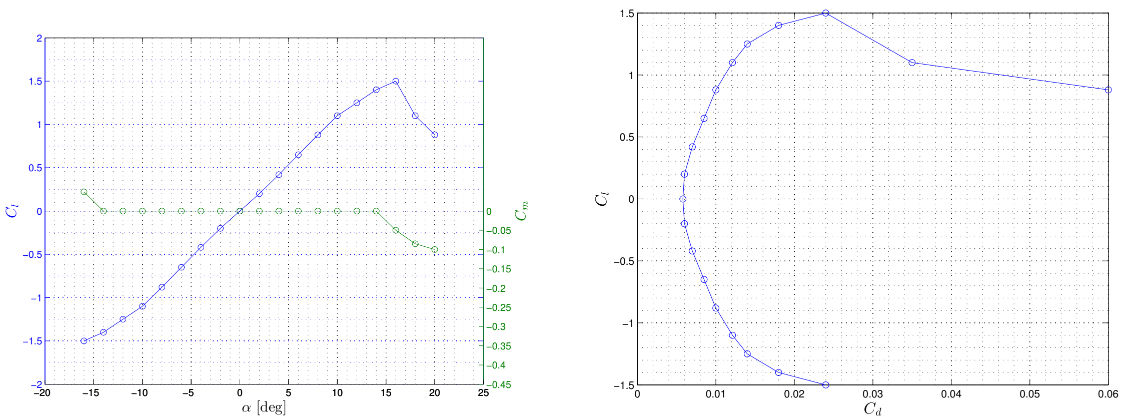
For the the horizontal stabilizer, rudder and the motor pylons, the aerodynamic properties are determined with the method described in section 3.2.1. Each of these aerodynamic surfaces has its own characteristics which will be explained in this section.

Aerodynamic properties of the M600 horizontal stabilizer

Unlike the main wing, the horizontal stabilizer has no high maximum lift coefficient design requirement. According to Vander Lind (2013a) '*every moderately thin and low drag airfoil has the same change in lift force with angle-of-attack and thus the same contribution to stability of the flight vehicle.*'. The horizontal stabilizer has no separate control surface, because the whole surface is hinged for control. In this design stage, the relatively thin NACA0012 airfoil is chosen. The aerodynamic characteristics for a specific Reynolds number are determined with Javafoil (Hepperle, 2006). The Reynolds number Re is calculated as:

$$\mathrm{Re} = \frac{Vc}{\nu} \tag{3.2}$$

With $V = 50m/s$, $c = 1.00m$ and $\nu = 10^{-5}m^2/s$, the Reynolds number is $\mathrm{Re} = 3.75 \cdot 10^6$. The aerodynamic properties are given in Figure 3.4.



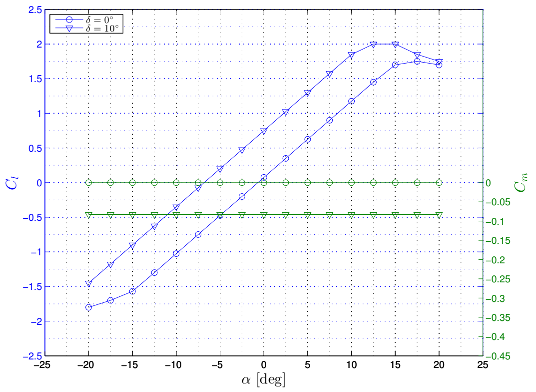
(a) Lift- and moment coeffecient curve M600 horizontal stabilizer

(b) Drag polar M600 horizontal stabilizer

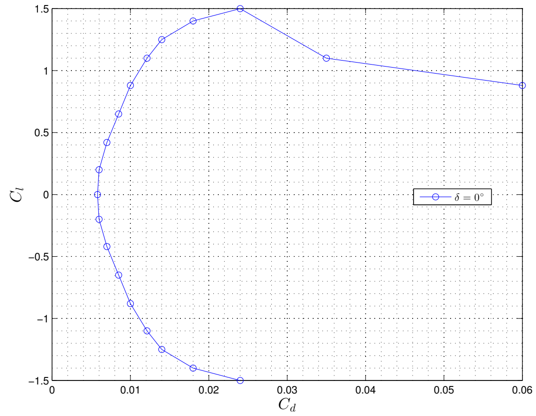
Figure 3.4: Aerodynamic properties M600 horizontal stabilizer

Aerodynamic properties of the M600 rudder

The aerodynamic design of the M600 rudder is optimized for control effectiveness by increasing control authority for a given rudder force (Vander Lind, 2013a). For the rudder, only the lift curve data is available. The moment and drag coefficient need to be derived. The rudder camber is very small and hence the moment coefficient is very close to $c_m = 0.0$, in the case of zero flap deflection. The flap deflection increases camber and shift the center of pressure forward resulting in a pitching down moment. In this design stage, it is assumed that the moment coefficient change with flap deflection, c_{m_δ} of the rudder is equal to c_{m_δ} of the main wing. The minimum vertical tail drag coefficient is assumed to be equal to the horizontal tail drag coefficient.



(a) Lift- and moment coeffecient curve M600 rudder

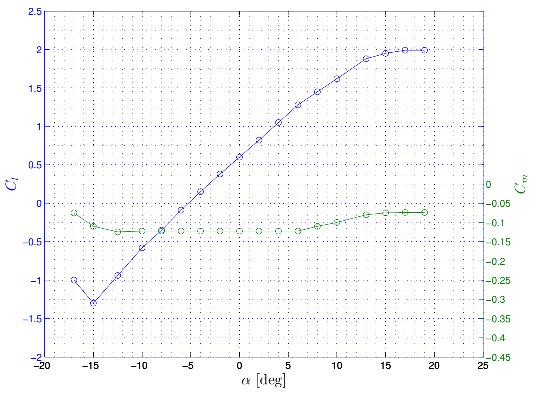


(b) Drag polar M600 rudder

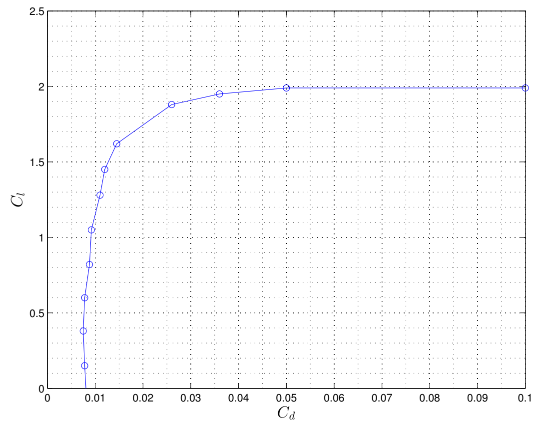
Figure 3.5: Aerodynamic properties M600 rudder

Aerodynamic properties of the M600 motor pylons

The aerodynamic design of the motor pylons is optimized for high maximum lift coefficient, benign stall, low drag and negative lift capability (Vander Lind, 2013a). The resulting lift- and moment coefficient curves are shown in Figure 3.6a. The drag polar is given in Figure 3.6b. With respect to the other lifting surfaces, this surface is characterized with the lowest drag coefficients.



(a) Lift- and moment coeffecient curve M600 motor pylons



(b) Drag polar M600 motor pylons

Figure 3.6: Aerodynamic properties M600 motor pylons

3.3 Structural properties

ASWING needs specific input to define structural properties of support beams (fuselages) and surface beams (lifting surfaces). These properties are derived from geometrical and material properties. As with the aerodynamic properties, the structural properties for the M600 are used as sample input. ASWING allows for the structural properties given in Table 3.3. All of these listed structural beam properties are covered in the next sections, but the extensional and shear-extensional damping time. In case a parameters is excluded from `asw.xxx` its default value is applied. For stiffnesses EI_{cc}, EI_{nn}, GJ, EA, GK_c and GK_n the default value is infinity. For all other structural properties the default is zero.

Table 3.3: ASWING structural inputs

ASWING keyword	Symbol	Description	Unit
mg	$\mu_1 g$	first section weight/length	$[N/m]$
Dmg	$\mu_2 g$	second section weight/length	$[N/m]$
Ccg	c_{cg_1}	c-location of section mass centroid	$[m]$
Ncg	n_{cg_1}	n-location of section mass centroid	$[m]$
DCcg	c_{cg_2}	c-location of additional-mass centroid	$[m]$
DNcg	n_{cg_2}	n-location of additional-mass centroid	$[m]$
Cea	c_{ea}	c-location of elastic axis	$[m]$
Nea	n_{ea}	n-location of elastic axis	$[m]$
Cta	c_{ta}	c-location of tension axis	$[m]$
Nta	n_{ta}	n-location of tension axis	$[m]$
EIcc	EI_{cc}	bending stiffness about c-axis	$[Nm^2]$
EInn	EI_{nn}	bending stiffness about n-axis	$[Nm^2]$
EIcn	EI_{cn}	bending cross-stiffness	$[Nm^2]$
GJ	GJ	torsional stiffness	$[Nm^2]$
EA	EA	extensional stiffness	$[N]$
GKc	GK_c	c-shear stiffness	$[N]$
GKn	GK_n	n-shear stiffness	$[N]$
mgcc	$\iota_{cc_1} g$	weight-inertia/span about c-axis	$[Nm]$
mgnn	$\iota_{nn_1} g$	weight-inertia/span about n-axis	$[Nm]$
Dmgcc	$\iota_{cc_2} g$	additional weight-inertia/span about c-axis	$[Nm]$
Dmgnn	$\iota_{nn_2} g$	additional weight-inertia/span about n-axis	$[Nm]$
tdeps	t_ϵ	extensional-strain damping time	$[s]$
tdgam	t_γ	shear-strain damping time	$[s]$

Most structural properties listed in Table 3.3, are dependent on the c,n-axis definition. In Drela (2009) three different representations of the airfoil geometric and structural properties are defined as given in Figure 3.7.

According to (Drela, 2009) the first representation is the best choice if the principal bending axis angle is known by inspection from symmetry. The second representation is usually the best choice if the principal axis is not obvious from inspection, so the principal bending axis is not immediately available. The third choice may be the most convenient if all the section quantities are to be computed in the global aircraft axes xyz.

For the design of the M600 wing the principal bending axis is not directly obvious from inspection, section quantities are not to be computed in the global aircraft axis. Hence representation 2 is chosen for further analysis.

3.3.1 Structural properties of the main wing

The geometrical and material properties for the M600 main wing are mostly available and hence the structural properties can be determined. Each wing half can be divided into two separate parts:

- **Root to half-way span;** the chord and spar position is constant. Material layers are removed stepwise and hence the skin and spar thickness is decreased stepwise with increased distance from the root.

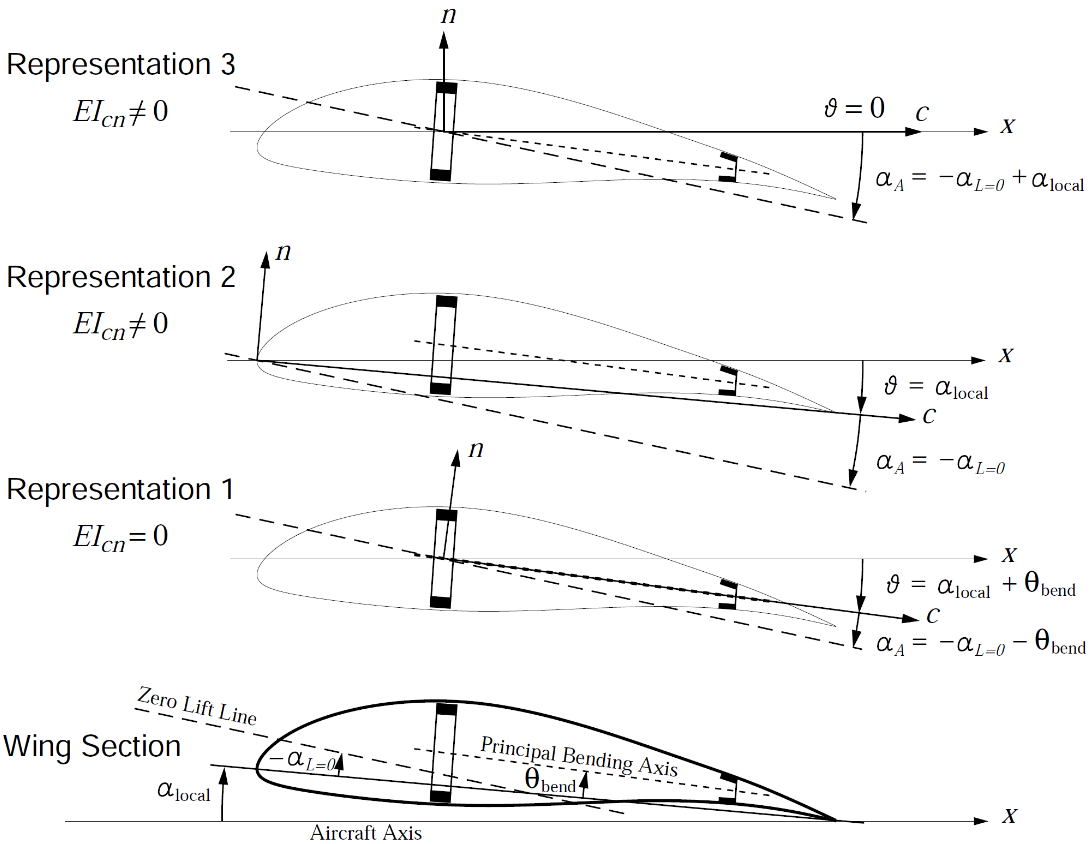


Figure 3.7: Airfoil representations (Drela, 2009)

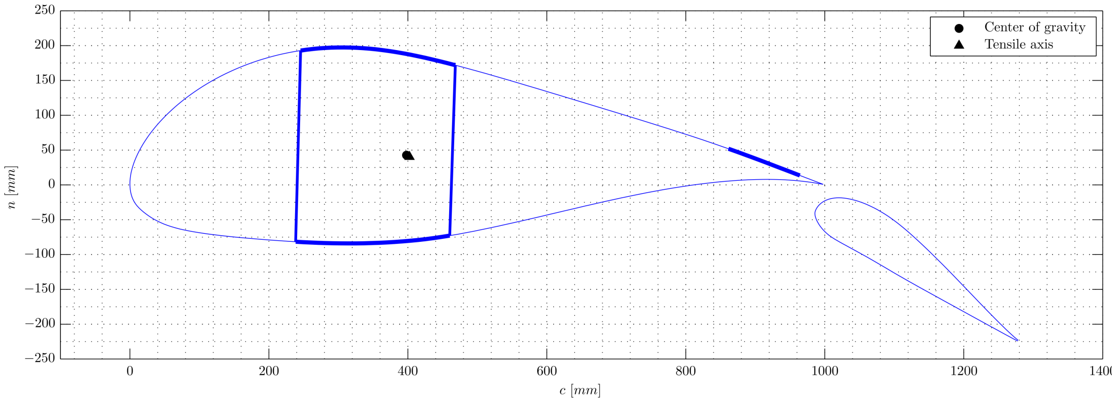
- **Half-way span to wing tip;** the chord at the wing tip is half of the chord with respect to the chord at half way span. Forward sweep is applied to keep the leading edge and the front spar straight. Again, material layers are removed stepwise.

The root and tip airfoil geometry are respectively given in Figure 3.8a and 3.8b. The flap contribution to structural properties is discarded. First the weight properties will be determined followed by the stiffness properties.

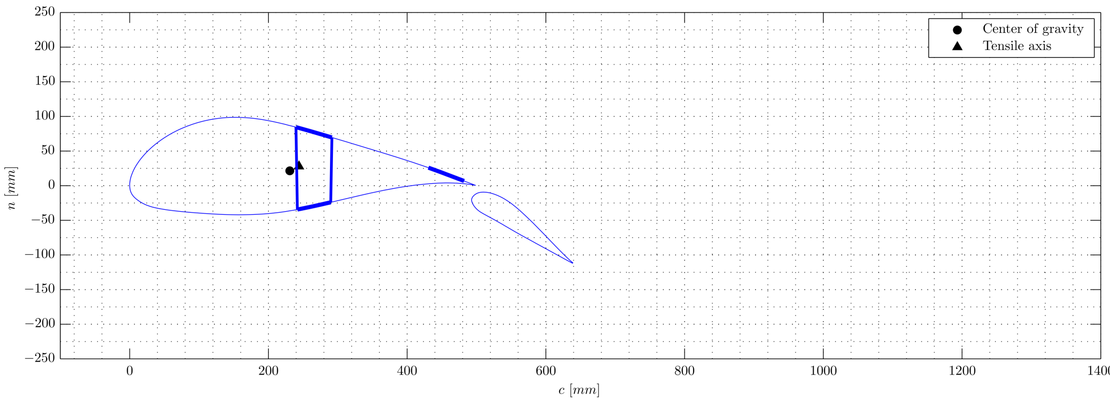
Weight properties of the main wing

The weight properties of the main wing are the weight per unit length (μg), the center of gravity ($\vec{r}_{cg}$) and the weight inertia per unit span (ιg). To determine those parameters, firs the wing weight per unit span is determined by discretization of the airfoil and the spars. The airfoil is split in several parts, with a unique fibre lay-up; the front skin, the top cap, the bottom cap, the shear-webs, the rear skin and the trailing edge cap. The top cap, bottom cap and the shear-webs form a torsion box and are designed to take most of the loads. The skin will take some part of the loads, but is mainly intended to maintain the aerodynamic shape of the airfoil.

Weight per unit length of the main wing(μg)



(a) root airfoil, including spars



(b) win tip airfoil, including spars

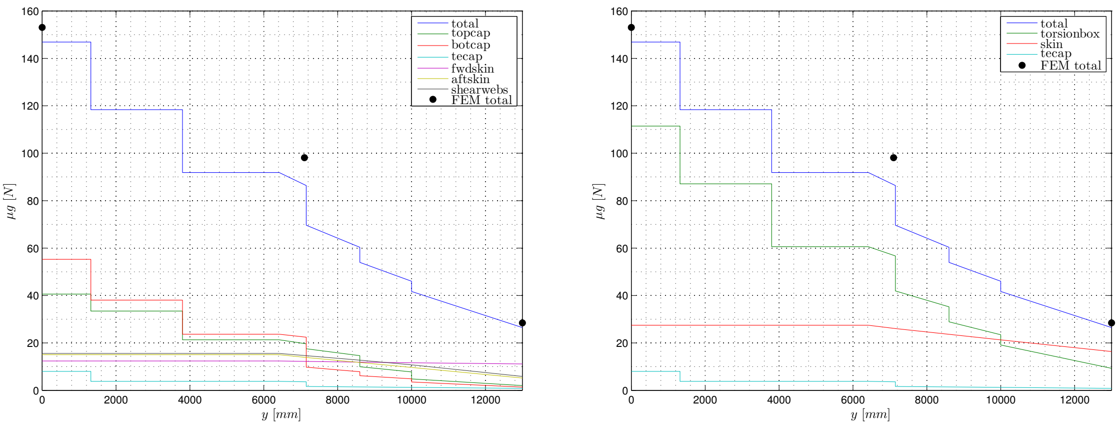
Figure 3.8: Airfoils of the main wing

The total mass per unit length is now determined as.

$$\mu_1 g = \sum \Delta\mu g \tag{3.3}$$

The weight per unit length for each airfoil segment is given in Figure 3.9a. From the root up to half-way span, weight is stepwise decreasing, because the chord is constant and material layers are removed stepwise. From half-way span to the tip, the weight is stepwise and linearly decreasing, because the chord is linearly decreasing and material layers are removed stepwise. The motors and bridles are attached from the root to half-way span and hence this is a high load regime. In this regime relatively thick top and bottom caps are used. After half-way span only aerodynamic loads need to be transferred and hence the thickness of stiffeners is decreased significantly. Figure 3.9b shows the weight impact of the torsion box (top cap, bottom cap and shear-webs), the skin and the trailing edge cap. For the final 3.00 meter towards the tip, the skin weight is higher than the weight of the torsion box. With respect to FEM results the weight per unit length is underestimated; at the root, at half-way span and at the tip the determined weight is 96%, 94% and 93% with respect to FEM determined numbers.

Center of gravity of the main wing ($\vec{r}_{cg}$)



(a) μg per airfoil segment

(b) μg for torsionbox, skin, trailing edge cap and total

Figure 3.9: Main wing airfoil weights

The airfoil center of gravity is is determined as

$$\vec{r}_{cg} = \frac{\int_r \vec{r}\mathrm{dm}}{\int_r \mathrm{dm}} \approx \frac{\sum_{i=0}^{n} \vec{r}_i \Delta\mu g}{\sum_{i=0}^{n} \Delta\mu g} \tag{3.4}$$

The airfoil center of gravity is shown in Figures 3.8a and 3.8b. At the root the weight contributions of the torsion box and hence the center of gravity is shifted forward with respect to the tip center of gravity.

Weight inertia per unit span (ι)

With the center of gravity the next property can be determined; the weight inertia per unit span. The weight inertia is a measure of the tendency to resist any change in motion and is defined by (Drela, 2009) as:

$$\iota_{cc} g = g \int \left(n - n_{cg}\right)^2 \mathrm{d}\mu \tag{3.5}$$

$$\iota_{nn} g = g \int \left(c - c_{cg}\right)^2 \mathrm{d}\mu \tag{3.6}$$

$$\iota_{ss} g = g \int \left[\left(n - n_{cg}\right)^2 + \left(c - c_{cg}\right)^2\right] \mathrm{d}\mu = g\left(\iota_{cc} + \iota_{nn}\right) \tag{3.7}$$

The weight inertia about the s-axis is the sum of the weight inertia about the c- and n-axis and there is need to specify this property separately in `xxx.asw` as ASWING will determine this $\iota_{ss} g$ from $\iota_{cc} g$ and $\iota_{nn} g$.

Stiffness properties of the main wing

The stiffness properties of the main wing are, the extensional stiffness (EA), the bending stiffness (EI), the shear stiffness (GK), the torsional stiffness (GJ), the tensile axis ($\vec{r}_{ta}$) and the elastic axis ($\vec{r}_{ea}$). The extensional stiffness with its related tensile axis, the uncoupled bending stiffness and the torsional stiffness are determined with a discretized system. For coupled bending stiffnesses, shear stiffnesses and the elastic axis a finite element program is used.

Calculation method to determine stiffnesses

Several principles exist to determine E-moduli, such as the Rule of Mixtures (ROM), the Hart-Smith 10% rule and the classical laminate analysis (CLA) (Richardson, 2013). All theories agree that material stiffness decreases rapidly with increasing inclination angle, but differ considerably about the rate of decrease as shown in Figure 3.10.

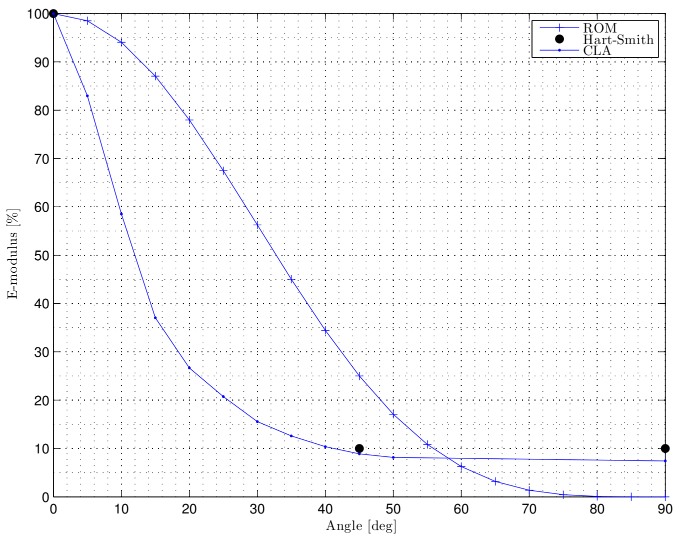


Figure 3.10: E-modulus with inclination angle for ROM, Hart-Smith and CLA

In general, CLA is considered as the most accurate method. As shown in Figure 3.10 the Hart-Smith 10% rule correlates well with CLA in case the lay-up consists of only 0/90 and 45/45 layers. The main advantage of the Hart-Smith 10% rule method with respect to CLA is its simplicity. For the M600, only 0/90 and 45/45 material layers are used and therefore the Hart-Smith method is considered as most appropriate.

Hart-Smith states that each 45° and 90° ply contributes to 10% of the stiffness of a 0° ply:

$$E_{11} = E_f V_f + E_m V_m \tag{3.8}$$

$$E_{cc} = E_{11}\left(0.1 + 0.9\frac{V_{0^\circ}}{V_f}\right) \tag{3.9}$$

$$E_{nn} = E_{11}\left(0.1 + 0.9\frac{V_{90^\circ}}{V_f}\right) \tag{3.10}$$

$$E_{ss} = E_{11}\left(0.1 + 0.9\frac{V_{0^\circ}}{V_f}\right) \tag{3.11}$$

$$G = E_{11}\left(0.028 + 0.234\frac{V_{45^\circ}}{V_f}\right) \tag{3.12}$$

In these equations EI_{11} is the composite E-modulus in fibre direction, E_f is the fibre E-modulus, V_f is the fibre volume fraction, E_m is the resin E-modulus and V_m is the resin volume fraction.

Extensional stiffness (EA) an tensile axis ($\tilde{\mathbf{r}}_{\mathbf{ta}}$)

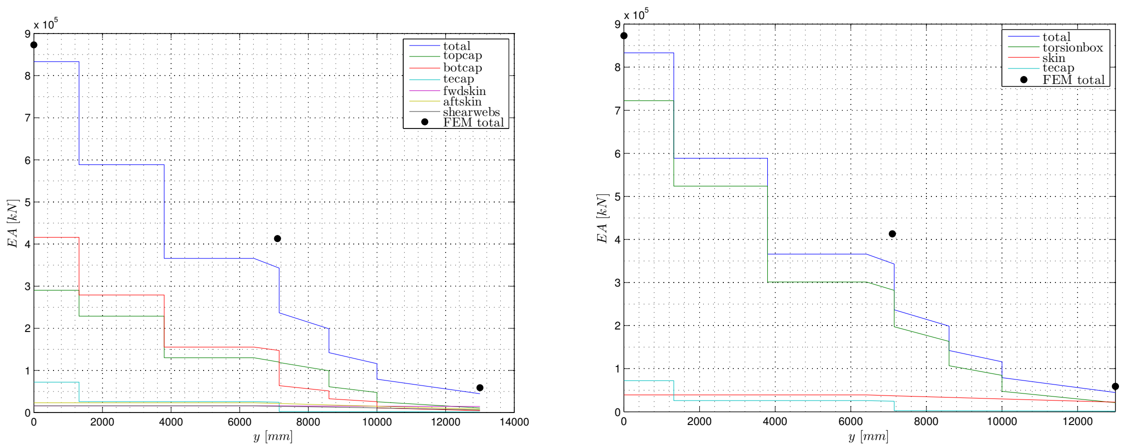
As with weight calculations, the airfoil skin and the stiffeners are discretized into multiple elements. The contribution of each element to the extensional stiffness, and the tensile axis is determined as:

$$E_{ss}A = \iint E_{ss}\ \mathrm{d}c\mathrm{d}n \approx \sum\sum E_{ss}\ \Delta c \Delta n \tag{3.13}$$

$$c_{ta} = \frac{1}{E_{ss}A}\iint E_{ss}\ c\ \mathrm{d}c\mathrm{d}n \approx \sum\sum \frac{1}{E_{ss}A} E_{ss}\ c\ \Delta c \Delta n \tag{3.14}$$

$$n_{ta} = \frac{1}{E_{ss}A}\iint E_{ss}\ n\ \mathrm{d}c\mathrm{d}n \approx \sum\sum \frac{1}{E_{ss}A} E_{ss}\ n\ \Delta c \Delta n \tag{3.15}$$

The extensional stiffness over the wingspan is given in Figures 3.11a and 3.11b. The calculated extensional stiffnesses at the root, at half-way span and at the wing tip are respectively 95%, 89% and 86% with respect to FEM.



(a) EA per airfoil segment over the wingspan

(b) EA for torsionbox, skin, trailing edge cap and total

Figure 3.11: Main wing airfoil EA

The location of the tensile axis is given in Figures 3.8a and 3.8b.

Bending stiffness of the main wing EI

With this tensile axis the bending stiffness is determined as:

$$EI_{cc} = \iint E_{cc}(n - n_{ta})^2\ \mathrm{d}c\mathrm{d}n \approx \sum\sum E_{cc}(n - n_{ta})^2\ \Delta c \Delta n \tag{3.16}$$

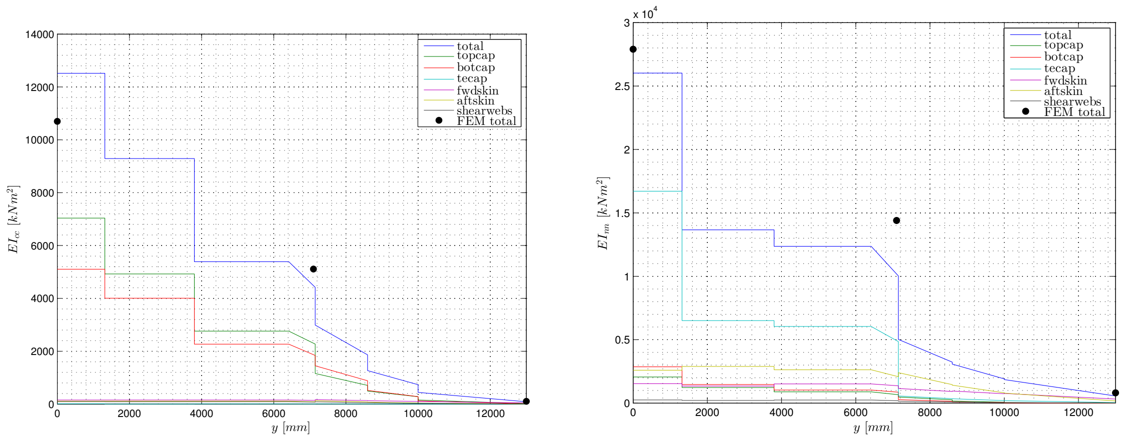
$$EI_{nn} = \iint E_{nn}(c - c_{ta})^2\ \mathrm{d}c\mathrm{d}n \approx \sum\sum E_{nn}(c - c_{ta})^2\ \Delta c \Delta n \tag{3.17}$$

$$EI_{cn} = \iint -E_{nn}(c - c_{ta})E_{cc}(n - n_{ta})\ \mathrm{d}c\mathrm{d}n \approx \sum\sum -E_{nn}(c - c_{ta})E_{cc}(n - n_{ta})\ \Delta c \Delta n \tag{3.18}$$

The EI_{cc} and EI_{nn} stiffness contributions for each airfoil segment are given in Figures 3.12a and 3.12b. The relatively stiff top and bottom caps are a relatively large distance away n from the tensile axis n_{ta}. Additionally these caps are designed to take the loads. Therefore the stiffness contribution of these stiffeners is 97.0% at the root at which the moment loads are maximum. Towards the tip, moment loads decrease and less strength and stiffness is required. The total bending stiffness about the c-axis at the tip is only 0.7%

with respect to the root. The trailing edge cap contributes significantly to the bending stiffness about the n axis, because the distance c from this stiffener to c_{ta} is relatively large.

At the root, at halfway span and at the tip the calculated EI_{cc} is respectively 117%, 86% and 86% with respect to the bending stiffness determined with FEM. For EI_{nn}, the determined stiffnesses are respectively 93%, 86% and 70% with respect to FEM.



(a) EI_{cc} per airfoil segment over the wingspan

(b) EI_{nn} per airfoil segment over the wingspan

Figure 3.12: Main wing airfoil EI

Torsional stiffness (GJ)

The torsional stiffness for closed shells is given by Peery and Azar (1982) as:

$$GJ = \frac{4A_{enc}^2}{\oint \mathrm{d}l/Gt} \tag{3.19}$$

In this equation A_{enc} is the enclosed area, which is determined as:

$$A_{enc} = \sum_{i=1}^{n-1} (c_{i+1} - c_i) \left| \frac{n_{i+1} + n_i}{2} \right| \tag{3.20}$$

The upper and lower spars contribute greatly to the torsional stiffness as a result of the increased thickness as compared to the front and the rear skin. Additionally these stiffeners have a significant proportion of bi-directional fibres oriented in the 45/45 direction. These layers contribute mostly to the torsional stiffness. The torsional stiffness of the torsion box, the front part of the airfoil and the rear part are given in Figure 3.13. At the root, at halfway span and at the tip, the torsional stiffness is respectively 154%, 93% and 60% with respect to the torsional stiffness determined from FEM. The method described in this thesis is not very accurate for determining the torsional stiffness. Therefore the interpolated torsional stiffnesses from FEM are used in the ASWING input file.

Elastic axis ($\tilde{\mathbf{r}}_{\mathbf{ea}}$)

The elastic axis is defined as the axis at which rotation will occur in case the wing is loaded in pure torsion. For a wing with uniform cross section over the entire span, the elastic axis is a straight line. The shear center of an airfoil is the point at which the resultant shear load must act to produce a wing deflection with no rotation. If the wing is an elastic structure then elastic axis coincides with the line joining the shear

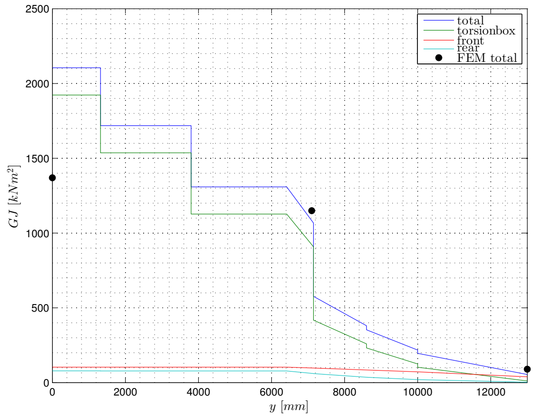


Figure 3.13: GJ per airfoil section over the wingspan

centres of the various cross-sections. The shear center is calculated as the position of the resultant shear force which yields a zero angle of twist: (Peery and Azar, 1982).

$$\sum_{i=1}^{N} \frac{q_i \Delta l_i L}{2 A t_i G} = 0 \tag{3.21}$$

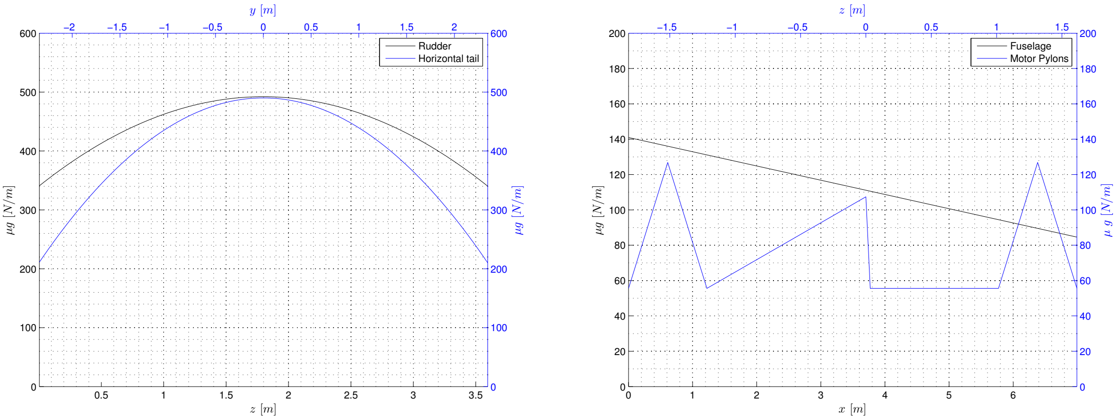
The shear flow is calculated as the product of shear stress and the thickness of the web. Open sections are able to resist shear forces applied at the shear center, but are unstable under torsional loads. Closed section box beams are able to resist these torsional loads. Peery and Azar (1982) describe analysis methods for single and multiple enclosed areas for thin airfoils with stiffeners. In this analysis the skin is assumed to transfer shear forces in between stiffeners, which take the axial loads. In this simplification the shear flow is written as a function of the shear flow in an arbitrary web, web 1.

$$q_N = q_1 + \sum_{i=1}^{N} \Delta F_i \tag{3.22}$$

In the analysis of the M600 airfoil, the top-, bottom and trailing edge cap contribute to both, the transfer of shear forces and additional take the loads. Close to the wing tip the skin is designed to take part of the loads and transfer the shear forces. Hence the shear flow analysis as described by Peery and Azar (1982) is not appropriate for elastic axis calculations. Therefore, the elastic axis determined from FEM is used in the ASWING input file. The horizontal stabilizer and rudder weight distribution are given in Figure 3.14a, the fuselage weight and motor pylon weight distribution in Figure 3.14b.

3.4 Engines

ASWING provides three different engine models, with increasing complexity; the simple proportional engine model, an actuator-disk model and an extended actuator-disk model with P-factor terms for propeller whirl prediction. The simple proportional model assumes that thrust and moment forces are directly proportional to the engine power parameter. Both actuator disk models determine the thrust and moment as a result of air density and the axial air velocity component. The side forces and moment as a results of the remaining velocity and rotation rate are referred to as the ‘P-factor’. All models take the downstream rotor effects into account. Propeller whirl prediction is outside the scope of this thesis, hence the actuator disk model is chosen as most appropriate. The parameters which need to be specified are listed in Table 3.4.



(a) Tail μg (b) Fuselage and motor pylon μg

Figure 3.14: Fuselage, tail and motor pylons μg

Table 3.4: ASWING engine inputs

ASWING keyword	Symbol	Description	Unit
`Tx, Ty, Tz`	$\vec{T}_{spec}$	engine axis vector	$[-]$
`Rdisk`	R_e	engine disk radius	$[m]$
`Omega`	Ω_e	rotational speed	$[\text{rad}/s]$
`cdA`	$(C_D A)_e$	total effective blade drag area	$[m^2]$
`Weight`	$m_e g$	weight of the motor	$[N]$
`Hxo, Hyo, Hzo`	$\vec{H}_{e,0}$	angular momentum in the undeformed state	$[N \cdot m \cdot s]$

The engine axis vector ($\vec{T}_{spec}$) in the undeformed state is assumed parallel to the x-axis of the aircraft. The total effective blade drag area is determined as:

$$(C_D A)_e = B R_e \bar{c} \bar{c}_d \tag{3.23}$$

In the configuration that the engine axis vector is aligned with the x-axis, the angular momentum about the y- and z-axis is equal to zero. The angular momentum about the x-axis is determined as the product of the area moment of inertia about the x-axis and its angular velocity about the x-axis:

$$H_{xx} = I_{xx} \omega_x \tag{3.24}$$

With these parameters, ASWING is able to calculate the engine power. For conventional aircraft the engines are designed to create thrust. However a negative power corresponds to an engine in windmill state. The minimum power corresponds to the Betz limit:

$$(P_i)_{min} = -\frac{8}{27} \rho V_r^3 A_r \tag{3.25}$$

P_i is the net inviscid power, the subscript $()_r$ corresponds to the rotor.

3.5 Conclusions

In this chapter the geometrical, aerodynamic and structural parameters of each M600 surface and fuselage beam element is calculated as well as the ASWING input parameters for the engines. The aerodynamic properties of the main wing, horizontal stabilizer, rudder and motor pylons are determined from the lift and moment coefficient curves and the drag polars. The structural properties of the main wing are determined with a simplified discretized model and compared with FEM results. The weight, extensional stiffness and the out-of-plane bending stiffness are within 15% range of FEM results. The torsional stiffness, the elastic axis, coupled bending and torsional stiffnesses and shear stiffnesses are not determined accurately with the simplified model and hence the FEM results will be used. The contribution of each airfoil segment on the stiffnesses is determined and these results can be used in a later design stage.

The input is automated with a MATLAB routine; the motor properties, the geometrical properties, the stiffnesses, elastic axis, the aerodynamic properties and the mass properties are automatically combined in this MATLAB script to set-up an ASWING input file. This allows the constant design changes to be immediately implemented in the latest aero-elastic analysis. The set-up of this MATLAB routine is given in Appendix A. The M600 input file (`M600.asw`) is given in Appendix B.1.

Chapter 4

Output ASWING

The goal of this chapter is (1) to present the ASWING output and (2) to verify this output. A subset of ASWING's capabilities have been verified against other calculation methods and flight data, such as: (1) for rigid structures the data is verified with lifting-line, vortex lattice, panel methods and classical flight dynamic analysis, (2) with no aerodynamic forces, the elasticity module is verified against NASTRAN, (3) high aspect ratio wing with simple elasticity flutter predictions are verified with Theodorsen theory and (4) the aileron and divergence speeds are verified with NASTRAN. However the results are as good as the user's understanding. In this chapter the linear and non-linear elastic response with no aerodynamic forces is respectively verified against analytical calculations and NASTRAN in section 4.1. The aerodynamic forces for the rigid structure are verified against the lifting line theory and XFLR5 in section 4.2. The flutter response is verified against Drela's test case and Jensen (2010) in section 4.3 and finally in section 4.4, the conclusions of this chapter are drawn.

4.1 Deflections

A 10 meter, solid, massless, squared beam is simulated. The beam is fixed at the root, no degrees of freedom in rotation and translation. The other end is free. The free end is loaded with respectively $F_{\text{tip}} = 1,000N$, $F_{\text{tip}} = 5,000N$ and $F_{\text{tip}} = 7,500N$. In case deflections are smaller than 10% with respect to beam length, linear deflections are assumed and analytical expressions are valid. In case deflections are larger than 10%, the linear relations are no longer valid and NASTRAN is used to verify the non-linear deflections determined from ASWING.

4.1.1 Linear deflections

The linear deflections of a cantilevered beam are determined as:

$$\delta_y = \frac{Fy^3}{3EI_{xx}} + \frac{My^2}{2EI_{xx}} \tag{4.1}$$

In this equation δ_y is the deflection at arbitrary y, L is the beam length, E is the modulus of elasticity and I_{xx} the area moment of inertia about the x-axis. For a solid rectangular block I_{xx} is calculated as:

$$I_{xx} = \frac{bh^3}{12} \tag{4.2}$$

For a cantilevered beam loaded at its tip, the moment at the tip is $M_{\text{tip}} = 0$ and is linearly increasing towards to root to $M_{\text{root}} = F_{\text{tip}}L$. The maximum deflection ($\delta_{y_{max}}$) is the deflection at the tip and is calculated as:

$$\delta_{y_{\max}} = \frac{F_{\text{tip}}L^3}{3EI_{xx}} \tag{4.3}$$

With $F_{\text{tip}} = 1000N$, $E = 70GPa$, $b = h = 0.10m$ and $L = 10m$, the tip deflection is equal to $\delta_{y_{max}} = 0.57m$. The tip deflection is 5.7% with respect to the length, hence the linear deflection assumption is valid. The beam deflection versus beam length is given in Figure 4.1. The ASWING tip deflection is 99.59% with respect to the tip deflection determined with the analytical method.

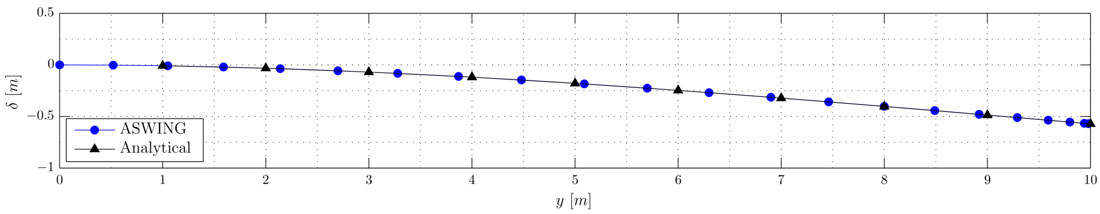


Figure 4.1: Deflected beam ASWING versus analytical method

An increase of tip load to $F_{\text{tip}} = 5,000N$, while other parameters kept constant, the tip deflection determined with the analytical expressions are equal to $\delta_{y_{max}} = 2.85m$. This is 28.5% with respect to the beam length. For this analysis, the assumption of linear deflection is no longer valid.

4.1.2 Non-linear deflections

The ASWING non-linear deflections are verified against NASTRAN. A beam, with equal specifications as the beam used for linear deflections, is loaded with $F_{\text{tip}} = 5,000N$ and $F_{\text{tip}} = 7,500N$. For the NASTRAN calculations the beam is split into 10 equally sized elements. ASWING has its own built-in routine to split the beam into element and there is no need to specify the number of elements. The beam deflections determined with ASWING and NASTRAN are shown in Figure 4.2. The tip deflection determined with ASWING is 99.59% with respect to the tip deflections determined with NASTRAN.

4.2 Lift forces

In ASWING the lift forces are determined with the theory from the lifting line theory. In this section, the lift forces are verified against the lifting line theory described by Van Garrel (2003) and XFLR5. Van Garrel (2003) and XFLR5 assume a perfectly rigid wing. This is modelled in ASWING as well.

4.2.1 Description of Van Garrel (2003) lifting line model

The lifting line theory is based on the theory that a flowfield around a wing can be described by sources and vortices as shown in Figure 4.3.

Van Garrel (2003): '*As the flow over an airfoil is started, the large velocity gradients at the sharp trailing edge result in the formation of a region of intense vorticity which rolls up downstream of the trailing edge,*

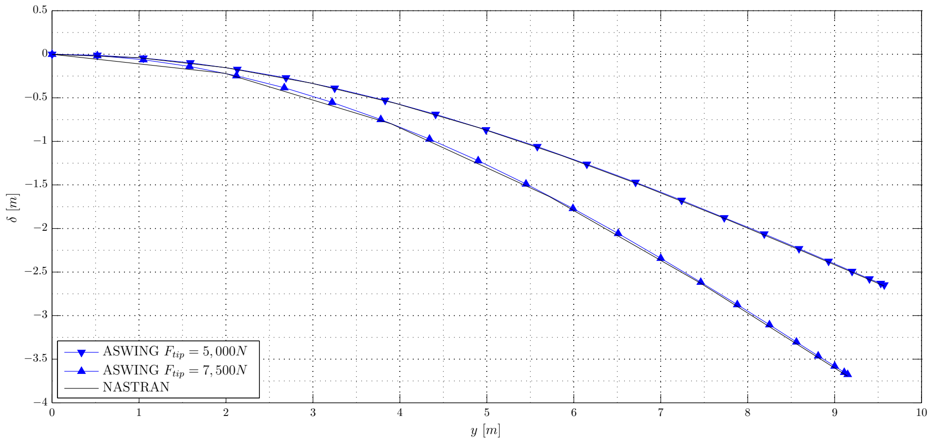


Figure 4.2: Deflected beam ASWING versus NASTRAN

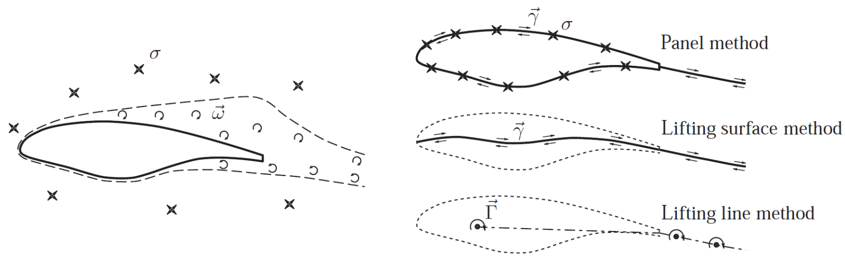


Figure 4.3: Flowfield representation (Van Garrel, 2003)

forming the starting vortex. This starting vortex has associated with it a counterclockwise circulation. Therefore, as an equal-and-opposite reaction, a clockwise circulation around the airfoil is generated. As the starting process continues, vorticity from the trailing edge is constantly fed into the starting vortex, making it stronger with a consequent larger counterclockwise circulation. In turn, the clockwise circulation around the airfoil become stronger, making the flow at the trailing edge more closely approach the Kutta condition, thus weakening the vorticity shed from the trailing edge. Finally, the starting vortex builds up to just the right strength such that the equal-and-opposite clockwise circulation around the airfoil leads to smooth flow from the trailing edge. When this happens, the vorticity shed from the trailing edge becomes zero, the starting vortex no longer grows in strength, and a steady circulation exist around the airfoil . '

For the panel method, all vorticity and source singularities are distributed on the configuration surface and in the wake. In the lifting surface method, no thickness effects are modelled and all surface vorticity is transferred to the mean line of the configuration. Lumping this mean line vorticity to a single point at quarter chord results in the lifting line method. The different methods are visualized in Figure 4.3.

The principle of the lifting line model originates from the conservation of circulation (Kelvin's theorem and Kutta condition) and the relation between circulation and lift per unit span (Kutta-Joukowski theorem):

$$L' = \rho_\infty V_\infty \Gamma_\infty \tag{4.4}$$

The initial lift and circulation are determined as:

$$L' = 2\pi\alpha_0 \frac{1}{2}\rho_\infty V_\infty^2 c \tag{4.5}$$

$$\Gamma = \frac{L'}{\rho_\infty V_\infty} \tag{4.6}$$

In these equations, L' is the lift per unit span, α_0 is the initial angle-of-attack, ρ_∞ is the undisturbed air density, V_∞, is the undisturbed flow velocity and Γ is the vortex strength.

The induction velocity at control point cp is determined as:

$$\vec{u_\Gamma}(\vec{x}_{cp}) = \frac{\Gamma}{2\pi} \frac{(r_1 + r_2)(\vec{r}_1 \times \vec{r}_2)}{r_1 r_2 (r_1 r_2 + \vec{r}_1 \cdot \vec{r}_2)} \tag{4.7}$$

In this equation $\vec{r_1}$, is the vector of the point of a vortex line element and r_1 is the magnitude of the vector. The induction velocity together with the wind velocity set the total velocity at the point of interest on the wing (x_{cp}). This velocity ($\vec{u_{cp}}$) defines the new angle-of-attack as:

$$\alpha_{cp} = \arctan\left(\frac{\vec{u}_{cp} \cdot \vec{a}_3}{\vec{u}_{cp} \cdot \vec{a}_1}\right) \tag{4.8}$$

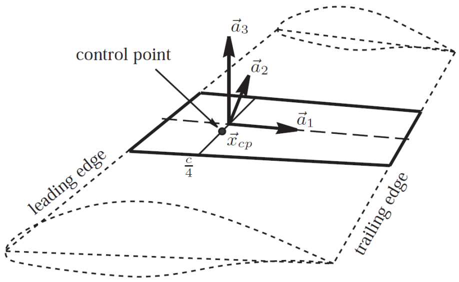


Figure 4.4: Wing strip geometry definitions (Van Garrel, 2003)

The wing strip geometry definitions are given in Figure 4.4. With the angle-of-attack the lift coefficient is determined from the lift curve and next the vortex strength is calculated as:

$$\Gamma_{cl} = C_l\left(\alpha_{cp}\right) \frac{\frac{1}{2}\left(\left(\vec{u}_{cp} \cdot \vec{a}_1\right)^2 + \left(\vec{u}_{cp} \cdot \vec{a}_3\right)^2\right) dA}{\sqrt{\left(\left(\vec{u}_{cp} \times d\vec{l}\right) \cdot \vec{a}_1\right)^2 + \left(\left(\vec{u}_{cp} \times d\vec{l}\right) \cdot \vec{a}_3\right)^2}} \tag{4.9}$$

At this point the difference between the old and the new rotation is calculated and an average is taken to increase the efficiency of the iteration process. These calculations are repeated in an iterative process until the maximum difference between two subsequent calculation circulations is smaller than $\epsilon = 10^{-6}$. At this point the result has converged. For a more detailed description, see Van Garrel (2003).

4.2.2 Lift forces against Van Garrel (2003)

A straight, rigid, single airfoil NACA0012 wing is simulated. This wing is characterized with a wingspan $b = 20.0m$ and chord $c = 0.50cm$. The flight speed is set at $V_{IAS} = 100m/s$ with $\alpha = 4°$. For this small angle-of-attack, a linear relation $C_l = 2\pi\alpha$ is used. The lift force distribution against Van Garrel (2003) is given in Figure 4.5. Both

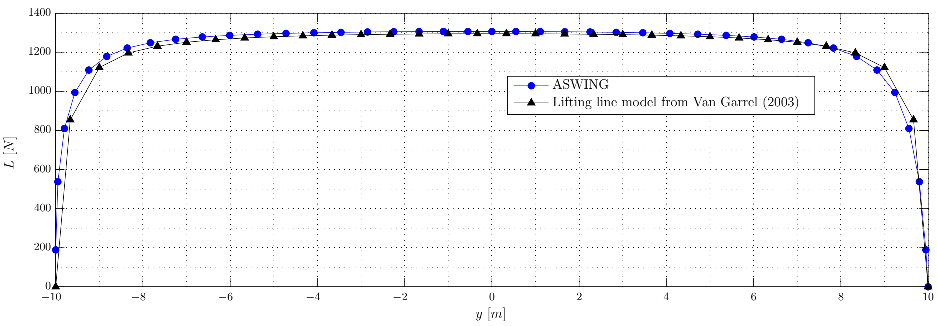


Figure 4.5: ASWING lift forces compared to Van Garrel (2003) lifting line method

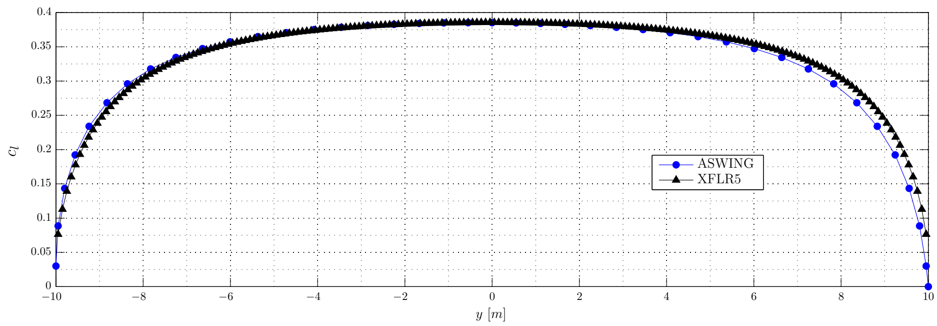


Figure 4.6: ASWING lift coefficient compared to XFLR5

4.2.3 ASWING lift forces compared to XFLR5

XFLR5 is an analysis tool for airfoils, wings and planes. Again a straight, rigid, single airfoil NACA0012 wing is analysed. This wing is characterized with a wingspan $b = 20.0m$ and chord $c = 2.00cm$. Again the flight speed is set at $V_{IAS} = 100m/s$ with $\alpha = 4°$.

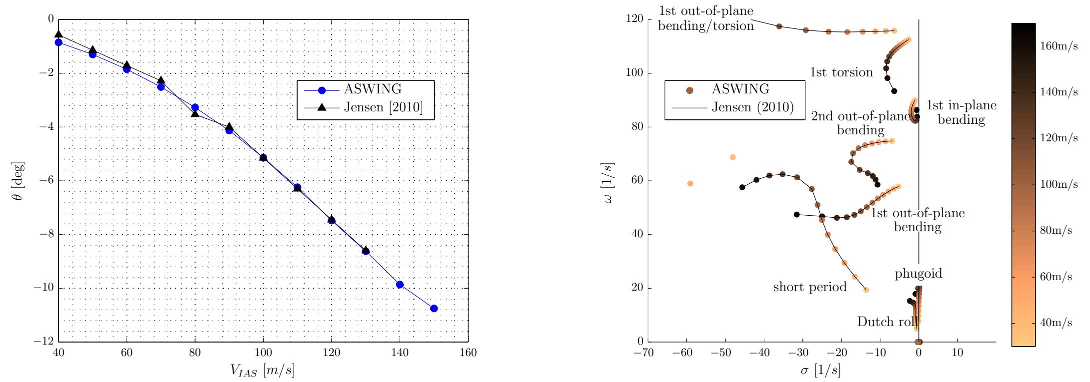
The lift coefficient distribution against XFLR5 is given in Figure 4.6. The ASWING lift coefficients at the root are equal to the lift coefficients determined with XFLR5. Closer to the wing-tips, a slight difference is shown between both analysis tools. This difference is only present at one wing-tip and therefore assumed to be caused by discretization errors.

4.3 Wing7 aero-elastic analysis

A subset of ASWING's capabilities have been presented and verified. The next step is the steady and and unsteady aero-elastic analysis. Jensen (2010) describes the results of the Wing 7 (W7) aero-elastic analysis. W7 is latest Makani Power prototype.

4.3.1 Wing7 divergence analysis

Jensen (2010) describes three different load cases to analyse W7 divergence: (1) tethered flight, (2) post-release, hight-G flight and (3) free flight. In the divergence analysis presented in this section, the free flight load cases are analysed. The wing is trimmed to steady, levelled flight for different flight speeds. For each flight speed, the maximum wing twist angle is plotted in Figure 4.7a. The Figure shows that the same results are determined with respect to Jensen (2010). Additionally the Figure shows that twist angles increase with flight speed. A sudden increase of twist angle with a small increment of flight speed indicates that the wing is approaching its divergence speed. This behaviour is not present in the examined flight speed regime and hence the same conclusion as Jensen (2010) is drawn: '*divergence does not appear to be an issue with W7*'.



(a) Verification divergence analysis against Jensen (2010)

(b) Verification flutter analysis against Jensen (2010)

Figure 4.7: Wing7 aero-elastic analysis

4.3.2 Wing7 flutter analysis

The ASWING Eigenmode analysis is performed with a root locus plot. Each mode is plotted against its frequency and growth rate. A negative growth rate (σ) indicates positive damping and hence a stable motion.

This ASWING version does not allow the input of a flexible tether-bridle system. Therefore Jensen (2010) analyses the limiting cases:

1. Infinitely soft tether, the tether is spring constant, $k_t = 0N/m$,
2. Infinitely stiff tether, the tether is spring constant, $k_t = \infty N/m$.

In this section, the results for an infinity stiff tether are analysed. The root locus plot is given in Figure 4.7b. This root locus plot is identical, and shows all modes which are also given in Jensen (2010). Dutch roll and phugoid are the modes close to the origin. These flight-modes also occur for a perfectly rigid aircraft. Further from the origin the short period, in-plane and out-of-plane bending modes, torsion modes and a bending/torsion mode are present. All modes are stable for the examined flight speeds, $40m/s \leq V_{IAS} \leq 160m/s$.

4.4 Conclusions

In this chapter a subset of ASWING's output is presented and verified; the linear and non-linear deflections are verified against respectively an analytical method and NASTRAN. ASWING's lift is verified against a lifting line method and XFLR5 and finally ASWING's steady and unsteady aero-elastic analysis is verified against Jensen (2010). All verification results are satisfactory.

ASWING's limitations for bridled flight appear in the verification case against Jensen (2010). This version of ASWING lacks the input of a flexible tether-bridle system and hence Jensen (2010) modelled the tether as perfectly rigid and infinitely soft. It is the goal of this thesis to solve this limitation of the current program and build an additional module that allows the input of a more realistic tether-bridle system.

Part II

ASWING Modifications

Chapter 5

Tether-bridle system

The goal of this chapter is to (1) determine the most appropriate method to define the tether-bridle system in ASWING and (2) to describe the tether-bridle system with its corresponding equations. In section 5.1 the approach to set-up the tether-bridle system is explained. In section 5.2 the tether-bridle model is explained. In section 5.3 the location of the tether-bridle attachment point is analytically derived. In section 5.4 the equations are set to calculate tether and bridle forces. In section 5.5 the constraints to set-up the equilibrium equations are explained and finally in section 5.6 the conclusions of this chapter are given.

5.1 Approach tether and bridle set-up

In the current ASWING version the most appropriate method to define the tether-bridle system is investigated. The use of existing ASWING routines to simulate the tether-bridle system is most time-efficient and investigated first. In case this approach is infeasible, the next step is examining the possibilities of modifying (part of) the routines. In case this is impossible, the final step is writing a new routine, which is compatible with ASWING.

5.1.1 Simulate the tether-bridle system with available routines

A possibly accurate tether-bridle system would contain a tether and multiple bridles like struts are defined in ASWING. These struts are assumed perfectly flexible in bending, but have a finite extensional stiffness, which allow the strut to change its length in response to extensional or compressive loads. Some issues are related with this strut; one strut end is attached at the aircraft, whereas the other is fixed in the aircraft reference frame. This definition does not allow (1) a moving tether-bridle attachment point and (2) multiple struts attached to each other. Hence the current strut definition does not allow the simulation of a tether-bridle system.

5.1.2 Simulate the tether-bridle system with adjusted routines

The strut definition could be adjusted. This adjusted model should allow a moving tether-bridle attachment point and multiple strut-strut connections. At the tether-bridle attachment location (explained in section

5.2) a minimum of three struts (two bridles and one tether) are connected. Drela (2009) discourages such a connection with:

> *if the beam has finite bending and extensional stiffness, it is permissible to have more than one #2 joints per beam, but this can give numerical difficulties and should be avoided if at all possible. In contrast, a beam can have an arbitrary number of #1 joint points with no ill effects, since these merely receive applied forces rather than kinematic constraints.*

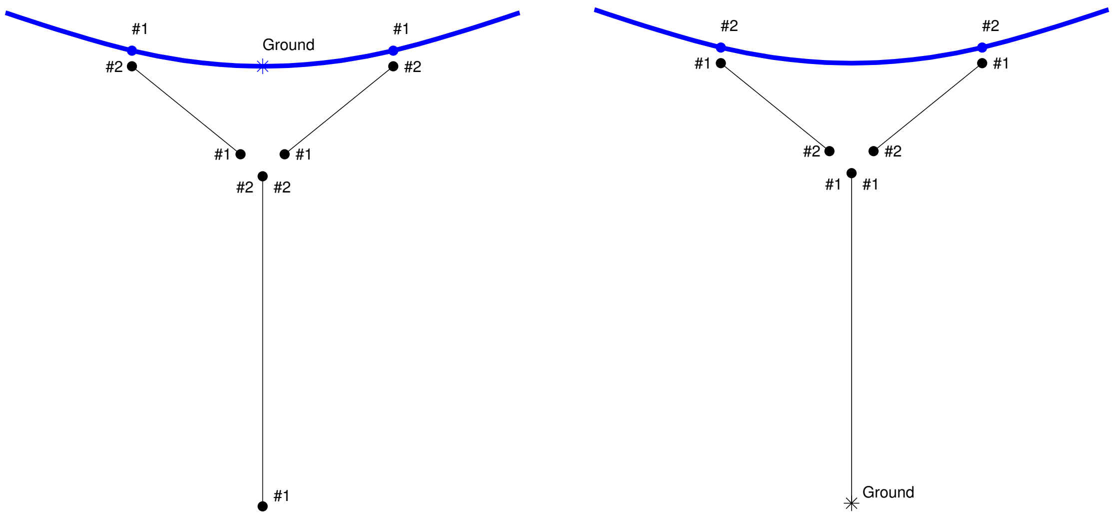
Each connection always contains one #*1* and one #*2* joint. Load resultants of a #1 joint are given by Drela (2009) as:

$$\Delta\vec{F}_{joint} = \vec{F}_J, \qquad \Delta\vec{M}_{joint} = \vec{M}_J, \tag{5.1}$$

A #2 joint implies kinematic constraints in lieu of moment equations:

$$\vec{r}_i = \vec{r}_{0_i} + \Delta\vec{r}_J, \qquad \vec{\theta}_i = \vec{\theta}_{0_i} + \Delta\vec{\theta}_J \tag{5.2}$$

Figures 5.1a and 5.1b show the #1 and #2 joint in case the wing is respectively grounded[1] with the aircraft and to the tether attachment at the ground-station. In case the aircraft is grounded, the tether has two #2 joints. In case the tether attachment to the ground-station is grounded, the aircraft wing has two #2 joints. Hence this approach, with tether and bridles simulated as adjusted struts, will not work properly.



(a) Joint connections with grounded wing

(b) Joint connections with grounded tether attachment to the ground-station

Figure 5.1: Joint connections tether-bridle system

5.1.3 Simulate tether-bridle system with a new routine

The available routines are unusable to simulate the tether-bridle system. Next three different options for the tether-bridle simulation are investigated, implemented and tested. The simulation of - (1) bridle forces with weights, (2) tether forces with a straight, massless, zero drag, axial stretchy tether, (3) the tether-bridle system with rigid bridles and a straight, axial stretchy tether.

[1] a ground joint is a joint to a fixed point in the xyz-axis. It serves to restrain the aircraft's rigid-body translation and rotation modes (Drela, 2009)

Simulation of bridle forces with weights

The forces at the bridles are simulated with the available point masses routine. In general the gravitational forces is aligned with Z_E. However this alignment does not hold for bridle forces. Therefore a new routine is written with a user defined gravity vector. The point mass simulation of bridle forces yields reasonable estimates for the bridled aircraft's static aero-elastic behaviour. For flutter analyses this approach is unusable, because (1) point masses are added to the mass inertia matrix and (2) the flexible dynamic tether behaviour is taken outside the analysis. The simulation of bridle forces with weights is concluded to be too inaccurate for the aero-elastic analyses for tethered flight. However, the routine, at which the user can modify the gravity direction, is used in the final program version to simulate tethered fight. This routine is explained in section 7.3.

Massless, zero drag, axial stretchy tether simulation

Next the tether is simulated with a spring system. One end of the tether is fixed in the Earth-reference frame and the other end is connected at one arbitrary aircraft location. In this model, no mass is added to the mass inertia matrix and the flexible dynamic tether behaviour is within the analysis. Hence the tether Jacobian need to be specified. With this representation the tether is attached to one point at the aircraft. In reality, the bridles are generally attached along the span, which tend to bend and twist the wing. This effect cannot be simulated with a massless, zero drag, flexible tether.

This system could give some valuable information about the flexible tether dynamic interaction with the wing. Therefore, the equations to calculate the tether force and its Jacobians are set-up. These equations are used in the final program version as well, and explained in respectively section 5.4 and 6.2.

Rigid bridles and a straight, axial stretchy tether

Finally the flexible tether is connected to the wing via two rigid bridles, which are free to rotate about all axis. This models couples the tether forces and Jacobian entries to the bridle forces and Jacobian entries. A detailed description is given in the next sections. This new tether-bridle routine is added in the original ASWING as shown in Figure 5.2. In the Eigenmode analysis, the new routine is added in a similar manner and given in the next chapter.

5.2 Description of tether-bridle model

As the point mass and the strut, the bridles are cantilevered from the beam's structural axis by a rigid pylon with geometric dimensions c_p, s_p, n_p. These pylon dimensions are determined via the transformation tensor.

$$\left\{c_{p,b}\ s_{p,b}\ n_{p,b}\right\}^T = \bar{\bar{T}}_0 \left\{\Delta\vec{r}_{p,b}\right\}_0 \tag{5.3}$$

The $()_0$ subscript denotes the undeformed state, which is known a priori, such that c_p, s_p, n_p are in effect fixed constants (Drela, 2009).

The location of the bridle attachment in airplane axes is given as:

$$\Delta\vec{r}_{p,b} = \bar{\bar{T}}^T \left\{c_{p,b}\ s_{p,b}\ n_{p,b}\right\}^T = \bar{\bar{T}}^T \bar{\bar{T}}_0 \Delta\vec{r}_{p0,b} \tag{5.4}$$

$$= \bar{\bar{T}}_{\text{net}} \Delta\vec{r}_{p0,b} \tag{5.5}$$

$$\vec{r}_{ap} = \vec{r}_{i,b} + \Delta\vec{r}_{p,b} \tag{5.6}$$

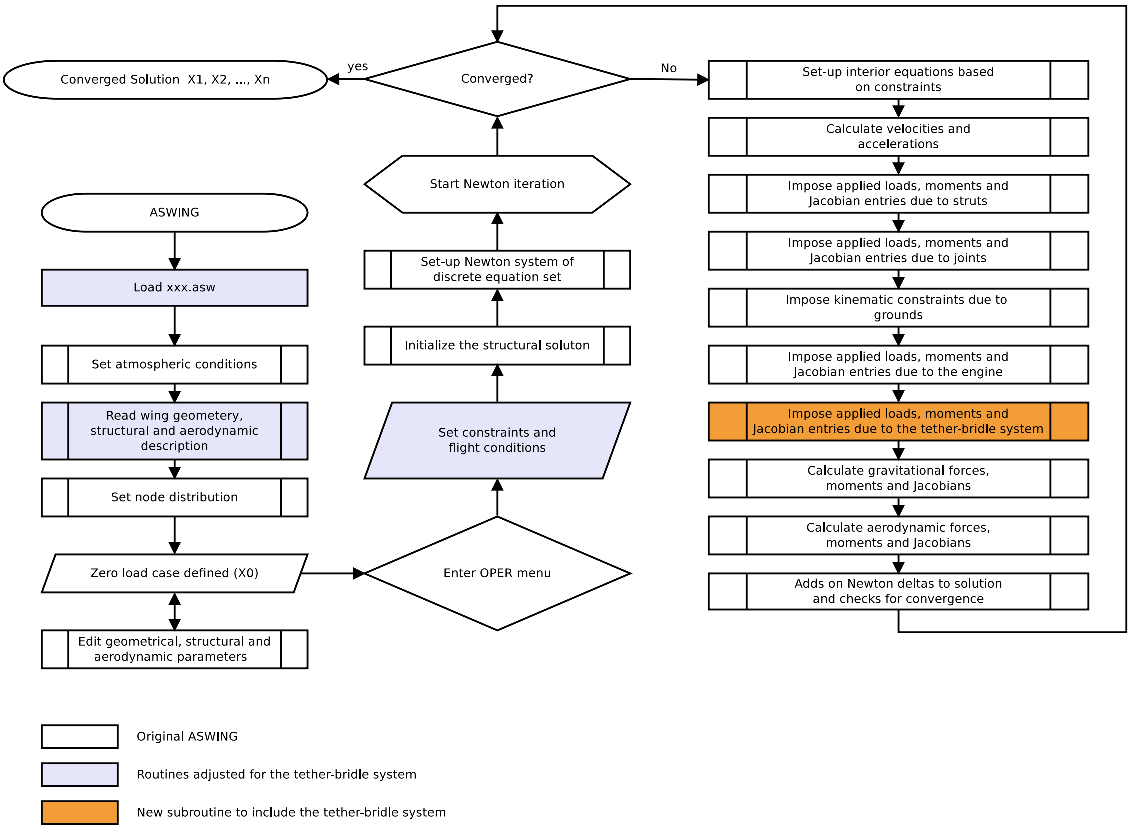


Figure 5.2: Flowchart ASWING including the tether-bridle routine

The set-up of the tether-bridle system is visualized in Figure 5.3. The offset between the aircraft reference frame and the Earth Reference frame is denoted as $\vec{R}$. The bridle 1 and bridle 2 attachment points to the wing are respectively called $\vec{r}_{i,b1}$ and $\vec{r}_{i,b1}$. The subscript $()_i$ denotes the node i at which the bridle is attached to the main wing. The location of the fixed attachment point to the ground-station is denoted as $\vec{R}_T$.

The tether force is dependent on the tether vector $\vec{r}_t$. With respect to the zero load length of the tether an extension of the tether results in a positive tether force. From Figure 5.3 follows that tether vector can de determined as

$$\vec{r}_t = \bar{\bar{T}}_E^T \vec{R}_T - \vec{r}_{tba} \tag{5.7}$$

With:

$$\vec{r}_{tba} = \vec{r}_{ap,b} + \vec{r}_b \tag{5.8}$$

With:

$$\vec{r}_{ap,b} = r_{i,b} + \left(\bar{\bar{T}}_{\text{net}} \Delta \vec{r}_{p0}\right)_b + \bar{\bar{T}}_E^T \vec{R} \tag{5.9}$$

Of these equation ($\vec{r}_{ap,b}$ is determined with current ASWING version. However the bridle vector $\vec{r}_b$ and the tether-bridle attachment vector $\vec{r}_{tba}$ need to be calculated.

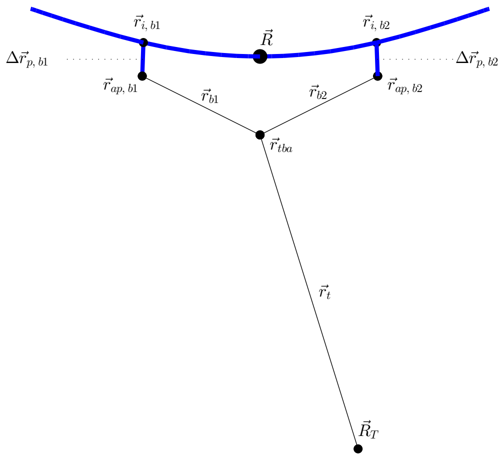


Figure 5.3: Tether-bridle system set-up

5.3 Definition of the tether-bridle attachment location $\vec{r}_{tba}$

To determine the tether-bridle attachment point, the bridles are assumed perfectly rigid. The bridle attachment points to the wing are known in priori and the bridles are free to rotate about all axis. Hence both bridle ends create a sphere as shown in Figure 5.4.

The intersection of both spheres is a circle. The tether-bridle attachment location is a specific point at this circle. The next sections will first determine the sphere-sphere intersection circle. Subsequently the point at this circle, which is the tether-bridle attachment point is analytically derived.

5.3.1 Sphere-sphere intersection circle created by two bridles

The intersection circle of both spheres represents all possible tether-bridle attachment points. Any arbitrary radii (bridle lengths) for both spheres can be chosen as long as both spheres intersect.

The circle midpoint and the radius are determined from geometrical relations. The distance between the bridle points ($d_{b1,\,b2}$) is calculated first. Figure 5.5 is used to clarify equations 5.10 - 5.12. The distance between the bridle attachment point ($\vec{r}_{ap,\,b}$ and the midpoint of the sphere-sphere intersection circle $\vec{r}_{ssic}$ is called d_b. The length of the bridles are denoted as L_b. The radius of the sphere-sphere intersection circle is denoted as R_{siic}.

$$R_{ssic}^2 + d_{b1}^2 = L_{b1}^2 \tag{5.10}$$

$$R_{ssic}^2 + d_{b2}^2 = L_{b2}^2 \tag{5.11}$$

$$\begin{aligned} d_{b1} + d_{b2} &= d_{b1,\,b2} \\ &= \sqrt{\left(\vec{r}_{ap,\,b1} - \vec{r}_{ap,\,b2}\right) \cdot \left(\vec{r}_{ap,\,b1} - \vec{r}_{ap,\,b2}\right)} \end{aligned} \tag{5.12}$$

This set of equations is solved to determine the radius of the circle (R_{ssic}),

$$R_{ssic} = \frac{\sqrt{(L_{b1} + L_{b2} + d_{b1,\,b2})\,(L_{b1} + L_{b2} - d_{b1,\,b2})\,(L_{b1} - L_{b2} + d_{b1,\,b2})\,(L_{b2} - L_{b1} + d_{b1,\,b2})}}{2d_{b1,\,b2}} \tag{5.13}$$

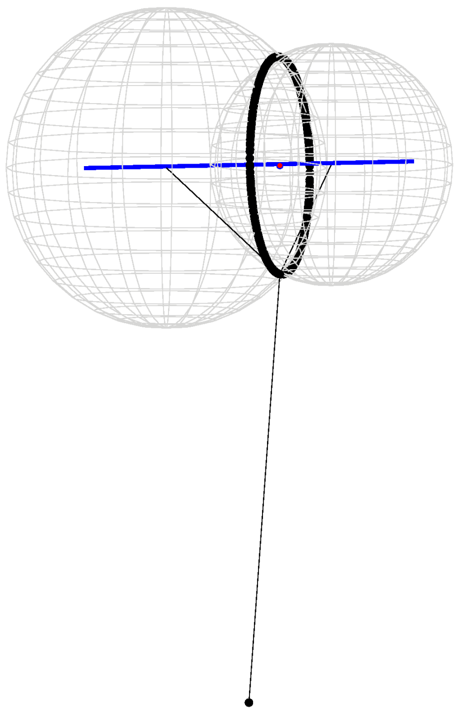


Figure 5.4: Sphere-sphere intersection circle created by two bridles

Substituting R_{ssic} in equation 5.10, the distance between $\vec{r}_{siic}$ and $\vec{r}_{b1}$ is determined and hence the location of the sphere-sphere intersection midpoint follows from the geometrical relation.

$$\vec{r}_{ssic} = \vec{r}_{ap,\,b1} + \frac{d_{b1}}{d_{b1,\,b2}}\left(\vec{r}_{ap,\,b1} - \vec{r}_{ap,\,b2}\right) \tag{5.14}$$

5.3.2 Exact location tether-bridle attachment point

The point at this circle that is the position of the tether-bridle attachment point ($\vec{r}_{tba}$) is based on the physical relation that the sum of forces is equal to zero at this node. In three dimensional space, three equilibrium equations exist with two unknowns. These are the equilibrium equations in x, y and z-direction and the unknown magnitude of both bridle forces:

$$\vec{F}_t + \frac{\vec{r}_{b1}}{l_{b1}}\,|F_{b1}| + \frac{\vec{r}_{b2}}{l_{b2}}\,|F_{b2}| = 0 \tag{5.15}$$

With:

$$\vec{r}_{b1} = \left(\vec{r}_{ap,\,b1} - \vec{r}_{tba}\right), \qquad \vec{r}_{b2} = \left(\vec{r}_{ap,\,b2} - \vec{r}_{tba}\right) \tag{5.16}$$

With two unknown (($|F_{b1}|$ and $|F_{b2}|$) and three equations, this is an overdetermined set of linear equations. Therefore another constraint is used: the tether and bridle forces need to be aligned. To solve this, the tether attachment point at the ground is projected in the plane of the sphere-sphere intersection circle. The following steps are taken:

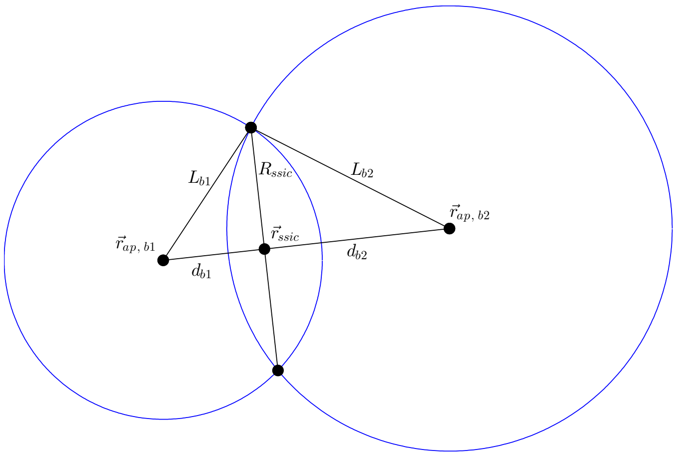


Figure 5.5: Geometrical parameters used in equations 5.10 - 5.12

1. Calculate the plane of the sphere-sphere intersection circle,
2. project the tether ground attachment point to this plane,
3. 'draw' a line from the midpoint of the sphere-sphere intersection circle to the projected tether ground attachment point point,
4. determine the point at which this line intersects with the sphere-sphere intersection circle.

Calculate the plane of the sphere-sphere intersection circle

The plane of the sphere-sphere intersection circle is determined from the sphere:

$$\left(x - x_{ap,\,b1}\right)^2 + \left(y - y_{ap,\,b1}\right)^2 + \left(z - z_{ap,\,b1}\right)^2 = L_{b1}^2 \tag{5.17}$$

$$\left(x - x_{ap,\,b2}\right)^2 + \left(y - y_{ap,\,b2}\right)^2 + \left(z - z_{ap,\,b2}\right)^2 = L_{b2}^2 \tag{5.18}$$

Subtract equation (5.18) from (5.17) to get the equation for the plane of the sphere-sphere intersection circle:

$$\begin{aligned} &-2x\left[x_{ap,\,b1} - x_{ap,\,b2}\right] + x_{ap,\,b1}^2 - x_{ap,\,b2}^2 \\ &-2y\left[y_{ap,\,b1} - y_{ap,\,b2}\right] + y_{ap,\,b1}^2 - y_{ap,\,b2}^2 \\ &-2z\left[z_{ap,\,b1} - z_{ap,\,b2}\right] + z_{ap,\,b1}^2 - z_{ap,\,b2}^2 = L_{b1}^2 - L_{b2}^2 \end{aligned} \tag{5.19}$$

Project the tether ground attachment point to this plane

For ease of notation, equation 5.19 is rewritten in terms of coefficients $\vec{C} = \{C_1\ C_2,\ C_3\}$ and D:

$$C_1 x + C_2 y + C_3 z + D = 0 \tag{5.20}$$

with:

$$\begin{aligned} C_1 &= -2\left[x_{ap,\,b1} - x_{ap,\,b2}\right] \\ C_2 &= -2\left[y_{ap,\,b1} - y_{ap,\,b2}\right] \\ C_3 &= -2\left[z_{ap,\,b1} - z_{ap,\,b2}\right] \\ D &= x^2_{ap,\,b1} - x^2_{ap,\,b2} + y^2_{ap,\,b1} - y^2_{ap,\,b2} + z^2_{ap,\,b1} - z^2_{ap,\,b2} - L^2_{b1} + L^2_{b2} \end{aligned} \tag{5.21}$$

The tether ground point is projected in this plane as:

$$\left(\vec{R}_t\right)^* = \vec{R}_t - \vec{C}\frac{\vec{C}\cdot\vec{R} + D}{\vec{C}\cdot\vec{C}} \tag{5.22}$$

In these equations the projection is denoted with superscript $()^*$. The location of the bridle attachment point in the aircraft reference axis $\vec{R}_t$ is determined from the location of the bridle attachment point in the Earth reference axis $\vec{R}_T$ as:

$$\vec{R}_t = \bar{\bar{T}}^T_E \vec{R}_T \tag{5.23}$$

'Draw' a line from the midpoint of the sphere-sphere intersection circle to the projected tether ground attachment point

The projected tether ground attachment point is connected to the midpoint of the sphere-sphere intersection circle by the projected tether vector r_t^*. This vector is determined as:

$$\left(\vec{r}_t\right)^* = \left(\vec{R}_t\right)^* - \vec{r}_{ssic} \tag{5.24}$$

Determine the point at which this line intersects with the sphere-sphere intersection circle

The position of the tether-bridle attachment point now determined as:

$$\vec{r}_{tba} = \vec{r}_{ssic} + \frac{R_{ssic}}{\left|\left(\vec{r}_t\right)^*\right|}\left(\left(\vec{R}_t\right)^* - \vec{r}_{ssic}\right) \tag{5.25}$$

With

$$\left|\left(\vec{r}_t\right)^*\right| = \sqrt{\left(\vec{r}_t\right)^* \cdot \left(\vec{r}_t\right)^*} \tag{5.26}$$

Figure 5.6 gives an overview of the steps taken to determine the tether-bridle attachment point.

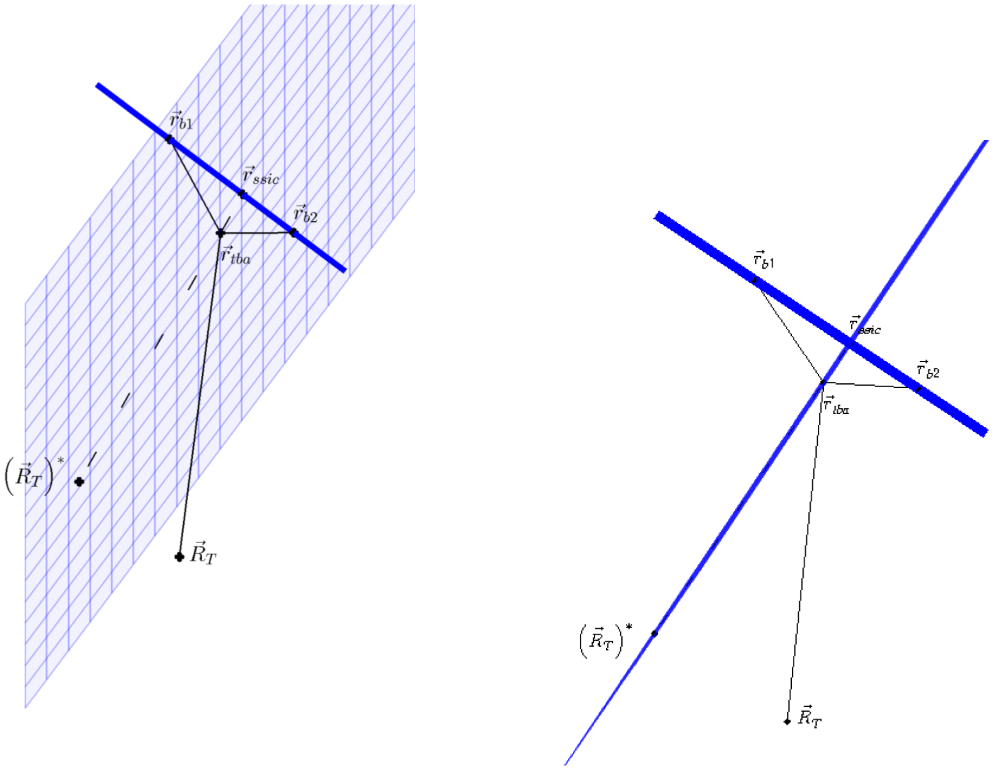


Figure 5.6: Visualization of the tether-bridle point determination

5.4 Tether and bridle Forces

The tether is simplified as a spring with spring constant k_t. Hookes law states that the reaction force is linearly proportional to the length of the spring minus the unstressed length. All input variables for the tether vector ($\vec{r}_t$), equation 5.8, are known and the stressed tether length is calculated with the inner product of the tether vector.

$$l_t = \sqrt{\vec{r}_t \cdot \vec{r}_t} \tag{5.27}$$

According to Hooke's law the magnitude of the tether force is equal to

$$|F_t| = k_t \left(l_t - l_{t0}\right) \tag{5.28}$$

In this equation $|F_t|$ is the magnitude of the tether force, k_t is the spring constant, l_t is the stressed tether length and l_{t0} in the unstressed tether length. The tether cannot withstand any bending, hence the tether force vector $\left(\vec{F}_t\right)$ is aligned with the tether direction vector. The tether force vector is calculated as:

$$\vec{F}_t = \frac{\vec{r}_t}{l_t} |F_t| = \hat{\vec{r}}_t |F_t| \tag{5.29}$$

With the tether force vector and force equilibrium equations at the tether-bridle attachment point, the bridle forces follow from any two of the three equilibrium equations (equation 5.15). Any two equilibrium equations calculate equal bridle force. However in case $x_{b1} = x_{b2} = x_{tba} = 0$ the solution is singular. This holds for y- and z-direction as well. This occurrence is less likely, because, in general, the bridles are attached along the wing span and the tether-bridle attachment point is generally located with an offset in z-direction with respect to the bridle attachment points at the wing. However at large Euler angles the same singularity in y- or z-direction can occur. In a bridle design at which the two bridles are not aligned, a simultaneous singularity in two direction cannot occur. To exclude the possibility of singularities the rules of equations 5.30 and 5.31 are applied.

$$|F_{b2}| = \begin{cases} \dfrac{F_{t_z} + \hat{\vec{z}}_{b1} \frac{F_{t_z} + F_{t_y}}{\hat{\vec{z}}_{b1} - \hat{\vec{y}}_{b1}}}{\hat{\vec{z}}_{b1} \left(\frac{\hat{\vec{z}}_{b2} - \hat{\vec{y}}_{b2}}{\hat{\vec{z}}_{b1} - \hat{\vec{y}}_{b1}} \right) - \hat{\vec{z}}_{b2}} & x_{b1} = x_{b2} = x_{tba} = 0 \\[2ex] \dfrac{F_{t_x} + \hat{\vec{x}}_{b1} \frac{F_{t_x} + F_{t_z}}{\hat{\vec{x}}_{b1} - \hat{\vec{z}}_{b1}}}{\hat{\vec{x}}_{b1} \left(\frac{\hat{\vec{x}}_{b2} - \hat{\vec{z}}_{b2}}{\hat{\vec{x}}_{b1} - \hat{\vec{z}}_{b1}} \right) - \hat{\vec{x}}_{b2}} & y_{b1} = y_{b2} = y_{tba} = 0 \\[2ex] \dfrac{F_{t_x} + \hat{\vec{x}}_{b1} \frac{F_{t_x} + F_{t_y}}{\hat{\vec{x}}_{b1} - \hat{\vec{y}}_{b1}}}{\hat{\vec{x}}_{b1} \left(\frac{\hat{\vec{x}}_{b2} - \hat{\vec{y}}_{b2}}{\hat{\vec{x}}_{b1} - \hat{\vec{y}}_{b1}} \right) - \hat{\vec{x}}_{b2}} & z_{b1} = z_{b2} = z_{tba} = 0 \end{cases} \tag{5.30}$$

$$|F_{b1}| = \begin{cases} -\dfrac{F_{t_z} + \hat{\vec{z}}_{b2} |F_{b2}|}{\hat{\vec{z}}_{b1}} & x_{b1} = x_{b2} = x_{tba} = 0 \\[2ex] -\dfrac{F_{t_x} + \hat{\vec{x}}_{b2} |F_{b2}|}{\hat{\vec{x}}_{b1}} & y_{b1} = y_{b2} = y_{tba} = 0 \\[2ex] -\dfrac{F_{t_y} + \hat{\vec{y}}_{b2} |F_{b2}|}{\hat{\vec{y}}_{b1}} & z_{b1} = z_{b2} = z_{tba} = 0 \end{cases} \tag{5.31}$$

Finally the bridle forces are calculated as:

$$\vec{F}_b = \frac{\vec{r}_{ap,b} - \vec{r}_{tba}}{l_b} |F_b| = \hat{\vec{r}}_b |F_b| \tag{5.32}$$

5.5 Tether force constraint

In the low speed flight regime, the wing is flying at its design lift coefficient $\left(C_{L_{design}}\right)$. The aerodynamic loads are balanced with the tether force. At high wind speeds these forces can increases the tether force to a value higher than the maximum tether design load. In this flight regime the lift coefficient should be decreased. For example, by decreasing the aircraft angle-of-attack or decreasing the flight speed. An extra module is created in the ASWING constraints GUI to allow an user defined maximum tether forcer. All ASWING variables are available to realize a stable flight with this maximum tether force. These variables include, but are not limited to: flight speed (V_{IAS}), angle-of-attack $\left(\alpha_{ref}\right)$, linear accelerations ($\vec{a}_0$) and angular accelerations ($\vec{\alpha}_0$).

5.6 Conclusions

In this chapter several methods were analysed to simulate the tether-bridle system. The most appropriate feasible option is to assume perfectly rigid bridles which are free to rotate about all axes. The tether is simulated as a massless spring with zero drag area. The tether and bridle forces are analytically derived and these equation are implemented in a new ASWING subroutine. The analytical expressions in this chapter are used in the next chapter to determine the corresponding Jacobian entries. The tether-bridle additions are programmed in FORTRAN77. This ASWING compatible subroutines are calculated in subroutine `SETTET` (set tether) and subroutine `SETBRI` (set bridle). The code can be found in Appendix C. The flowchart of its main functionalities is given in Figure 5.7.

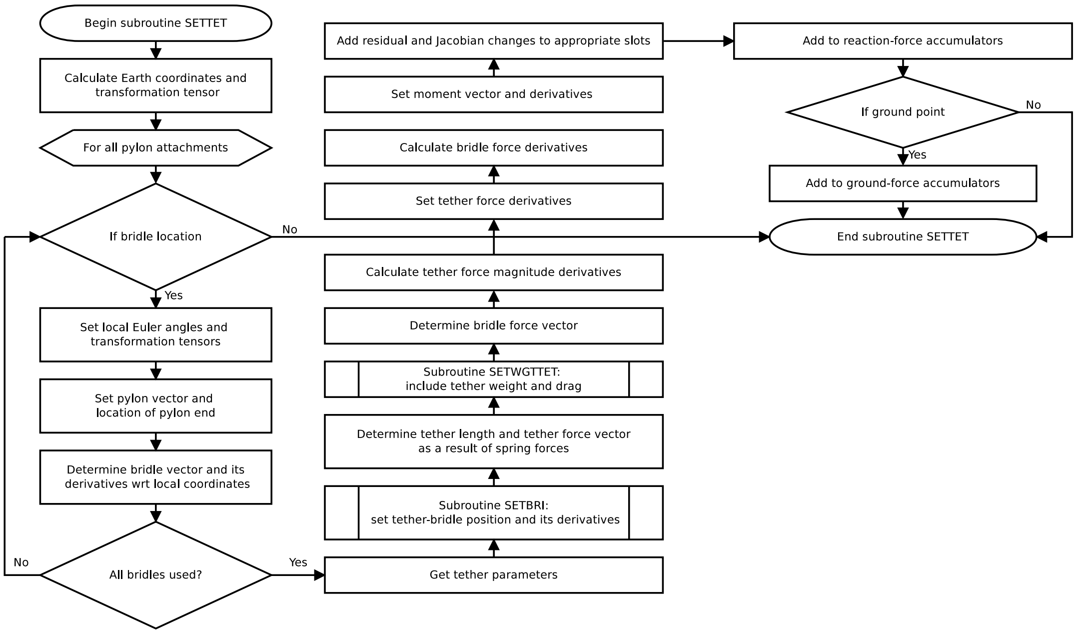


Figure 5.7: Flowchart ASWING tether-bridle routine

In this chapter the tether force as a result of spring forces are calculated. In chapter 7, the tether aerodynamic and gravitational forces are implemented in the system of equations.

Chapter 6

Bridle force Jacobian entries

It is the goal of this chapter to determine the ASWING compatible Jacobian entries for the tether-bridle system. Section 2.2 described that flutter modes are determined from Jacobian matrices $\frac{\partial \mathbf{r}}{\partial \mathbf{x}}$ and $\frac{\partial \mathbf{r}}{\partial \dot{\mathbf{x}}}$. ASWING is written in the character-based program language FORTRAN77 at which the initial state variables and the Jacobian entries need to be specified separately. The initial state variables were discussed in chapter 5. This chapter will determine the Jacobian entries analytically. The bridle force Jacobian entries tend to become comprehensive and hence the calculation of Jacobian entries in split into several parts. In section 6.1 the bridle force derivatives are written as a function of the tether force and its derivatives. In section 6.2 these tether force derivatives are determined. Finally in section 6.3 the conclusions of this chapter are drawn.

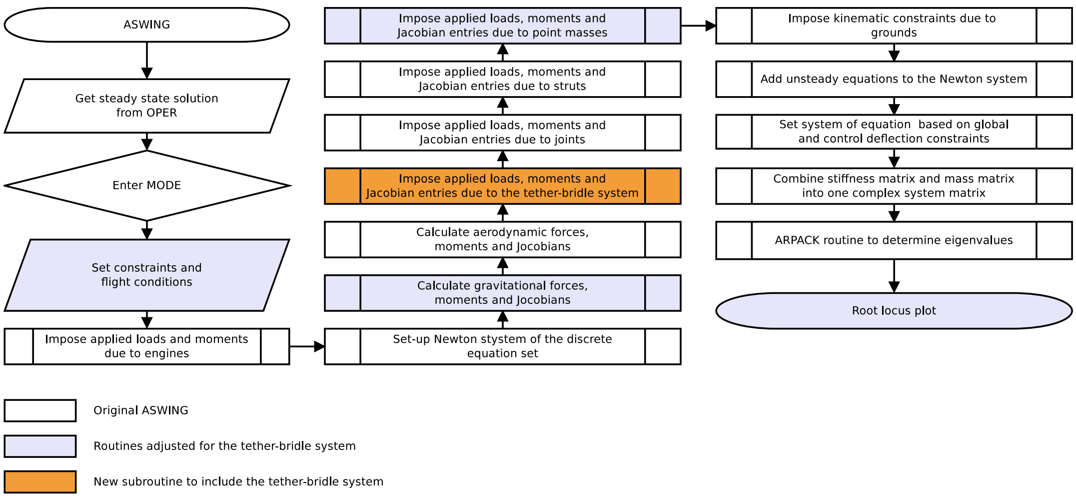


Figure 6.1: Flowchart Eigenmode analysis including the tether-bridle routine

6.1 Bridle force derivatives as a function of the tether

From equation 5.32, bridle force partial derivatives with respect to an arbitrary variable V_i are determined as:

$$\frac{\partial \vec{F}_b}{\partial V_i} = \frac{\partial \hat{\vec{r}}_b}{\partial V_i} |F_b| + \hat{\vec{r}}_b \frac{\partial |F_b|}{\partial V_i} \tag{6.1}$$

The bridle force magnitude partial derivatives $\left(\frac{\partial |F_b|}{\partial V_i}\right)$ are determined next in section 6.1.1 and 6.1.2. The partial derivatives of the normalized bridle vector is expanded as:

$$\frac{\partial \hat{\vec{r}}_b}{\partial V_i} = \frac{\frac{\partial \vec{r}_{ap,b}}{\partial V_i} - \frac{\partial \vec{r}_{tba}}{\partial V_i}}{L_b} \tag{6.2}$$

These partial derivatives, $\left(\frac{\partial \vec{r}_{ap,b}}{\partial V_i}\right)$ and $\left(\frac{\partial \vec{r}_{tba}}{\partial V_i}\right)$, determined in section 6.2.1.

6.1.1 Bridle 2 derivatives as a function of the tether

The derivatives of the magnitude of bridle force 2, $|F_{b2}|$, with respect to an arbitrary variable V_i is determined with the quotient rule:

$$|F_{b2}| = \frac{|F_{b2}|^{TP}}{|F_{b2}|^{BT}} \tag{6.3}$$

$$\frac{\partial |F_{b2}|}{\partial V_i} = \frac{\frac{\partial |F_{b2}|^{TP}}{\partial V_i} |F_{b2}|^{BT} - |F_{b2}|^{TP} \frac{\partial |F_{b2}|^{BT}}{\partial V_i}}{\left(|F_{b2}|^{BT}\right)^2} \tag{6.4}$$

In this equation, the superscript $()^{TP}$ denotes the top part of the equation, and superscript $()^{BT}$ denotes the bottom part of the equation. Substitute equation 5.30 into equation 6.4. The bridle 2 force magnitude partial derivatives as a function of tether force and its derivatives are calculated as:

$$\frac{\partial |F_{b2}|}{\partial V_i} = \frac{\partial F_{t,z}}{\partial V_i} - \frac{\partial \left(\hat{z}_{b1} \frac{F_{t,z} - F_{t,y}}{(\hat{z}_{b1} - \hat{y}_{b1})}\right)}{\partial V_i} \tag{6.5}$$

$$= \frac{\partial F_{t,z}}{\partial V_i} - \frac{\left[\frac{\partial \hat{z}_{b1}}{\partial V_i}\left(F_{t,z} - F_{t,y}\right) + \hat{z}_{b1}\left(\frac{\partial F_{t,z}}{\partial V_i} - \frac{\partial F_{t,y}}{\partial V_i}\right)\right](\hat{z}_{b1} - \hat{y}_{b1}) - \left(\hat{z}_{b1}\left(F_{t,z} - F_{t,y}\right)\right)\left[\frac{\partial \hat{z}_{b1}}{\partial V_i} - \frac{\partial \hat{y}_{b1}}{\partial V_i}\right]}{(\hat{z}_{b1} - \hat{y}_{b1})^2} \tag{6.6}$$

6.1.2 Bridle 1 derivatives as a function of the tether

Substitute equation 5.31 into equation 6.4. The bridle 1 partial derivatives as a function of tether force and its derivatives are determined as:

$$\frac{\partial |F_{b1}|}{\partial V_i} = \frac{\partial \left(\frac{-F_{t,y} - \hat{y}_{b2}|F_{b2}|}{\hat{y}_{b1}}\right)}{\partial V_i} \tag{6.7}$$

$$= \frac{\left[-\frac{\partial F_{t,y}}{\partial V_i} - \frac{\partial \hat{y}_{b2}}{\partial V_i}|F_{b2}| - \hat{y}_{b2}\frac{\partial |F_{b2}|}{\partial V_i}\right]\hat{y}_{b1} - \left[-F_{t,y} - \hat{y}_{b2}|F_{b2}|\right]\frac{\partial \hat{y}_{b1}}{\partial V_i}}{(\hat{y}_{b1})^2} \tag{6.8}$$

The bridle force magnitudes partial derivatives are determined as a function of the tether force and its derivatives. The tether force is calculated in section 5.4 and in the next section the tether force partial derivatives are determined.

6.2 Tether force partial derivatives

The partial derivatives of the tether force vector are determined from equation 5.29 as:

$$\frac{\partial \vec{F}_t}{\partial V_i} = \frac{\partial \vec{F}_t}{\partial \vec{r}_t}\frac{\partial \vec{r}_t}{\partial V_i} + \frac{\partial \vec{F}_t}{\partial l_t}\frac{\partial l_t}{\partial V_i} + \frac{\partial \vec{F}_t}{\partial |F_t|}\frac{\partial |F_t|}{\partial V_i} \tag{6.9}$$

In this equation $\frac{\partial \vec{F}_t}{\partial \vec{r}_t}$, $\frac{\partial \vec{F}_t}{\partial l_t}$ and $\frac{\partial \vec{F}_t}{\partial |F_t|}$ are termed the 'tether force general partial derivatives', because these derivatives are general for each derivative with respect to the tether force. These tether force general partial derivatives are equal to:

$$\frac{\partial \vec{F}_t}{\partial \vec{r}_t} = \frac{|F_t|}{l_t}, \qquad \frac{\partial \vec{F}_t}{\partial l_t} = -\frac{\vec{r}_t}{l_t^2}|F_t|, \qquad \frac{\partial \vec{F}_t}{\partial |F_t|} = \frac{\vec{r}_t}{l_t} \tag{6.10}$$

Next the specific partial derivatives, the derivatives with respect to a specific variable V_i, are determined $\left(\frac{\partial \vec{r}}{\partial V_i}\right)$, $\left(\frac{\partial l_t}{\partial V_i}\right)$ and $\left(\frac{\partial |F_t|}{\partial V_i}\right)$.

6.2.1 Tether position vector

The partial derivative of the tether vector with respect to an arbitrary variable is equal to the sum of the partial derivative of the tether attachment to the ground-station and the tether-bridle attachment location:

$$\frac{\partial \vec{r}_t}{\partial V_i} = \frac{\partial\left(\bar{\bar{T}}_E^T \vec{R}_T\right)}{\partial V_i} - \frac{\partial \vec{r}_{tba}}{\partial V_i} \tag{6.11}$$

This partial derivative equation contains two parts, (1) the tether attachment point to the ground-station partial derivative $\left(\frac{\partial\left(\bar{\bar{T}}_E^T \vec{R}_T\right)}{\partial V_i}\right)$ and (2) the tether-bridle attachment point partial derivative $\left(\frac{\partial \vec{r}_{tba}}{\partial V_i}\right)$. Both partial derivatives are determined next.

Tether attachment point to the ground-station

The position of the ground attachment point is fixed in the Inertial reference frame. Hence the partial derivative of $\vec{R}_T$ with respect to any arbitrary variable is zero, $\frac{\partial \vec{R}_T}{\partial V_i} = 0$. The tether force partial derivatives with respect to the tether attachment point to the ground-station is reduced to:

$$\frac{\partial\left(\bar{\bar{T}}_E^T \vec{R}_T\right)}{\partial V_i} = \frac{\partial \bar{\bar{T}}_E^T}{\partial V_i}\vec{R}_T + \frac{\partial \vec{R}_T}{\partial V_i}\bar{\bar{T}}_E^T = \frac{\partial \bar{\bar{T}}_E^T}{\partial V_i}\vec{R}_T \tag{6.12}$$

The transformation tensor $\bar{\bar{T}}_E^T$ is solely dependent on the Euler angles $\vec{\Theta} = \{\Phi\ \Theta\ \Psi\}^T$, hence:

$$\frac{\partial \bar{\bar{T}}_E^T}{\partial V_i} = \frac{\partial \bar{\bar{T}}_E^T}{\partial \vec{\Theta}} \tag{6.13}$$

As a final partial derivative for this part of the equation, the transformation tensor partial derivatives with respect to $\vec{\Theta} = \{\Phi\ \Theta\ \Psi\}^T$ are derived from equation 2.2 as:

$$\frac{\mathrm{d}\bar{\bar{T}}_E^T}{\mathrm{d}\Phi} = \begin{bmatrix} 0 & 0 & 0 \\ -\cos\Phi\sin\Theta\cos\Psi - \sin\Phi\sin\Psi & \sin\Phi\sin\Theta\cos\Psi - \cos\Phi\cos\Psi & -\cos\Phi\cos\Theta \\ -\sin\Phi\sin\Theta\cos\Psi + \cos\Phi\sin\Psi & \sin\Phi\sin\Theta\sin\Psi + \cos\Phi\cos\Psi & -\sin\Phi\cos\Theta \end{bmatrix} \tag{6.14}$$

$$\frac{\mathrm{d}\bar{\bar{T}}_E^T}{\mathrm{d}\Theta} = \begin{bmatrix} -\sin\Theta\cos\Psi & \sin\Theta\sin\Psi & -\cos\Theta \\ -\sin\Phi\cos\Theta\cos\Psi & \sin\Phi\cos\Theta\sin\Psi & \sin\Phi\sin\Theta \\ \cos\Phi\cos\Theta\cos\Psi & -\cos\Phi\cos\Theta\sin\Psi & -\cos\Phi\sin\Theta \end{bmatrix} \tag{6.15}$$

$$\frac{\mathrm{d}\bar{\bar{T}}_E^T}{\mathrm{d}\Psi} = \begin{bmatrix} -\cos\Theta\sin\Psi & -\cos\Theta\cos\Psi & 0 \\ \sin\Phi\sin\Theta\sin\Psi + \cos\Phi\cos\Psi & \sin\Phi\sin\Theta\cos\Psi - \cos\Phi\sin\Psi & 0 \\ -\cos\Phi\sin\Theta\sin\Psi + \sin\Phi\cos\Psi & -\cos\Phi\sin\Theta\cos\Psi - \sin\Phi\sin\Psi & 0 \end{bmatrix} \tag{6.16}$$

Tether-bridle attachment point

The tether-bridle attachment point partial derivatives follow from equation 5.25 as:

$$\frac{\partial \vec{r}_{tba}}{\partial V_i} = \frac{\partial \vec{r}_{ssic}}{\partial V_i} + \frac{\partial\left[\frac{R_{ssic}}{|(\vec{r}_t)^*|}\left(\left(\vec{R}_t\right)^* - \vec{r}_{ssic}\right)\right]}{\partial V_i} \tag{6.17}$$

This partial derivative equation contains two distinct parts, the first being the sphere-sphere intersection circle partial derivatives $\left(\frac{\partial \vec{r}_{ssic}}{\partial V_i}\right)$, which is determined next.

First part derivative equations $\frac{\partial \vec{\mathbf{r}}_{\mathbf{tba}}}{\partial \mathbf{V_i}}$

The sphere-sphere intersection circle partial derivatives $\left(\frac{\partial \vec{r}_{ssic}}{\partial V_i}\right)$ follow from equation 5.14 as:

$$\frac{\partial \vec{r}_{ssic}}{\partial V_i} = \frac{\partial \vec{r}_{ap,\,b1}}{\partial V_i} + \frac{\partial\left[\frac{d_{b1}}{d_{b1,\,b2}}\left(\vec{r}_{ap,\,b1} - \vec{r}_{ap,\,b2}\right)\right]}{\partial V_i} \tag{6.18}$$

Again this equation contains two distinct parts. The derivative of the attachment point of bridle 1 follows from equation 5.9 as:

$$\frac{\partial \vec{r}_{ap,\,b1}}{\partial V_i} = \frac{\partial r_{i,\,b1}}{\partial V_i} + \frac{\partial\left(\bar{\bar{T}}_{\text{net}}\Delta\vec{r}_{p0}\right)_{b1}}{\partial V_i} + \frac{\partial\left(\bar{\bar{T}}_E^T\vec{R}\right)}{\partial V_i} \tag{6.19}$$

In this equation the pylon-offset $\left(\Delta\vec{r}_{p0}\right)$ is constant; independent of any variable. Equation 6.19 is written as:

$$\frac{\partial \vec{r}_{ap,\,b1}}{\partial V_i} = \frac{\partial r_{i,\,b1}}{\partial V_i} + \frac{\partial\left(\bar{\bar{T}}_{\text{net}}\right)_{b1}}{\partial V_i}\Delta\vec{r}_{p0_{b1}} + \frac{\partial\bar{\bar{T}}_E^T}{\partial V_i}\vec{R} + \frac{\partial\vec{R}}{\partial V_i}\bar{\bar{T}}_E^T \tag{6.20}$$

Note that $\vec{r}_i$, $\bar{\bar{T}}_{\text{net}}$ and $\Delta\vec{r}_{p0}$ are denoted with subscript $()_{b1}$, because these are local beam coordinates and hence dependent on the state of the beam. $\vec{R}$ and $\bar{\bar{T}}_E^T$ are variables in the Inertial reference frame and independent on the local beam coordinate system.

The partial derivative of the bridle attachment location of the wing is only dependent on r_i. The derivative $\frac{\partial r_{i,b1}}{\partial V_i} = \frac{\partial r_{i,b1}}{\partial \vec{r}_i}$ is given in matrix form as:

$$\frac{\partial r_{i,b1}}{\partial V_i} = \frac{\partial r_{i,b1}}{\partial r_i} = \begin{bmatrix} \frac{\partial x}{\partial x} & \frac{\partial y}{\partial x} & \frac{\partial z}{\partial x} \\ \frac{\partial x}{\partial y} & \frac{\partial y}{\partial y} & \frac{\partial z}{\partial y} \\ \frac{\partial x}{\partial z} & \frac{\partial y}{\partial z} & \frac{\partial z}{\partial z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.21}$$

The partial derivatives of the local net transformation tensor $\frac{\partial \left(\bar{\bar{T}}_{\text{net}}\right)_{b1}}{\partial V_i}$ is a function of the local Euler beam transformation tensor and the Euler transformation tensor for the undeformed state, which is constant. Hence the partial derivatives of the local net transformation tensor $\frac{\partial \left(\bar{\bar{T}}_{\text{net}}\right)_{b1}}{\partial V_i}$ follow from equations 5.4 and 5.5 as:

$$\frac{\partial \left(\bar{\bar{T}}_{\text{net}}\right)_{b1}}{\partial V_i} = \frac{\partial \left(\bar{\bar{T}}^T \bar{\bar{T}}_0\right)_{b1}}{\partial V_i} = \frac{\partial \left(\bar{\bar{T}}^T\right)_{b1}}{\partial V_i} \bar{\bar{T}}_0 \tag{6.22}$$

The Euler transformation tensor is dependent on $\vec{\theta} = \{\varphi\ \vartheta\ \psi\}^T$. The transformation tensor partial derivatives with respect to φ, ϑ and ψ are derived from equation 2.4 as:

$$\frac{\partial \bar{\bar{T}}^T}{\partial \varphi} = \begin{bmatrix} 0 & 0 & 0 \\ -\cos\vartheta\sin\psi\sin\varphi + \sin\vartheta\cos\varphi & -\cos\psi\sin\varphi & -\sin\vartheta\sin\psi\sin\varphi - \cos\vartheta\cos\varphi \\ \cos\vartheta\sin\psi\cos\varphi + \sin\vartheta\sin\varphi & \cos\psi\cos\varphi & \sin\vartheta\sin\psi\cos\varphi - \cos\vartheta\sin\varphi \end{bmatrix} \tag{6.23}$$

$$\frac{\partial \bar{\bar{T}}^T}{\partial \vartheta} = \begin{bmatrix} -\sin\vartheta\cos\psi & 0 & \cos\vartheta\cos\psi \\ -\sin\vartheta\sin\psi\cos\varphi + \cos\vartheta\sin\varphi & 0 & \cos\vartheta\sin\psi\cos\varphi + \sin\vartheta\sin\varphi \\ -\sin\vartheta\sin\psi\sin\varphi - \cos\vartheta\cos\varphi & 0 & \cos\vartheta\sin\psi\sin\varphi - \sin\vartheta\cos\varphi \end{bmatrix} \tag{6.24}$$

$$\frac{\partial \bar{\bar{T}}^T}{\partial \psi} = \begin{bmatrix} -\cos\vartheta\sin\psi & -\cos\psi & -\sin\vartheta\sin\psi \\ \cos\vartheta\cos\psi\cos\varphi & -\sin\psi\cos\varphi & \sin\vartheta\cos\psi\cos\varphi \\ \cos\vartheta\cos\psi\sin\varphi & -\sin\psi\sin\varphi & \sin\vartheta\cos\psi\sin\varphi \end{bmatrix} \tag{6.25}$$

The partial derivatives of the Earth transformation tensor are derived at equations 6.14, 6.15 and 6.16. Next the derivatives of the aircraft coordinate system with respect to the Inertial reference frame ($\vec{R}$) is determined. This partial derivative of $\vec{R}$ is solely dependent on $\vec{R}$:

$$\frac{\partial \vec{R}}{\partial V_i} = \frac{\partial \vec{R}}{\partial \vec{R}} = \begin{bmatrix} \frac{\partial X}{\partial Z} & \frac{\partial Y}{\partial X} & \frac{\partial Z}{\partial X} \\ \frac{\partial X}{\partial Y} & \frac{\partial Y}{\partial Y} & \frac{\partial Z}{\partial Y} \\ \frac{\partial X}{\partial Z} & \frac{\partial Y}{\partial Z} & \frac{\partial Z}{\partial Z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.26}$$

This equation finalizes the input for the derivative $\frac{\partial \vec{r}_{ap,b1}}{\partial V_i}$ (equation 6.20) and the first part of $\frac{\partial \vec{r}_{ssic}}{\partial V_i}$ (equation 6.18). The second part of $\frac{\partial \vec{r}_{ssic}}{\partial V_i}$ is derived as:

$$\frac{\partial \left[\frac{d_{b1}}{d_{b1,b2}}\left(\vec{r}_{ap,b1} - \vec{r}_{ap,b2}\right)\right]}{\partial V_i} = \frac{\partial \left(\frac{d_{b1}}{d_{b1,b2}}\right)}{\partial V_i}\left(\vec{r}_{ap,b1} - \vec{r}_{ap,b2}\right) + \frac{d_{b1}}{d_{b1,b2}} \frac{\partial \left(\vec{r}_{ap,b1} - \vec{r}_{ap,b2}\right)}{\partial V_i} \tag{6.27}$$

From equation 5.12 follows that the term $\left(\frac{d_{b1}}{d_{b1,\,b2}}\right)$ is solely dependent on $\vec{r}_{ap,\,b1}$ and $\vec{r}_{ap,\,b2}$. Therefore the derivatives given in equation 6.27 are a function of the $\vec{r}_{ap,\,b}$ and its partial derivatives only:

$$\frac{\partial\left(\frac{d_{b1}}{d_{b1,\,b2}}\right)}{\partial V_i} = \frac{\frac{\partial\left(\vec{r}_{ap,\,b1}-\vec{r}_{ap,\,b2}\right)}{\partial V_i}\left(\vec{r}_{ap,\,b1}-\vec{r}_{ap,\,b2}\right)}{\sqrt{\left(\vec{r}_{ap,\,b1}-\vec{r}_{ap,\,b2}\right)\cdot\left(\vec{r}_{ap,\,b1}-\vec{r}_{ap,\,b2}\right)}} \tag{6.28}$$

These partial derivatives are derived in section 6.2.1. This finalizes the derivative equation with respect to the sphere-sphere intersection circle vector, which is the first part of the tether-bridle attachment point partial derivative equation.

Second part derivative equations $\frac{\partial \mathbf{\vec{r}_{tba}}}{\partial \mathbf{V_i}}$

The second part of these $\vec{r}_{tba}$ derivative equations is given as:

$$\frac{\partial\left[\frac{R_{ssic}}{\left|(\vec{r}_t)^*\right|}\left(\left(\vec{R}_t\right)^* - \vec{r}_{ssic}\right)\right]}{\partial V_i} = \frac{\partial\left(\frac{R_{ssic}}{\left|(\vec{r}_t)^*\right|}\right)}{\partial V_i}\left(\left(\vec{R}_t\right)^* - \vec{r}_{ssic}\right) + \frac{R_{ssic}}{\left|(\vec{r}_t)^*\right|}\frac{\partial\left(\left(\vec{R}_t\right)^* - \vec{r}_{ssic}\right)}{\partial V_i} \tag{6.29}$$

In this equation:

$$\frac{\partial\left(\frac{R_{ssic}}{\left|(\vec{r}_t)^*\right|}\right)}{\partial V_i} = \frac{\frac{\partial R_{ssic}}{\partial V_i}\left|(\vec{r}_t)^*\right| - R_{ssic}\frac{\partial\left|(\vec{r}_t)^*\right|}{\partial V_i}}{\left|(\vec{r}_t)^*\right|^2} \tag{6.30}$$

The partial derivative equations $\frac{\partial R_{ssic}}{\partial V_i}$, $\frac{\partial\left|(\vec{r}_t)^*\right|}{\partial V_i}$ and $\frac{\partial\left(\vec{R}_t\right)^*}{\partial V_i}$ follow from respectively equation 5.13, 5.24 and 5.22:

$$\frac{\partial R_{ssic}}{\partial V_i} = \frac{\frac{\partial (R_{ssic})^{\mathrm{TP}}}{\partial V_i} d_{b1,\,b2} - (R_{ssic})^{\mathrm{TP}}\, d_{b1,\,b2}}{2\,(d_{b1,\,b2})^2} \tag{6.31}$$

With the top part of the equation for the sphere-sphere intersection circle equation denoted as:

$$(R_{ssic})^{\mathrm{TP}} = \sqrt{(L_{b1}+L_{b2}+d_{b1,\,b2})(L_{b1}+L_{b2}-d_{b1,\,b2})(L_{b1}-L_{b2}+d_{b1,\,b2})(L_{b2}-L_{b1}+d_{b1,\,b2})} \tag{6.32}$$

The partial derivative of the projected tether attachment point to the ground is equal to:

$$\frac{\partial (R_t)^*}{\partial V_i} = \frac{\partial R_t}{\partial V_i} - \frac{\partial \vec{C}}{\partial V_i}\frac{\vec{C}\cdot\vec{R}_t + D}{\vec{C}\cdot\vec{C}} - \vec{C}\cdot\frac{\left(\frac{\partial \vec{C}}{\partial V_i}\cdot\vec{R}_t + \vec{C}\cdot\frac{\partial \vec{R}_t}{\partial V_i} + D\right)\vec{C}\cdot\vec{C} - 2\left(\vec{C}\cdot\vec{R}_t + D\right)\frac{\partial \vec{C}}{\partial V_i}\cdot\vec{C}}{\left(\vec{C}\cdot\vec{C}\right)^2} \tag{6.33}$$

The partial derivative of the projected tether vector is given as:

$$\frac{\partial \vec{r}_t^*}{\partial V_i} = \frac{\partial (R_t)^*}{\partial V_i} - \frac{\partial \vec{r}_{ssic}}{\partial V_i} \tag{6.34}$$

Finally $\frac{\partial|(\vec{r}_t)^*|}{\partial V_i}$ is determined as:

$$\frac{\partial\left|(\vec{r}_t)^*\right|}{\partial V_i} = \frac{\frac{\partial(\vec{r}_t)^*}{\partial V_i}\cdot(\vec{r}_t)^*}{\sqrt{(\vec{r}_t)^*\cdot(\vec{r}_t)^*}} \tag{6.35}$$

This equation concludes the derivation of the tether position vector and next the tether length and force partial derivatives can be determined.

6.2.2 Tether length and force

The tether length partial derivatives follow from the tether location vector partial derivatives and equation 5.27 as:

$$\frac{\partial l_t}{\partial V_i} = \frac{\frac{\partial \vec{r}_t}{\partial V_i}\cdot\vec{r}_t}{\sqrt{\vec{r}_t\cdot\vec{r}_t}} \tag{6.36}$$

The tether force magnitude is a function of the tether length (l_t) a constant initial tether length (l_{t_0}) and a constant spring constant (k_t). The tether force is linearly related to the tether length, hence the tether force magnitude derivatives are linearly related to the tether length partial derivatives as:

$$\frac{\partial\left|F_t\right|}{\partial V_i} = k_t\frac{\partial l_t}{\partial V_i} \tag{6.37}$$

6.3 Conclusions

This chapter showed that ASWING compatible bridle force Jacobian entries can be determined analytically as a function of (1) the local beam location at which the bridle is attached $\vec{r}_i = \{x\ y\ z\}^T$, (2) the local beam Euler angles, $\vec{\theta}_i = \{\varphi\ \vartheta\ \psi\}^T$, (3) the distance from the aircraft reference frame with respect to the Inertial reference frame, $\vec{R} = \{X\ Y\ Z\}^T$ and (4) the roll, pitch and yaw angles $\vec{\Theta} = \{\Phi\ \Theta\ \Psi\}^T$.

The ASWING Jacobian matrix already defied entries for state vector derivatives with respect to $\vec{r}_i$, $\vec{\theta}_i$, $\vec{R}$ and $\vec{\Theta}$ and hence the tether-bridle Jacobian entries will be added to this Jacobian matrix. These ASWING compatible subroutines are calculated in subroutine `SETTET` and subroutine `SETBRI`, which can be found at Appendix C. These subroutines exclude the aerodynamic and gravitational tether contributions. These are determined in the next chapter.

Chapter 7

Tether aerodynamic and gravitational loads

The goal of this chapter is to include the aerodynamic and gravitational loads into the tether-bridle system. The relative small bridles have a negligible contribution to the drag and weight of the system. However the tether weight and drag are responsible for about one-third of the total weight and drag of the system (wing+tether). In this chapter the tether gravitational and aerodynamic forces are derived, such that these forces can be implemented in ASWING. The tether path and gravitational loads are determined first in section 7.1. In section 7.2 the tether aerodynamic loads are calculated. In section 7.3 the routine to change gravity direction is explained and finally in section 7.4 the conclusions of this chapter are drawn.

7.1 Tether gravitational loads

The tether sag is solved analytically by Noom (2013). In this model a uniform distributed load is considered. A dynamic discretized system such as described by Leuthold (2013) are more accurate. However to gain full benefits of the dynamic discretized approach, wind shear should be taken into account. In the current version of ASWING, wind shear is not (yet) implemented. Therefore a dynamic discretized tether system is considered outside the scope of this thesis. First the tether path is defined and next the tether gravitational loads are implemented in the system of equations.

7.1.1 Tether path

The tether sag model of Noom (2013) is based on distributed gravitational loads. In that case the tether path is a function of the tether loading constant; the tether weight divided by the force in horizontal direction:

$$c_t = \frac{\rho_t \pi r_t^2 g}{F_{t,x}} \tag{7.1}$$

Where c_t is the tether loading constant, ρ_t is the density of the tether, r_t is the tether radius and $F_{t,x}$ is the tether force in x-direction. For an elevation angle $\beta_t = 30°$, the tether sag for various loading constants is given at Figure 7.1.

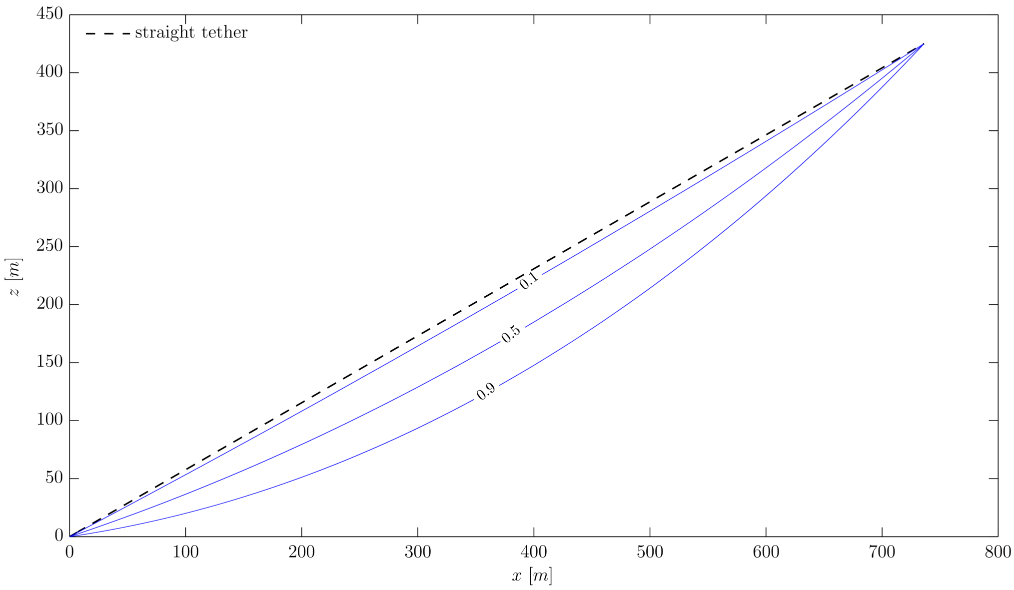


Figure 7.1: Tether sag for various c_t

In this Master's thesis, Makani Power's M600 system is analysed. For this system the tether length is typically 420 meter, tether force in x-directional is $F_{t,x} > 10kN$ and hence the tether loading constant, $c_t < 0.1$. From Figure 7.1 follows that the straight tether assumption can be applied with acceptable error.

7.1.2 Tether gravitational loads implementation

With the straight tether assumptions and a constant weight distribution, the tether center of gravity is halfway the tether length. This center of gravity position ($\vec{r}_{t,\,\text{cog}}$)is determined as:

$$\vec{r}_{t,\,\text{cog}} = \frac{1}{2}\left[\vec{r}_{tba} + \bar{\bar{T}}_E\left(\vec{R}_t - \vec{R}\right)\right] \tag{7.2}$$

Airborne wind systems with rigid wings, such as AmpyxPower and Makani Power, are characterized by high lift over weight ratios (L/W) in the order of, $L/W = 10 - 20$ (Agten, 2012). In flight conditions with a typical lift over weight ratio, $(L/W) = 15$ and a tether weight, which is half the weight of wing ($W_t = 0.5W_w$), the tether weight is 2.2% with respect to the total tether force. Hence it is assumed that the tether-bridle attachment location remains unchanged. The forces are added to the tether force as:

$$\vec{F}_{t,\,\text{new}} = \vec{F}_{t,\,\text{old}} + \vec{W}_t \tag{7.3}$$

With the tether weight defined as:

$$\vec{W}_t = \vec{g}m_t \tag{7.4}$$

7.2 Tether aerodynamic loads

The flight velocity along the tether differs along the tether length. Two initial conditions are known (1) at the ground, the tether velocity is $\vec{v}_t = 0$ and at the attachment location to the wing the tether velocity is close to the flight velocity of the wing, $\vec{v}_t \approx \vec{v}_{\text{wing}}$. Noom (2013) derives the tether drag analytically with the assumptions from Argatov et al. (2009) that:

1. *the apparent wind velocity experienced at the tether $\vec{v}_{a,\,t}$ increases linearly along the tether from zero at the ground station to the magnitude of the apparent wind velocity at the kite*
2. *the direction of $\vec{v}_{a,\,t}$ is constant along the tether and equals the direction of $\vec{v}_{a,\,t}$ at the kite*

These assumptions are valid for the straight tether and hence the tether drag is derived as:

$$D_t = C_{D,t}\frac{1}{4}l_t d_t \frac{1}{2}\rho\vec{v}_a^2 \tag{7.5}$$

The drag force for a point mass is determined in a similar way in ASWING. Drela (2009) gives the aerodynamic drag force for a point mass with effective drag area $(C_D A)_{pm}$ as:

$$D_{pm} = \frac{1}{2}\rho\left|\vec{V}_{pm}\right|\vec{V}_{pm}\,(C_D A)_{pm} \tag{7.6}$$

The subscript $()_{pm}$ denotes a point mass. The effective drag area $(C_D A)$ can be written as a function of the tether diameter, length and drag coefficient. Note the factor $(1/4)$, between both drag equations. This factor is related to the flight velocity along the tether. With exception to the tether attachment point to the wing, the wind velocity experienced by the tether is lower than the wind velocity experienced by the wing. Therefore, aerodynamic forces are reduced. The point mass routine is adjusted to include the tether drag:

$$D_t = \frac{1}{2}\rho\left|\vec{v}_a\right|\vec{v}_a\left(\frac{C_D A}{4}\right)_t \tag{7.7}$$

In flight conditions with a typical lift over drag ratio, $(L/D) = 10$ and a tether drag, which is half the drag of wing ($D_t = 0.5D_w$), the tether drag is 3.3% with respect to the total tether force. As with the tether gravitational force it is assumed that the tether-bridle attachment location remains unchanged. The forces are added to the tether force as:

$$\vec{F}_{t,\,\text{new}} = \vec{F}_{t,\,\text{old}} + \vec{D}_t \tag{7.8}$$

7.3 Gravitational change

With respect to the tether and lift forces, the gravitational forces are relatively low. However, the gravity forces are in the drag force order of magnitude. With respect to the flight velocity the tethered wing gravity directional vector constantly changes over the flight loop. Hence, it would be useful to adjust the gravity directional vector for each flight regime. The adjusted gravity force vector is always given in the Inertial reference frame, hence the gravitational vector in the aircraft reference frame is determined as:

$$\begin{Bmatrix} g_x \\ g_y \\ g_z \end{Bmatrix} = \begin{bmatrix} & \bar{\bar{T}}_E^T & \end{bmatrix} \begin{Bmatrix} g_X \\ g_Y \\ g_Z \end{Bmatrix} \tag{7.9}$$

With this system the user can manipulate the direction of the gravity vector and hence simulate the flight at which the gravity force is aligned with the flight velocity vector.

7.4 Conclusions

In the typical M600 flight regime, the straight tether assumption is valid for tether weight and aerodynamic forces. Readily available routines are adjusted to determine the drag forces and finally a method is implemented to change the direction of gravity. With these modifications the flight along the entire flight loop can be simulated.

In this system, the tether force is first calculated with a massless, drag-less spring as explained in chapter 5. Next the tether gravitation and aerodynamic loads are added. This finalizes the tether-bridle routine. The modified ASWING will be termed ASWINGb, with a b for 'bridled'. The tether aerodynamic and gravitational force additions are programmed in subroutine `SETWGTTET` and can be found at Appendix C.

Part III

Model Verification and Validation

Chapter 8

Tether and bridle force verification

It is the goal of this chapter (1) to verify the tether and bridle force magnitudes and their position in the state matrix $\mathbf{x}$ *and (2) to verify the Jacobian entries and its position in the Jacobian matrix. In section 8.1 the ASWINGb bridle force additions are verified by simulating the bridle force with weights in the original program ASWING. Next a more quantitative approach is taken in section 8.2 by setting up the force and moment equilibrium equations. In section 8.3 tether force constraint is verified. These verifications all assume that tether force and bridle force are determined appropriately in ASWINGb. The verification of the tether and bridle force and the Jacobian entries are given in section 8.4. Section 8.5 and 8.6 respectively verify that the tether aerodynamic and gravitational loads are appropriately determined in ASWINGb.*

8.1 Bridle force verification with ASWING weights

For this analysis the ASWING standard input file `hawk.asw` is used to verify that bridle forces follow correctly from the tether forces and additionally verify that the bridle forces are implemented at the right position in the Newton system of the discrete equation set; that is, the position in state matrix $\mathbf{x}$. The unmodified ASWING program will be used in this verification case. The input file `hawk.asw` can be found in Appendix B.2.

In ASWING the standard `hawk.asw` file is adjusted by adding two equal point mass weights of ($W_{pm} = 150N$) at about half-span location. These weights simulate the bridle forces.

Additionally, in ASWINGb the `hawk.asw` input file is adjusted by attaching two bridles at the same location as the point masses. These bridles have a length equal to four times the full span of the aircraft to ensure that the bridle force vector and the gravity vector are nearly aligned. With this set-up the bridle forces in gravitational direction is $F_{b,Z_E} = 0.992\,|F_b|$. Next the tether force is increased to $|F_t| = 300N$ with the constraint in the ASWINGb constraint GUI. For this symmetrical aircraft with symmetrical tether-bridle system, the tether force is equally divided over both bridles such that $F_{b1,z} = F_{b2,z} \approx |F_b| \approx 0.5\,|F_t|$. For both aircraft set-ups (`hawk.asw` with point masses and `hawk.asw` with bridles) the aircraft flight speed is $V_{IAS} = 0m/s$, hence aerodynamic forces are negligible[1].

[1] ASWING and ASWINGb always set the initial airspeed to a minimum $V_{IAS} = 10^{-6}m/s$ to avoid numerical errors. However for $V_{IAS} = 10^{-6}m/s$ aerodynamic forces are negligible

For both set-ups the deflection of the main wing is measured and given in Figure 8.1. In this Figure y is the wing span location and δ is the deflection. The deflections due to point masses in ASWING and bridle forces in ASWING are equal.

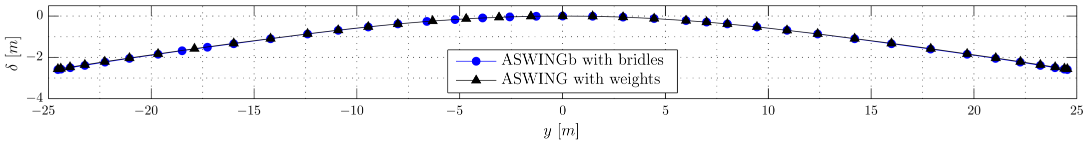


Figure 8.1: Bending of hawk's main wing for a wing loaded with point masses and with bridles

8.2 Force and moment additions to x

The bridle forces and derivatives are added in the Newton system of the discrete equation set. To verify this part of the code the sum of forces and moments are determined for two cases.

1. An anchored flying wing with no tether and bridle forces and moments,
2. an anchored flying wing with tether and bridle forces and moments.

In case 2 the sum of forces and moments should be equal to the case 1 sum of forces and moments plus the forces and moments induces by the bridles. The force and moment equations are verified with analytical equations at which the bridle forces are input values determined with ASWINGb. This sum of forces or moments should be equal to the the sum of forces and moments of the unbridled wing plus the bridle additions.

A variety of tests cases is run with different wind velocities, angles-of-attack, center of gravity of the aircraft and Euler angles. These characteristics are given in Table 8.1.

Table 8.1: Flight parameters for ASWINGb bridle force verification for analytical determined force and moment equilibria

	$\mathbf{V_{IAS}}$ [m/s]	α [deg]	**X** [m]	**Y** [m]	**Z** [m]	$\mathbf{\Phi}$ [deg]	$\mathbf{\Theta}$ [deg]	$\mathbf{\Psi}$ [deg]
1	50.0	4.0	0.0	0.0	0.0	0.0	0.0	0.0
2	50.0	5.0	0.0	5.0	0.0	0.0	0.0	0.0
3	50.0	5.0	10.0	5.0	0.0	0.0	0.0	0.0
4	60.0	5.0	10.0	5.0	1.0	0.0	0.0	0.0
5	60.0	5.0	10.0	5.0	1.0	10.0	0.0	0.0
6	60.0	5.0	10.0	5.0	1.0	10.0	5.0	0.0
7	60.0	5.0	10.0	5.0	1.0	10.0	5.0	12.0

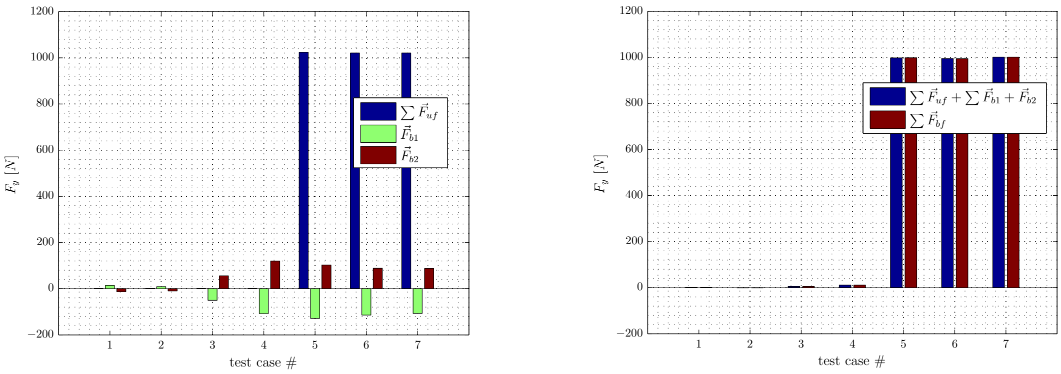
8.2.1 Force additions to x

The sum of forces for the bridled case is equal to the sum of forces of the unbridled case plus the bridle additions:

$$\sum \vec{F}_{bf} = \sum \vec{F}_{uf} + \sum \vec{F}_{b} \tag{8.1}$$

In this equation, $()_{bf}$ denotes bridled flight, $()_{uf}$ unbridled flight and $()_b$ bridle forces and moments.

For these seven cases the sum of forces in x, y and z-direction are determined for the unbridled case and the bridled case. The results in y-direction are visualized with a bar-chart in Figures 8.2a and 8.2b.



(a) Force equilibrium y-direction

(b) [ASWING vs analytical, forces in y-direction

Figure 8.2: Comparison sum of forces in y-direction for the unbridled and bridled case

In the first 4 test cases the sum of forces in y-direction for the unbridled case is close to zero. In case 5-7 a roll angle $\Phi = 10°$ is applied and forces in y-direction increase. The bridle y-forces are opposite in direction and different in magnitude. Therefore the total sum of forces for the bridled wing is different with respect to the unbridled wing. The sum of forces for the unbridled flying wing plus the additional bridle forces is equal to the ASWINGb determined sum of forces in y-direction for the bridled case as shown in Figure 8.2b.

The force equilibrium graphs in x and z-direction show equal results and can be found in Appendix D.1. This verifies that the forces in x, y and z-direction are appropriately added to the Newton system of equations.

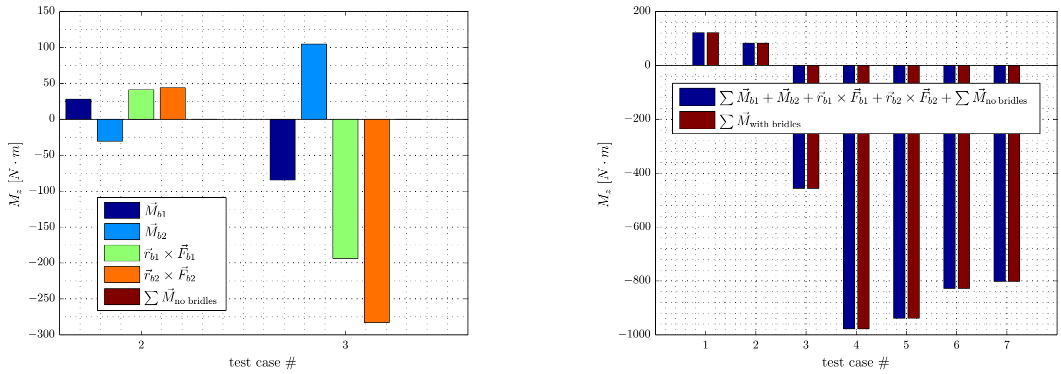
8.2.2 Moment additions to x

The sum of moments for the bridled case is equal to the sum of moments for the unbridled case plus the moments induced by the bridles. These moments are split in two parts; (1) the first part $\vec{M}_b$ contains the moments induced at the bridle attachment point of the wing. This moment is created by the bridle force in combination with the rigid pylon offset to the wing. The second part $\vec{F}_b \times \vec{r}_{bi}$ is the moment induced due to bridle force and the offset between the wing attachment point and the location at which the moment is determined.

In an equation:

$$\sum \vec{M}_{bf} = \sum \vec{M}_{uf} + \sum \vec{M}_b + \sum \vec{F}_b \times \vec{r}_{i,b} \tag{8.2}$$

To determine the correct moments: (1) the moments induced by the bridles $\vec{M}_b$ should be properly loaded into the Newton system, (2) the bridle forces $\vec{F}_b$ should be properly loaded into the system of equations and (3) these bridle forces should be loaded at the proper location into the Newton system ($\vec{r}_{i,b}$). The components of the moments about the z-axis are given in Figures 8.3a and 8.3b. The moments of test case



(a) Moment equilibrium z-axis

(b) ASWING vs analytical, moments about z-axis

Figure 8.3: Comparison sum of moments about z-axis for the unbridled and bridled case

2 and test case 3 are split into the components given at equation 8.2. The moments about the z-axis are mainly a results of bridle components.

From Figure 8.3b follows that the sum of moments determined by ASWINGb is equal to the sum of moments about the z-axis determined analytically. The moment equilibrium graphs about the x and y-axis show equal results and can be found in Appendix D.1. This verifies that the forces and moments in x, y and z-direction are appropriately added to the Newton system of equations.

8.3 Tether force constraint and equilibrium set-up

The previous section verified that the bridle forces are appropriately loaded into the Newton system of equations. The next step is verifying the tether force constraint and resulting force and moment equilibrium equations. For this verification the ASWING standard file for a flying wing (`fw.asw`) is adjusted for tethered flight. This input file can be found in Appendix B.3.

The ASWING module for force equilibrium equations is modified, such that bridle forces are included as well. In this verification case, three different flight modes are used: anchored mode with zero tether force $|F_t| = 0.0N$, the anchored mode with tether force $|F_t| = 4000.0N$ and force equilibrium mode with $\sum \vec{F} = 0$ and $|F_t| = 4000.0N$. In ASWINGb the anchored mode means that linear and angular accelerations are respectively $\vec{a}_0 = 0$ and $\vec{\alpha}_0 = 0$ and sum of force is not necessarily zero. Anchored mode is best comparable with a wind tunnel test in which the model is fixed.

For all three flight modes, the flying wing flight speed is $V_{IAS} = 30m/s$, with angle-of-attack $\alpha_{ref} = 2°$. The tether spring constant is $k_t = 100N/m$. The bridles are symmetrically attached at half wing span ($b = 24m$). The tether is attached to the ground at $\vec{R}_T = [0.0 \;\; 0.0 \;\; -75.0]$. The aircraft reference frame is located at $\vec{R} = [0.0 \;\; 0.0 \;\; 0.0]$.

For the second mode (anchored mode with $|F_t| = 4000.0N$) the ASWINGb constraint GUI is used with constraint $|F_t| = 4000.0N \rightarrow Z_E$. This means that ASWINGb calculates a solution at which $|F_t| = 4000.0N$ by changing Z_E (the Z-coordinate of the aircraft reference frame). From Table 8.2 follows that $Z_E = 40.0m$. This corresponds to $F_t = k_t \Delta Z$.

For the third mode $\left(\sum \vec{F} = 0.0N, \quad |F_t| = 4000.0N\right)$ the force equilibrium is given as:

$$\sum \vec{F} = \vec{F}_{aero} + \vec{F}_{mass} + \vec{F}_t + \vec{F}_{acc} \tag{8.3}$$

In this equation, $\vec{F}_{aero}$ are forces due to aerodynamic loads, $\vec{F}_{mass}$ are forces due to mass, $\vec{F}_t$ are tether forces and $\vec{F}_{acc}$ are forces induced by accelerations. The sum of forces is evaluated in x, y and z-direction.

With the ASWINGb determined forces (Table 8.2) and ASWINGb determined components: $L = 9857.0N$, $L/D = 32.65$, $\alpha = 2°$, $\dot{U}_x = -0.07040 m/s^2$ and $\dot{U}_z = -0.06154 m/s^2$, the sum of forces equals:

$$\sum F_x = \frac{D}{L}L - \sin\alpha\,(W + |F_t|) - \dot{U}_x \frac{W}{g} = -1.3 \tag{8.4}$$

$$\sum F_y = 0.0 \tag{8.5}$$

$$\sum F_z = -\cos\alpha\,(W + |F_t|) + L - \dot{U}_z)\frac{W}{g} = 0.0 \tag{8.6}$$

The sum of forces in x-direction is unequal to zero due to rounding errors. The lift drag ratio is given with an 0.01 accuracy. Hence the drag force is determined with an error ΔD of

$$D = D \pm \Delta D = \frac{D}{L}L \pm 0.005L = D \pm 49.3N \tag{8.7}$$

Table 8.2: ASWINGb tether force verification with sum of forces

	$\mathbf{Z_E}$ **[m]**	$\sum \mathbf{F_x}$ **[N]**	$\sum \mathbf{F_y}$ **[N]**	$\sum \mathbf{F_z}$ **[N]**
Anchored mode, $\lvert F_t\rvert = 0.0N$	0.0	-42.73	0.00	3997.0
Anchored mode, $\lvert F_t\rvert = 4000.0N$	40.0	-42.33	0.00	-38.43
$\sum \vec{F} = 0.0N \ \lvert F_t\rvert = 4000.0N$	40.0	0.00	0.00	0.00

From equation 8.4 - 8.7 follows that the force equilibrium equations are properly set-up with the newly implemented tether and bridle forces, and hence the tether force constraint and the force and moment equilibrium set-up is verified.

8.4 Verification tether and bridle force calculation

The previous sections verified that bridle forces are appropriately loaded into the Newton system of equations for a given tether force. This section will verify that tether forces and Jacobian entries are calculated correctly by ASWINGb.

The bridle forces are a function of the distance between the aircraft - and Inertial reference frame $\left(\vec{R}\right)$, the roll, pitch and yaw angles $\left(\vec{\Theta}\right)$, the location of the attachment point of the bridles $(\vec{r}_i)$ and the local beam Euler angles $\left(\vec{\theta}_i\right)$. With two bridle attachment locations, 18 variables will influence the Jacobian entries. Three variables from $\vec{R}$ and $\vec{\Theta}$ and six variables from $\vec{r}_i$ and $\vec{\theta}_i$. Each variable contributes to the force in x, y and z direction. Hence $18 \cdot 3 = 54$ different Jacobian entries are specified in ASWINGb. From chapters 5 and 6 follow that these 54 Jacobian entries are dependent on the tether force and its derivatives. For a given tether force and its derivatives the bridle forces and the bridle Jacobian entries are determined with the same equations.

In this section first the tether force and its derivatives with respect to $\vec{R}$, $\vec{\Theta}$, $\vec{r}_i$ and $\vec{\theta}$ are verified. Next the general equations are verified, which calculate the bridle forces and their derivatives from the tether force and its derivatives.

To verify the output an independent MATLAB program is written. The integrated MATLAB routine `solve` calculates the bridle forces from equation 5.15. The integrated MATLAB routine `diff` calculates the Jacobian entries which are explained in chapter 6.

8.4.1 Tether forces as a function of $\vec{R}$

The influence of the distance between the aircraft - and Inertial reference frame $\vec{R}$ with respect to the tether force and its derivatives is determined with the ASWING standard `hawk.asw` input file. This file can be found in Appendix B.2. To determine only the $\vec{R}$ influence the tethered aircraft has an infinitely stiff main wing, and additionally the roll, pitch and yaw angles are equal to zero $\left(\vec{\Theta} = 0\right)$. The bridles are attached at about half-way span at $\vec{r}_{ap,b1} = \{0.00, \ -12.40, \ 0.00\}^T$ and $\vec{r}_{ap,b2} = \{0.00, \ 12.40, \ 0.00\}^T$. The tether-bridle attachment point with zero tether force is equal to $\vec{r}_{tba} = \{0.00, \ 0.00, \ -10.00\}^T$. The tether attachment point to the groundstation is $\vec{R}_T = \{0.00, \ 0.00, \ -300.00\}^T$, hence the unstressed tether lengths is $l_{t,0} = 290.00m$. The tether spring constant used in this verification case is $k_t = 100N/m$.

With these simplifications the tether locational vector follows from equations 5.7 - 5.9 as:

$$\vec{r}_t = \vec{R}_T - \vec{R} - \vec{r}_{tba} \tag{8.8}$$

Note that the Earth transformation tensors are left outside equation 8.8, because the Euler angles are zero and hence the Earth transformation tensor is a unity matrix.

Another simplification is needed to determine the influence of $\vec{R}$ analytically. The tether-bridle attachment point is fixed in the aircraft reference frame and assumed independent of $\vec{R}$. With these simplifications the tether force is solely a function of a constant $\vec{R}_T$, a constant $\vec{r}_{tba}$ and a varying $\vec{R}$. The tether force is now determined as:

$$|F_t| = k_t \left(\sqrt{\left(\vec{C} - \vec{R}\right) \cdot \left(\vec{C} - \vec{R}\right)} - l_{t,0} \right) \tag{8.9}$$

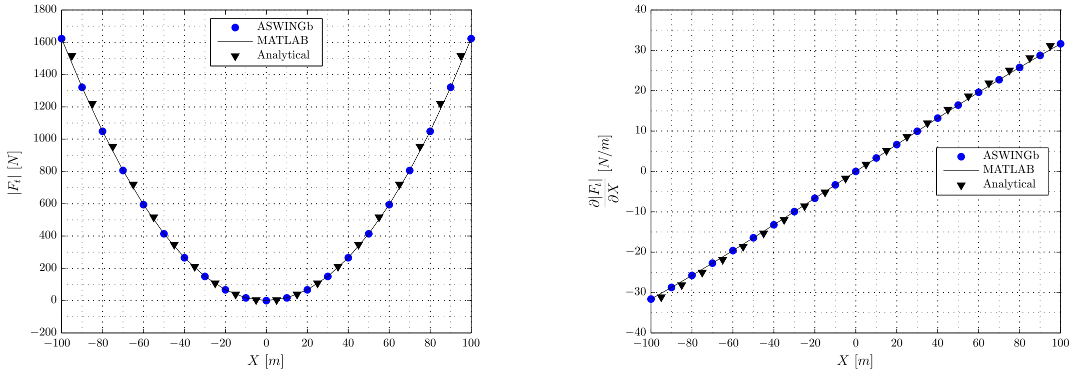
With constant $\vec{C}$ as:

$$\vec{C} = \vec{R}_T - \vec{r}_{tba} \tag{8.10}$$

Tether force as a function of X

The ASWINGb, MATLAB and analytically calculated tether force and its derivative with respect to X are given in Figures 8.4a and 8.4b.

At $X = 0.0m$, $|F_t| = 0.0N$, the analytical results are equal to the MATLAB and ASWINGb results, because equal $\vec{r}_{tba}$ is used. At $X = 100.0m$ the hand calculated tether force $|F_t| = 1,675.7N$, whereas ASWINGb and MATLAB determine both $|F_t| = 1,6228N$; a 3.3% difference. The tether-bridle attachment point determined by ASWINGb and MATLAB is $\vec{r}_{tba} = \{-3.16 \ 0.00 \ -9.49\}$, whereas the analytical method applied



(a) $|F_t|$ versus X

(b) $\frac{\partial |F_t|}{\partial X}$ versus X

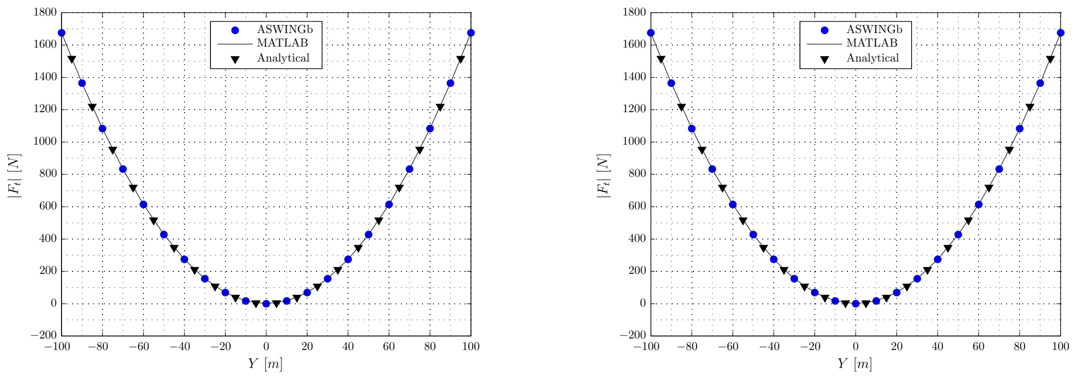
Figure 8.4: Tether force and its derivative versus X

$\vec{r}_{tba} = \{0.00\ 0.00\ -10.00\}$.

The tether force derivatives with respect to X are given in Figure 8.4b. This Figure shows that the ASWING Jacobian $\left(\frac{\partial |F_t|}{\partial X}\right)$ is equal to the Jacobian determined with the MATLAB integrated `diff` routine.

Tether force as a function of Y

The ASWINGb calculated tether force and its derivative with respect to Y are given in Figures 8.5a and 8.5b. The analytical results are equal to the MATLAB and ASWINGb results, because a shift of Y does not change the position of $\vec{r}_{tba}$.



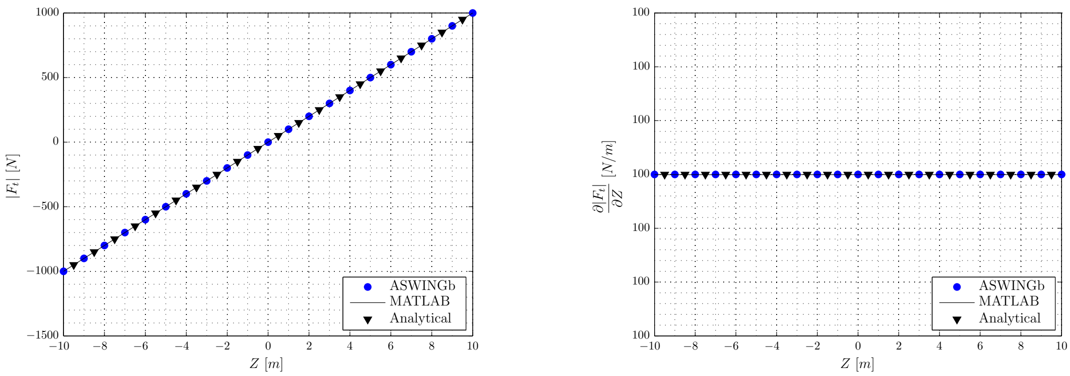
(a) $|F_t|$ versus Y

(b) $\frac{\partial |F_t|}{\partial Y}$ versus Y

Figure 8.5: Tether force and its derivative versus Y

Tether force as a function of Z

The ASWINGb calculated tether force and its derivative with respect to Z are given in Figures 8.6a and 8.6b. The analytical results are equal to the MATLAB and ASWINGb results, because a shift of Z does not change the position of $\vec{r}_{tba}$.



(a) $|F_t|$ versus Z

(b) $\frac{\partial |F_t|}{\partial Z}$ versus Z

Figure 8.6: Tether force and its derivative versus Z

The tether force $|F_t|$ and Z are positively linearly related, because the tether positional vector $x_t = y_t = 0$ and tether force is given as:

$$|F_t| = k_t \left(\sqrt{(Z_T - Z - z_{tba}) \cdot (Z_T - Z - z_{tba})} - l_{t,0} \right) = k_t \left(Z_T - Z - z_{tba} - l_{t,0} \right) \tag{8.11}$$

In this verification case Z_T, z_{tba} and $l_{t,0}$ are kept constant and hence the tether force is linearly related with respect to position Z. In this case the tether force derivative with respect to Z is equal to the spring constant $k_t = 100N/m$.

$$\frac{\partial |F_t|}{\partial Z} = k_t \tag{8.12}$$

8.4.2 Tether forces as a function of Euler angles $\left(\vec{\Theta}\right)$

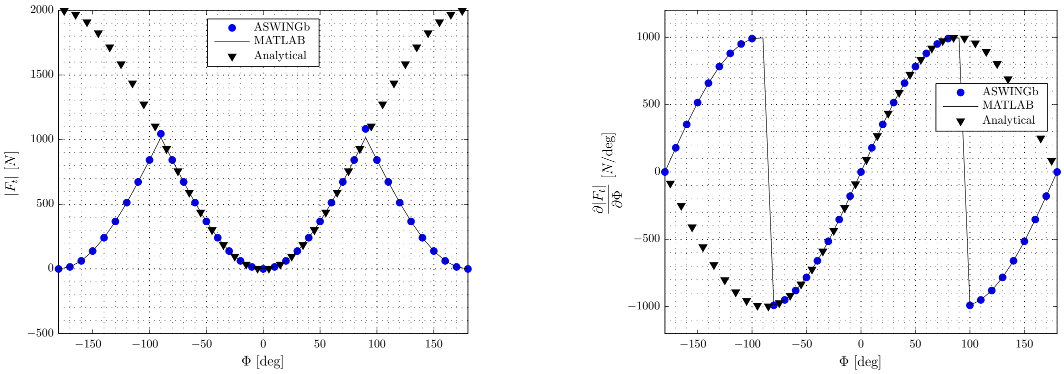
For the verification analyses for the roll, pitch and yaw angles the ASWING input file for the rigid hawk is used (Appendix B.2). The rigid hawk is applied to ensure zero wing bending and torsion. The wing flexibility does not influence the tether force and its derivative and hence the Euler angles' influences can be determined independently. As with the verification for $\vec{R}$ the MATLAB integrated `solve` and `diff` routines are used to determine tether forces and derivatives. With an analytical method a more intuitive approach is added.

Tether force as a function of bank angle (Φ)

For the analytical method the tether-bridle attachment point $\vec{r}_{tba}$ depends on the bank angle transformation tensor as:

$$\begin{Bmatrix} x_{tba} \\ y_{tba} \\ z_{tba} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{bmatrix} \begin{Bmatrix} x_{tba,0} \\ y_{tba,0} \\ z_{tba,0} \end{Bmatrix} \tag{8.13}$$

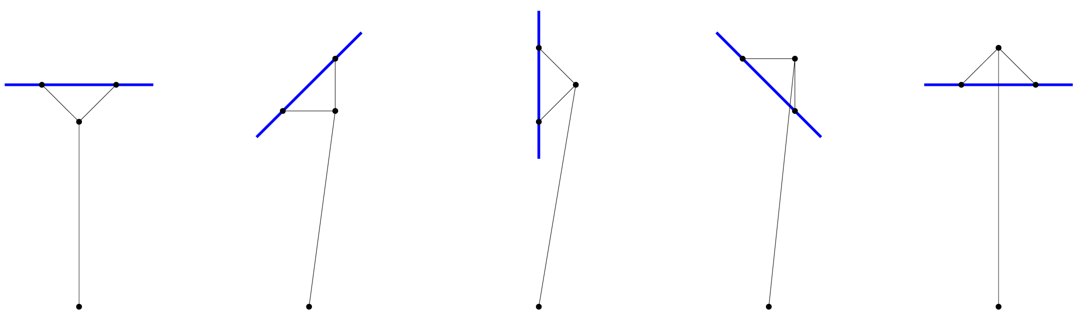
In this equation the subscript $()_0$ denotes the zero tether force case. The ASWINGb, MATLAB and analytically calculated tether force and its derivative with respect to the bank angle are given in Figures 8.7a and 8.7b.



(a) $|F_t|$ versus Φ (b) $\frac{\partial |F_t|}{\partial \Phi}$ versus Φ

Figure 8.7: Tether force and its derivative versus Φ

For bank angles $|\Phi| < \pm 90°$, the analytically calculated tether force and its derivative are close to the tether force calculated by ASWINGb and MATLAB. For increasing bank angles the tether-bridle attachment points determined by equation 8.13 are not between the aircraft and the tether-bridle attachment point, whereas ASWINGb and MATLAB determine the tether-bridle attachment point between the aircraft and the tether attachment to the ground. Hence for $|\Phi| > \pm 90°$ the analytical method is invalid. In the extreme case $\Phi = \pm 180°$, the aircraft is flipped upside-down and hence the tether-bridle attachment point determined with equation 8.13 is mirrored with respect to the zero bank angle case and also mirrored with respect to the tether-bridle attachment point determined with MATLAB and ASWINGb. The tether-bridle attachment points as determined with equation 8.13 are visualized in Figure 8.8.



(a) $\Phi = 0°$ (b) 45° (c) 90° (d) 135° (e) 180°

Figure 8.8: Position of the analytically determined tether-bridle attachment point for various bank angles Φ

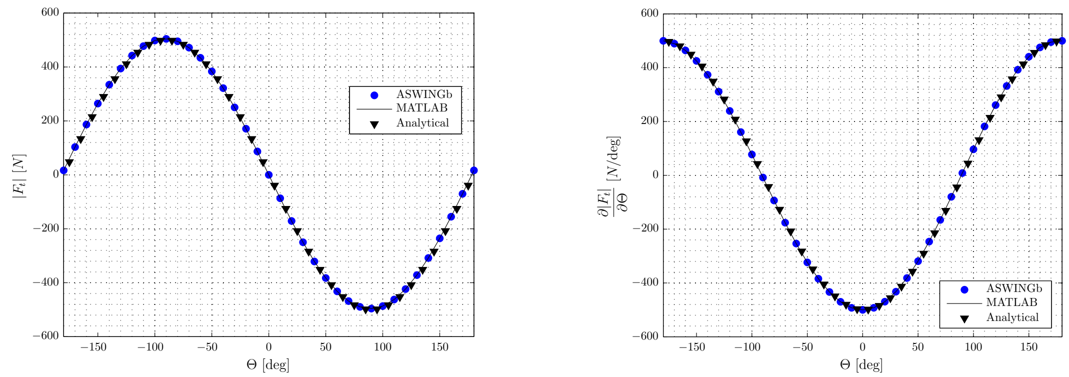
From this Figure follows that the tether forces determined analytically for $|\Phi| < \pm 90°$ are physically impossible. Most important, the ASWINGb and MATLAB calculated tether force and its derivative with respect to the bank angle are equal.

Tether for as a function of elevation angle (Θ)

For the analytical method, assume that the tether-bridle attachment point $\vec{r}_{tba}$ will change according to:

$$\begin{Bmatrix} x_{tba} \\ y_{tba} \\ z_{tba} \end{Bmatrix} = \begin{Bmatrix} x_{tba,0} \\ y_{tba,0} \\ z_{tba,0} \end{Bmatrix} + \begin{bmatrix} \cos\Theta & 0 & \sin\Theta \\ 0 & 1 & 0 \\ -\sin\Theta & 0 & \cos\Theta \end{bmatrix} \begin{Bmatrix} x_i + \Delta x_p \\ 0 \\ z_i + \Delta z_p \end{Bmatrix} \tag{8.14}$$

In this equation $\vec{r}_i + \Delta\vec{r}_p$ is the distance from the aircraft reference frame R to the bridle attachment locations (Euler angles relative to $\vec{R}$). The tether-bridle attachment y-position is assumed independent of the elevation angle. Figures 8.9a and 8.9b show the tether force and its derivative with respect to the elevation angle. From these Figures follow that the ASWINGb and MATLAB calculated tether force and its derivatives are equal and analytical results are slightly off with respect to the ASWINGb and MATLAB determined values.



(a) $|F_t|$ versus Θ

(b) $\frac{\partial |F_t|}{\partial \Theta}$ versus Θ

Figure 8.9: Tether force and its derivative versus Θ

Tether force as a function of yaw angle (Ψ)

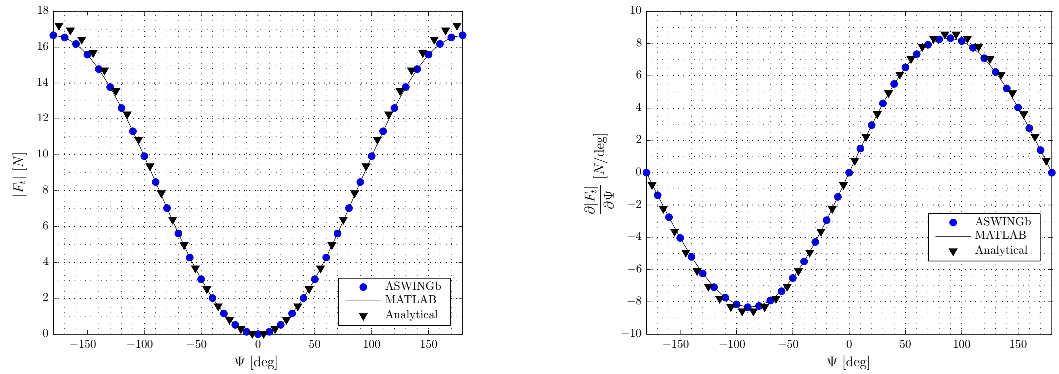
For the analytical method, assume that the tether-bridle attachment point $\vec{r}_{tba}$ will change with the yaw angle transformation tensor as:

$$\begin{Bmatrix} x_{tba} \\ y_{tba} \\ z_{tba} \end{Bmatrix} = \begin{bmatrix} \cos\Psi & \sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x_{tba,0} \\ y_{tba,0} \\ z_{tba,0} \end{Bmatrix} \tag{8.15}$$

The tether force and its derivative with respect to the yaw angle are given in Figures 8.10a and 8.10b. For large positive or negative yaw angles ($|\Psi| > \pm 90°$) the analytically calculated tether forces are slightly different with respect to the ASWINGb and MATLAB calculated tether forces, because the tether-bridle attachment location is slightly different than proposed by equation 8.15. Most important, the ASWINGb and MATLAB calculated tether force and its derivative with respect to the yaw angle are equal.

8.4.3 Tether forces as a function of local beam variables $\vec{r}_i$ and $\vec{\theta}_i$

For this analysis the standard ASWING input for a flying wing is adjusted for tethered flight. This ASWINGb input file can be found in Appendix B.3. The tether-bridle characteristics are given in Table 8.3.



(a) $|F_t|$ versus Ψ

(b) $\frac{\partial|F_t|}{\partial\Psi}$ versus Ψ

Figure 8.10: Tether force and its derivative versus Ψ

Table 8.3: Tether-bridle characteristics for flying wing (Appendix B.3)

$\vec{r}_{i,\,b1}$	=	$\{0.0,\ \ -6.0,\ \ 0.0\}^T$	$[m]$
$\vec{r}_{ap,\,b1}$	=	$\{-6.0,\ \ -12.0,\ \ 6.0\}^T$	$[m]$
$\Delta\vec{r}_{p,\,b1}$	=	$\{-6.0,\ \ -6.0,\ \ 6.0\}^T$	$[m]$
l_{b1}	=	22.4	$[m]$
$\vec{r}_{i,\,b2}$	=	$\{0.0,\ \ 6.0,\ \ 0.0\}^T$	$[m]$
$\vec{r}_{ap,\,b1}$	=	$\{12.0,\ \ 12.0,\ \ -12.0\}^T$	$[m]$
$\Delta\vec{r}_{p,\,b1}$	=	$\{12.0,\ \ 6.0,\ \ -12.0\}^T$	$[m]$
l_{b2}	=	17.0	$[m]$
$\vec{R}_T$	=	$\{0.0,\ \ 0.0,\ \ 75\}^T$	$[m]$
k_t	=	100.0	$[N/m]$
$l_{t,\,0}$	=	63.0	$[m]$

In the previous sections each individual parameter could be varied independently to determine its effect on the tether force. Unfortunately this approach is impossible for a flexible wing and each beam variable is dependent on wing bending flexibility effects and thus airspeed. To verify the tether force for the local beam variables $\vec{r}_i$ and $\vec{\theta}_i$ the wing is flown at different airspeeds. The tether force as a function of flight speed is given in Figure 8.11. These tether forces are determined with constant $\vec{R}$ and $\vec{\Theta}$. Next the tether force derivatives with respect to local beam coordinates $(\vec{r}_i)$ and local beam Euler angles $\left(\vec{\theta}_i\right)$ are determined .

Tether force derivatives with respect to local beam coordinates $\vec{r}_i$

With increasing airspeed, the lift force increases, which increases wing flexibility effects. This wing bending influences the tether force. Recall that the tether force is determined from the locational difference between the tether-bridle attachment point and the tether attachment at the ground. The wing bending and torsion influence the position of the tether-bridle attachment point. The tether force derivatives with respect to the local beam coordinates $\vec{r}_{i,\,b1}$ and $\vec{r}_{i,\,b2}$ are given in Figures 8.12a, 8.12b and 8.12c.

Bridle 1 is attached at negative $x_{i,\,b1}$ and $y_{i,\,b1}$, whereas bridle 2 is attached at positive $x_{i,\,b2}$ and $y_{i,\,b2}$. The tether-bridle attachment point is in between these bridle attachment points at the wing. Moving a bridle

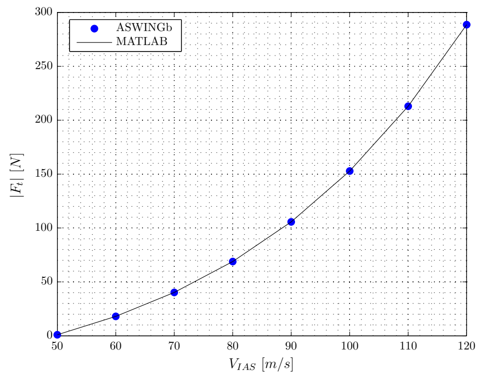


Figure 8.11: Tether force versus air speed for flexible flying wing

attachment point towards the tether-bridle attachment point will decrease the tether force and vice versa. This is shown in Figures 8.12a and 8.12b; bridle 1 is characterized by a negative $\frac{\partial |F_t|}{\partial x_{i,b1}}$ and $\frac{\partial |F_t|}{\partial y_{i,b1}}$, whereas bridle 2 is characterised by a positive $\frac{\partial |F_t|}{\partial x_{i,b2}}$ and $\frac{\partial |F_t|}{\partial y_{i,b2}}$. For bridle 1 the partial derivative with respect to $z_{i,b1}$ is close to the spring constant, which means that a difference in $z_{i,b1}$, will results in an almost equal difference in z_{tba}. It is interesting to note that the tether force derivative with respect to $z_{i,b2}$ is close to zero. The bridle 2 attachment z-position is close to the tether-bridle attachment z-position. The x and y differences between $\vec{r}_{ap,b2}$ and $\vec{r}_{tba}$ are relatively large and hence an increment in $z_{i,b2}$ has a small influence on z_{tba}.

Most important: for all points the MATLAB determined Jacobians are equal to values determined by ASWINGb.

Tether force derivatives with respect to local beam Euler angles $\vec{\theta}_i$

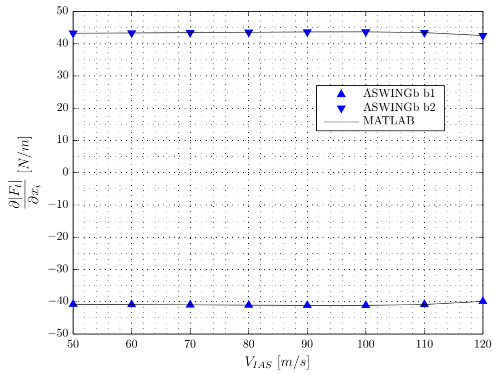
The bridle-attachment location is determined as: (1) determine local attachment location at the wing ($\vec{r}_i$), and next (2) multiply local Euler angles with the rigid pylon offset $\left(\Delta r_p\right)$. This rigid pylon dimensions are given in Table 8.3 and the tether force derivatives with respect to the local beam Euler angels are given in Figure 8.13a, 8.13b and 8.13c.

For all points the MATLAB determined derivatives are equal to the derivatives calculated with ASWINGb.

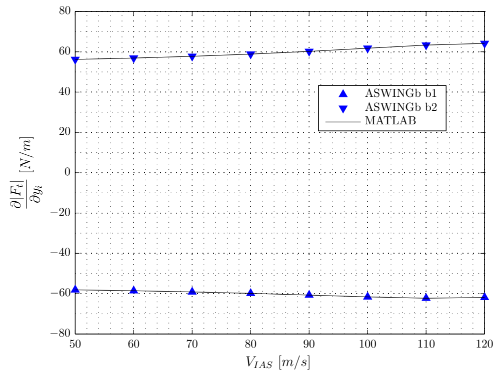
8.4.4 Bridle forces as a function of tether force

The previous sections verified that the tether force magnitude and its derivatives are determined correctly by ASWINGb. This section will verify that the bridle forces and their derivatives follow correctly from the tether force.

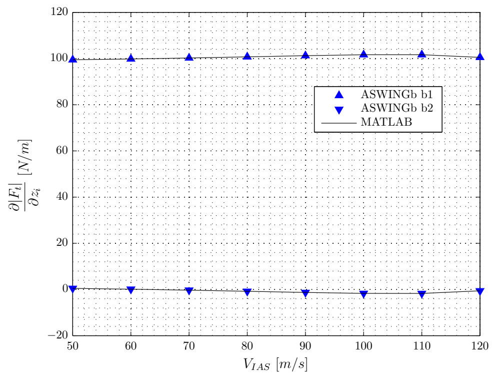
The tether force vector is determined from the tether positional vector and the tether force magnitude, see equation 5.29. This tether force vector is an input for the bridle force magnitude and subsequently for bridle force vectors, see equations 5.30, 5.31 and 5.32. To verify that these calculation steps are appropriately implemented in the ASWINGb code, the bridle force vectors as a function of the tether force magnitude are given in Figures 8.14a, 8.14b and 8.14c.



(a) $\frac{\partial|F_t|}{\partial x_i}$ versus V_{IAS}



(b) $\frac{\partial|F_t|}{\partial y_i}$ versus V_{IAS}



(c) $\frac{\partial|F_t|}{\partial z_i}$ versus V_{IAS}

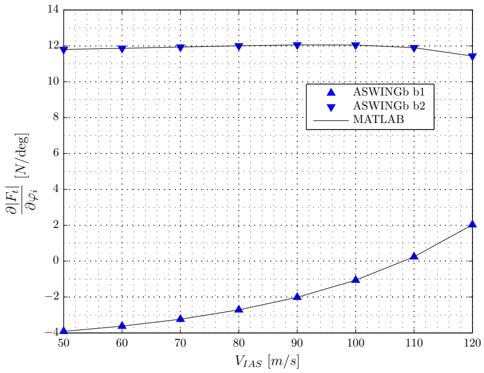
Figure 8.12: Tether force derivative with respect to local beam coordinates

From these figures follows that the tether x and y forces are relatively small compared to the tether force in z-direction. This is caused by the small x and y directional components of the tether; the aircraft is almost straight above the tether attachment at the ground station. The bridle force in x and y direction are almost equal in magnitude, but opposite in direction. The bridle 2 z-component is relatively small compared to the bridle 1 and tether force component in z-direction, because the z-difference between the bridle attachment point and the tether-bridle attachment location is relatively small. Hence bridle 1 takes most of tether loads.

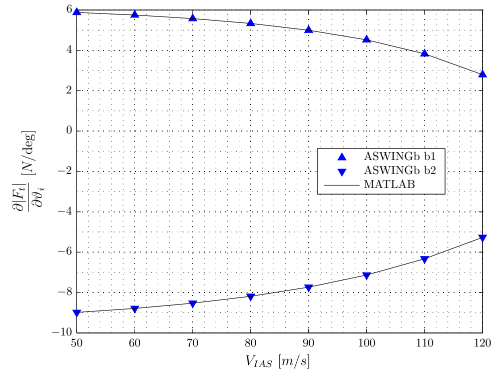
The sum of forces in x, y and z are all equal to zero and most important, the tether and bridle force components determined with ASWINGb are equal to the force components determined with MATLAB.

8.4.5 Bridle force Jacobian entries

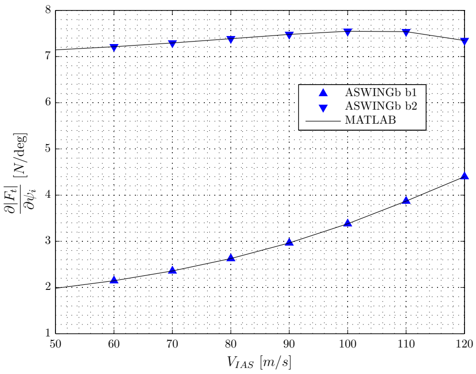
The bridle Jacobian entries are a function of tether force $\left(\vec{F}_t\right)$, the tether force derivatives $\left(\frac{\partial \vec{F}_t}{\partial V_i}\right)$, the bridle attachment point $\left(\vec{r}_{ap}\right)$ and the tether-bridle attachment point $(\vec{r}_{tba})$. The individual input is all verified in the previous sections. Finally the equations are verified to determine the bridle force derivatives from all individual components (equations 6.1, 6.2, 6.6 and 6.8). The derivative with respect to any arbitrary variable, $\vec{R}$, $\vec{\Theta}$, $\vec{r}_{i,b1}$, $\vec{r}_{i,b2}$, $\vec{\theta}_{i,b1}$ or $\vec{\theta}_{i,b2}$, can be used to verify this part of the ASWINGb code. The bridle force derivatives with respect to the bridle 1 x position at the wing $(x_{i,b1})$ are given in Figures 8.15a, 8.15c and 8.15c.



(a) $\frac{\partial|F_t|}{\partial\varphi_i}$ versus V_{IAS}



(b) $\frac{\partial|F_t|}{\partial\vartheta_i}$ versus V_{IAS}



(c) $\frac{\partial|F_t|}{\partial\psi_i}$ versus V_{IAS}

Figure 8.13: Tether force derivative with respect to local beam Euler angles

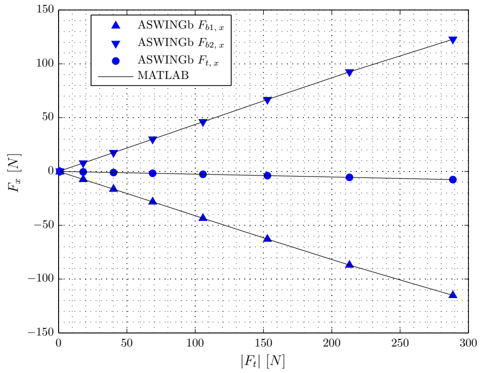
These Figures show the bridle force derivatives determined with ASWINGb are equal to the values determined with MATLAB and hence this final part of the code is verified as well. For redundancy the other bridle force Jacobian entries are verified as well. These results can be found in Appendix D.2.

8.5 Implementation of tether aerodynamic drag force

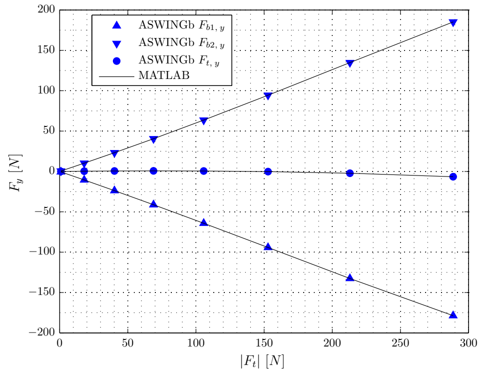
For the verification of the tether aerodynamic drag the M600 configuration is used as given in Appendix B.4. For a M600 representative tether with tether diameter $d_t = 25mm$, tether length $l_t = 400m$ and tether effective drag area $(C_D A)_t = 8.00m^2$ the drag force is determined with equation 7.7 as $D_t = 4410.0N$.

In ASWINGb the analysis is run twice. Once excluding the tether drag and once with tether drag included. The difference in the sum of forces in x-direction ($\Delta \sum F_x$) should be equal to tether drag force. The sum of forces in x-direction are determined by ASWINGb as:

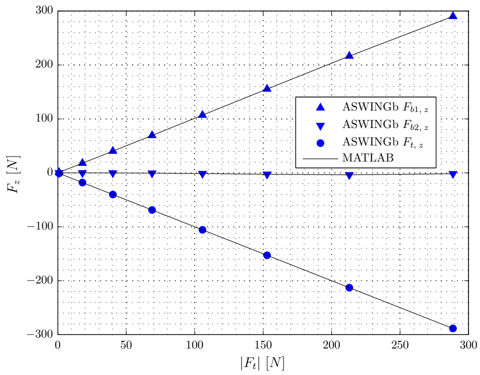
	$\sum \mathbf{F_x}$
$(C_D A)_t = 0.00m^2$	8,582.0
$(C_D A)_t = 8.00m^2$	12,990.0



(a) F_x versus $|F_t|$



(b) F_y versus $|F_t|$



(c) F_z versus $|F_t|$

Figure 8.14: Tether and bridle force vector verification

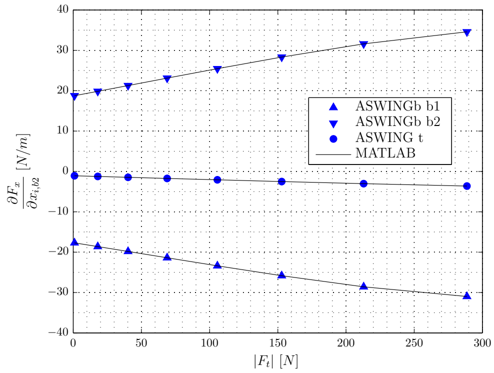
The difference in sum of forces in x-direction is $\Delta \sum F_x = 4408.0N$, which is 99.95% with respect to the hand calculated drag force.

8.6 Implementation of tether gravity

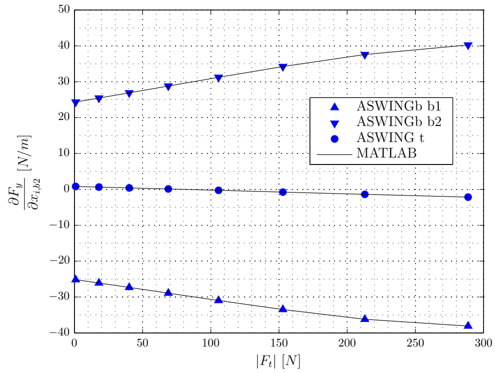
In case Euler angles are equal to zero, $\left(\vec{\Theta} = 0\right)$, gravitational forces are aligned with the aircraft z-axis. The ASWINGb simulation is run with and without tether weight. The tether-bridle connection and the bridle wing connection is free to rotate around all axis, hence the additional tether weight does not create any moments. Only the tether force in z-direction is changed as:

	$F_{t,z}$
$W_t = 0.00N$	-154,600.0
$W_t = 4,000N$	-150,600.0

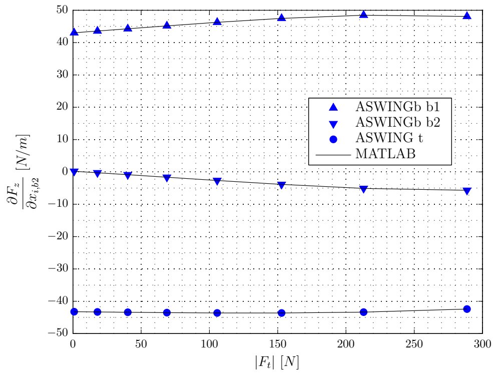
The tether force difference in z-direction is $\Delta F_{t,z} = 4000N$, which is equal to the tether weight. The negative tether force is less, because part of the aerodynamic lift forces are balanced with the tether weight.



(a) $\frac{\partial F_x}{\partial x_{i,\,b2}}$ versus $|F_t|$



(b) $\frac{\partial F_y}{\partial x_{i,\,b2}}$ versus $|F_t|$



(c) $\frac{\partial F_z}{\partial x_{i,\,b2}}$ versus $|F_t|$

Figure 8.15: Verification bridle force Jacobian entries

8.7 Conclusions

The bridle force magnitudes and position in the state matrix are verified. Additionally the Jacobian entries are verified for all independent variables which influence the bridle force. Also the tether-force constraint and the implementation of the tether aerodynamic and gravitational loads are verified. These verification cases give confidence in the program and to increase confidence more, wind tunnel tests are used for validation.

Chapter 9

Wind tunnel test

It is the goal of this chapter to validate the dynamic aero-elastic modes of the ASWING tether-bridle addition with a wind tunnel test. Many different dynamic aero-elastic modes exist and the validation of all individual modes with wind tunnel data is outside the scope of this research. The tether and bridles influences of one relatively simple wing configuration are examined. As a benchmark, first this model is tested without the tether-bridle system. Second the bridles and tether are attached. The wind tunnel results are compared to the ASWING and ASWINGb results. In section 9.1 the wind tunnel is briefly described. The design of the wind tunnel model is explained in section 9.2. The wind tunnel model weights and stiffnesses are validated in section 9.3. In section 9.5 the results of the wind tunnel test are given and finally in section 9.6 the conclusions are drawn.

9.1 Wind tunnel description

The Delft University of Technology (TU Delft) low speed low turbulence wind tunnel (LLT) has a test section of 1.80 meter wide, 1.25 meter high and 2.60 meters long. The low turbulence intensity (0.015% at 20 m/s to 0.07% at 75 m/s (Aerodynamics Research Group)) make this wind tunnel very suitable to determine 2D airfoil lift curve slopes. Hence wind turbine blade and airfoil design are an ongoing research at this wind tunnel. A picture of this wind tunnel is shown in Figure 9.1.

The TU Delft focusses on many field of research, such as but not limited to: '*research on laminar airfoils for sail planes and wind turbines*', '*boundary layer suction*', '*flow control of separation on wing-flap systems*' and '*education: analysis on airfoils, wings and aircraft models with propeller propulsion*' (Aerodynamics Research Group).

This wind tunnel is chosen, because of its relative high maximum velocity ($V_{\max, LLT} = 100 m/s$) and additionally two movable wing tip constraints can be added. These constraints limit the vibrational motion to about $5mm$ and hence will prevent the wing from destructive failure. The wind tunnel width, which is $1.80m$ should be sufficiently large for the wind tunnel model.

9.2 Design of the wind tunnel model

The design requirement for the wind tunnel model are summarized as:

Figure 9.1: TU Delft LLT with 2D test set-up

The model

1. should be able to resist the loads at $V_{IAS} = 80m/s$ with $\alpha = 10°$,
2. may not cause any damage to the wind tunnel in case flutter strikes,
3. should show aero-elastic modes at low wind speeds,
4. should be easy to build,
5. should be easily transportable from California to Delft.

To satisfy requirement 4, a straight, symmetrical, single airfoil, $1.20m$ span wing is chosen as a starting point. Requirement 1 and 3 are a possible set of conflicting requirements; a stiff and strong wing can cope with the loads, but is less susceptible for aero-elastic modes. To satisfy both requirement a specific airfoil design and fibre lay-up are designed as described in the next section.

9.2.1 Airfoil design

In general; long, slender, thin wings with a center of gravity either aft or forward the aerodynamic center are more susceptible to any dynamic or static aero-elastic modes. With these considerations and to satisfy the easy to build requirement, a relatively thin off the shelf available NACA0012 airfoil profile is chosen, see Figure 9.2. The lift and drag curve for Reynolds numbers 150,000 - 350,000 are created with the Javafoil web applet (Hepperle, 2006) and given in Figure 9.3. With a characteristic chord $\bar{c} = 3$" ($76.2mm$), these Reynolds numbers correspond to a wind velocity range $V_w = 30m/s - 65m/s$.

Fiber lay-up

In section 3.3.1 several different methods are explained to determine material stiffnesses for given fibre lay-up. To calculate the stiffnesses for the wind tunnel model, the Hart-Smith method is considered the most appropriate method, because of its simplicity with respect to CLA and its accuracy in case the lay-up consists of only 0/90 and 45/45 layers, which is the case for the wind tunnel model.

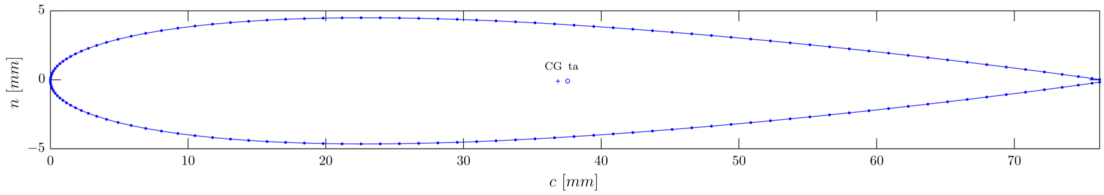
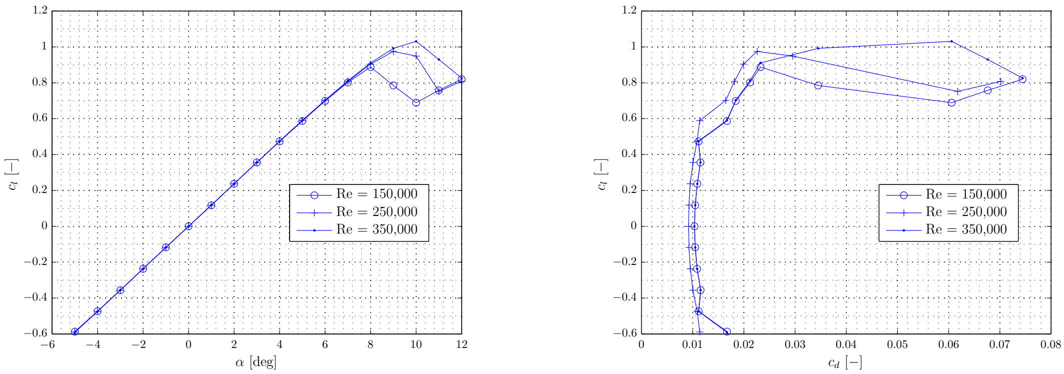


Figure 9.2: NACA0012 airfoil



(a) Lift polar NACA0012

(b) Lift-drag curve NACA0012

Figure 9.3: Aerodynamic characteristics NACA0012 airfoil

Airfoil Properties

The wind tunnel model should not break in case of flutter. To satisfy this requirement the wing tip movement is restricted by thin walled airfoil shaped steel constraints. The wing tip upward - and downward movement is restricted to a few millimetre. However in case of unexpected destructive failure, this should cause the minimum damage possible. Highly brittle carbon fibres will splinter. Glass fibre as well, however in a less extend. Kevlar fibres are ductile and hence will not splinter apart after destructive failure. A lightweight wing is more susceptible to aero-elastic modes, and hence the main disadvantage of Kevlar with respect to carbon fibre is its lower stiffness-density. To satisfy the safety and the susceptibility to aero-elastic modes a hybrid lay-up is chosen. From the surface to the foam:

For $y = 0.0b - 0.6b$
- 2 layers 0.0889 mm Kevlar 0/90
- 2 layers 0.2000 mm Carbon unidirectional
- 2 layers 0.0889 mm Kevlar 0/90
- 1 layer Glass

For $y = 0.6b - 1.0b$
- 2 layers 0.0889 mm Kevlar 0/90
- 1 layer 0.2000 mm Carbon unidirectional
- 2 layers 0.0889 mm Kevlar 0/90
- 1 layer Glass

Kevlar is abrasive resistant. The outermost thin Glass layer is (easier) sand-able and added for a smooth surface finish and a solid glue connection with the rods (explained in section 9.2.2). This Glass layer is thin and no additional stiffness/strength is assumed from this layer. Two unidirectional carbon fibres add strength to the wing from the root to the bridle connection at about half-way span, whereas one layer unidirectional layer is sufficient from about half-way span towards the tip. These carbon layers are sandwiched between two layers Kevlar 0/90.

This fibre lay-up results from the combination of requirement 1 *'the model should be able to resist the* *loads at $V_{IAS} = 80m/s$ with $\alpha = 10°$'* requirement 2 *'the model may not cause any damage to the wind tunnel in case flutter strikes'* and requirement 3 *'The model should show aero-elastic modes at low wind speeds'*. The unidirectional Carbon and the 0/90 Kevlar add strength to take up bending loads, whereas these layers minimally increase the torsional stiffness. Additionally the ductile Kevlar fibres keep the wing together in case of destructive failure. The resulting properties are summarized in Table 9.2. The relevant material characteristics are given in Table 9.1 (NorthwestFoam.com, 2007; AircraftSpruce; FlyingFoam).

Table 9.1: Material properties used for wind tunnel model

Symbol	Description	Kevlar	Carbon	surfboard	Unit
ρ	density	1,373.0	1,271.1	35.2	$[kg/m^3]$
E	Youngs modulus of elasticity	131.0	230.3	<0.01	$[GPa]$

Table 9.2: Airfoil characteristics wind tunnel model

Symbol	Description	Value		Unit
		$\mathbf{y = 0.0(b/2) - 0.6(b/2)}$	$\mathbf{y = 0.6(b/2) - 1.0(b/2)}$	
W	weight per unit span	1.6843	1.2968	$[N/m]$
n_{cg}	n-position center of gravity	0.0000	0.0000	$[m]$
c_{cg}	c-position center of gravity	0.0370	0.0369	$[m]$
n_{ea}	n-position elastic center	0.0000	0.0000	$[m]$
c_{ea}	c-position elastic center	0.0376	0.0376	$[m]$
EI_{nn}	bending stiffness about n-axis	1,349.9	1,171.8	$\left[N/m^2\right]$
EI_{cc}	bending stiffness about c-axis	103.8	63.1940	$\left[N/m^2\right]$
GJ	torsional stiffness	11.4	7.6040	$\left[N/m^2\right]$
EA	extensional stiffness	$9.1477 \cdot 10^6$	$5.5692 \cdot 10^6$	$[N]$
$i_{cc}g$	weight-inertia/span about c-axis	$1.6913 \cdot 10^{-4}$	$1.2599 \cdot 10^{-4}$	$[N \cdot m]$
$i_{nn}g$	weight-inertia/span about n-axis	0.0074	0.0055	$[N \cdot m]$

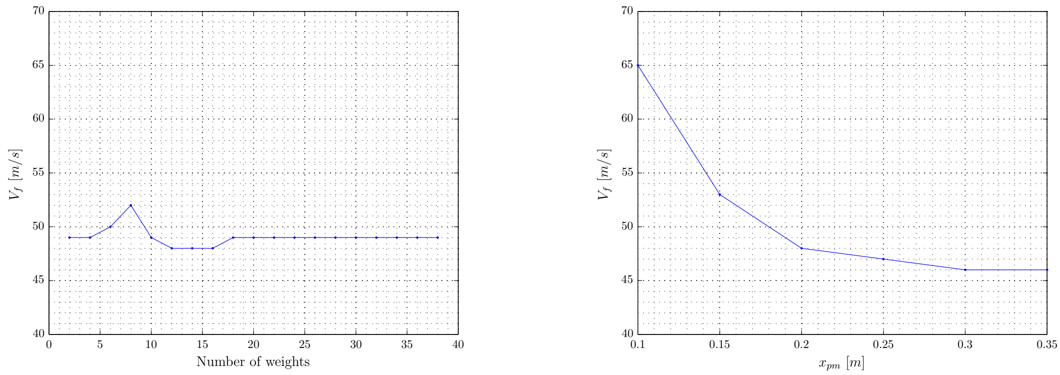
9.2.2 Weight-rod design

The straight, symmetrical, single airfoil, $1.20m$ span wing is stable in case no modifications are made. Therefore aero-elastic modes are induced by (1) shifting the airfoil center of gravity aft and (2) placing mass inertia behind the wing's aerodynamic center. The mass (inertia) is induced with a glued rod at the wing, which is oriented towards the aft. As a second purpose the bridles could be attached to these rods. This section describes this weight-rod design, which should decrease the flutter speed. The number of weights, the magnitude of the weights and the location of the weights are determined in this section.

Number of weights

The number of weights along the wing span is varied between 38 and 2. In the 38 weights case the weights are evenly distributed along the span. The weights are removed from the root of the wing. Hence in the 2 weights case there is one weight at each wing tip. The flutter speed as a function of the number of weights is given in Figure 9.4a. In each case, the magnitude of each point mass weight is $W_{pm} = 0.12N$ and located at $x_{pm} = 15cm$ from the airfoil aerodynamic center.

As shown in Figure 9.4a the flutter speed is slightly dependent on the number of weights. For ease of manufacturing a design with four weights is used. The analyses of the remaining of this chapter are performed for a wing with four point mass weights.



(a) Flutter speed as a function of the number of weights

(b) Flutter speed as a function of the position of the weights

Figure 9.4: Flutter speed as a function of the weights

Position of weights

The wing is more susceptible to flutter the further aft the weights are attached. The x-position of the weights is varied from $x_{pm} = 10cm$ to $x_{pm} = 35cm$ behind the quarter chord line. The flutter speed as a function of the position of the weights is given in Figure 9.4b.

As shown in Figure 9.4b, the further aft the weights, the lower the flutter speed. Hence the weights should be positioned as far aft as practically possible.

Magnitude of weights

Without weight no flutter will occur for this specific wind tunnel model. With (too) heavy weights; (1) the wing may break due to its own weight and (2) the high wing mass inertia stabilizes the wing and hence increases the flutter speed. The magnitude of each weight is varied from $W_{pm} = 0.01N$ to $W_{pm} = 0.30N$.

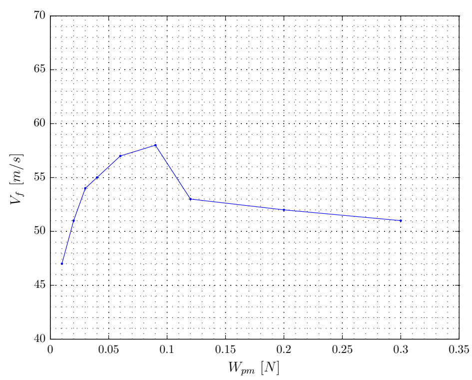


Figure 9.5: Flutter speed as a function of the magnitude of weights

Figure 9.5 shows that small magnitude weights result in the lowest flutter speeds. These small weights are practically impossible, because the rod should be able to transport the bridle forces to the wing.

A lightweight, strong, thin and stiff carbon rod is chosen as the optimum. The carbon rod is $30cm$ long and weights about $20g$. In ASWINGb this rod is simulated as a $0.2N$ point mass at $x_b = 15cm$.

9.2.3 Tether-bridle system

The tether is simulated with a spring scale and the bridles are made from nylon strings. The two bridles can be attached to multiple locations at the wing. At each wing-half one rod is glued at half-way span and another close to the wing tip. The bridles can be attached to multiple locations at these rods.

The tether is simulated with a spring scale, which is set to a certain pre-stress to simulate the tether forces. The initial pre-stress is varied from $F_{t,\,ini} = 25N$ to $F_{t,\,ini} = 75N$. For these initial spring forces the initial wing bending is acceptable. The main requirement for the spring stiffness is its sensitivity to linear deflections. A spring scale is chosen with a spring stiffness equal to $k_t = 1000N/m$ and a maximum linear deflection equal to $\Delta x_{t,\,\max} = 20cm$. For the initial spring forces the linear deflections will vary from $\Delta x_t = 2.5cm$ to $\Delta x_t = 7.5cm$.

This finalizes the design of the wind tunnel model. A picture of this model is given in Figure 9.6.

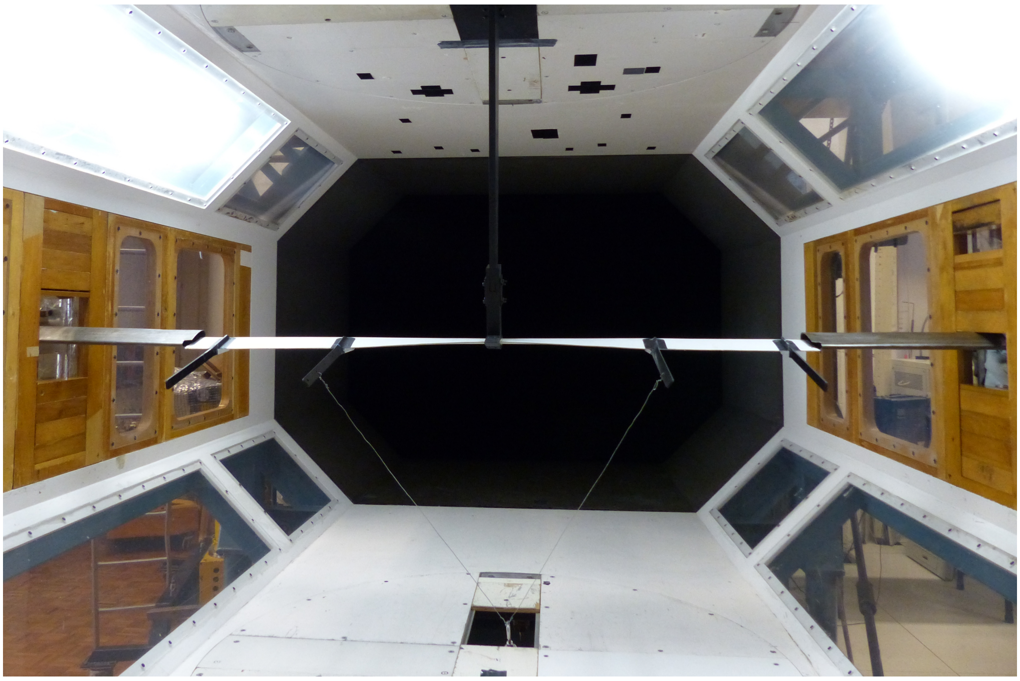


Figure 9.6: Wind tunnel model in the TU Delft LLT

9.3 Stiffness and weight validation

In this section the two most important inputs of the stiffness matrix are determined from static tests. First the bending stiffness, EI_{cc} is analysed, second the torsional stiffness GJ and finally the weight of the wing is determined.

9.3.1 Bending stiffness EI_{cc}

The test set-up is rather simple; a series of $500g$ ($= 4.9N$) weights are placed at the connection rods close to the wing tips and the z-deflection is measured. The measurement tool is calibrated for zero deflections at the zero load case. The test set-up is shown in Figure 9.7.

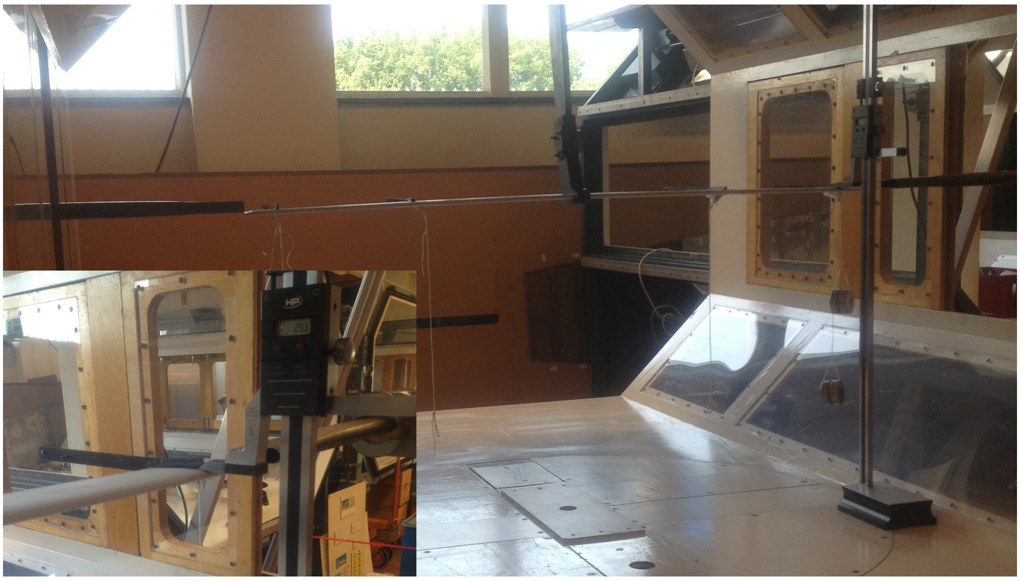


Figure 9.7: Test set-up deflection measurements

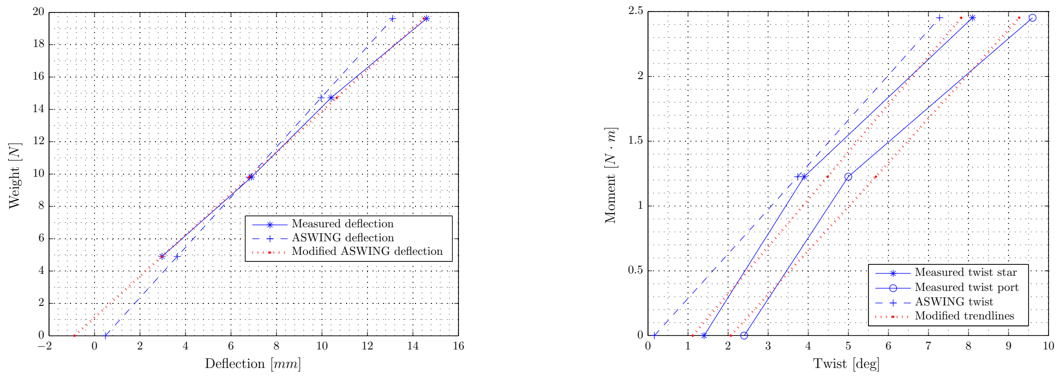
In ASWING, the deflections for the same load cases are determined. The measured and the ASWING calculated deflections are plotted as a function of the load in Figure 9.8a. The slope of the curves represent the stiffness. The offset of the measured deflections with respect to the ASWING calculated deflections can be explained by (1) deflections due to the weight of the wing and (2) imperfections in the manufacturing process. This offset does not influence the stiffness parameters of the wing. The slope of the ASWING deflection line and the offset are modified to fit the experimental data. With respect to the original determined bending stiffness, the real stiffness is 17.7% lower. The difference in stiffness is explained by the sensitivity to fibre orientation.

9.3.2 Torsional stiffness GJ

The angle of twist and the torsional stiffness are linearly related; a measurement of the angle of twist represents the torsional stiffness. A torsional load is induced by weights attached to the outer most edge of the carbon connection rods at $x = 25cm$ and $y = \pm 55cm$ from the quarter chord line. Again the same load cases are evaluated with ASWING. The twist angles as a function of torsional load are given in Figure 9.8b. Due to imperfections in the manufacturing process, the zero load angle of twist is unequal at the wing tips. To fit the experimental data, the ASWING torsional stiffnesses are adjusted with respectively a factor 1.06 and 0.99. Hence the torsional stiffness is on average 3% lower with respect to the torsional stiffness determined with the model.

9.3.3 Weight and weight inertia

The real weight of the wing excluding the carbon struts is $W_{\text{wing}} = 2.45N$. In ASWING the total wing weight is determined as $W_{\text{wing}} = 1.84N$. To account for this difference a weight and weight inertia correc-



(a) Measured deflections compared to ASWING determined deflections

(b) Measured angle of twist compared to ASWING determined angle of twist

Figure 9.8: Measured deflection and twist compared to ASWING

tion factor $\gamma_{w,\mu} = 1.33$ is applied. The weight difference is explained by the manufacturing process; extra epoxy is added to the foam to ensure a solid fibre-foam connection.

9.4 Test set-up

First the model is attached in the wind tunnel as shown in Figure 9.6. Next the bridles are attached to a certain position at the wing. Subsequently the spring is pre-stressed to a certain force. Due to the spring force the wing bends downward, and a positive real angle-of-attack is induced by the twist angle. The thin walled airfoil shaped steel constraints cover about $1cm$ of the wing tips. This is the starting point for each test case. Next the wind velocity is gradually increased. The aerodynamic loads bend the wind upward and additionally increase the real angle-of-attack. The wing-tip constraints are constantly adjusted to ensure that the wing-tips are free to translate and rotate in all directions. The flutter speed is defined as the minimum wind speed at which the model constantly flickers against the constraints. The tests are repeated for:

- spring forces, $F_{t,\text{ini}} = 25N - 100N$
- bridles attached to half-way span and at the wing tips
- bridles attached to the carbon connection rod, $x_b = 1.0cm - 22.5cm$
- tether angle with the ground, $\beta = 0.0° - 30.0°$

The complete test schedule can be found in Appendix E.

9.5 Results wind tunnel test

To get benchmark results, this model is first tested without the tether-bridle system. Second the bridles are attached to about half-way wing span. In the last analysis case the bridles are attached to the wing-tips.

For all cases the wind tunnel results are compared to the ASWING and ASWINGb results.

9.5.1 ASWING benchmark run

With the original, unmodified ASWING the flutter speed for the wind tunnel model is calculated as $V_f = 47.5 m/s$. The wind tunnel tests showed a flutter speed $V_f = 58 m/s$. The difference can be explained by imperfections in the airfoil shape resulting in a difference in aerodynamic center, center of gravity, aero-elastic axis, distributed weight inertia and measurement errors determining the torsional and bending stiffness. The ASWING root locus plot of the four most interesting modes is given in Figure 9.9. The two lower frequency modes get unstable and cross the $\sigma = 0$ line at about equal velocity.

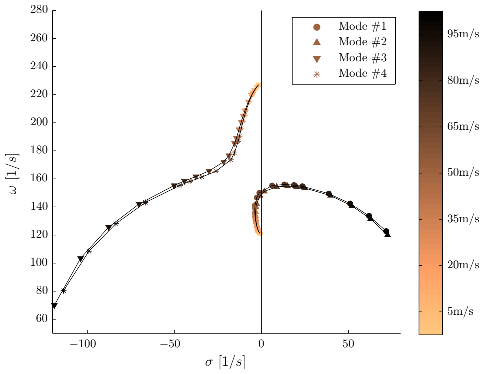
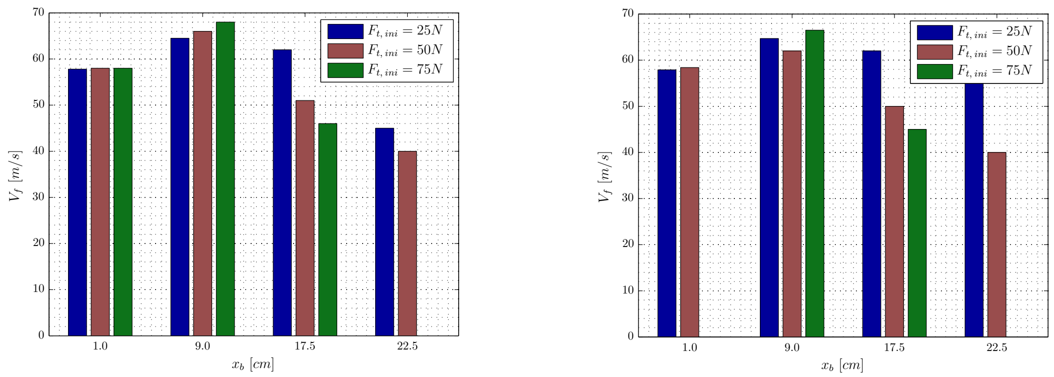


Figure 9.9: Root locus plot unbridled flight with original ASWING

9.5.2 Bridles at halfway span

The bridles are attached to the carbon fibre rods at half-way span and this bridle attachment point is varied in x-direction between $x_b = 0.0 - 22.5 cm$. For each attachment point the initial spring (tether) force is adjusted between $F_{t,\text{ini}} = 25N - 75N$. For the first set of measurements the tether angle is set to $\beta_t = 0°$. For the second set $\beta_t = 30°$. Independent of the tether angle, an interesting phenomenon occurred. At $x_b \leq 9cm$ the wing is very stable up to the flutter speed and flutter strikes 'suddenly'. At higher $x_b \geq 9cm$ the wing starts to wiggle, starts to hit the constraints when trying to get into full flutter mode and finally hits full flutter mode. In case of an unconstrained wing it is likely that the initial wiggling is the first unstable flutter mode and hence this first flutter speed is shown in Figure 9.10.



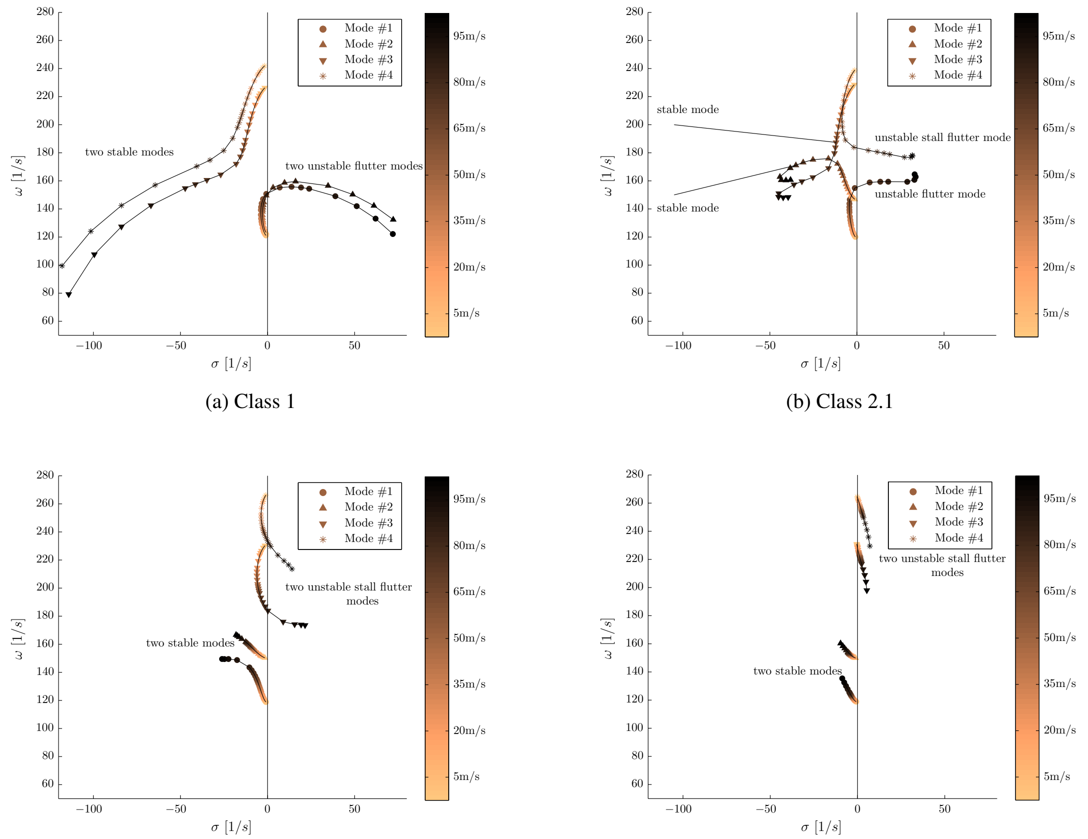
(a) $\beta_t = 0°$ (b) $\beta_t = 30°$

Figure 9.10: Wind tunnel results bridles half-way span

Comparison wind tunnel results with ASWINGb

For bridle attachment points $x_b = 17.5cm$ and $x_b = 22.5cm$ the wind tunnel results show that flutter speed decreases with tether force, whereas this phenomenon does not occur at bridle attachment points $x_b = 1.0cm$ and $x_b = 9.0cm$. The next section will explain these results with the results from ASWINGb.

For increasing x_b and constant F_t, the ASWINGb root locus plots change in a repetitive pattern, see Figure 9.11.



(a) Class 1

(b) Class 2.1

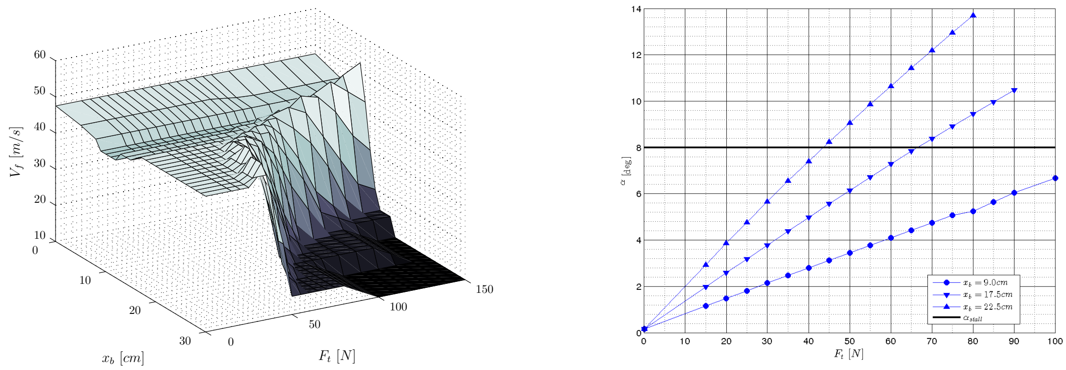
(c) Class 2.2

(d) Class 3

Figure 9.11: Root locus plot with modified ASWINGb

In case the bridles are attached close to the airfoil aero-dynamic center the root locus plot is similar to the unbridled root locus plot (Compare the bridled case (Figure 9.11a) to the unbridled case (Figure 9.9). Increasing the distance of the bridle attachment point from the airfoil aero-dynamic center toward the rear changes the root locus plot first to Figure 9.11b and subsequently to Figure 9.11c. These two root locus plots correspond to slightly higher flutter speeds, in the order of $10 - 20\%$. Increasing the bridle attachment point further aft results in root locus plots given in Figure 9.11d. At first glance this root locus plot is similar to Figure 9.11c, however the flutter speed has decreased with 40% with respect to the unbridled flutter speed.

The flutter speed as a function of bridle attachment point and tether force is given in Figure 9.12a. From this Figure follows that the flutter speed suddenly decreases for specific x_b and F_t combinations. This ‘sudden’ drop occurs at about the $x_b - F_t$ combinations at which the real angle-of-attack is close to the stall angle-of-attack, see Figure 9.12b. These results suggest that stall flutter strikes. A sudden drop of flutter speed due to stall effects is in consensus with the research from Yung (2002). Stall flutter phenomena occur due to the stall induced vortices.



(a) Flutter speed as a function of bridle attachment point and tether force

(b) Angle of attack as a result of wing twist

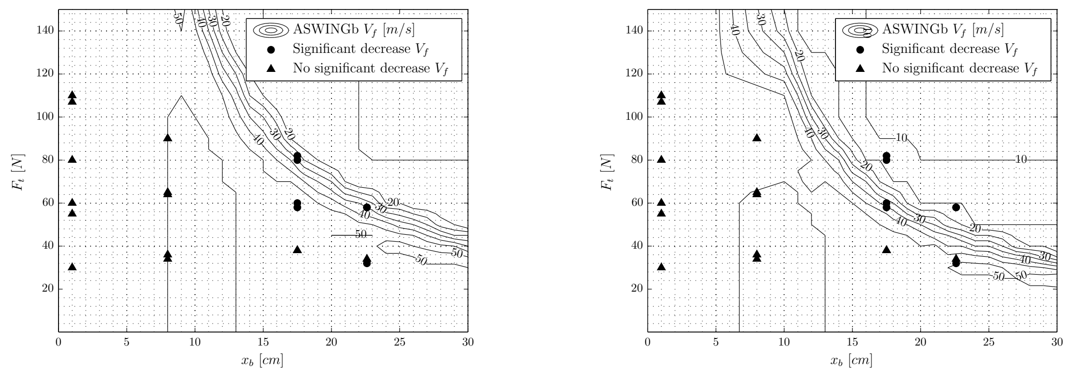
Figure 9.12: Flutter speed as a function of bridle attachment point and tether force, the bridles are attached halfway span

As an example take the the $x_b = 22.5cm$ case. The ASWINGb calculated flutter speed decreased from $V_f = 50m/s$ to $V_f = 20m/s$ with a tether force increase from $F_t = 32N$ to $F_t = 57N$. The wind tunnel results show decreasing flutter speed with increasing tether tension for this case as well. However the drop is less severe. Yung (2002) found that stall flutter usually decreases flutter speed to a minimum and then rises again as the wing is completely stalled. ASWING is unreliable in this full stall region and no modifications are made with respect to this aspect in ASWINGb. The slight increase in flutter speed between $F_t = 20N$ to $F_t = 32N$ is shown in both, the wind tunnel and the ASWINGb results and is explained by the decreasing lift curve slope in the $c_l - \alpha$ region.

To compare the wind tunnel results with ASWINGb results, Figure 9.15a plots the wind tunnel results in a contour plot of Figure 9.12a. An equal trend is shown; higher F_t and x_b result in lower V_f. However not all wind tunnel data points correspond well with the model. This could possibly be caused by a difference in aerodynamic properties, a difference in torsional and bending stiffness or wing shape. Another possible explanation is found in the non-perfect alignment of the model, resulting in an initial angle-of-attack. A change from $c_{l,\max} = 1.0$ to $c_{l,\max} = 0.9$ results in the contour plot given in Figure 9.13b which fits the measured wind tunnel data. Taken the manufacturing process errors into account this 10% decrease of aerodynamic performance is likely and in the remaining of this chapter, $c_{l,\max} = 0.9$ is used for further data analysis.

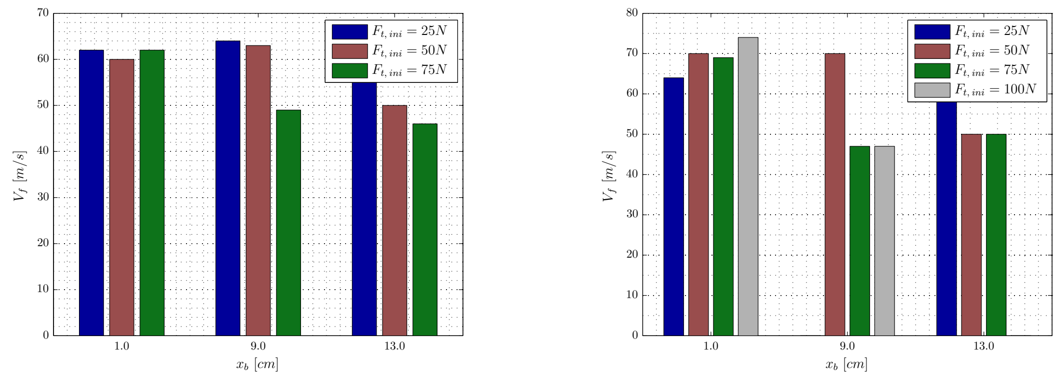
9.5.3 Bridles at wing tips

The wind tunnel results again follow the trend; more tether tension (and thus bridle tension) and the further behind the quarter chord line the bridles are attached, the lower the flutter speed. In some specific cases the flutter speed is considerably higher with respect to the unbridled flutter speed. The results are summarized in Figure 9.14.



(a) $c_{l,\max} = 1.0$ (b) $c_{l,\max} = 0.9$

Figure 9.13: Contour plot ASWINGb flutter speed and wind tunnel results, bridles attached halfway span



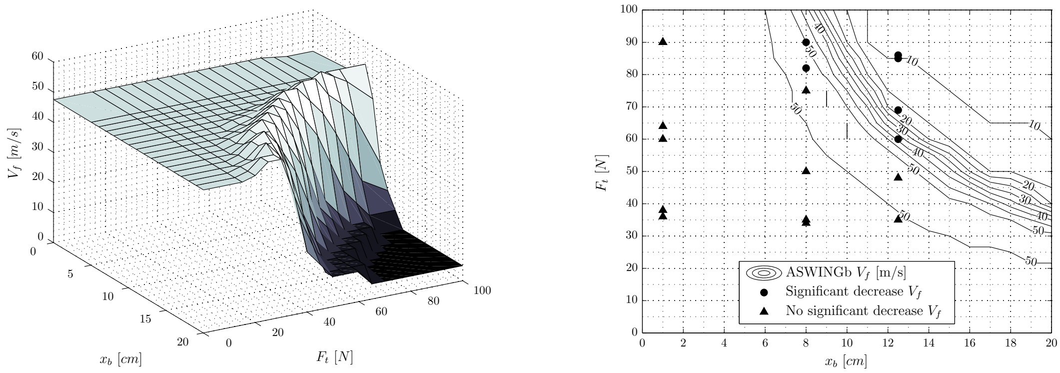
(a) $\beta_t = 0°$ (b) $\beta_t = 30°$

Figure 9.14: Wind tunnel results bridles at wing-tips

Comparison wind tunnel results with ASWINGb

The general flutter behaviour is equal to the 'bridles at half-way span'-case; the change of ASWINGb root locus plots with F_t and x_b is equal to the change given in Figure 9.11. The main difference is the magnitude of F_t and x_b at which the changes of root locus plots occur. For constant x_b, a lower F_t results in a change of root locus plot. The change of root locus plots is physically a change of aerodynamic state of the wing; a change from a non-stalled to a semi-stalled to a fully-stalled mode. For constant F_t and x_b, more wing twist is induced in the 'bridles at the wing tips'-case with respect to the 'bridles at half-way span'-case. The flutter speed as a function of F_t and x_b is given in Figure 9.15a.

In Figure 9.15b the wind tunnel results are plotted in a contour plot. Valuable information is given by the measurements at $x_b = 9cm$. Between $F_t = 75 - 90N$ the transition between non-stalled flutter and stalled flutter occurs. The measurements and the ASWINGb results correspond well except one wind tunnel measurement. ASWINGb determined no significant decrease of flutter speed, whereas the wind tunnel measurements show that a significant decrease of flutter speed has occurred. This measurement is relatively close to the edge at which stall flutter occurs. Measurement errors are the a plausible explanation of this discrepancy.



(a) Flutter speed as a function of bridle attachment point and tether force

(b) Contour plot ASWINGb flutter speed and wind tunnel results

Figure 9.15: Flutter speed as a function of bridle attachment point and tether force, bridles attached to wing tips

9.6 Conclusions

The wind tunnel test showed decreasing flutter speed with increasing tether force and increasing bridle x-attachment. ASWINGb calculated similar results and flutter speed decreased dramatically after the stall angle-of-attack has reached, which suggests stall flutter has occurred. Wind tunnel tests which include lift and drag measurement devices as well as angle-of-attack measurements could give valuable information about this phenomenon. Additionally flutter mode frequencies could be measured to validate the flutter mode frequencies determined with ASWINGb.

Despite the limitations of the wind tunnel test, this test validated that the tether-bridle addition is implemented correctly in ASWING. The equations are implemented appropriately and ASWINGb appropriately calculates deflections which could case divergence and influence the control effectiveness. Additionally ASWINGb calculates flutter modes, as a results of the tether-bridle system.

Part IV

Results

Chapter 10

Results M600 aero-elastic analysis

The goal of this chapter is to (1) present the results of the M600 aero-elastic analysis and (2)ti give design recommendations based on the aero-elastic analysis. The M600 input file as described in chapter 3 was used for the analysis. Three different aero-elastic phenomena are examined; torsional divergence, control effectiveness and reversal and flutter. For each aero-elastic phenomenon the effect of the main wing's stiffness parameters, the bridle attachment point and wing geometry is examined. In section 10.1 the M600 main wing torsional divergence is analysed, in section 10.2 aileron effectiveness and reversal and in section 10.3 the M600 susceptibility to flutter. In section 10.4 the conclusions are drawn.

10.1 Main wing torsional divergence

This section examines the divergence effects of several different main wing parameters. With an analytical expression, the torsional divergence speed is derived as a function of the main wing parameters. Next the effect of these parameters is examined in more detail. Also the effect of bridle location is examined. The results from ASWINGb are compared with the analytical expression.

For the M600, the maximum flight speed at cut-out wind speed is specified as $V_{\text{max}} = 95m/s$ (Vander Lind, 2013b). In the torsional divergence analysis a maximum flight speed of $V_{\text{IAS}} = 130/s$ is used.

10.1.1 Analytical torsional divergence speed

For an unswept, rectangular wing, the torsional divergence dynamic pressure is given by Jensen (2010) and Hulshoff (2011) as:

$$q_{\text{div}} = \begin{cases} \frac{K_\theta}{C_{L_\alpha} ecS} & \text{Hulshoff (2011)} \\ \frac{\pi^2}{4} \frac{GJ}{l^2 ec^2 C_{L_\alpha}} & \text{Jensen (2010)} \end{cases} \tag{10.1}$$

In this equation, K_θ is the torsional stiffness, C_{L_α} the lift curve slope, e the eccentricity factor, c the wing chord, S the wing surface and l the half-span length. The eccentricity factor is defined as the normalized distance between the aerodynamic center and the elastic axis:

$$e = \frac{c_{ac} - c_{ea}}{c} \tag{10.2}$$

From these equations follow that the torsional stiffness and elastic axis are linearly related to the torsional divergence dynamic pressure.

In a subsequent section the torsional stiffness parameter is used. This stiffness is defined from equation 10.1 as:

$$K_\theta = \pi^2 \frac{GJ}{b} \tag{10.3}$$

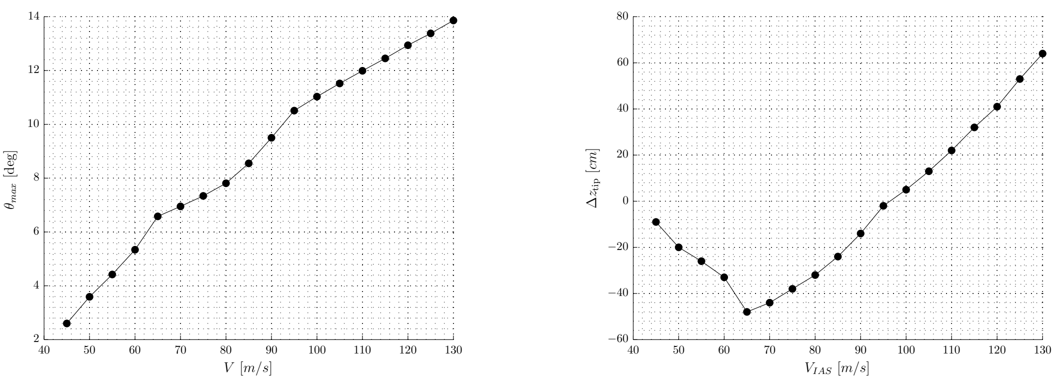
10.1.2 Benchmark run

In the benchmark case, the M600 geometrical, aerodynamic and structural parameters are used, as defined is chapter 3. For all operating points The aircraft is trimmed for straight, horizontal and steady flight. At each individual operating point the airspeed, the main wing flap deflections and the angle-of-attack are variable. Eight flaps ($f_1 - f_8$) are attached at the main wing, which can deflect independently of each other. The flaps at the wing tips (the ailerons) trim the aircraft in roll motion.

For all operating point the lift and drag forces are balanced with the tether force. In the low flight speed regime, the wing is flying with zero flap deflections and zero angle-of-attack. At about $V_{IAS} = 65m/s$, the tether force reaches its maximum allowable value, $F_{t,\max} = 250kN$. Increasing the flight speed further, with constant flap deflection and angle-of-attack, would increase the tether force to an unacceptable high value. Hence aerodynamic forces need to be decreased. First negative flap deflections de-power the main wing. The flap deflections range from from $\Delta f_1 - \Delta f_8 = 0.00°$ at $V_{IAS} = 65m/s$ to its minimum value $\Delta f_1 - \Delta f_8 = -30.00°$ at $V_{IAS} = 95m/s$. For higher flight speeds, the main wing is de-powered with negative angles-of-attack.

The secondary effect of negative flap deflections, is an upstream shift of aerodynamic center. This induces a nose-up pitching moment. Negative angles-of-attack of the main wing do not significantly influence the aerodynamic center position.

For this benchmark run, the airspeed versus wing twist angle and the airspeed versus tip deflection are given in respectively Figure 10.1a and 10.1b.



(a) Airspeed vs twist angle

(b) Airspeed vs tip deflection

Figure 10.1: Divergence benchmark run

A sudden increase in twist angle with a small increment of airspeed indicates that the flight speed is close to the divergence speed. From Figure 10.1a follows that divergence is not a problem for flight speeds up to $V_{IAS} = 130m/s$. However some interesting phenomena can be described in three different flight speed regimes:

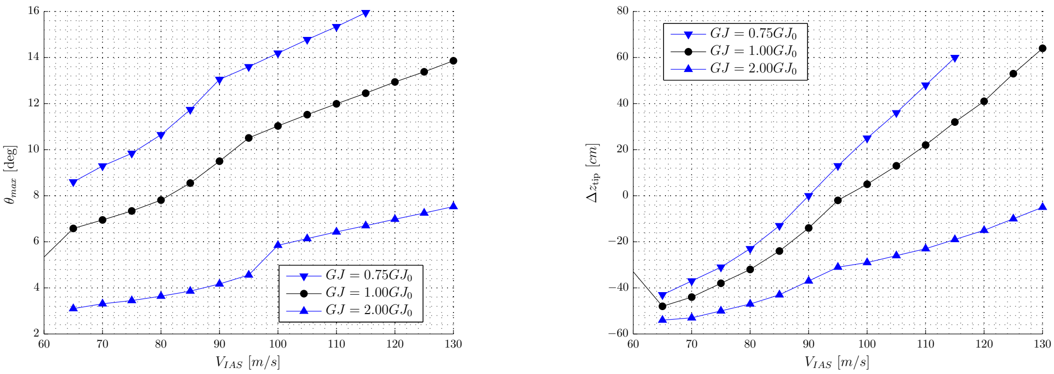
- **Flight regime 1:** $V_{IAS} \leq 65m/s$: flap deflections and angle-of-attack are equal to zero.
  - The bridle forces act at the bridle attachments at the wing, that is, at half-way span,
  - as a result of wing taper, the wing's net aerodynamic force acts closer to the root with respect to the bridle forces,
  - aerodynamic and tether forces increase quadratically with flight speed,
  - the balance of forces results in negative wing tip deflection, which increase with velocity,
  - the increase in force magnitudes increase the twist angle.
- **Flight regime 2:** $65m/s > V_{IAS} \leq 95m/s$: constant tether force and main wing flap deflections increase linearly with airspeed.
  - About constant tether and aerodynamic force magnitudes,
  - towards the tip, lower torsional stiffness and hence twist angles increase,
  - lift force distribution shift towards the tips,
  - less negative flap deflections,
  - flap deflections shift the aerodynamic center upstream,
  - increase of twist angle and tip deflection with flight speed.
- **Flight regime 3:** $V_{IAS} > 95m/s$: constant flap deflections at $\Delta f = \Delta f_{min} = -30.00°$, decreasing angle-of-attack with flight speed.
  - About constant tether and aerodynamic force magnitudes,
  - towards the tip, lower torsional stiffness and hence twist angles increase,
  - lift force distribution shifts towards the tips,
  - with increasing flight speed, the difference between the root and tip angle-of-attack is increasing,
  - increase of twist angle and tip deflection with flight speed.

In the remaining of this chapter, the benchmark run is denoted with subscript $()_0$.

10.1.3 Torsional stiffness effects

The effect of torsional stiffness on divergence speed is examined by scaling the benchmark torsional stiffness. The torsional stiffness is increased to 200% and decreased to 75% with respect to its benchmark value. The maximum twist angles as a function of flight speed is given in Figure 10.2.

None of the examined flight cases show a sudden increase in twist angle. Hence the flight speed is not close to its divergence speed. However the graph shows higher maximum twist angles for lower torsional stiffness and lower maximum twist angles for higher torsional stiffness. Compare the results with the perfect linear twisting model. In that case, the bending twist angles, of the $GJ = 0.75GJ_0$ and $GJ = 2.00GJ_0$ case, are respectively 133% and 50%, with respect to the benchmark case. The calculated results range from respectively 128 – 137% and 44 – 54%. Calculated twist angles are fairly close to the linear twisting model.

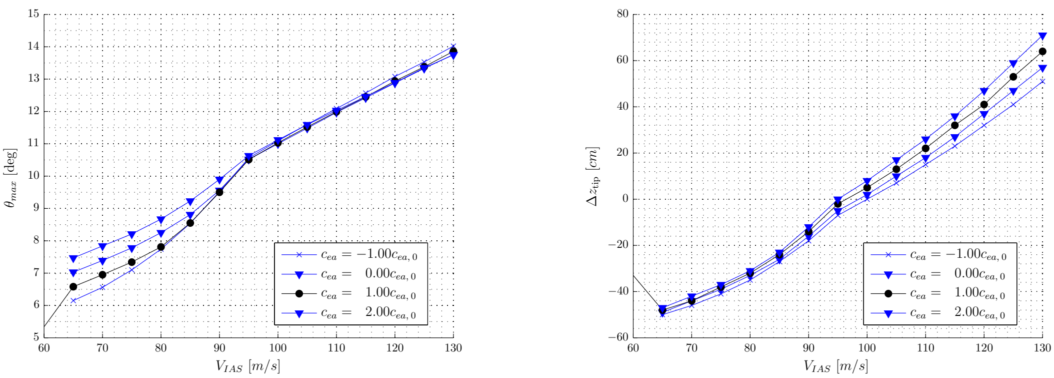


(a) GJ effect on twist angle (b) GJ effect on tip deflection

Figure 10.2: The effect of torsional stiffness with respect to the divergence speed; subscript $()_0$ denotes the benchmark run

10.1.4 Elastic axis effects

The benchmark elastic axis is scaled to determine its effects on divergence speed. The effect of the elastic axis c_{ea} position with respect to the maximum twist angle and wing tip deflection are given in Figures 10.3a and 10.3b. For a conventional (untethered) aircraft, an elastic axis shift further away from the aerodynamic center will increase the aerodynamic moment about the elastic axis and hence increase the twist angle. However for tethered flight, not only the distance from the aerodynamic center and the elastic axis is increased, but also the distance from the bridle force vector and the elastic axis. The magnitude the bridle forces are about equal to the aerodynamic forces, but opposite in direction. Hence the effect, of increasing twist angles with increasing c_{ea}, is diminished.



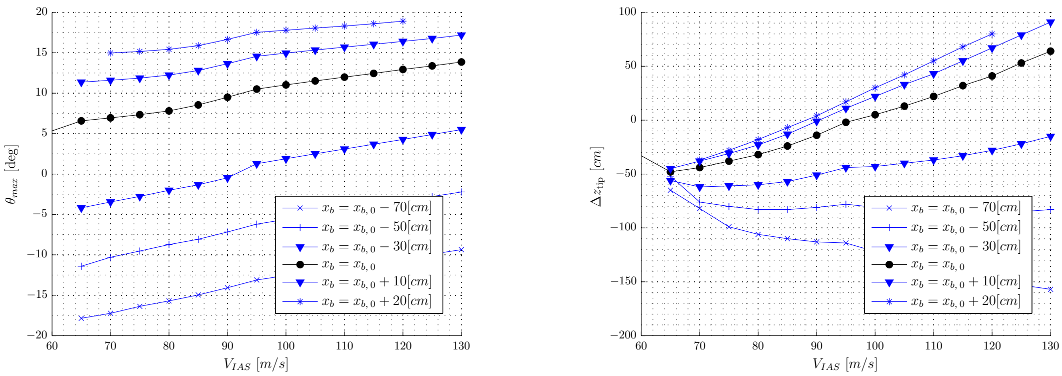
(a) c_{ea} effect on twist angle (b) c_{ea} effect on tip deflection

Figure 10.3: The effect of the location of the elastic axis with respect to the divergence speed; subscript $()_0$ denotes the benchmark run

10.1.5 Bridle attachment effects

The bridle attachment point along the chord determines moment arm between the bridle force and the elastic axis. For the benchmark run, the bridle is attached 6.91cm upstream of the main wing quarter-chord line. This bridle forces result in a nose-down pitching moment. In case the bridle attachment points are moved further downstream, higher twist angles are expected at equal force (and thus flight velocity). An opposite effect is expected in case the bridle points are attached further upstream. These effect are shown in Figure 10.4a. Large twist angles, up to $\theta_{max} = 19°$ are present in case the bridle attachment point is moved $20cm$ upstream. However there is no indication that the wing is close to divergence.

An upstream shift of bridle attachment position, results in a twist angle increase. This increase in real angle-of-attack increases aerodynamic forces and thus tip deflection. This effect is shown in Figure 10.4b.



(a) x_b effect on twist angle

(b) x_b effect on tip deflection

Figure 10.4: The effect of the location of bridle attachment point with respect to the divergence speed; subscript $()_0$ denotes the benchmark run

10.1.6 The effect of other main wing parameters

The effect of bending stiffness EI_{cc} and EI_{nn}, center of gravity position c_{cg} and n_{cg} and the elastic axis n_{ea} are examined. These parameters are outside the analytical divergence speed expression. As expected, these parameters have no significant effect on maximum twist angles and thus divergence speed. For these parameters, the flight speed versus maximum twist angle and flight speed versus tip deflection are shown in Appendix F.1.

10.2 Control effectiveness and reversal

Control reversal is the phenomenon that the effect of the deflection of an arbitrary control area is actually reversed. The horizontal and vertical tail plane are assumed perfectly rigid. Therefore the effect of rudder and elevator control reversal and effectiveness is left outside this analysis. The control surface efficiency (η_{cs}) is defined by Bisplinghoff et al. (1996) as the fraction of the control surface lift force with respect to the rigid control surface lift force:

$$\eta_{cs} = \frac{L}{L^R} \tag{10.4}$$

Vander Lind (2013b) specifies a minimum control effectiveness of 75% for all flight speeds. For the M600, the maximum flight speed at cut-out wind speed is specified as $V_{\max} = 95m/s$.

A downward deflection of the aileron increases the 'real' camber of the airfoil, increasing the lift coefficient and hence the lift per unit span over that part of the wing. To roll, one aileron is deflected upwards and the other is deflected downwards. This induces a rolling moment about the aircraft's x-axis. For the aileron deflected downwards, the produced moment tends to twist the airfoil more nose down, reducing the angle-of-attack and thereby the lift. The twisting moments of the wing increase with the square of the speed, whereas the elastic restoring torques remain constant with speed. Hence at higher speeds the rolling moments decrease up. The speed at which aileron deflections do not produce any net rolling moment is called the aileron reversal speed. Beyond this speed a deflection of ailerons results in a rolling moment opposite than that of a rigid wing (Bisplinghoff et al., 1996). According to Bisplinghoff et al. (1996) the wing's torsional stiffness should be increased in case the aileron reversal speed is lower than the operational flight speed.

10.2.1 Analytical control reversal speed

The control reversal dynamic pressure is given by Hulshoff (2011):

$$q_{\text{rev}} = -\frac{C_{L_\delta} K_\theta}{C_{L_\alpha} C_{M_{ac_\delta}} cS} \tag{10.5}$$

In this equation C_{L_δ} is the change of lift coefficient with flap deflection δ and $C_{M_{ac_\delta}}$ is the change of moment coefficient with flap deflection. Recap that the torsional stiffness K_θ is dependent on the torsional stiffness and the wingspan, see equation 10.3. In the next sections the effect of the parameters listed in equation 10.5 will be analysed in more detail.

10.2.2 Benchmark run

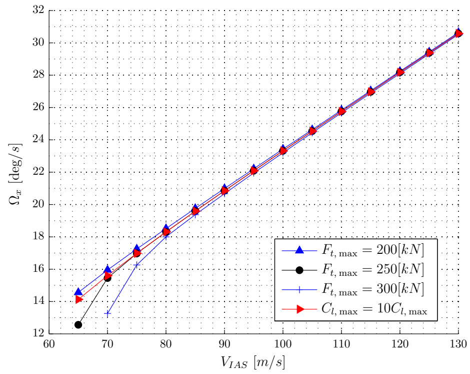
The M600 main wing is equipped with four control surfaces at each side. These control surfaces can be moved independently of each other. For this analysis the outer control surface is used as the aileron. In ASWING the aileron reversal speed is calculated for a fixed lift force, a fixed aileron deflection angle and a fixed rolling moment $\sum M_x = 0N \cdot m$. ASWING's solution converges to a steady state roll rate Ω_x to balance the moment. This approach is in consensus with the linear control, which states that the roll rate is linearly related to the airspeed:

$$\Omega_x = CV_{IAS} \tag{10.6}$$

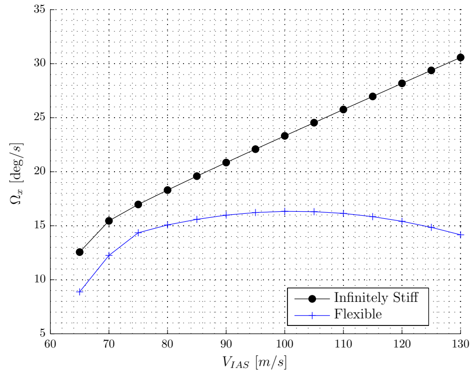
In this equation Ω_x is the roll rate, C is an arbitrary constant and V_{IAS} is the indicated air speed.

Initially this roll rate will increase linearly with speed, then round over and eventually go to zero at the aileron reversal speed (Drela, 2008b).

The roll rate increases linearly with the control surface lift force. Therefore the control efficiency is defined as the fraction of the flexible wing roll rate and the infinitely stiff roll rate. First the M600 rigid wing roll rate is determined. This roll rate is independent of lift and tether forces in case the aircraft is operating in the linear lift curve slope regime. In case the aircraft is operating with a lift coefficient relatively close to the maximum lift coefficient, a flap deflection results in a lower increment of lift with respect to the aircraft operating in the linear lift curve slope regime. This smaller increment in lift results in a smaller increment in roll rate. Tether forces are balanced with aerodynamic forces and vice versa. Hence, for relatively low flight speeds in combination with a high tether force, the aircraft is operating close to its maximum lift coefficient and the roll rate for given flap deflection is decreased. This effect is shown in Figure 10.5a. At flight speed $V_{IAS} = 70m/s$, compare the roll rate for user defined tether force $F_{t,\max} = 300N$, $F_{t,\max} = 250N$ and $F_{t,\max} = 200N$. The M600 maximum allowable tether force is equal to $F_{t,\max} = 250N$, hence this maximum tether force will be used in the remaining of this section.



(a) Ω_x vs V, infinitely stiff wing



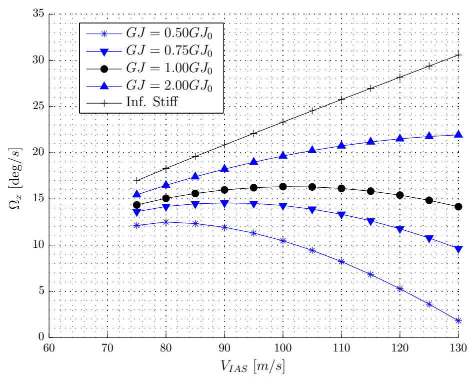
(b) Ω_x vs V benchmark

Figure 10.5: Benchmark results control reversal analysis

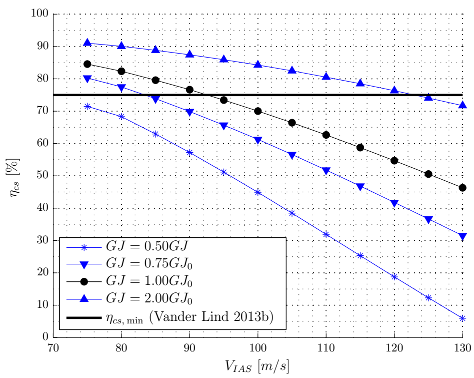
Figure 10.5b shows that the control efficiency is 72% at $V_{IAS} = 95m/s$, which is slightly lower than the requirements from Vander Lind (2013b).

10.2.3 Torsional stiffness effect

A more torsionally flexible wing is more susceptible to wing twist. Hence a decrease in torsional stiffness will decrease the control effectiveness. To visualize the torsional stiffness effect on control effectiveness, the torsional stiffness is decreased to 50% and increased to a maximum of 200% with respect to the benchmark GJ.



(a) Ω_x vs V, varying GJ



(b) η_{cs} vs V, varying GJ

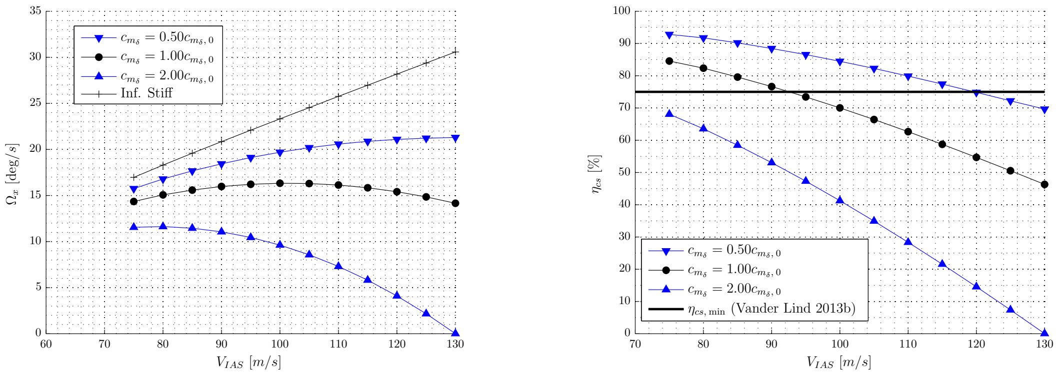
Figure 10.6: GJ effect on control effectiveness; subscript $()_0$ denotes the benchmark run

From Figure 10.6 follows an increase of torsional stiffness, increases the control effectiveness. This effect is more apparent with increasing velocity. At $V_{IAS} = 75m/s$, the control effectiveness for $GJ = 0.50GJ_0$ and $GJ = 2.00GJ_0$ are respectively 71.5% and 91.0%. At $V_{IAS} = 130m/s$, the control effectiveness for

$GJ = 0.50GJ$ has decreased to 5.9%, whereas the control effectiveness for the $GJ = 2.00GJ_0$ case is 71.7%. A slight increase of $GJ = 1.10GJ_0$ will satisfy the control effectiveness requirement, $\eta_{cs} = 75\%$ at $V_{IAS} = 75m/s$.

10.2.4 Flap aerodynamics effect

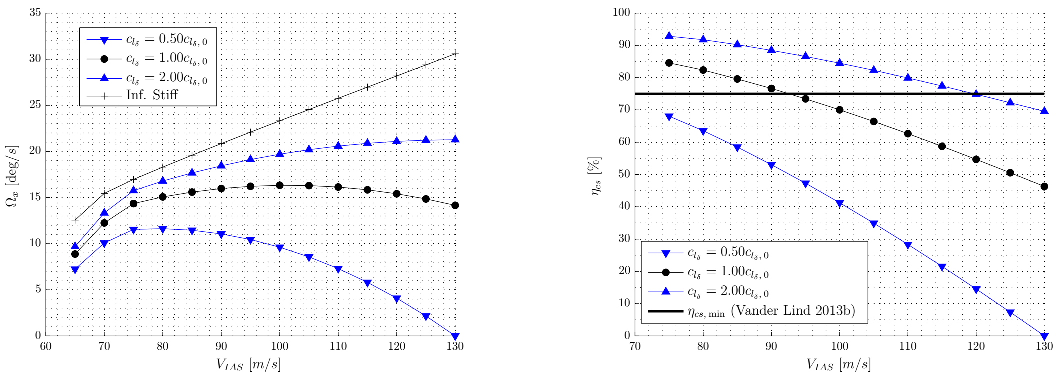
The pitching moment caused by the flap deflection is the main reason for wing twist which results in a loss of control effectiveness and eventually control reversal. A decrease of pitching moment should increase the control effectiveness and vice versa. This intuitive approach is in accordance with equation 10.5. The roll rate versus airspeed is plotted for varying flap moment coefficients in Figure 10.7.



(a) Ω_x vs V, varying $C_{m_{ac_\delta}}$

(b) η_{cs} vs V, varying $C_{m_{ac_\delta}}$

Figure 10.7: Flap $C_{m_{ac_\delta}}$ effect on control effectiveness; subscript $()_0$ denotes the benchmark run



(a) Ω_x vs V, varying C_{l_δ}

(b) η_{cs} vs V, varying C_{l_δ}

Figure 10.8: Flap C_{l_δ} effect on control effectiveness; subscript $()_0$ denotes the benchmark run

For a flap moment coefficient which is equal to twice the benchmark flap moment coefficient the control reversal speed is about $V_{\text{rev}} = 130m/s$. That is, control effectiveness is 0%. A decrease to $C_{m_{ac_\delta}} = 0.50C_{m_{ac_\delta}}$ increases the control effectiveness from 46.3% to 67.7% at $V_{IAS} = 130m/s$.

In an ideal case a downward flap deflection solely increases the lift forces at the aileron, with no resulting pitching moment which could twist the wing. Hence an increase of resultant lift force for given flap deflection should increase aileron efficiency. For an infinitely stiff wing, the roll rate increases linearly with aileron lift force and hence with C_{l_δ}. To analyse the roll rate versus airspeed, for varying C_{l_δ}, the flap deflection is scaled with C_{l_δ}. That is; in case C_{l_δ} is increased with a factor 2, the flap deflection is divided with a factor 2 and vice versa. The resulting roll rate versus airspeed for varying C_{l_δ} is shown in Figure 10.8.

Note the similarity in control effectiveness effect for $C_{m_{ac_\delta}}$ and C_{l_δ}. Those two Figures suggest what also equation 10.5 suggests; the factor $\left(C_{l_\delta}/C_{m_{ac_\delta}}\right)$ is most important. An increase of C_{l_δ} with a factor 2 has the same effect as dividing $C_{m_{ac_\delta}}$ with a factor 2. That is; for a given $C_{m_{ac_\delta}}$ and flap deflection, a certain pitching moment twists the wing and thereby decreases the angle-of-attack. In case C_{l_δ} is increased, a smaller flap deflection results in equal lift increment and roll rate. Keeping $C_{m_{ac_\delta}}$ constant, the smaller flap deflection results in a smaller pitching moment, wing twist and thus change of angle-of-attack.

An optimization of flap aerodynamics in the order $\left(C_{l_\delta}/C_{m_{ac_\delta}}\right) = 1.10\left(C_{l_\delta}/C_{m_{ac_\delta}}\right)_0$, will satisfy the control effectiveness requirements.

10.2.5 The effect of other main wing parameters

From the theory for an unswept, straight rectangular wing follows that control reversal and effectiveness is solely dependent on aerodynamic properties and the torsional stiffness GJ. However the M600 is a swept forward, tapered wing at which bridles are attached. Therefore the effect of the center of gravity c_{cg} and n_{cg}, bending stiffness, EI_{cc} and EI_{nn} and bridle attachment point x_b is analysed as well. It is concluded that these parameters only have a minor effect and do not significantly influence the aileron control effectiveness. See Appendix F.2 for more details.

10.3 Flutter

The flutter point is defined by Hulshoff (2011) as the point at which *the self-sustained oscillation transitions from convergent to divergent motion*. In general flutter occurs when the frequencies of two modes interact with each other[1]. Many modes exist which could possibly interact and start a flutter mode. The description and analytical derivation of all modes is outside the scope of this thesis. However some valuable insight is gained with the classical bending-torsion flutter mode. This example is used to explain the effect of certain parameters on flutter behaviour.

10.3.1 Analytical torsion-bending flutter mode

In the classical torsion-bending flutter mode, the twisting mode is coupled to the out-of-plane bending mode. First examine stable flight: aerodynamic torsional forces, twist the wing. This twist increases the real angle-of-attack and the wing gains lift which bends the wing upward. This upward motion decreases the effective angle-of-attack and stabilizes the wing.

In an unstable torsion-bending mode, the phases of one of the modes is shifted with respect to the other mode. The twist induced angle-of-attack occurs simultaneously with the wing downward motion, which increases the angle-of-attack further. This dynamic motion, increases its amplitude at each oscillation and is dynamically unstable.

[1] Hulshoff (2011) lists some control surface, single degree of freedom flutter modes. These modes are in the transonic flight regime and not expected in this analysis.

For a straight wing with a simple, uncambered airfoil, the equation of motion for this two degree of freedom system is given by Hulshoff (2011) as:

$$\begin{bmatrix} m & mx_\theta b \\ mx_\theta b & I_\theta \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{\theta} \end{Bmatrix} + \left(\begin{bmatrix} K_h & 0 \\ 0 & K_\theta \end{bmatrix} - q \begin{bmatrix} 0 & -SC_{L_\alpha} \\ 0 & 2S\,ebC_{L_\alpha} \end{bmatrix} \right) \begin{Bmatrix} x \\ \theta \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \tag{10.7}$$

In this equation m is the mass, x_θ is the displacement of the center of gravity from the elastic axis, I_θ is the mass moment of inertia with respect to the elastic axis, $\ddot{x}$ is the linear acceleration, $\ddot{\theta}$ is the angular acceleration, K_h is the bending stiffness and K_θ is the torsional stiffness.

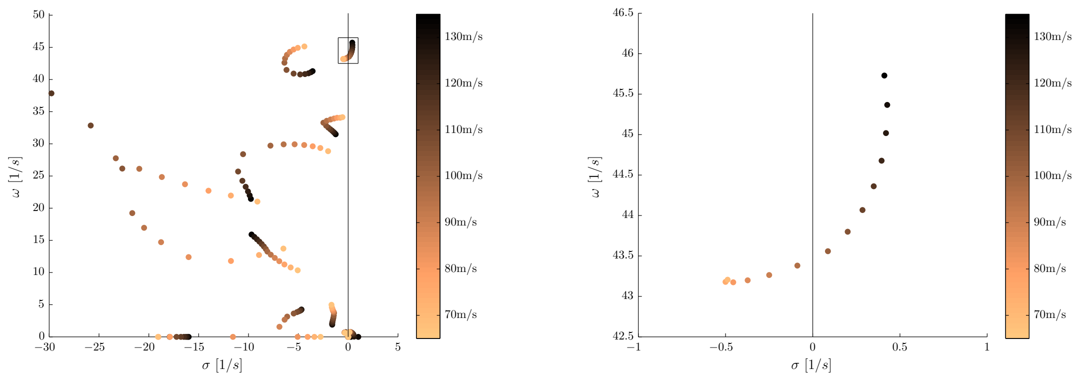
From this simplified equation for one mode follows that the mass distribution, the mass inertia, the torsional stiffness, the bending stiffness, the center of gravity positions, the elastic axis position, the wing's geometrical properties and the wing's aerodynamic properties all influence the flutter behaviour.

10.3.2 Benchmark run flutter

In ASWING the effect of flutter is investigated with root locus plots. The two axis of the root locus plot define (1) the growth rate σ and (2) the frequency ω. A non-zero frequency and a negative growth rate indicate positive, converging damping and thus a stable mode. A positive growth rate in combination with a non-zero frequency indicate unstable, diverging mode.

Vander Lind (2013b) specifies a minimum apparent wind speed at which flutter is predicted $V_{\text{flutter}} \geq 120m/s$.

For the benchmark case, first the wing is trimmed with the same constraints as in the divergence case. The benchmark root locus plot is given in Figure 10.9a. The root locus plot shows two different unstable modes; one unstable motions close to $\omega < 2Hz$ and an unstable mode at about $\omega \approx 42Hz$. The unstable mode at $\omega < 2Hz$ is an unstable flight mode and not a structural flutter mode. The analysis of unstable flight modes is not a structural, but a control problem and therefore left outside the scope of this research. However, the unstable mode at $\omega \approx 42Hz$ is an unstable flutter mode. To examine this mode in more detail, Figure 10.9b zooms into this part of the root locus plot. From this Figure follows that flutter strikes at an unacceptable flight speed, at about $V_{IAS} = V_{flutter} \approx 90m/s$. This flutter mode is a combination of in-plane-bending, out-of-plane bending and torsion modes.



(a) Root locus benchmark

(b) Zoom in interesting points root locus benchmark

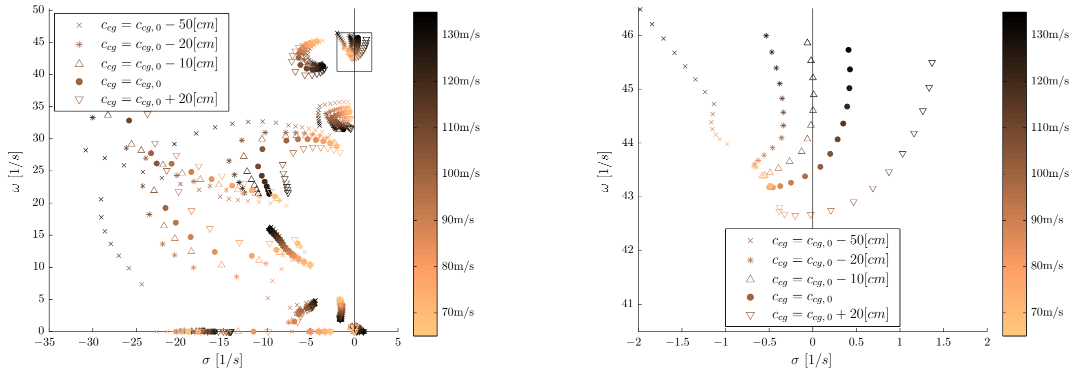
Figure 10.9: Flutter benchmark run

10.3.3 Center of gravity effect

The immediate effect on the overall flutter speed does not follow directly from equation 10.7. However this equation indicates that the center of gravity position can have a significant effect on the flutter behaviour.

c_{cg} effect

The root locus plots with shifting center of gravity are given in Figure 10.10a. From these Figures follows that an upstream center of gravity shift will increase the stability of the wing; all flutter modes move toward a more negative growth rate. To indicate the effect of the center of gravity shift on the mode which was unstable in the benchmark run, Figure 10.10b zooms in on the interesting part. From this Figure follows that this mode can be stabilized with a $10cm$ upstream center of gravity shift. The stability can be increased further by increasing the center of gravity further upstream. Moving the center of gravity downstream will decrease the stability of the system.



(a) Root locus with varying c_{cg}

(b) Zoom in interesting points root locus with varying c_{cg}

Figure 10.10: c_{cg} effect on flutte; a shift in negative c direction indicates an upstream shift and vice versa ; subscript $()_0$ denotes the benchmark run

n_{cg} effect

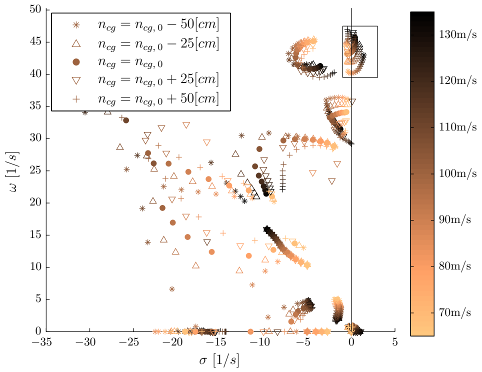
The root locus plot with changing n_{cg} is given in Figure 10.11a, zooming in at the interesting part in Figure 10.11b. These Figures suggest that an upward shift of the center of gravity has a stabilizing effect. At a point between $25cm$ and $50cm$, above the benchmark center of gravity location, the motion is stabilized. However the wing is only about $20cm$ thick.

10.3.4 Elastic axis effect

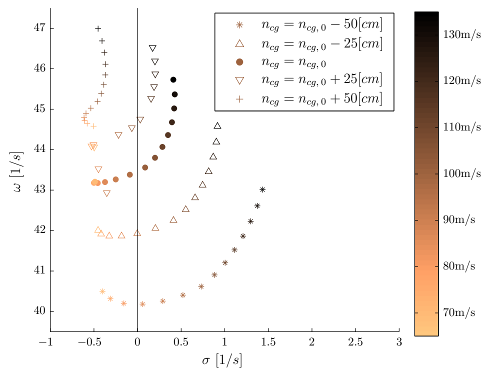
The position of the elastic axis can have a significant effect on the flutter behaviour as shown in equation 10.7. As with the center of gravity the quantitative effect on the overall flutter behaviour does not follow from this equation, but this equation suggest that the elastic axis position can influence the flutter behaviour.

c_{ea} effect

The overall root locus plot with varying c_{ea} is given in Figure 10.12a and Figure 10.12b zooms in to the most interesting part; the part of the plot with an unstable flutter mode. The effect of a $20cm$ shift either



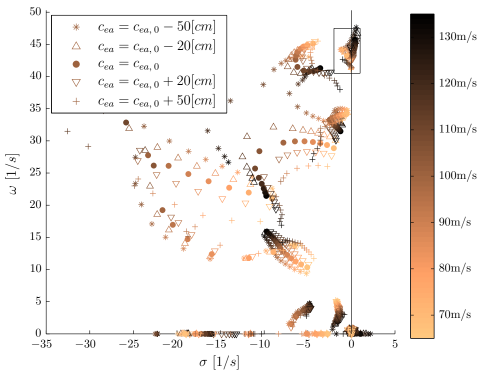
(a) Root locus with varying n_{cg}



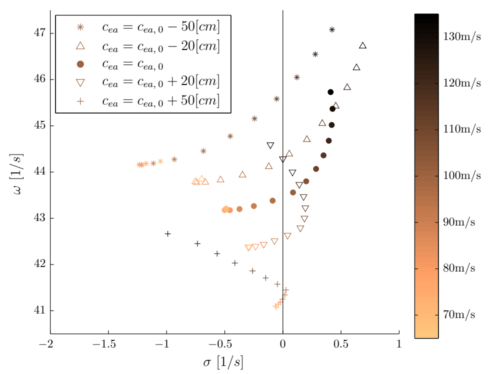
(b) Zoom in interesting points root locus with varying n_{cg}

Figure 10.11: n_{cg} effect on flutter; subscript $()_0$ denotes the benchmark run

upstream or downstream is not significant. A $50cm$ shift upstream, significantly increases the stability and thus flutter speed.



(a) Root locus with varying c_{ea}



(b) Zoom in interesting points root locus with varying c_{ea}

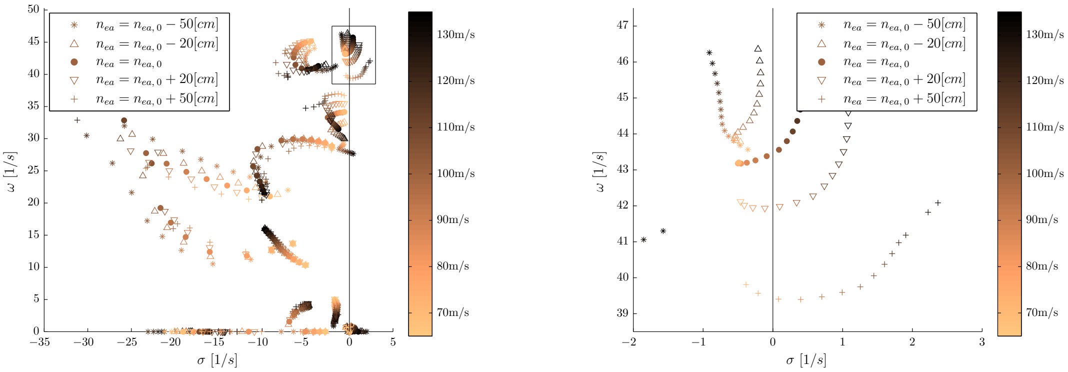
Figure 10.12: c_{ea} effect on flutter, a shift in negative c direction indicates an upstream shift and vice versa; subscript $()_0$ denotes the benchmark run

n_{ea} effect

The overall root locus plots with varying n_{ea} are given in Figure 10.13. Note the similarities with Figure 10.11 at which the center of gravity n position is varied. An elastic axis n shift, is similar with respect to a center of gravity n shift, but has an opposite effect; a downward n_{ea} shift increases flutter speed and vice versa.

10.3.5 Bending stiffness effect

The bending stiffness influences the degree of bending for a given load case and from equation 10.7 follows that bending stiffness has an influence on the classical bending-torsion flutter example and it expected to



(a) Root locus with varying n_{ea}

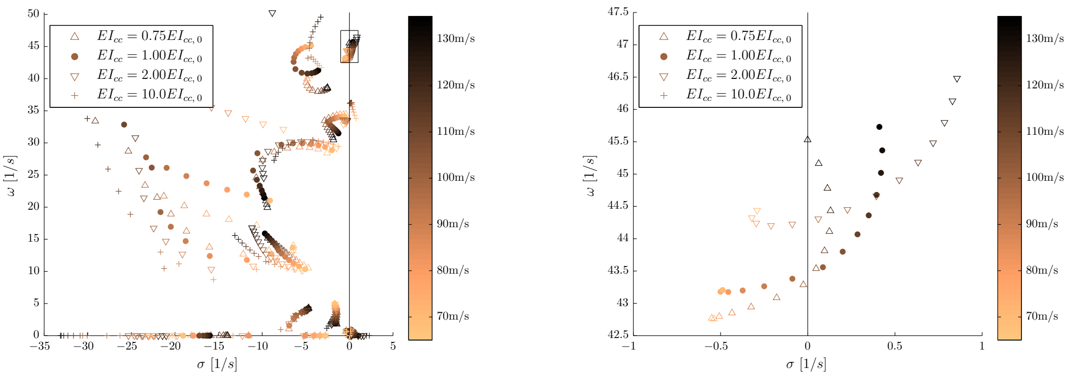
(b) Zoom in interesting points root locus with varying n_{ea}

Figure 10.13: n_{ea} effect on flutter; subscript $()_0$ denotes the benchmark run

influence the other modes as well.

EI_{cc} effect

The effect of bending stiffness about the c axis on the flutter behaviour is given in Figures 10.14a and 10.14b. For increasing EI_{cc}, the wing is destabilized and flutter speed is decreasing. In case EI_{cc} is increased by a factor 10, flutter occurs already at very low flight speeds. This behaviour is counter intuitive at first, but can be explained by the nature of the flutter mode. The flutter mode is a combination of in-plane-bending, out-of-plane-bending and a torsion mode. An increase of EI_{cc} will decrease effect of the the out-of-plane mode. The results suggest that this mode damps the coupling between the in-plane-bending and torsion mode and hence an increase of EI_{cc} will decrease this damping.



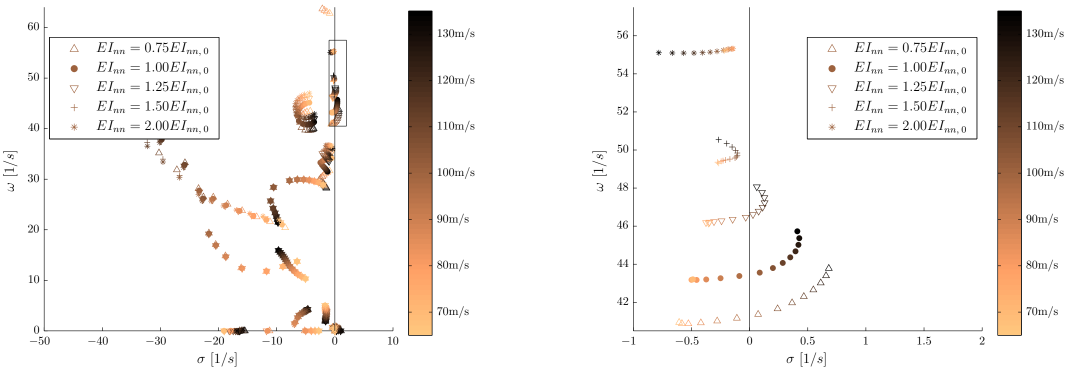
(a) Root locus with varying EI_{cc}

(b) Zoom in interesting points root locus with varying EI_{nn}

Figure 10.14: EI_{cc} effect on flutter; subscript $()_0$ denotes the benchmark run

EI_{nn} effect

The effect of bending stiffness about the n axis on the flutter behaviour is given in Figures 10.15a and 10.15b. With increasing stiffness EI_{nn} the flutter speed is increased, which is intuitively correct. An increase of 50% with respect to the benchmark EI_{nn} vanishes the susceptibility to flutter to at least $V_{IAS} = 130m/s$. Although the data is not available, the Figures suggest that the critical flutter mode is stabilized for higher flight speeds as well. In case $EI_{nn} \geq 1.50EI_{cc,0}$, the critical mode growth rate is decreasing with increasing flight speeds.



(a) Root locus with varying EI_{nn}

(b) Zoom in interesting points root locus with varying EI_{nn}

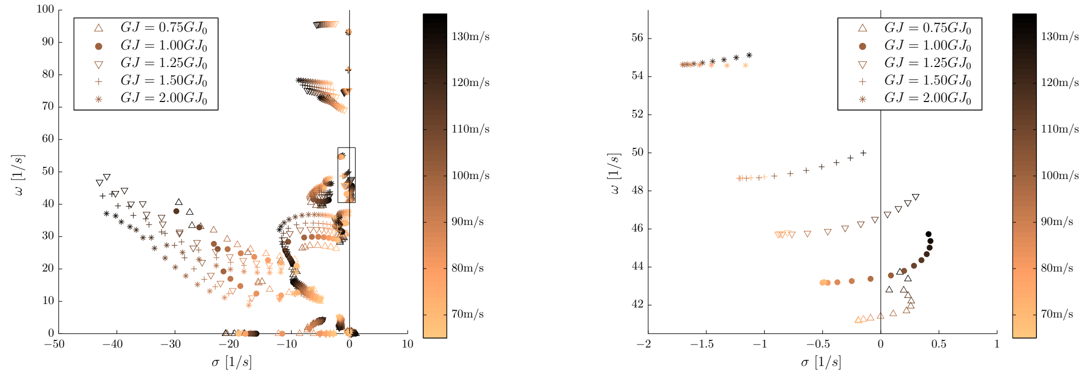
Figure 10.15: EI_{nn} effect on flutter; subscript $()_0$ denotes the benchmark run

10.3.6 Torsional stiffness effect

As with the torsional divergence and control effectiveness, the torsional stiffness (GJ) usually greatly influence the flutter behaviour. This is in consensus with the classical torsion-bending flutter equation 10.7. From Figure 10.16b follows that an increase of torsional stiffness increases the flutter speed for the critical mode. An increase with 50% with respect to the benchmark GJ increases the flutter speed to at least $130m/s$. Despite the lack of data Figure 10.16b suggest that the flutter speed is about $140m/s$ for this case.

10.3.7 Flap deflection effect

At high flight speeds, negative flap deflections, δ_F, depower the main wing. See section 10.1.2 for more details. Next to depowering the wing, negative flap deflections shift the aerodynamic center aft and hence influence flutter behaviour. The root locus plot for zero negative flap deflections and the benchmark case are shown in Figure 10.18. Recall from section 10.1.2 that flap deflections can be as low as $\delta_F = -30°$. This Figure shows two modes which are highly influenced by the flap deflections. One mode at about $\omega = 32Hz$ and another mode at about $\omega = 45Hz$. Figures 10.18a and 10.18b zoom in these two modes. Figure 10.18a shows a shift from a stable mode with flap deflections to an unstable flutter mode without flap deflections. Figure 10.18b shows an unstable mode with flap deflections to a stable mode without flap deflections. The nature of the critical mode has changed, but the magnitude of the flutter speed remain unchanged at $V_{flutter} = 90m/s$.



(a) Root locus with varying GJ

(b) Zoom in interesting points root locus with varying GJ

Figure 10.16: GJ effect on flutter; subscript $()_0$ denotes the benchmark run

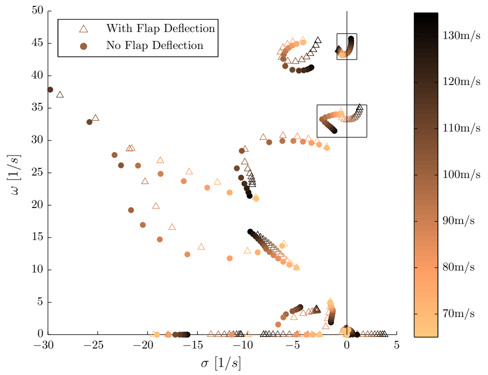


Figure 10.17: δ_F effect on flutter

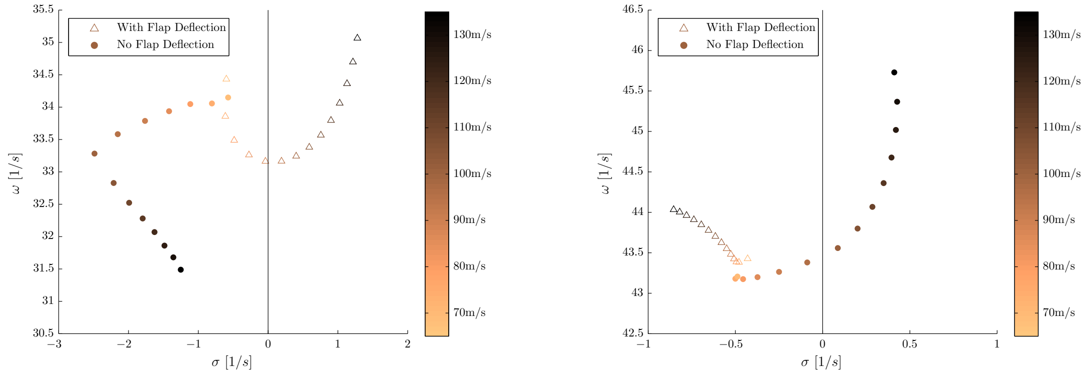
10.3.8 Bridle attachment effects

The bridle attachment point in x-direction (x_b), determines the moment arm between the aerodynamic center and the bridle forces. The bridle forces are at the same order of magnitude with respect to the aerodynamic forces and it is likely that a change of moment arm influences the wing flutter behaviour. The bridle attachment point is shifted up to $70cm$ upstream and $20cm$ downstream with respect to the benchmark case. Figure 10.19 shows two interesting areas at which Figures 10.20a and 10.20b zoom in.

The mode given in Figure 10.20a is stabilized with an upstream shift of bridle attachment point, whereas the mode given in Figure 10.20b is destabilized with an upstream shift of bridle attachment point. With a $50cm$ upstream shift of x_b with respect to the benchmark $x_{b,0}$, the mode given in Figure 10.20b is still stable and the mode given in Figure 10.20a gained stability up to $V_{IAS} \approx 110m/s$. Hence a shift of bridle attachment location could stabilize the wing.

10.3.9 Fuselage stiffness effect

In the previous simulations the fuselage was simplified as perfectly rigid, because of lack of stiffness information. However a flexible fuselage could influence flutter behaviour, because (1) the vertical and horizontal tail mass influence the system mass and mass inertia matrix and (2) the aerodynamic effect of



(a) Zoom in interesting area root locus with varying δ_F (1)

(b) Zoom in interesting area root locus with varying δ_F (2)

Figure 10.18: Zoom in interesting area root locus with varying δ_F

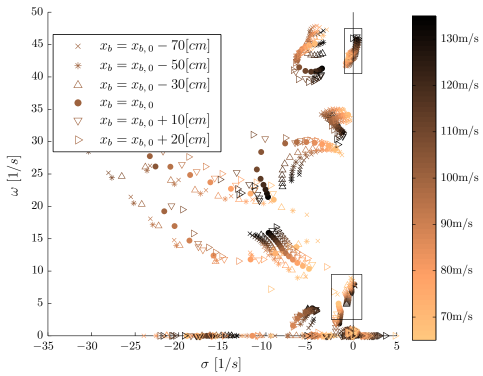


Figure 10.19: x_b effect on flutter

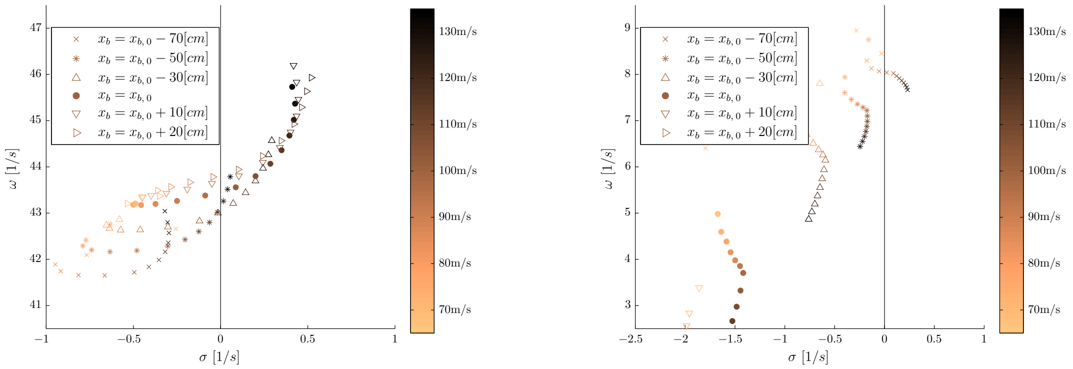
the vertical and horizontal tail influence the overall flight performance and could induce Body Freedom Flutter (BBF) at which a flight mode interacts with a structural mode.

The fuselage stiffness is decreased from infinitely stiff to a minimum acceptable stiffness parameter. That is, the minimum fuselage stiffness at which the flutter speed is not influenced by fuselage flexibility.

The fuselage thickness is assumed constant and the fuselage radius is assumed to vary linearly from its maximum at the wing attachment to its minimum at the tail attachment. From thin wall theory follows that the stiffness is proportional to:

$$EI_{cc} = EI_{nn} = \pi r^3 t \tag{10.8}$$

In this equation r is the fuselage radius and t is the material thickness. The fuselage-tail connection and wing-tail connection fuselage radii are respectively $28.11cm$ and $16.87cm$, hence the stiffness at the fuselage-tail connection as a function of the stiffness at the wing-tail connection is calculated as:



(a) Zoom in interesting area root locus with varying x_b (1)

(b) Zoom in interesting area root locus with varying x_b (2)

Figure 10.20: Zoom in interesting area root locus with varying x_b; subscript $()_0$ denotes the benchmark run

$$EI_{f,\,\text{tail}} = \left(\frac{r_{f,\,\text{tail}}}{r_{f,\,\text{wing}}}\right)^3 EI_{f,\,\text{wing}} = 0.216 EI_{f,\,\text{wing}} \tag{10.9}$$

In this equation $EI_{f,\,\text{tail}}$ is the fuselage bending stiffness at the fuselage-tail connection, $EI_{f,\,\text{wing}}$ is the fuselage bending stiffness at the fuselage-wing connection, $r_{f,\,\text{tail}}$ is the fuselage radius at the fuselage-tail conncetion, $r_{f,\,\text{wing}}$ is the fuselage radius at the the fuselage-wing connection.

The maximum and minimum root stiffnesses applied in this analysis are respectively $EI_{f,\,\text{wing}} = 10^8 N \cdot m^2$ and $EI_{f,\,\text{wing}} = 10^6 N \cdot m^2$. The root locus plot with varying fuselage stiffness is given in Figure 10.21. Four interesting areas are distinguished and zoomed in in Figures 10.22a - 10.22d.

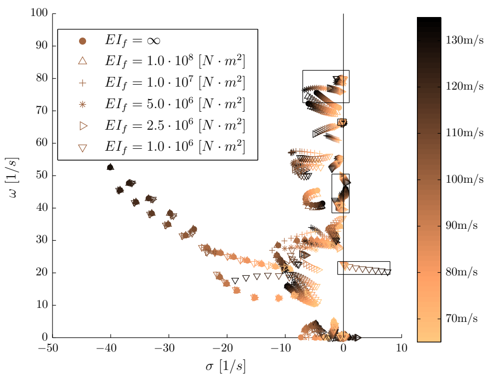


Figure 10.21: EI_f effect on flutter

The mode given in Figure 10.22c shows an interesting phenomenon; for an infinite stiff fuselage the flutter speed is about $V_{flutter} \approx 90m/s$. With increasing flexibility the mode gains stability and is completely stabilized in case the fuselage stiffness at the fuselage-wing connection is as low as $EI_{f,\,\text{wing}} = 5 \cdot 10^6 N \cdot m^2$. Decreasing the stiffness below $EI_{f,\,\text{wing}} = 5 \cdot 10^6 N \cdot m^2$ decreases the stability of this mode again. Hence there is only a small range of fuselage stiffnesses at which this mode is stabilized.

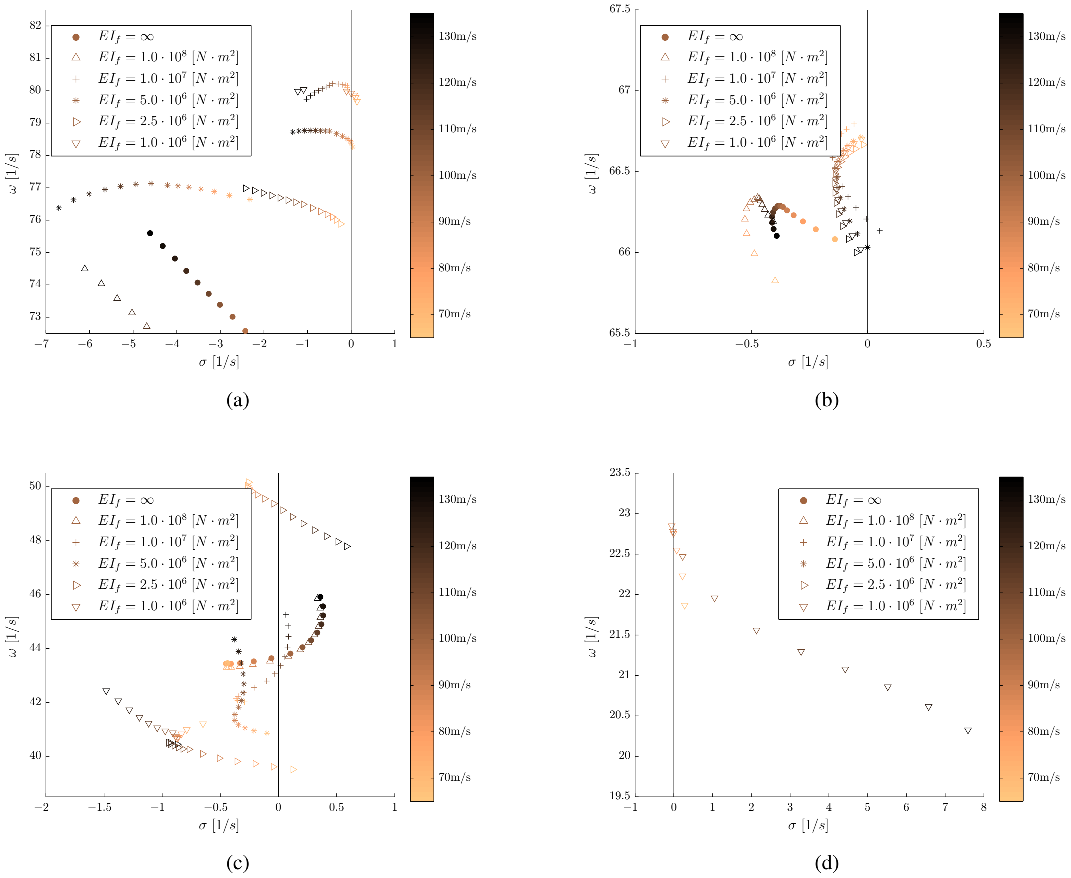


Figure 10.22: Zoom in interesting area root locus with varying EI_f

Figures 10.22a, 10.22b and 10.22d show flutter modes, which are stable in the rigid fuselage case. With decreasing fuselage stiffness, the stability of these modes is decreasing as well. A fuselage stiffness of at least $EI_{f,\text{ wing}} \geq 10^7$ stabilizes these modes.

10.3.10 The effect of other main wing and tether parameters

The flutter behaviour for the tether stiffness, k_t, the effective tether drag area $(C_DA)_t$, bridle attachment point z_b the mass inertia $\iota_{cc}g$ and $\iota_{nn}g$ and engine settings Δ_E is investigated as well. From the analyses follow that these parameters have no significant effect on flutter behaviour. The root locus plots with these varying parameters can be found in Appendix F.2.

10.4 Conclusions

The M600 divergence speed is beyond the flight speed regime. Destabilizing the wing with a decrease of torsional stiffness or a shift of the elastic axis does not results in divergence. Hence divergence is not a critical mode for the M600.

Control reversal is not a critical mode for the M600. Even in case the torsional stiffness or the ratio $\left(C_{l_\delta}/C_{m_{ac_\delta}}\right)$ is decreased with 50%, the control reversal speed is still $V_{\text{rev}} = 130 m/s$.

However the control effectiveness is critical. A slight increase in torsional stiffness of $GJ \approx 1.10 GJ_0$ will satisfy this requirement. Increasing the torsional stiffness to gain control effectiveness is in consensus with Hulshoff (2011) and Drela (2008b). Alternatively the flap aerodynamics could be optimized for higher $\left(C_{l_\delta}/C_{m_{ac_\delta}}\right)$.

The analyses showed an unacceptably low flutter speed. The critical flutter mode is an in-plane, out-of-plane, torsion mode, which strikes at $90 m/s$. Several approaches are feasible to increase the flutter speed. The flutter speed can be increased by an:

1. upstream center of gravity shift. This approach is in consensus with Jensen (2010). A $10 cm$ shift upstream will stabilize the critical flutter mode.
2. increase of in-plane-bending stiffness. An increase of $EI_{nn} = 1.50 EI_{nn,0}$ stabilizes the critical mode; this modes gains negative growth rate with increasing velocity.
3. increase of torsional stiffness. An increase of $GJ = 1.50 GJ_0$ stabilizes the wing up to about $V_{\text{flutter}} = 140 m/s$
4. upstream shift of bridle attachment point. A $50 cm$ shift upstream will stabilize the wing up to a flutter speed $V_{\text{flutter}} = 110 m/s$

The effect of tether drag and spring constant on flutter behaviour is examined as well. In the analysis for the M600 these parameters do not significantly influence the flutter speed.

Tail flutter does not influence the flutter speed in case the bending stiffness at the wing-tail connection is $EI_{f,\text{wing}} \geq 10^7$.

Chapter 11

Conclusions and recommendations

11.1 Conclusions

The objective of this research is to analyse the aero-elastic behaviour of the next generation airborne wind turbine designed by Makani Power. The M600, which is currently in the planning stage, uses a tethered wing of 28m span with wing mounted small on-board turbines to harvest the kinetic energy of the relative wind during crosswind flight manoeuvres. Due to the large size and the lightweight constructing, fluid-structure interaction with considerable deformation can play a decisive role.

The analysis is based on computational simulation as well as wind tunnel measurements. An existing software framework, ASWING, is extended with a sub-module for the tether and bridle line system. This module is based on a straight, axial stretchy tether with user defined mass, aerodynamic drag properties and spring constant. The bridle lines are assumed massless and perfectly rigid. The tether and bridle forces are dependent on the wing flexibility as well as its orientation in space. The dynamic aero-elastic response (flutter) is investigated by means of an Eigenmode analysis. In this analysis, the initial state vector and its derivatives influence the flutter modes. To include the tether-bridle system in the eigenvalue problem, the state derivatives are added. The wind tunnel test at TU Delft validated the tether-bridle system induced static elasticity effects as well as the flutter modes.

The M600 torsional divergence, aileron reversal and effectiveness, and flutter behaviour are analysed for flight speeds to to 130m/s. In this first design iteration, the turn rate is left outside the analysis and hence constant apparent wind velocity over the span is assumed. The maximum flight speed at cut-out wind speed is equal to 95m/s. The requirements state a minimum apparent wind speed, at which flutter is predicted, of 120m/s. From these analyses follow that,

- divergence and aileron reversal are no critical modes. Not even in case of (1) a 50% decrease in torsional stiffness, (2) a 20cm upstream or downstream shift of elastic axis, or (3) a 50% decrease in the aileron, lift-moment ratio.
- the minimum 75% control efficiency is reached at 92m/s. A 10% increase in torsional stiffness or a 10% increase in lift-moment ratio with aileron deflection, will satisfy this requirement.
- the predicted flutter speed is equal to 90m/s. This susceptibility can be reduced by various design and construction measurements such as, (1) a 50% increase in torsional stiffness, (2) a 50% increase in in-plane-bending stiffness or (3) a 10cm upstream shift of the center of gravity.

The M600 carbon fibre design allows for stiffness adjustments with fibre lay-up. A torsional stiffness increase can be realized with the addition of bi-directional carbon fibre in the 45/45 orientation. The in-plane-bending stiffness can be increased with the addition of extra fibres oriented in the chord direction. The center of gravity shift can be realized by adding more material in the nose of the wing, for example cabling. Alternatively the position of the motor pylons could be shifted further upstream.

The effects of the tether-bridle system on the static and aero-elastic effects are summarized as:

- the tether aerodynamic drag and gravity loads are respectively 3.3% and 2.2% with respect to the tether force and hence do not significantly contribute to the M600's static and dynamic aero-elastic behaviour.
- it is likely that the tether spring constant could induce tether-wing modes. However in the specific M600 case the effect of the spring constant on flutter behaviour is minor.
- the chord-wise position of the bridle-wing attachment location,
  - is linearly related to the distributed and maximum wing twist angles. These angles influence the real angles-of-attack, subsequently the aerodynamic forces and finally the wing deflections. A change of chord-wise attachment is useful is case maximum wing twist angle or tip deflection are a serious design consideration.
  - does not significantly influence the aileron control effectiveness,
  - can change the nature of the Eigenmode responses. In the M600 case, an unstable flutter mode transitioned to a stable mode and vice versa. For a $50cm$ upstream shift, the flutters speed was increased from $90m/s$ to $110m/s$.

The developed aero-elastic modelling program for AWE systems is one of the first to determine the tether-bridle effects on aero-elastic behaviour of a rigid with AWT. The program demonstrated that the tether-bridle system significantly contributes to the wing twist and bending behaviour in case of static aero-elasticity. Additionally the bridle attachment position significantly influences the dynamic aero-elastic behaviour.

11.2 Recommendations for future research

Over the last decade, the average size of wind turbines has doubled. In the future, it is expected that wind turbine size will increase further. The airborne wind energy industry seems to follow this trend. In example, in the future, Makani aims to develop a $5MW$ system with a $65m$ wing span. With this ever increasing size, a realistic aero-elastic modelling tool will be ever more important. The current program can be improved by increasing its validity, by increasing its applicability and by increasing its computational accuracy. The aero-elastic analysis of the M600 can be improved by including more accurate structural design parameters and flight characteristics.

11.2.1 Validation

Ground vibration tests (GVTs) are applied to experimentally determine the aero-elastic behaviour and can be used to validate the structural frequency responses. For a ground vibration set-up shakers are used to realize structural vibrations and accelerometers to measure the wing response (Dunbar).

The performed wind tunnel tests validated the working principle of the developed tether-bridle system. Both, the outcome of the wind tunnel tests and the program, suggest that stall flutter strikes for certain combinations of tether force and bridle attachment position. However measurement data is unavailable for flutter frequencies and aerodynamic forces. Therefore hard conclusions cannot be drawn. Wind tunnel tests at which these measurements are included could further validate the program. Additionally wind tunnel tests with a free flying wing could validate the body freedom flutter modes.

11.2.2 Applicability increase

In the current program, any wing with a tether-bridle system with two bridle lines can be modelled. The generalisability of the program could be improved by allowing more than two bridle lines. In case three or more lines are used, the tether-bridle node is a statically overdetermined system. The laws of displacement compatibility could be used to solve the system of equations for force equilibrium. This addition is not compatible with the current ASWING version, which strongly discourages the use of overdetermined systems and hence a new module should be developed.

11.2.3 Accuracy increase

The implementation of flexible bridle lines would increase the accuracy of the system. It is questionable if this upgrade significantly increases the aero-elastic modelling behaviour, because the bridle lines are relatively short and stiff with respect to the tether. Additionally it is likely that flexible bridle lines contribute to a significant increase of computational time, which is disadvantageous.

The straight tether assumptions is applied. The tether aerodynamic drag and weight each account for respectively 3.3% and 2.2% with respect to the tether force. This deviates the tether from a straight line. To improve the current system, a discretized tether could be applied as described by Breukels (2010) and Leuthold (2013). In continuation to this improvement, wind shear could be included to determine a more realistic wind velocity at each discretized tether element, and hence determine aerodynamic drag more accurately.

11.2.4 Aero-elastic analysis M600

The structural design properties of the fuselage, the horizontal stabilizer and the rudder were unavailable when this research was performed. Therefore these systems were modelled as perfectly rigid [1]. The M600 is under constant development and it is expected that more structural design properties become available in the near future. This will improve the quality of the analyses.

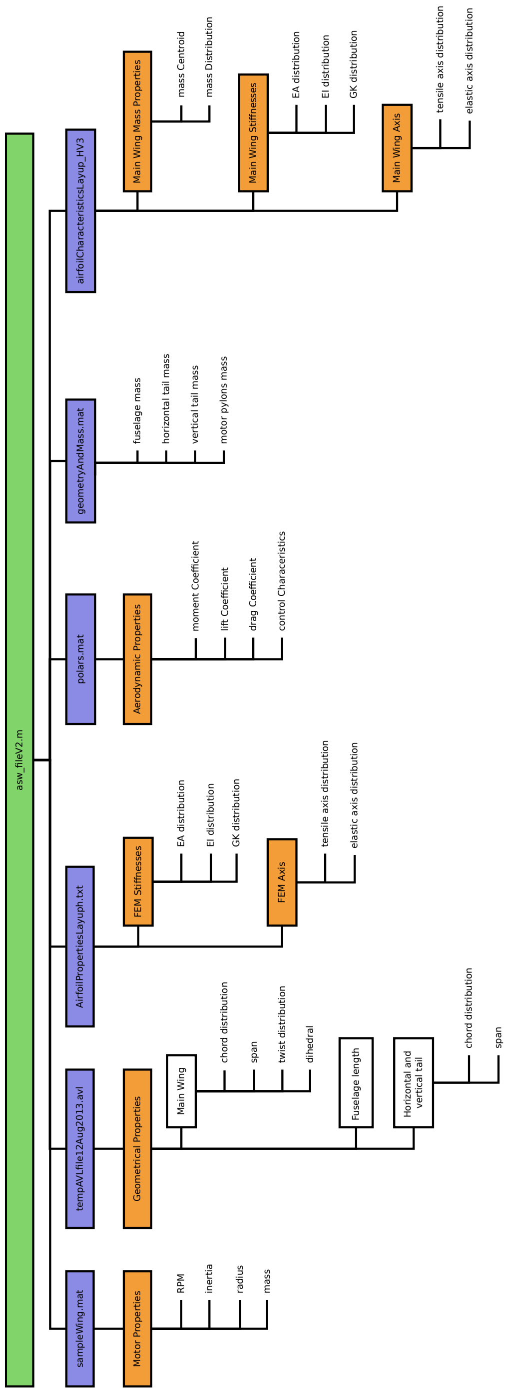
In this first design iteration, a constant apparent wind speed over span width is assumed. In crosswind flight, the flight loop radius is about $150m$. From wing tip to wing tip, the apparent wind speed can deviate with almost 20%. This asymmetry in wind speed will result in an asymmetric aerodynamic force distribution, which could influence the aero-elastic effects. The accuracy of the analyses can be improved by including the turn rate of the wing.

[1] The effect of fuselage stiffness is explored in section 10.3.9, but for the remaining of the analyses the fuselage was simulated as perfectly rigid.

Appendices

Appendix A

Overview MATLAB routine to determine ASWING input file


asw_fileV2.m
sampleWing.mat
Motor Properties
RPM
inertia
radius
mass
tempAVLfile12Aug2013.avl
Geometrical Properties
Main Wing
chord distribution
span
twist distribution
dihedral
Fuselage length
Horizontal and vertical tail
chord distribution
span
AirfoilPropertiesLayuph.txt
FEM Stiffnesses
EA distribution
EI distribution
GK distribution
FEM Axis
tensile axis distribution
elastic axis distribution
polars.mat
Aerodynamic Properties
moment Coefficient
lift Coefficient
drag Coefficient
control Characeristics
geometryAndMass.mat
fuselage mass
horizontal tail mass
vertical tail mass
motor pylons mass
airfoilCharacteristicsLayup_HV3
Main Wing Mass Properties
mass Centroid
mass Distribution
Main Wing Stiffnesses
EA distribution
EI distribution
GK distribution
Main Wing Axis
tensile axis distribution
elastic axis distribution

Appendix B

ASWING and ASWINGb input files

B.1 Input File ASWING

```
#============
Name
M600
End
#============
Units
L 1.0 m
T 1.0 s
F 1.0 N
End
#============
Constant
# g rho v_sound
9.81 1.225 340.29
End
#============
Reference
# Sref Cref Bref
38.4450 1.3865 27.7290
End
#============
Ground
# Nbeam t Kground
2 0.0 0
End
#============
Joint
#
# nBeam1 nBeam2 t1 t2
```

```
2 1 0.0000 0 ! main wing to fuselage
2 4 6.9323 0.0000 ! fuselage to vertical tail
4 3 4.8540 0.0000 ! vertical tail to horizontal tail
1 5 1.6922 0.0508 ! main wing to star inner motor pylon
1 6 -1.6922 0.0508 ! main wing to port inner motor pylon
1 7 5.0766 0.1523 ! main wing to star outer motor pylon
1 8 -5.0766 0.1523 ! main wing to port outer motor pylon
End
#============
Weight
# nBeam t X0 Y0 Z0 weight CDA Vol Hxo Hyo Hzo
* 1.0 1.0 1.0 1.0 100 0.1 1.0 1.0e3 1.0e3 1.0e3
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1 0.0 0.0 0.0 0.0 2500.0 0.0 0.0 0.0 0.0 0.0
7 1.6922 -1.5251 5.0766 1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
5 1.6922 -1.5251 1.6922 1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
6 1.6922 -1.5251 -1.6922 1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
8 1.6922 -1.5251 -5.0766 1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
8 -1.6922 -1.5251 -5.0766 -1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
6 -1.6922 -1.5251 -1.6922 -1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
5 -1.6922 -1.5251 1.6922 -1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
7 -1.6922 -1.5251 5.0766 -1.6922 1.2979 6.2652 0.0 0.4834 0.0 0.0
End
#============
Beam 1
main Wing
#
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0000 -0.3961 0.0000 0.0000 1.5845 13.0000 0.25
6.9323 -0.3961 6.9323 0.0000 1.5845 13.0000 0.25
14.2804 -0.1981 13.8645 0.4159 0.7923 3.0000 0.25
14.2804 -0.1981 13.8745 0.4159 0.7923 3.0000 0.25
15.5977 -0.0990 13.8845 1.7132 0.3961 -2.0000 0.25
#
#Aerodynamic Properties
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0000 7.0712 -0.1900 0.0185 2.6252 -1.0000 6.1850
15.5977 7.0712 -0.1900 0.0185 2.6252 -1.0000 6.1850
#
#Control settings
t  dCLdF1 dCMdF1
* 1.0    0.01 0.01
+ 0.0  0.0 0.0
-15.5977  0.0 0.0
-14.2804  0.0 0.0
-14.2804  -4.972 0.829
0.0  -4.972 0.829
0.0000  4.9720 -0.8290
14.2804  4.9720 -0.8290
```

```
14.2804  0.0 0.0
15.5977  0.0 0.0
#
#Structural properties
t EIcc EInn EIcn GJ EA
* 1.0 1.e6 1.e7 1.e6 1.e6 1.e8
0.0000 26.0304 15.3760 8.0742 6.7718 11.1924
6.9323 26.0304 15.3760 8.0742 6.7718 11.1924
14.2804 3.2544 1.9224 1.0095 0.8466 5.5966
14.2904 3.2544 1.9224 1.0095 0.8466 5.5966
15.5977 0.4066 0.2402 0.1261 0.1058 2.7979
#
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 10.0 10.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
0.0000 5.1132 8.2888 0.2631 1.6116
0.3183 5.1132 8.2888 0.2631 1.6116
0.6366 5.1132 8.2888 0.2631 1.6116
0.9550 5.1132 8.2888 0.2631 1.6116
1.2733 5.1132 8.2888 0.2631 1.6116
1.5916 5.1132 8.2888 0.2631 1.6116
1.9099 5.1132 8.2888 0.2631 1.6116
2.2282 5.1132 8.2888 0.2631 1.6116
2.5466 5.1132 8.2888 0.2631 1.6116
2.8649 5.1132 8.2888 0.2631 1.6116
3.1832 5.1132 8.2888 0.2631 1.6116
3.5015 5.1132 8.2888 0.2631 1.6116
3.8198 5.1132 8.2888 0.2631 1.6116
4.1382 5.1132 8.2888 0.2631 1.6116
4.4565 5.1132 8.2888 0.2631 1.6116
4.7748 5.1132 8.2888 0.2631 1.6116
5.0931 5.1132 8.2888 0.2631 1.6116
5.4114 5.1132 8.2888 0.2631 1.6116
5.7298 5.1132 8.2888 0.2631 1.6116
6.0481 5.1132 8.2888 0.2631 1.6116
6.3664 5.1132 8.2888 0.2631 1.6116
6.6847 5.1132 8.2888 0.2631 1.6116
7.0030 5.1132 8.2888 0.2631 1.6116
7.3214 5.1118 8.2444 0.2631 1.6116
7.6397 5.1042 8.1614 0.2631 1.6116
7.9580 4.5747 8.0785 0.2631 1.6116
8.2763 4.2826 7.9612 0.2631 1.6116
8.5947 4.0189 7.7953 0.2631 1.6116
8.9130 3.7650 7.6294 0.2631 1.6116
9.2313 3.5172 7.4635 0.2631 1.6116
9.5496 3.2758 7.2974 0.2631 1.6116
9.8679 3.0411 7.1314 0.2631 1.6116
10.1863 2.8135 6.9653 0.2631 1.6116
10.5046 2.5932 6.7994 0.2631 1.6116
10.8229 2.3806 6.6335 0.2631 1.6116
11.1412 2.1761 6.4676 0.2631 1.6116
11.4595 1.9801 6.3018 0.2631 1.6116
```

```
11.7779 1.7930 6.1359 0.2631 1.6116
12.0962 1.6152 5.9700 0.2631 1.6116
12.4145 1.4472 5.8040 0.2631 1.6116
12.7328 1.2895 5.6379 0.2631 1.6116
13.0511 1.1424 5.4718 0.2631 1.6116
13.3695 1.0066 5.3059 0.2631 1.6116
13.6878 0.8827 5.1400 0.2631 1.6116
14.0061 0.7712 4.9741 0.2631 1.6116
14.3244 0.6728 4.8082 0.2631 1.6116
14.6427 0.5881 4.6423 0.2631 1.6116
14.9611 0.5181 4.4764 0.2631 1.6116
15.2794 0.4635 4.3116 0.2631 1.6116
#
End
#============
Beam 2
fuselage
#
t x y z radius mg
* 1.0 1.0 1.0 1.0 1.0 10.0
0.0 0.0 0.0 0.0 0.2811 14.3711
6.9323 6.9323 0.0 0.4853 0.1687 8.6226
End
#============
Beam 3
horziontal tail
#
t x y z chord Xax
* 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0
0.0000 6.7243 0.0000 4.8540 0.8319 0.25
2.7729 6.7243 2.7729 4.8540 0.8319 0.25
#
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
0.0000 0.0 43.5167 0.0 0.0
2.7729 0.0 43.5167 0.0 0.0
#
#Aerodynamic Properties
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0000 0.0000 0.0000 0.0058 1.5000 -1.5000 6.1184
2.7729 0.0000 0.0000 0.0058 1.5000 -1.5000 6.1184
#
#Control settings
t dCLdF3 dCMdF3
* 1.0 1.0 0.00
+ 0.0 0.00 0.00
0.0000 0.1068 -0.3917
2.7729 0.1068 -0.3917
#
```

```
End
#============
Beam 4
vertical tail
#
t x y z chord Xax
* 1.0 1.0 1.0 1.0 1.0 1.0
+ 1.0 0.0 0.0 0.0 0.0 0.0
0.0000 6.8611 0.0000 0.4853 0.7972 0.25
1.4562 6.8611 0.0000 1.9415 0.7972 0.25
4.3687 6.8611 0.0000 4.8540 0.7972 0.25
#
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
0.4853 0.0 41.7015 0.0 0.0
1.9415 0.0 59.5728 0.0 0.0
4.8540 0.0 41.7015 0.0 0.0
#
#Aerodynamic Properties
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0000 0.0000 0.0000 0.0058 1.5000 -1.5000 6.1184
4.3687 0.0000 0.0000 0.0058 1.5000 -1.5000 6.1184
#
#Control settings
t dCLdF2 dCMdF2
* 1.0 0.02 0.10
0.0000 -2.3500 0.3917
4.3687 -2.3500 0.3917
#
End
#============
Beam 5
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6922 -1.1438 1.6922 1.6922 0.7087 0.0000 0.2500
0.0423 -1.1543 1.6922 0.0423 0.7087 0.0000 0.2500
0.0423 -1.1543 1.6922 0.0423 0.7087 0.0000 0.2500
0.0000 -1.1543 1.6922 0.0000 1.5009 0.0000 0.2500
0.0000 -1.1543 1.6922 0.0000 1.5009 0.0000 0.2500
-1.9742 -1.1438 1.6922 -1.9742 0.7087 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 10.0 10.0 0.0 0.0
1.6922 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0000 7.8515 3.9257 0.0 0.0
```

```
0.0000 7.8515 3.9257 0.0 0.0
-1.9742 3.7072 1.8536 0.0 0.0
#
End
#============
Beam 6
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6922 -1.1438 -1.6922 1.6922 0.7087 0.0000 0.2500
0.0423 -1.1543 -1.6922 0.0423 0.7087 0.0000 0.2500
0.0423 -1.1543 -1.6922 0.0423 0.7087 0.0000 0.2500
0.0000 -1.1543 -1.6922 0.0000 1.5009 0.0000 0.2500
0.0000 -1.1543 -1.6922 0.0000 1.5009 0.0000 0.2500
-1.9742 -1.1438 -1.6922 -1.9742 0.7087 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 10.0 10.0 0.0 0.0
1.6922 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0000 7.8515 3.9257 0.0 0.0
0.0000 7.8515 3.9257 0.0 0.0
-1.9742 3.7072 1.8536 0.0 0.0
#
End
#============
Beam 7
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6922 -1.1438 5.0766 1.6922 0.7087 0.0000 0.2500
0.0423 -1.1543 5.0766 0.0423 0.7087 0.0000 0.2500
0.0423 -1.1543 5.0766 0.0423 0.7087 0.0000 0.2500
0.0000 -1.1543 5.0766 0.0000 1.5009 0.0000 0.2500
0.0000 -1.1543 5.0766 0.0000 1.5009 0.0000 0.2500
-1.9742 -1.1438 5.0766 -1.9742 0.7087 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 10.0 10.0 0.0 0.0
1.6922 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0000 7.8515 3.9257 0.0 0.0
0.0000 7.8515 3.9257 0.0 0.0
-1.9742 3.7072 1.8536 0.0 0.0
#
End
#============
```

```
Beam 8
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6922 -1.1438 -5.0766 1.6922 0.7087 0.0000 0.2500
0.0423 -1.1543 -5.0766 0.0423 0.7087 0.0000 0.2500
0.0423 -1.1543 -5.0766 0.0423 0.7087 0.0000 0.2500
0.0000 -1.1543 -5.0766 0.0000 1.5009 0.0000 0.2500
0.0000 -1.1543 -5.0766 0.0000 1.5009 0.0000 0.2500
-1.9742 -1.1438 -5.0766 -1.9742 0.7087 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 1 10.0 10.0 0.0 0.0
1.6922 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0423 3.7072 1.8536 0.0 0.0
0.0000 7.8515 3.9257 0.0 0.0
0.0000 7.8515 3.9257 0.0 0.0
-1.9742 3.7072 1.8536 0.0 0.0
#
End
#============
```

B.2 ASWINGb input file rigid hawk.asw

```
#============
Name
Light Hawk
End
#============
Units
L 1.0 ft
T 1.0 s
F 1.0 lb
End
#============
Constant
#   g    rho_SL     V_sound
  32.18  0.002378   1115.0
End
#============
Reference
# Sref  Cref  Bref
 125.0  3.0   49.2
#
#  Xmom  Ymom  Zmom
  -0.40  0.0   0.0
  -0.40  0.0   0.0
  -0.40  0.0   0.0
```

```
End
#============
Joint
#  Nbeam1  Nbeam2    t1     t2
    4      1          0.0    0.0
    4      3         11.4    1.0
    3      2          6.0    0.0
End
#============
Ground
#  Nbeam  t     Kground
!    1     0.0   0
!    2     0.0   0
!    3     0.0   0
    4      0.0   0
End
#=====================
Weight
#  Nbeam  t     Xp     the constraints are adjusted suchYp     Zp      Mg      CDA     Vol     Hx    H
*         1.    1.     1.     1.      1.     1.     1.     1.   1.   1.
#    4   -1.83  -2.83   0.0    0.0   225.0    0.0    0.0    0.0  0.0  0.0
    4   -1.83  -2.83   0.0    0.0   160.0    0.0    0.0    0.0  0.0  0.0
#    4   -1.83  -2.83   0.0    0.0   140.0    0.0    0.0    0.0  0.0  0.0
    4    0.00   0.00   0.0   -1.0     4.0    0.0    0.0    0.0  0.0  0.0
    4    0.00   0.00   0.0   +0.5    15.0    0.0    0.0    0.0  0.0  0.0
    4    0.00   1.50   0.0    0.0     6.0    0.0    0.0    0.0  0.0  0.0
#    4   11.40  11.40   0.0    0.0    15.0    0.0    0.0    0.0  0.0  0.0
    4   11.40  11.40   0.0    0.0     5.0    0.0    0.0    0.0  0.0  0.0
End
#=====================
Sensor
# KS Nb    t     Xp     Yp     Zp     Vx     Vy     Vz      Ax     Ay     Az
*          1.    1.     1.     0.0412  1.    1.     1.      1.     1.     1.
   1  1  -19.7  -1.40 -19.7   19.7   1.0    0.0    0.0     0.0    0.0    1.0
   2  1  -8.01  -1.40 -8.0    8.0    1.0    0.0    0.0     0.0    0.0    1.0
   3  1   0.0   -1.40  0.0    0.0    1.0    0.0    0.0     0.0    0.0    1.0
   4  1   8.01  -1.40  8.0    8.0    1.0    0.0    0.0     0.0    0.0    1.0
   5  1   19.7  -1.40  19.7   19.7   1.0    0.0    0.0     0.0    0.0    1.0

   6  4    0.0   0.0   0.0    0.0    1.0    0.0    0.0     0.0    0.0    1.0

!
!   1  1  -24.0  -1.40 -24.0   24.0   1.0    0.0    0.0     0.0    0.0    1.0
!   2  1  -20.0  -1.40 -20.0   20.0   1.0    0.0    0.0     0.0    0.0    1.0
!   3  1  -16.0  -1.40 -16.0   16.0   1.0    0.0    0.0     0.0    0.0    1.0
!   4  1  -12.0  -1.40 -12.0   12.0   1.0    0.0    0.0     0.0    0.0    1.0
!   5  1  -8.01  -1.40 -8.01    8.0   1.0    0.0    0.0     0.0    0.0    1.0
!   6  1  -4.0   -1.40 -4.0     4.0   1.0    0.0    0.0     0.0    0.0    1.0
!
!   7  1   4.0   -1.40  4.0     4.0   1.0    0.0    0.0     0.0    0.0    1.0
!   8  1   8.01  -1.40  8.01    8.0   1.0    0.0    0.0     0.0    0.0    1.0
!   9  1   12.0  -1.40  12.0   12.0   1.0    0.0    0.0     0.0    0.0    1.0
!  10  1   16.0  -1.40  16.0   16.0   1.0    0.0    0.0     0.0    0.0    1.0
```

```
!  11  1   20.0  -1.40  20.0   20.0   1.0   0.0   0.0    0.0   0.0   1.0
!  12  1   24.0  -1.40  24.0   24.0   1.0   0.0   0.0    0.0   0.0   1.0
End
#============
Engine
# KPeng IEtyp Nbeam t    Xp     Yp Zp  Tx  Ty Tz   dFdPe dMdPe Rdisk Omega cdA
 1     1     4     1.0 -6.975  0. 0.  -1. 0. -.1  1.0   1.0   3.0   100.  0.0
  1     0     4     1.0 -6.975  0. 0.  -1. 0. -.1  1.0   0.01  0.0   100.  0.0
  2     0     1     6.0 -0.30   6. -3.  0. 0. 1.   1.    0.    0.0   0.0   0.0
End
#============
tether
# Nbeam t xt  yt zt Xe Ye      Ze Dl Kspr Wtet CDA
  1     0.   5.000 0.the constraints are adjusted such0   -10.  5.000 0. -300.0  0.0    100
0.0 0.00
end
#============
bridle
# Nbeam t xb yb zb xt yt    zt
  1    -12.40  5.0 -12.40 -0.0 5.000 0.0   -10.
  1     12.40  5.0 12.40  -0.0 5.000   0.0   -10.
end
#============
Beam 1
Wing
      t           x           y           z           chord       alpha
*     1.          1.0         1.          0.0412       1.0         5.0
!*     1.          1.0         1.          0.0         1.0         5.0
      0.00000     0.00000     0.00000   0.00000   3.00000     1.00000
      1.81241    -0.06198     1.81241   1.81241   2.99672     1.00000
      3.61364    -0.12086     3.61364   3.61364   2.98667     1.00000
      5.39259    -0.17600     5.39259   5.39259   2.96923     1.00000
      7.13829    -0.22665     7.13829   7.13829   2.94344     1.00000
      8.83999    -0.27193     8.83999   8.83999   2.90807     1.00000
     10.48718    -0.31090    10.48718  10.48718   2.86169     1.00000
     12.06972    -0.34263    12.06972  12.06972   2.80287     1.00000
     13.57784    -0.36626    13.57784  13.57784   2.73027     1.00000
     15.00225    -0.38102    15.00225  15.00225   2.64283     1.00000
     16.33417    -0.38637    16.33417  16.33417   2.53992     1.00000
     17.56538    -0.38202    17.56538  17.56538   2.42157     1.00000
     18.68830    -0.36806    18.68830  18.68830   2.28864     1.00000
     19.69599    -0.34502    19.69599  19.69599   2.14306     1.00000
     20.58225    -0.31276    20.58225  20.58225   1.99172     1.00000
     21.34162    -0.27222    21.34162  21.34162   1.84220     1.00000
     21.96941    -0.22670    21.96941  21.96941   1.70013     1.00000
     22.46175    -0.18081    22.46175  22.46175   1.57303     1.00000
     22.81560    -0.14051    22.81560  22.81560   1.47041     1.00000
     23.02879    -0.11245    23.02879  23.02879   1.40278     1.00000
     23.10000    -0.10233    23.10000  23.10000   1.37903     1.00000
!     23.10000    -0.10233    23.10000  23.10000   1.37903     1.00000
     23.17427    -0.09133    23.17427  23.17427   1.35359     1.00000
     23.38238    -0.05779    23.38238  23.38238   1.27810     1.00000
     23.68311    -0.00043    23.68311  23.68311   1.15535     1.00000
```

```
     24.01689     0.08177    24.01689  24.01689   0.99057     1.00000
     24.31762     0.18842    24.31762  24.31762   0.79199     1.00000
     24.60000     0.31657    24.60000  24.60000   0.55000     1.00000
!    24.60000     0.46037    24.60000  24.60000   0.34120     1.00000

      t           Cshell      Nshell      Atshell
*     1.          0.35        0.09        0.0001
      0.00000     3.00000     3.00000     3.00000
      1.81241     2.99672     2.99672     2.99672
      3.61364     2.98667     2.98667     2.98667
      5.39259     2.96923     2.96923     2.96923
      7.13829     2.94344     2.94344     2.94344
      8.83999     2.90807     2.90807     2.90807
     10.48718     2.86169     2.86169     2.86169
     12.06972     2.80287     2.80287     2.80287
     13.57784     2.73027     2.73027     2.73027
     15.00225     2.64283     2.64283     2.64283
     16.33417     2.53992     2.53992     2.53992
     17.56538     2.42157     2.42157     2.42157
     18.68830     2.28864     2.28864     2.28864
     19.69599     2.14306     2.14306     2.14306
     20.58225     1.99172     1.99172     1.99172
     21.34162     1.84220     1.84220     1.84220
     21.96941     1.70013     1.70013     1.70013
     22.46175     1.57303     1.57303     1.57303
     22.81560     1.47041     1.47041     1.47041
     23.02879     1.40278     1.40278     1.40278
     23.10000     1.37903     1.37903     1.37903
!    23.10000     1.37903     1.37903     1.37903
     23.17427     1.35359     1.35359     1.35359
     23.38238     1.27810     1.27810     1.27810
     23.68311     1.15535     1.15535     1.15535
     24.01689     0.99057     0.99057     0.99057
     24.31762     0.79199     0.79199     0.79199
     24.60000     0.55000     0.55000     0.55000
!    24.60000     0.34120     0.34120     0.34120


#      t      mg      mgnn    EIcc    EInn    GJ        EIcs    EIsn
#*    1.     0.9     0.9     1.0E5   5.0E6   1.0E5     4.0E4   0.0E5
#*     1.     0.9     0.9     1.0E5   5.0E6   1.0E5     0.      0.
#    -24.6    1.2     1.2     0.005   0.1     0.05      -0.005  -0.1
#    -24.0    1.21    1.21    0.006   0.105   0.0505    -0.006  -0.105
#    -20.0    1.5     1.5     0.04    0.20    0.12      -0.04   -0.20
#    -10.0    3.0     3.0     0.65    0.75     0.5      -0.65   -0.75
#      0.0    5.0     5.0     2.0     2.0     1.0      -2.0    -2.0
#      0.0    5.0     5.0     2.0     2.0     1.0       2.0     2.0
#     10.0    3.0     3.0     0.65    0.75     0.5       0.65    0.75
#     20.0    1.5     1.5     0.04    0.20    0.12       0.04    0.20
#     24.0    1.21    1.21    0.006   0.105   0.0505     0.006   0.105
#     24.6    1.2     1.2     0.005   0.1     0.05       0.005   0.1
```

```
      t       Ccg
*     1.      0.
      0.0     1.0
     10.0     1.0
     20.0     1.0
     24.0     1.0
     24.6     1.0

      t       CLmax     CLmin     Cdf     Cdp
*     1.      1.        1.        1.      1.
      0.0     2.1      -1.0       0.006   0.004
     10.0     2.1      -1.0       0.0064  0.0044
     20.0     2.1      -1.0       0.0068  0.0048
     24.0     2.1      -1.0       0.009   0.006
     24.6     2.1      -1.0       0.012   0.008

!       t       dCLdF4  dCLdF1   dCMdF4  dCMdF1    Cm
!*      1.      1.      1.      -0.25   -0.25     1.0
!     -24.6     0.0     0.0      0.0     0.0      -0.19
!     -24.0     0.0     0.0      0.0     0.0      -0.19
!     -24.0     0.04    0.03     0.04    0.03     -0.19
!     -16.0     0.04    0.03     0.04    0.03     -0.19
!     -16.0     0.06    0.02     0.06    0.02     -0.19
!      -8.01    0.06    0.02     0.06    0.02     -0.19
!      -8.01    0.07    0.01     0.07    0.01     -0.19
!       0.0     0.07    0.01     0.07    0.01     -0.19
!       0.0     0.07   -0.03     0.07   -0.03     -0.19
!       8.01    0.07   -0.03     0.07   -0.03     -0.19
!       8.01    0.06   -0.06     0.06   -0.06     -0.19
!      16.0     0.06   -0.06     0.06   -0.06     -0.19
!      16.0     0.04   -0.09     0.04   -0.09     -0.19
!      24.0     0.04   -0.09     0.04   -0.09     -0.19
!      24.0     0.0     0.0      0.0     0.0      -0.19
!      24.6     0.0     0.0      0.0     0.0      -0.19

      t       dCLdF4  dCLdF1   dCMdF4  dCMdF1    Cm
*     1.      1.      1.      -0.25   -0.25     1.0
    -24.6     0.0     0.0      0.0     0.0      -0.19
    -24.0     0.0     0.0      0.0     0.0      -0.19
    -24.0     0.04    0.06     0.04    0.06     -0.19
    -16.0     0.04    0.06     0.04    0.06     -0.19
    -16.0     0.06    0.04     0.06    0.04     -0.19
     -8.01    0.06    0.04     0.06    0.04     -0.19
     -8.01    0.07    0.02     0.07    0.02     -0.19
      0.0     0.07    0.02     0.07    0.02     -0.19
      0.0     0.07   -0.02     0.07   -0.02     -0.19
      8.01    0.07   -0.02     0.07   -0.02     -0.19
      8.01    0.06   -0.04     0.06   -0.04     -0.19
     16.0     0.06   -0.04     0.06   -0.04     -0.19
     16.0     0.04   -0.06     0.04   -0.06     -0.19
     24.0     0.04   -0.06     0.04   -0.06     -0.19
     24.0     0.0     0.0      0.0     0.0      -0.19
     24.6     0.0     0.0      0.0     0.0      -0.19
```

```
End
#===========
Beam 2
Horizontal Stab
      t          x          y          z          chord      alpha
+     0.         11.4       0.         5.0        0.        -0.0
*     1.0        1.0        1.0        1.0        1.0        1.0
      0.00000    0.24000    0.00000    0.00000    1.40000    0.00000
      0.78316    0.24254    0.78316    0.00000    1.39275    0.00000
      1.55291    0.25036    1.55291    0.00000    1.37041    0.00000
      2.29610    0.26404    2.29610    0.00000    1.33132    0.00000
      3.00000    0.28443    3.00000    0.00000    1.27306    0.00000
      3.65257    0.31244    3.65257    0.00000    1.19302    0.00000
      4.24264    0.34888    4.24264    0.00000    1.08898    0.00000
      4.76012    0.39665    4.76012    0.00000    0.96666    0.00000
      5.19615    0.45761    5.19615    0.00000    0.83173    0.00000
      5.54328    0.53104    5.54328    0.00000    0.68599    0.00000
      5.79556    0.61538    5.79556    0.00000    0.53243    0.00000
      6.00000    0.70823    6.00000    0.00000    0.37513    0.00000
!     6.00000    0.80662    6.00000    0.00000    0.21892    0.00000

      t      mg     mgnn
*     1.     1.2    1.0
      0.0    0.80   0.10
      3.0    0.55   0.04
      6.0    0.40   0.015

      t      CLmax   CLmin    Cm       dCLdF2   dCMdF2
*     1.     1.      1.       1.       1.       1.
      0.0    1.2    -1.2      0.0      0.05    -0.03
      6.0    1.2    -1.2      0.0      0.05    -0.03

      t      EIcc     EInn     GJ
*     1.0    1.0E4    1.0E5    4.0E4
      0.0    1.0      1.0      1.0
      3.0    0.4      0.6      0.8
      6.0    0.2      0.2      0.6
End
#===========
Beam 3 2
Vertical Stab
      t          x          z        chord    Xax
+     1.         11.4       0.0      0.       0.0
*     1.         1.         1.       1.       1.
      0.0000     0.40       0.0      3.0      0.80
      2.5000     0.40       2.5      2.25     0.7333
      4.0000     0.40       4.0      1.8      0.6667
      5.0000     0.40       5.0      1.5      0.60

      t         mg      mgnn
+     1.        0.0     0.0
*     1.        1.0     1.0
      0.0000    1.5     0.20
```

```
      2.5000    1.25    0.12
      4.0000    1.1     0.07
      5.0000    1.0     0.04

      t       CLmax   CLmin    Cm       dCLdF3   dCMdF3
+     1.      0.0     0.0      0.       0.       0.
*     1.      1.      1.       1.       1.       1.
      0.0     1.2    -1.2      0.0     -0.05     0.03
      5.0     1.2    -1.2      0.0     -0.05     0.03

      t       EIcc     EInn     GJ
*     1.0     3.0E4    3.0E5    3.0E4
      0.0     1.0      1.0      1.0
      2.5     0.9      0.7      0.8
      5.0     0.8      0.4      0.6
End
#===========
Beam 4
Fuselage
       t           x       radius     mg
*    1.0        1.0       1.0        1.5
    -7.5000    -7.5000    0.07680    0.07680
    -7.4797    -7.4797    0.09301    0.09301
    -7.4000    -7.4000    0.15360    0.15360
    -7.2751    -7.2751    0.23941    0.23941
    -7.0000    -7.0000    0.38981    0.38981
    -6.5866    -6.5866    0.54262    0.54262
    -6.0000    -6.0000    0.70361    0.70361
    -5.0000    -5.0000    0.92558    0.92558
    -4.0000    -4.0000    1.05922    1.05922
    -2.0000    -2.0000    1.17204    1.17204
     0.0000     0.0000    1.09189    1.09189
     3.0000     3.0000    0.70182    0.70182
     5.0000     5.0000    0.49062    0.49062
     8.0000     8.0000    0.29618    0.29618
    11.4000    11.4000    0.19990    0.19990

      t       EIcc     EInn     GJ
*     1.0     1.0E5    1.0E5    1.0E5
     -7.5     0.6      0.6      0.3
     -6.0     0.9      0.9      0.5
     -3.0     1.5      1.5      0.8
      0.0     1.5      1.5      0.8
      6.0     0.9      0.9      0.6
     11.4     0.4      0.4      0.4

      t       Cdf      Cdp
*     1.      1.       1.
     -7.5     0.003    0.3
     -6.0     0.003    0.3
     -3.0     0.003    0.3
      0.0     0.003    0.3
      6.0     0.003    0.3
```

```
     11.4    0.003    0.3
End
```

B.3 ASWINGb input file for flying wing

```
#============
Name
Swept Flying Wing METRIC
End
#============
Units
L 1.0 m
T 1.0 s
F 1.0 N
End
#============
Constant
  9.81  1.225   340.29
End
#============
Reference
# Sref   Cref   Bref
 24.0   1.00   24.0
#
#  Xmom  Ymom   Zmom
#  3.26   0.0    0.125
   0.0    0.0    0.0
End
#=====================
#Weight
#  Nbeam  t     Xp    Yp    Zp      Mg    CDA    Vol   Hx   Hy   Hz
#*         1.0   1.0   1.0   1.0     1.0   1.0    1.0   1.   1.   1.
#   1      0.0   2.20  0.0   0.0   150.0   0.     0.    0.0  0.0  0.0
#End
#============
Ground
#  Nbeam  t     Kground
    1     0.    0
End
#============
tether
# Nbeam t xt  yt zt Xe Ye Ze Dl Kspr
  1     0.   0.0   0.0     -12. 0.   0. -75.   0.0     100
end
#============
bridle
# Nbeam t x  y z xt yt zt
  1     -5.   -6.0   -12.0  6.0    0.   0.  -12.
  1     5.   12.0   12.0   -12.0    0.      0.  -12.
end
#============
Beam 1
Wing
#
   t     chord    x      y      z
#* 1.0    1.0     1.0    1.0    1.0
```

```
#  0.0    3.00    0.0    0.0    0.0 !Original
# 10.0    2.10    8.0   40.0    0.5 !Original
# 10.0    2.10    8.0   40.0    0.5 !Original
# 12.0    1.20    8.2   39.2    3.0 !Original

  0.0    1.00    0.0    0.0    0.0 !Adjusted
 10.0    1.00    0.0   12.0    0.0 !Adjusted
# 10.0    3.00    0.0   40.0    0.0 !Adjusted
# 12.0    1.20    8.2   39.2    3.0 !Adjusted
#
   t     Cshell   Nshell    Atshell
* 1.0    0.0021   0.0021    1.524e-5
  0.0    0.50     0.50      0.025
  3.0    0.50     0.50      0.015
  3.0    0.375    0.375     0.015
 10.0    0.375    0.375     0.010
# 10.0    0.375    0.375     0.010
# 11.5    0.375    0.375     0.010
#
   t     alpha    Cm      Cdf
* 1.0    1.0      1.0     1.0
  0.0    6.0      0.05    0.015
 10.0    6.0      0.05    0.015
# 10.0    2.0      0.0     0.015
# 11.5    2.0      0.0     0.015
#
  t   twist
* 1.0  0.5  !Original
#* 1.0  0.0 !Adjusted
  0.0  0.0
  2.0  0.0
  5.0 -0.2
  7.0 -1.0
  8.5 -2.4
 10.0 -5.0
# 10.0 -0.0
# 11.5 -0.0
#
  t    dCLdF1   dCMdF1
* 1.0  1.0      1.0
#-11.5  0.0      0.0
#-10.0  0.0      0.0
-10.0  0.07    -0.03
 -9.0  0.07    -0.03
 -6.0  0.07    -0.03
 -6.0  0.0      0.0
  6.0  0.0      0.0
  6.0 -0.07     0.03
  9.0 -0.07     0.03
 10.0 -0.07     0.03
# 10.0  0.0      0.0
# 11.5  0.0      0.0
#
```

```
  t    dCLdF2   dCMdF2
* 1.0  1.0      1.0
  0.0  0.07    -0.01
  4.0  0.07    -0.01
  4.0  0.0      0.0
 10.0  0.0      0.0
# 10.0  0.0      0.0
# 11.5  0.0      0.0
#
  t    dCLdF3   dCMdF3
* 1.0  1.0      1.0
#-11.5  0.07    -0.03
#-10.0  0.07    -0.03
-10.0  0.0      0.0
 -9.0  0.0      0.0
 -6.0  0.0      0.0
 -6.0  0.0      0.0
  6.0  0.0      0.0
  6.0  0.0      0.0
  9.0  0.0      0.0
 10.0  0.0      0.0
# 10.0 -0.07    -0.03
# 11.5 -0.07    -0.03
#
  t    dCLda
* 1.0  1.0
  0.0  6.2
 10.0  6.2
# 10.0  6.2
# 11.5  6.2
#
  t    CLmax  CLmin
* 1.0  5.0    5.0
  0.0  1.1   -1.1
 10.0  1.1   -1.1
# 10.0  1.1   -1.1
# 11.5  1.1   -1.1
#
  t     Xax   Ccg   Cea   Cta
* 1.0   1.0   0.03   0.03   0.03
  0.0   0.35  3.00  3.00  3.00
 10.0   0.35  0.70   0.70    0.70
# 10.0   0.35  2.10  2.10  2.10
# 11.5   0.35  1.40  1.40  1.40
#
   t   EIcc    EInn     GJ
* 1.0  6e4     25e7    40e3
#* 1.0  0.      0.       0.  !Adjusted for rigid wing
#-10.0   0.0     0.       0.    !Adjusted to make one side of the wing rigid
#0.0     0.0     0.       0.    !Adjusted to make one side of the wing rigid
  0.0 180.0    1.0      40.0
  2.0 120.0    0.7      36.0
  4.0  70.0    0.38     30.0
```

```
  7.0  33.0    0.17     16.0
 10.0  20.0    0.08      5.0
# 10.0  20.0    0.08      5.0
# 11.5   5.0    0.02      2.0
#
  t     mg    mgnn
* 1.0  400.    0.08
#* 1.0  0.0    0.0 !adjusted for rigid wing
  0.0  0.95   1.0
  1.0  0.95   1.0
  1.0  0.75   1.0
  7.5  0.50   1.0
 10.0  0.40   1.0
#10.0  0.40   1.0
# 11.5  0.10   1.0
End
#============
```

B.4 ASWINGb input M600 with bridles

```
#============
Name
M600
End
#============
Units
L 1.0 m
T 1.0 s
F 1.0 N
End
#============
Constant
# g rho v_sound
9.81 1.225 340.29
End
#============
Reference
# Sref Cref Bref
32.9285 1.2831 25.6626
End
#============
Ground
# Nbeam t Kground
1. 0.0 0
End
#============
Gravity
# Gx Gy Gz
0. 0. -1.
End
#============
tether
# Nbeam t xt  yt zt Xe Ye      Ze Dl Kspr Wtet CDA
    1     0.   0.000 0.0   -4.6188  0.000 0. -300.0  -2.0    6.e4 4000.0 8.00 !Original
#    1     0.   0.000 0.0   -4.6188  0.000 0. -300.0  -2.0    12.e4 4000.0 8.00 !Adjusted
end
#============
bridle
# Nbeam t xb yb zb xt yt    zt
  1    -6.4157  -0.0691 -6.4157 -0.2961 0.000 0.0   -4.6188  !Original
  1     6.4157  -0.0691 6.4157  -0.2961 0.000   0.0   -4.6188  !Original

#  1    -6.4157  -0.4691 -6.4157 -0.2961 0.000 0.00   -4.6188  !Adjusted for experiment
#  1     6.4157  -0.4691 6.4157  -0.2961 0.000   0.00   -4.6188  !Adjusted for experiments
end
#============
Joint
#
# nBeam1 nBeam2 t1 t2
1 2 0.0000 0.0000 ! main wing to fuselage
2 4 7.0572 1.4500 ! fuselage to vertical tail
```

```
4 3 3.5958 0.0000 ! vertical tail to horizontal tail
1 5 -3.6393 0.0364 ! main wing to star inner motor pylon
1 6  3.6393 0.0364 ! main wing to port inner motor pylon
1 7 -1.2131 0.1092 ! main wing to star outer motor pylon
1 8  1.2131 0.1092 ! main wing to port outer motor pylon
End
#============
Weight
# nBeam t X0 Y0 Z0 weight CDA Vol Hxo Hyo Hzo
* 1.0 1.0 1.0 1.0 100 1.0 1.0 1.0e3 1.0e3 1.0e3
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
7 1.3142 -1.6424 -1.2131 1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
7 1.3142 -1.6424 -1.2131 1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
5 1.3142 -1.6424 -3.6393 1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
5 1.3142 -1.6424 -3.6393 1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
6 1.3142 -1.6424 3.6393 1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
6 1.3142 -1.6424 3.6393 1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
8 1.3142 -1.6424 1.2131 1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
8 1.3142 -1.6424 1.2131 1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
8 -1.3142 -1.6424 1.2131 -1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
8 -1.3142 -1.6424 1.2131 -1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
6 -1.3142 -1.6424 3.6393 -1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
6 -1.3142 -1.6424 3.6393 -1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
5 -1.3142 -1.6424 -3.6393 -1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
5 -1.3142 -1.6424 -3.6393 -1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
7 -1.3142 -1.6424 -1.2131 -1.3142 0.7847 0.0000 0.0 0.3385 0.0 0.0 !motor
7 -1.3142 -1.6424 -1.2131 -1.3142 4.2244 0.0000 0.0 0.0000 0.0 0.0 !generator
# 1 0. 0. 0. 0. 0. 2.0000 0.0 0.0000 0.0 0.0 !added to simulate tether drag
# 1 0.0000 0.0 0.0 -150. 32. 2.00 0. 0. 0. 0.
End
#============
Engine
# Keng  IEtyp  Nbeam   t    Xo    Yo    Zo     Tx   Ty   Tz     dFdPe   dMdPe   Rdisk  Omega   cdA
*    1.0 1.0 1.0 1.0 1.0 1. 1.0 1. 1. 1.0 1.0    1.0
+           0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0. 0.
1 1       7 1.3142 -1.6424 -1.2131 1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
2 1 5 1.3142 -1.6424 -3.6393 1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
3 1 6 1.3142 -1.6424 3.6393 1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
4 1 8 1.3142 -1.6424 1.2131 1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
5 1 8 -1.3142 -1.6424 1.2131 -1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
6 1 6 -1.3142 -1.6424 3.6393 -1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
7 1 5 -1.3142 -1.6424 -3.6393 -1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
8 1 7 -1.3142 -1.6424 -1.2131 -1.3142 -1. 0.0 0.0 1.0 1.0 1.03 120. 0.0089
End
#============
Beam 1
main Wing
#
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0000 -0.3636 0.0000 0.0000 1.4543 12.0000 0.25
# 6.4157 -0.3636 6.4157 0.0000 1.4543 12.0000 0.25
```

```
6.4157 -0.3636 6.4157 0.0000 1.4543 12.0000 0.25
9.6235 -0.2780 9.6235 0.1925 1.1120 8.2500 0.25
12.8313 -0.1924 12.8313 0.3849 0.7696 4.5000 0.25
12.8313 -0.1924 12.8313 0.3849 0.7696 4.5000 0.25
13.0228 -0.0962 12.8513 1.4546 0.3848 -5.5000 0.25

# 9.6235 -0.4636 9.6235 0.1925 1.1120 8.2500 0.25 !forward sweep values, straight leading edge
# 12.8313 -0.5636 12.8313 0.3849 0.7696 4.5000 0.25 !forward sweep values, straight leading edge
# 12.8313 -0.5636 12.8313 0.3849 0.7696 4.5000 0.25 !forward sweep values, straight leading edge
# 13.0228 -0.5636 12.8513 1.4546 0.3848 -5.5000 0.25 !forward sweep values, straight leading edge
#
#Stifnesses and FEM properties
t EIcc EInn EA GJ Cta Nta Cea Nea
* 1.0 1.0e6 1.0e6 1.0e8 1.0e6 1.0 1.0 1.0 1.00
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
# 0. 10.7 27.9 8.73 1.37 0.404 0.033 0.404 0.033 !original values Gregor
# 7.1 5.11 14.4 4.13 1.15 0.4 0.045 0.4 0.045 !original values Gregor
# 13.0228 0.104 0.816 0.588 0.09 0.24 0.026 0.24 0.026 !original values Gregor

0. 10.7 27.9 8.73 1.37 0.0404 0.033 0.0404 0.033 !change of elastic axis (actually account for c=(
7.1 5.11 14.4 4.13 1.15 0.0404 0.045 0.0404 0.045 !change of elastic axis (actually account for c=
# 7.1 5.11 14.4 4.13 1.15 0.0404 0.045 0.0404 0.045 !change of elastic axis (actually account for
12.8313 0.104 0.816 0.588 0.09 0.0476 0.026 0.0476 0.026 !change of elastic axis (actually accoun
12.8313 0.104 0.816 0.588 0.09 0.0476 0.026 0.0476 0.026 !change of elastic axis (actually accoun
13.0228 0.104 0.816 0.588 0.09 0.0476 0.026 0.0476 0.026 !change of elastic axis (actually accoun

#
# 0. 13.3750 34.8750 10.9125 1.7125 0.404 0.033 0.404 0.033 !new airfoil 16aug2013 Damon
# 7.1 6.3875 17.4240 5.45 1.426 0.4 0.045 0.4 0.045 !new airfoil 16aug2013 Damon
# 13.0228 0.09256 0.49931 0.588 0.08271 0.24 0.026 0.24 0.026 !new airfoil 16aug2013 Damon

# 0. 0.0 0.0 0.0 0.0 0.404 0.033 0.404 0.033 !infinitely stiff
# 7.1 0.0 0.0 0.0 0.0 0.4 0.045 0.4 0.045 !infinitely stiff
# 13.0228 0.0 0.0 0.0 0.0 0.24 0.026 0.24 0.026 !infinitely stiff
#
#Aerodynamic Properties
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0000 10.0712 -0.1900 0.0185 3.4252 -1.0000 6.1850
12.8313 10.0712 -0.1900 0.0185 3.4252 -1.0000 6.1850
12.8313  0.0 -0.000 0.0185 1.00 -1.0000 6.1850
13.0228 0.0 -0.000 0.0185 1.00 -1.0000 6.1850
#
#Control settings
t  dCLdF9 dCMdF9
* 1.0  0.01 0.01
+ 0.0  0.0 0.0
-13.0228 0.0000 0.0000
-12.8313 0.0000 0.0000
-12.8313 -5.888 0.829
-11.8813 -5.888 0.829
-11.8813 0.0000 0.0000
```

```
0.0000 0.0000 0.0000
11.8813 0.0000 0.0000
11.8813 5.888 -0.829
12.8313 5.888 -0.829
12.8313 0.0000 0.0000
13.0228 0.0000 0.0000

t dCLdF1 dCMdF1
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
# -12.8313 0.0000 0.0000
-13.0228 0.0000 0.0000
-11.8813 0.0000 0.0000
-11.8813 5.888 -1.0
-9.6813 5.888 -1.0
-9.6813 0.0000 0.0000
0.0000 0.0000 0.0000
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000

t dCLdF2 dCMdF2
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
-13.0228 0.0000 0.0000
# -12.8313 0.0000 0.0000
-9.6813 0.0000 0.0000
-9.6813 5.888 -1.0
-6.4813 5.888 -1.0
-6.4813 0.0000 0.0000
0.0000 0.0000 0.0000
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000
#
t dCLdF3 dCMdF3
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
-13.0228 0.0000 0.0000
# -12.8313 0.0000 0.0000
-6.4813 0.0000 0.0000
-6.4813 5.888 -1.0
-3.2813 5.888 -1.0
-3.2813 0.0000 0.0000
0.0000 0.0000 0.0000
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000

t dCLdF4 dCMdF4
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
-13.0228 0.0000 0.0000
# -12.8313 0.0000 0.0000
-3.2313 0.0000 0.0000
-3.2313 5.888 -1.0
```

```
0.0000 5.888 -1.0
0.0000 0.0000 0.0000
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000

t dCLdF5 dCMdF5
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
-13.0228 0.0000 0.0000
# -12.8313 0.0000 0.0000
0.0000 0.0000 0.0000
0.0000 5.888 -1.0
3.2313 5.888 -1.0
3.2313 0.0000 0.0000
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000

t dCLdF6 dCMdF6
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
-13.0228 0.0000 0.0000
# -12.8313 0.0000 0.0000
0.0000 0.0000 0.0000
3.2813 0.0000 0.0000
3.2813 5.888 -1.0
6.4813 5.888 -1.0
6.4813 0.0000 0.0000
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000

t dCLdF7 dCMdF7
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
-13.0228 0.0000 0.0000
# -12.8313 0.0000 0.0000
0.0000 0.0000 0.0000
6.4813 0.0000 0.0000
6.4813 5.888 -1.0
9.6813 5.888 -1.0
9.6813 0.0000 0.0000
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000

t dCLdF8 dCMdF8
* 1.0 0.01 0.01
+ 0.0 0.0 0.0
-13.0228 0.0000 0.0000
# -12.8313 0.0000 0.0000
0.0000 0.0000 0.0000
9.6813 0.0000 0.0000
9.6813 5.888 -1.0
11.8813 5.888 -1.0
11.8813 0.0000 0.0000
```

```
13.0228 0.0000 0.0000
# 12.8313 0.0000 0.0000
#


#
#Mass properties
t Ccg Ncg mg mgcc mgnn Dmg DCcg DNcg !Dmg represent the servo mass
* 1.0 1.0 1.0 9.81 1.0 1.0 9.81 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0000 0.3968 0.0348 14.9734 1.9472 5.6514 1.0 1.2 0.0348
7.1000 0.3981 0.0427 9.3642 1.0806 4.3344 1.0 1.2 0.0427
12.8313 0.2308 0.0215 2.7845 0.0562 0.4332 1.0 0.5 0.0215
# 13.0228 0.2308 0.0215 2.7845 0.0562 0.4332 1.0 0.5 0.0215
#
End
#============
Beam 2
fuselage
#
t x y z radius mg EIcc   EInn
* 1.0 1.0 1.0 1.0 1.0 10.0 2.5e6   2.5e6
0.0 0.0 0.0000 0. 0.2811 14.3711 1.     1.
1.0000 6.9933 0.0000 1.45000 0.1687 8.6226 0.216  0.216

End
#============
Beam 3
horizontal tail
#
t x y z chord Xax
* 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0
-1.0 7.1309 -2.3484 3.5958 0.4423 0.25
0.0000 6.9933 0.0000 3.5958 1.0319 0.25
1.0 7.1309 2.3484 3.5958 0.4423 0.25
#
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
-1.0 0.0 21.4114 0.0 0.0
 0.0 0.0 49.9579 0.0 0.0
 1.0 0.0 21.4114 0.0 0.0
#
#Aerodynamic Properties
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
-1.0 0.0000 0.0000 0.0058 1.5000 -1.5000 6.1184
0.0000 0.0000 0.0000 0.0058 1.5000 -1.5000 6.1184
1.0 0.0000 0.0000 0.0058 1.5000 -1.5000 6.1184
#
#Control settings
```

```
t dCLdF10 dCMdF10
* 1.0 1.0 0.01
+ 0.0 0.00 0.00
-1.0 0.1068 -0.3917
0.0000 0.1068 -0.3917
1.0 0.1068 -0.3917
#
End
#============
Beam 4
vertical tail
#
t x y z chord Xax
* 1.0 1.0 1.0 1.0 1.0 1.0
+ 1.0 0.0 0.0 0.0 0.0 0.0
0.0000 6.9933 0.0000 0.0000 0.7164 0.25
1.4383 6.6862 0.0000 1.4383 1.0235 0.25
3.5958 6.9933 0.0000 3.5958 0.7164 0.25
#
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
0.0000 0.0 34.6849 0.0 0.0
1.4383 0.0 49.5512 0.0 0.0
3.5958 0.0 34.6849 0.0 0.0
#
#Aerodynamic Properties
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0000 0.0000 0.0000 0.008 2.000 -1.3000 6.1184
3.5958 0.0000 0.0000 0.008 2.000 -1.3000 6.1184
#
#Control settings
t dCLdF11 dCMdF11
* 1.0 1.0 0.01
0.0000 -0.80 0.3917
3.5958 -0.80 0.3917
#
End
#============
Beam 5
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6142 -1.2318 -3.6393 1.6142 0.7805 0.0000 0.2500
1.3142 -1.6318 -3.6393 1.3142 1.7805 0.0000 0.2500
1.3142 -1.6318 -3.6393 1.3142 1.7805 0.0000 0.2500
1.0142 -1.2318 -3.6393 1.0142 0.7805 0.0000 0.2500
1.0142 -1.2318 -3.6393 1.0142 0.7805 0.0000 0.2500
0.0329 -1.1869 -3.6393 0.0329 0.7805 0.0000 0.2500
0.0329 -1.1869 -3.6393 0.0329 0.7805 0.0000 0.2500
```

```
0.0000 -1.1869 -3.6393 0.0000 1.5077 0.0000 0.2500
0.0000 -1.1869 -3.6393 0.0000 1.5077 0.0000 0.2500
-1.2164 -1.2318 -3.6393 -1.2164 0.7805 0.0000 0.2500
-1.2164 -1.2318 -3.6393 -1.2164 0.7805 0.0000 0.2500
-1.5164 -1.6318 -3.6393 -1.5164 1.7805 0.0000 0.2500
-1.5164 -1.6318 -3.6393 -1.5164 1.7805 0.0000 0.2500
-1.8164 -1.2318 -3.6393 -1.8164 0.7805 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 10.0 10.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
1.6142 3.7787 1.8894 0.0 0.0
1.3142 8.6200 4.3100 0.0 0.0
1.3142 8.6200 4.3100 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
0.0000 7.2993 3.6497 0.0 0.0
0.0000 7.2993 3.6497 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.8164 3.7787 1.8894 0.0 0.0

#Aerodynamic Properties  !rh8
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.6142 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
0.0000 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
-1.8164 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
#
End
#============
Beam 6
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6142 -1.2318 3.6393 1.6142 0.7805 0.0000 0.2500
1.3142 -1.6318 3.6393 1.3142 1.7805 0.0000 0.2500
1.3142 -1.6318 3.6393 1.3142 1.7805 0.0000 0.2500
1.0142 -1.2318 3.6393 1.0142 0.7805 0.0000 0.2500
1.0142 -1.2318 3.6393 1.0142 0.7805 0.0000 0.2500
0.0329 -1.1869 3.6393 0.0329 0.7805 0.0000 0.2500
0.0329 -1.1869 3.6393 0.0329 0.7805 0.0000 0.2500
0.0000 -1.1869 3.6393 0.0000 1.5077 0.0000 0.2500
0.0000 -1.1869 3.6393 0.0000 1.5077 0.0000 0.2500
-1.2164 -1.2318 3.6393 -1.2164 0.7805 0.0000 0.2500
-1.2164 -1.2318 3.6393 -1.2164 0.7805 0.0000 0.2500
-1.5164 -1.6318 3.6393 -1.5164 1.7805 0.0000 0.2500
-1.5164 -1.6318 3.6393 -1.5164 1.7805 0.0000 0.2500
```

```
-1.8164 -1.2318 3.6393 -1.8164 0.7805 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 10.0 10.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
1.6142 3.7787 1.8894 0.0 0.0
1.3142 8.6200 4.3100 0.0 0.0
1.3142 8.6200 4.3100 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
0.0000 7.2993 3.6497 0.0 0.0
0.0000 7.2993 3.6497 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.8164 3.7787 1.8894 0.0 0.0
#
#Aerodynamic Properties  !rh8
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.6142 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
0.0000 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
-1.8164 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
#
End
#============
Beam 7
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6142 -1.2318 -1.2131 1.6142 0.7805 0.0000 0.2500
1.3142 -1.6318 -1.2131 1.3142 1.7805 0.0000 0.2500
1.3142 -1.6318 -1.2131 1.3142 1.7805 0.0000 0.2500
1.0142 -1.2318 -1.2131 1.0142 0.7805 0.0000 0.2500
1.0142 -1.2318 -1.2131 1.0142 0.7805 0.0000 0.2500
0.0329 -1.1869 -1.2131 0.0329 0.7805 0.0000 0.2500
0.0329 -1.1869 -1.2131 0.0329 0.7805 0.0000 0.2500
0.0000 -1.1869 -1.2131 0.0000 1.5077 0.0000 0.2500
0.0000 -1.1869 -1.2131 0.0000 1.5077 0.0000 0.2500
-1.2164 -1.2318 -1.2131 -1.2164 0.7805 0.0000 0.2500
-1.2164 -1.2318 -1.2131 -1.2164 0.7805 0.0000 0.2500
-1.5164 -1.6318 -1.2131 -1.5164 1.7805 0.0000 0.2500
-1.5164 -1.6318 -1.2131 -1.5164 1.7805 0.0000 0.2500
-1.8164 -1.2318 -1.2131 -1.8164 0.7805 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 10.0 10.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
1.6142 3.7787 1.8894 0.0 0.0
```

```
1.3142 8.6200 4.3100 0.0 0.0
1.3142 8.6200 4.3100 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
0.0000 7.2993 3.6497 0.0 0.0
0.0000 7.2993 3.6497 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.8164 3.7787 1.8894 0.0 0.0
#
#Aerodynamic Properties  !rh8
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.6142 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
0.0000 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
-1.8164 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
#
End
#============
Beam 8
Motor Pylons
t x y z chord twist Xax
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6142 -1.2318 1.2131 1.6142 0.7805 0.0000 0.2500
1.3142 -1.6318 1.2131 1.3142 1.7805 0.0000 0.2500
1.3142 -1.6318 1.2131 1.3142 1.7805 0.0000 0.2500
1.0142 -1.2318 1.2131 1.0142 0.7805 0.0000 0.2500
1.0142 -1.2318 1.2131 1.0142 0.7805 0.0000 0.2500
0.0329 -1.1869 1.2131 0.0329 0.7805 0.0000 0.2500
0.0329 -1.1869 1.2131 0.0329 0.7805 0.0000 0.2500
0.0000 -1.1869 1.2131 0.0000 1.5077 0.0000 0.2500
0.0000 -1.1869 1.2131 0.0000 1.5077 0.0000 0.2500
-1.2164 -1.2318 1.2131 -1.2164 0.7805 0.0000 0.2500
-1.2164 -1.2318 1.2131 -1.2164 0.7805 0.0000 0.2500
-1.5164 -1.6318 1.2131 -1.5164 1.7805 0.0000 0.2500
-1.5164 -1.6318 1.2131 -1.5164 1.7805 0.0000 0.2500
-1.8164 -1.2318 1.2131 -1.8164 0.7805 0.0000 0.2500
#Mass properties
t mg Dmg mgcc mgnn
* 1.0 10.0 10.0 1.0 1.0
+ 0.0 0.0 0.0 0.0 0.0
1.6142 3.7787 1.8894 0.0 0.0
1.3142 8.6200 4.3100 0.0 0.0
1.3142 8.6200 4.3100 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
1.0142 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
0.0329 3.7787 1.8894 0.0 0.0
```

```
0.0000 7.2993 3.6497 0.0 0.0
0.0000 7.2993 3.6497 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.2164 3.7787 1.8894 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.5164 8.6200 4.3100 0.0 0.0
-1.8164 3.7787 1.8894 0.0 0.0
#
#Aerodynamic Properties  !rh8
t alpha Cm Cdf CLmax CLmin dCLda
* 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.6142 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
0.0000 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
-1.8164 1.5000 0.0500 0.0058 1.8000 -1.000 6.1184
#
End
#============
```

Appendix C

FORTRAN77 source code additions to ASWING

C.1 Tether-bridle subroutine

```fortran

      SUBROUTINE SETTET(IS,IPNT, RTBA)
      INCLUDE 'ASWING.INC'
c-----------------------------------------------------------------------
c
c     This subroutine accounts for all loads and and damping as a
c     resulting from the tether. One tether end is fixed at the 'aircraft'
c     and the other end is fixed in the Earth-reference frame (the
c     ground).
c
c     The parameters listed below are new parameters which are created
c     for SUBROUTINE SETTET.
c
c-----------------------------------------------------------------------
c
c    NBRID            = number of bridles
c    NCBRID           = bridle number
c    TETR(3,3)        = TE transpose matrix
c     (...)_ANG       = derivative wrt to aicraft Euler angles
c     (...)_ANGE      = derivative wrt to Earth Euler angles
c     (...)_R         = derivative wrt aircraft COG position
c    DRPBR(NBRID,3)   = pylon offset vector dr_p
c    RBRI(NBRID,3)    = real attachment of the bridle r_bi
c    LBRID(NBRID)     = bridle length
c    RE0TET(3)        = tether ground attachment point, Earth coordinates
c    TETMASS          = tether mass
c    CDA              = drag area coefficient
c    RTET(3)          = tether vector
c    LTET             = tether lenght
c    KSPR             = spring constant
c    TETF             = tether force magnitude
c    TETFOR(3)        = tether force vector
c    RBRIST(NBRID,3)  = normalized r_bri
c    BRIDF(2)         = bridle force magnitude
c    BRIDFOR(2,3)     = bridle force vectors
c    SUMFXTBA         = sum for forces in x at tether-bridle attachment
```

```
c     SUMFYTBA           = sum for forces in y at tether-bridle attachment
c     SUMFZTBA           = sum for forces in z at tether-bridle attachment
c     RTBA(3)            = position vector tether-bridle attachment
c-----------------------------------------------------------------------
       REAL XYZ(3), ANG(3), POS(3),
     &          T0(3,3), T(3,3), T_ANG(3,3,3),
     &          TNET(3,3), TNET_ANG(3,3,3)
c
       REAL DRP(3), DRP_ANG(3,3), RP(3), RE0TET(3)
c
       REAL TE(3,3), TE_A(3,3,3), TETR(3,3), TETR_ANGE(3,3,3)
c
       REAL RTET(3), RTET_POS(3,3), TETFOR_LTET(3), TETFOR_TETF(3),
     &          RTET_RTET(3,3), RTET_ANG(2,3,3), RTET_ANGE(3,3),
     &          LTET_RTET(3), TETF_RTET(3), LTET_ANG(2,3),
     &          TETF_ANG(2,3),
     &          LTET_POS(3), TETF_POS(3), LTET_ANGE(3), TETF_ANGE(3),
     &          TETFOR_RTET(3,3), TETFOR_ANG(2,3,3), TETFOR_POS(3,3),
     &          TETFOR_ANGE(3,3), DRTET(3),DRP_ANGE(3,3),
     &          TNET_ANGE(3,3,3), RTE(3), RTE_TEA(3,3), RTET_TEA(3,3)

       REAL DRPBR_ANG(NBRID,3,3), BRIDF(NBRID), ALPHA(NBRID),
     & DRPBR(NBRID,3), RBRI_R(NBRID,3,3), BRIDFOR(NBRID,3),
     & RBRI_ANG(NBRID,3,3), RBRI_Q(NBRID,3,3), RTBA_ANG(2,3,3),
     & RBRIST(NBRID,3), RTBA_R(2,3,3)

       REAL RBRI_TETFOR(NBRID,3,3), BRIDFOR_POS(NBRID,3,3),
     & BRIDFOR_TETFOR(NBRID,3,3), BRIDF_TETFOR(NBRID,3),
     & BRIDFOR_ANGE(NBRID,3,3), BRIDFOR_R(NBRID,3,3),
     & BRIDFOR_ANG(NBRID,3,3)


       REAL LTET_R(2,3), TETF_R(2,3),TETFOR_R(2,3,3), EUL(3)

       INTEGER  IEB(NBRID), NBRI

       DIMENSION F(3), F_Q(3,18), F_GL(3,0:NGLX), F_FRP(3,0:NFRLX),
     &          M(3), M_Q(3,18), M_GL(3,0:NGLX), M_FRP(3,0:NFRLX),
     &                F_QT(3,6), F_GLT(3,0:NGLX),
     &                M_QT(3,6), M_GLT(3,0:NGLX),
     &        FK_QO(18)   ,FK_QP(18)   ,FK_GL(NGLX), FK_GLT(NGLX),
     &        MK_QO(18)   ,MK_QP(18)   ,MK_GL(NGLX), MK_GLT(NGLX),
     &        FK_QTO(6)   ,FK_QTP(6)   ,FK_FRP(NFRLX),
     &        MK_QTO(6)   ,MK_QTP(6)   ,MK_FRP(NFRLX)
c
       REAL KSPR, LTET, DLEN, LTET0, TETFOR_RTETK, TETF_TEST, RTET1_PSI,
     & RTET2_PSI, RTET3_PSI, SUMFXTBA, SUMFYTBA, SUMFZTBA

       REAL BRIDFTOP(2), VARA, VARATOP, VARATOP_R(3), VARABOT,
     & VARABOT_R(3), VARA_R(3), VARBTOP, VARBTOP_R(3), VARBBOT,
     & VARBBOT_R(3), VARB_R(3), BRIDFTOP_R(2,3), BRIDFBOT_R(2,3),
     & BRIDF_R(2,3),  RBRIST_R(2,3,3), BRIDFBOT(2), BRIDFOR1_R(2,3,3),
     & TEMPRBRI_R(2,3,3), TEMPRBRI_ANG(2,3,3), BRIDFOR2_R(2,3,3),
     & RBRIST_POS(2,3,3), VARATOP_POS(3), VARABOT_POS(3), VARA_POS(3),
     & VARBTOP_POS(3), VARBBOT_POS(3), VARB_POS(3), BRIDFTOP_POS(2,3),
     & BRIDFBOT_POS(2,3), BRIDF_POS(2,3), BRIDFOR1_POS(2,3,3),
     & TEMPRBRI_POS(2,3,3), BRIDFOR2_POS(2,3,3), RTBA_POS(2,3,3),
     & VARATOP_ANG(3), VARABOT_ANG(3), VARA_ANG(3), VARBTOP_ANG(3),
     & VARBBOT_ANG(3), VARB_ANG(3), BRIDFTOP_ANG(2,3),BRIDFBOT_ANG(2,3),
     & BRIDF_ANG(2,3),  RBRIST_ANG(2,3,3), BRIDFOR1_ANG(2,3,3),
     &  BRIDFOR2_ANG(2,3,3)

       DIMENSION ICRS(3), JCRS(3)

       REAL RBRIST_ANGE(2,3,3), VARATOP_ANGE(3), VARABOT_ANGE(3),
     & VARA_ANGE(3), VARBTOP_ANGE(3), VARBBOT_ANGE(3), VARB_ANGE(3),
     & BRIDFTOP_ANGE(2,3), BRIDFBOT_ANGE(2,3), BRIDF_ANGE(2,3),
```

```
     & BRIDFOR1_ANGE(2,3,3), TEMPRBRI_ANGE(2,3,3),
     & BRIDFOR2_ANGE(2,3,3), RTBA_ANGE(2,3,3), IB(2)

      REAL DRPBR_ANGE(2,3,3)

      REAL TEMPRE0TET(3), TEMPPOS(3)

      REAL TETMASS, CDA

c
      DATA ICRS / 2, 3, 1 / , JCRS / 3, 1, 2 /

      DATA DEGREES / 0.0174532925 /
c

c----------------------------------------------------------------------
c     Determine the transformation tensor and the partial derivatives.
c     This part is mainly copied from other subroutines and can be found
c     in the theory document at ....
c----------------------------------------------------------------------
      BANK   = PARAM(KPBANK,IPNT)
      ELEV   = PARAM(KPELEV,IPNT)
      HEAD   = PARAM(KPHEAD,IPNT)

      DO K = 1, 3
         POS(K)  = PARAM(KPPOS(K),IPNT)
         TEMPPOS(K) = POS(K)
         IF (POS(K).EQ.0) THEN
           POS(K) = 1e-10
         ENDIF
      ENDDO

      ANG(1) = BANK
      ANG(2) = ELEV
      ANG(3) = HEAD

      CALL ROTENS3(ANG,TE,TE_A)

      EUL(1) = ANG(1)
      EUL(2) = ANG(2)
      EUL(3) = ANG(3)

      DO K = 1,3
        DO L = 1,3
           TETR(K,L) = TE(L,K)
        ENDDO
      ENDDO

      DO J=1,3
         DO L=1,3
           DO K=1,3
              TETR_ANGE(K,L,J) = TE_A(L,K,J)
           ENDDO
         ENDDO
      ENDDO

      NCBRID = 0

      DO 200 KP=1, NPYLO
       IF(KPTYPE(KP).NE.4   .OR.
     &    ISPYLO(KP).NE.IS       ) GO TO 200

      NCBRID = NCBRID+1

C---- interval where pylon is attached
      IEB(NCBRID) = IPYLO(KP)
C
```

```
C---- node to which pylon is effectively attached
      IB(NCBRID) = MIN( MAX(IEB(NCBRID) ,IFRST(IS)) , ILAST(IS) )

       IEQ = IEB(NCBRID)
       I   = IB(NCBRID)

c---- set tether force derivatives to zero
      DO L = 1, 18
         TETF_Q(L,I) = 0
      ENDDO

      DO L= 1,6
         TETF_UT(L,I) = 0
      ENDDO

      DO L=1, NRHS
         TETF_GL( L) = 0
         TETF_GLT(L) = 0
      ENDDO
C
C---- set local Euler angles and transformation tensor for undeformed state
      ANG(1) = Q0(4,I)
      ANG(2) = Q0(5,I)
      ANG(3) = Q0(6,I)
      CALL ROTENS(ANG, T0,T_ANG, KBTYPE(IS))
C
C---- set local Euler angles and transformation tensor
      ANG(1) = Q(4,I,IPNT)
      ANG(2) = Q(5,I,IPNT)
      ANG(3) = Q(6,I,IPNT)
      CALL ROTENS(ANG, T  ,T_ANG, KBTYPE(IS))

C
C          t
C---- set T To matrix
      DO K = 1, 3
        DO L = 1, 3
          TNET(K,L) = T(1,K)*T0(1,L)
     &              + T(2,K)*T0(2,L)
     &              + T(3,K)*T0(3,L)

          DO J = 1, 3
            TNET_ANG(K,L,J) = T_ANG(1,K,J)*T0(1,L)
     &                      + T_ANG(2,K,J)*T0(2,L)
     &                      + T_ANG(3,K,J)*T0(3,L)

          ENDDO
        ENDDO
      ENDDO
C
C---- set pylon vector and location of pylon end
      DO K = 1, 3
        DRPBR(NCBRID,K) = TNET(K,1)*(QPYLO(1,KP)-Q0(1,I))
     &                  + TNET(K,2)*(QPYLO(2,KP)-Q0(2,I))
     &                  + TNET(K,3)*(QPYLO(3,KP)-Q0(3,I))

        DO L = 1, 3
          DRPBR_ANG(NCBRID,K,L) = TNET_ANG(K,1,L)*(QPYLO(1,KP)-Q0(1,I))
     &                          + TNET_ANG(K,2,L)*(QPYLO(2,KP)-Q0(2,I))
     &                          + TNET_ANG(K,3,L)*(QPYLO(3,KP)-Q0(3,I))

          DRPBR_ANGE(NCBRID,K,L) = TE_A(K,1,L)*(QPYLO(1,KP))
     &                           + TE_A(K,2,L)*(QPYLO(2,KP))
     &                           + TE_A(K,3,L)*(QPYLO(3,KP))
        ENDDO
      ENDDO
```

```
c------------------------------------------------------------------
c                                _          _       =        _      c
c       determine the bridle vector (r_bri = r_i + T_net Dr_b0) and the c
c       derivatives of the bridle verctor with respect to r and theta.  c
c                                                                   c
c------------------------------------------------------------------

      DO K =1, 3
         RBRI(NCBRID,K) = Q(K,I,IPNT) + DRPBR(NCBRID,K)

         DO L =1,3
            RBRI_ANG(NCBRID,K,L) = DRPBR_ANG(NCBRID,K,L)

            IF (K .EQ. L) THEN
               RBRI_R(NCBRID,K,L) = 1
            ELSE
               RBRI_R(NCBRID,K,L) = 0
            ENDIF
         ENDDO
      ENDDO

c--- determine the lenght of the bridles. The bridles are assumed rigid
c    and hence got a constant length
      LBRID(NCBRID) = SQRT((QPYLO(1,KP)-QPYLO(4,KP))**2 +
     &                     (QPYLO(2,KP)-QPYLO(5,KP))**2 +
     &                     (QPYLO(3,KP)-QPYLO(6,KP))**2 )

       IF (NCBRID.EQ.NBRID) THEN

c------------------------------------------------------------------
c        determine some bridle parameters needed in the tether loop
c------------------------------------------------------------------

      DO K = 1, NPYLO
        IF(KPTYPE(K).EQ.3) THEN
           RE0TET(1) = QPYLO(4,K)
           RE0TET(2) = QPYLO(5,K)
           RE0TET(3) = QPYLO(6,K)
           TETMASS   = QPYLO(9,K)
           CDA       = QPYLO(10,K)
        ENDIF
      ENDDO

      DO L = 1,2
         DO K =1,3
            TEMPRE0TET(K) = RE0TET(K)
            TEMPPOS   (K) = POS   (K)
         ENDDO
      ENDDO

c--- set the bridle postion and determine bridle derivatives
      CALL SETBRI(RBRI, LBRID, IS,EUL,POS, RTBA, RE0TET, RBRI_ANG,
     & RBRI_R, RTBA_ANG, RTBA_R, RTBA_POS, RTBA_ANGE, DRPBR_ANGE)


c--- set position RE0TET back again
      DO K = 1, NPYLO
        IF(KPTYPE(K).EQ.3) THEN
           RE0TET(1) = QPYLO(4,K)
           RE0TET(2) = QPYLO(5,K)
           RE0TET(3) = QPYLO(6,K)
        ENDIF
      ENDDO

c--- determine RBRI derivatives wrt to local parameters
      DO N=1,2
        DO K=1,3
```

```
          DO L=1,3
             RBRI_ANG(N,K,L) = DRPBR_ANG(N,K,L)

              IF (K .EQ. L) THEN
                 RBRI_R(N,K,L) = 1
              ELSE
                 RBRI_R(N,K,L) = 0
              ENDIF
              TEMPRBRI_R   (N,K,L) = RBRI_R   (N,K,L)
              TEMPRBRI_ANG(N,K,L) = RBRI_ANG(N,K,L)
           ENDDO
         ENDDO
       ENDDO

c---- derivatives RTET with respect to RTET(K)
      DO K =1,3
         DO J=1,3
            IF (K .EQ. J) THEN
               RTET_RTET(K,J) = 1
            ELSE
               RTET_RTET(K,J) = 0
            ENDIF
         ENDDO
      ENDDO


c--- determine the lenght of the tether in aircraft xyz coordinates
      DO K =1,3
         RTET(K) = (TE(1,K)*(TEMPRE0TET(1) - TEMPPOS(1)))
     &            + (TE(2,K)*(TEMPRE0TET(2) - TEMPPOS(2)))
     &            + (TE(3,K)*(TEMPRE0TET(3) - TEMPPOS(3))) - RTBA(K)

      ENDDO

      LTET = SQRT(RTET(1)**2+RTET(2)**2+RTET(3)**2)

c--- get the unstressed tether length and spring constant
      DO K = 1, NPYLO
        IF(KPTYPE(K).EQ.3) THEN
          LTET0 = SQRT((QPYLO(1,K)-QPYLO(4,K))**2 +
     &                 (QPYLO(2,K)-QPYLO(5,K))**2 +
     &                 (QPYLO(3,K)-QPYLO(6,K))**2)
         ENDIF
       ENDDO

c--- determine tether force magnitude and vector
       TETF = KSPR*(LTET-LTET0)

       WRITE(*,*) 'ASWINGb.TETF(i) =', TETF, ';'

      DO K = 1,3
         TETFOR(K) = (RTET(K)/LTET) * TETF
      ENDDO

c--- include tether weight and drag
      CALL SETWGTTET(IS,IPNT,TETMASS, CDA, TE,TEMPRE0TET, TEMPPOS,
     & RTBA_ANG, RTBA, TETFOR)

c--- determine tether force magnitude and vector
       TETF =SQRT(TETFOR(1)**2+TETFOR(2)**2+TETFOR(3)**2)
       PARAM(KPTETF,IPNT) = TETF

c                    *
c--- determine r_bri; normalized r_bri
      IF (TETF .NE. 0) THEN
        DO K = 1,2
           DO L = 1,3
```

```
                DO N = 1,3
                   RBRIST(K,L) = (RBRI(K,L)-RTBA(L))/LBRID(K)

                   RBRIST_R   (K,L,N) = (RBRI_R   (K,L,N)-RTBA_R   (K,L,N))
     &                                / LBRID(K)
                   RBRIST_ANG(K,L,N) = (RBRI_ANG(K,L,N)-RTBA_ANG(K,L,N))
     &                                / LBRID(K)

                ENDDO
              ENDDO
           ENDDO

c--- determine magnitude of bridle forces
            BRIDF(2) = (TETFOR(1) - (RBRIST(1,1)*(TETFOR(1)-TETFOR(2))/
     & (RBRIST(1,1)-RBRIST(1,2)))) / ((RBRIST(1,1)*(RBRIST(2,1)-
     & RBRIST(2,2))/(RBRIST(1,1)-RBRIST(1,2))) - RBRIST(2,1))

           BRIDF(1) = -(TETFOR(2) + (((RBRI(2,2)-RTBA(2))/LBRID(2))
     &                 *BRIDF(2))) /
     &                 ((RBRI(1,2)-RTBA(2))/LBRID(1))

       ELSE
           BRIDF(1) = 0
           BRIDF(2) = 0
       ENDIF

c--- determine bridle force vector
       DO K = 1,2
          DO L = 1,3
             BRIDFOR(K,L) = RBRIST(K,L)*BRIDF(K)
          ENDDO
       ENDDO

c-----------------------------------------------------------------------
c
c     derivatives with respect to the tether force vector.
c
c-----------------------------------------------------------------------

c---- derivative of RTET wrt POS and ANGE
       DO K = 1,3
          DO L = 1, 3
             RTET_POS(K,L) = -TETR(L,K)

             RTET_ANGE(K,L) =
     & (((TEMPRE0TET(1)-TEMPPOS(1))*TE_A(1,K,L))+
     &  ((TEMPRE0TET(2)-TEMPPOS(2))*TE_A(2,K,L))+
     &  ((TEMPRE0TET(3)-TEMPPOS(3))*TE_A(3,K,L))) - RTBA_ANGE(1,K,L)

          ENDDO
       ENDDO

c--- derivatives of LTET wrt POS and ANGE
       DO K = 1,3

           LTET_POS(K) = ((RTET_POS(1,K)*RTET(1))
     &     + (RTET_POS(2,K)*RTET(2)) + (RTET_POS(3,K) * RTET(3))) / LTET

           LTET_ANGE(K) = ((RTET_ANGE(1,K)*RTET(1))
     &     + (RTET_ANGE(2,K)*RTET(2)) + (RTET_ANGE(3,K) * RTET(3))) /LTET

c--- derivatives TETF wrt POS and ANGE
           TETF_POS  (K)      = KSPR * LTET_POS  (K)
           TETF_ANGE(K)       = KSPR * LTET_ANGE(K)

c--- set the derivatives TETF wrt POS into a more appropriate array
           TETF_GL(LPOS(K)) = TETF_POS(K)
```

```fortran
      ENDDO

c--- set the derivatives TETF wrt ANGE into a more appropriate array
       TETF_GL(LBANK)    = TETF_ANGE(1)  !PHI
       TETF_GL(LELEV)    = TETF_ANGE(2)  !THETA
       TETF_GL(LHEAD)    = TETF_ANGE(3)  !PSI

c--- derivative of RBRI with respect to r and ANG

      DO N=1,2

      DO NBRI=1,2
        DO K =1,3
          DO L=1,3
             RBRI_R   (NBRI,K,L) = TEMPRBRI_R   (NBRI,K,L)
             RBRI_ANG(NBRI,K,L) = TEMPRBRI_ANG(NBRI,K,L)
          ENDDO
        ENDDO
      ENDDO

      IF (N.EQ.1) THEN
         DO K=1,3
            DO L =1,3
              RBRI_ANG(2,K,L) = 0
              RBRI_R   (2,K,L) = 0
            ENDDO
         ENDDO
      ELSEIF (N.EQ.2) THEN
         DO K=1,3
            DO L =1,3
              RBRI_ANG(1,K,L) = 0
              RBRI_R   (1,K,L) = 0
            ENDDO
         ENDDO
      ENDIF

c---- derivative of RBRIST
        DO NBRI = 1,2
           DO K = 1,3
             DO L = 1,3
               RBRIST  (NBRI,K  ) = (RBRI  (NBRI,K  )-RBTA  (K    ))
     &                                    / LBRID(NBRI)

          RBRIST_R   (NBRI,K,L) = (RBRI_R   (NBRI,K,L)-RTBA_R   (N,K,L))
     &                          / LBRID(NBRI)
          RBRIST_ANG(NBRI,K,L) = (RBRI_ANG(NBRI,K,L)-RTBA_ANG(N,K,L))
     &                          / LBRID(NBRI)

         RBRIST_POS(NBRI,K,L) = ((-1)*RTBA_POS(N,K,L)) /LBRID(NBRI)

         RBRIST_ANGE(NBRI,K,L) = ((-1)*RTBA_ANGE(N,K,L))
     &                          / LBRID(NBRI)

             ENDDO
           ENDDO
        ENDDO

c------------------------------------------------------------------------
c      tether force magnitude derivatives wrt R and ANG
c------------------------------------------------------------------------
      DO K= 1,3
         LTET_R   (N,K)=-((RTBA_R(N,1,K)*RTET(1))
     &                  +(RTBA_R(N,2,K)*RTET(2))
     &                  +(RTBA_R(N,3,K)*RTET(3)) ) / LTET

         LTET_ANG(N,K)=  -((RTBA_ANG(N,1,K)*RTET(1))
```

```
     &                    +(RTBA_ANG(N,2,K)*RTET(2))
     &                    +(RTBA_ANG(N,3,K)*RTET(3))  )  /  LTET

          TETF_R   (N,K)  = KSPR * LTET_R   (N,K)
          TETF_ANG(N,K)  = KSPR * LTET_ANG(N,K)

          TETF_Q(K,I)    = TETF_R   (N,K)  !R
          TETF_Q(K+3,I)  = TETF_ANG(N,K)  !THETA

      ENDDO

c-----------------------------------------------------------------------
c     set tether force derivatives
c-----------------------------------------------------------------------


c---- general partial derivatives
      DO K = 1,3

          TETFOR_RTETK      = TETF / LTET
          TETFOR_LTET(K)    = (-1)*(RTET(K)/LTET**2)*TETF
          TETFOR_TETF(K)    =  RTET(K)/LTET

      ENDDO

c---- avoid singularities
      DO K = 1, 3
          IF (RTET(K).EQ.0) THEN
            RTET(K) = 1e-12
          ENDIF
      ENDDO

      DO K = 1,3
         DO L = 1,3

             TETFOR_R    (N,L,K) = ((-1)*TETFOR_RTETK * RTBA_R(N,L,K))
     &       + (TETFOR_LTET(L)   * LTET_R(N,K))
     &       + (TETFOR_TETF(L)   * TETF_R(N,K))

              TETFOR_ANG(N,L,K) = ((-1)*TETFOR_RTETK * RTBA_ANG(N,L,K))
     &       + (TETFOR_LTET(L)   * LTET_ANG(N,K))
     &       + (TETFOR_TETF(L)   * TETF_ANG(N,K))

             TETFOR_POS(L,K) = (TETFOR_RTETK * (RTET_POS(L,K) -
     &                                         RTBA_POS(N,L,K)))
     &       + (TETFOR_LTET(L)   * LTET_POS(K))
     &       + (TETFOR_TETF(L)   * TETF_POS(K))

             TETFOR_ANGE(L,K) = (TETFOR_RTETK * (RTET_ANGE(L,K)))
     &       + (TETFOR_LTET(L)   * LTET_ANGE(K))
     &       + (TETFOR_TETF(L)   * TETF_ANGE(K))

         ENDDO
      ENDDO

c-----------------------------------------------------------------------
c      determine the derivatives of the bridle forces. This part of the
c      code is less intuitive to follow without any reference. Please
c      use part ... of report ... as a reference.
c-----------------------------------------------------------------------

      DO K =1,3

c--- derivatives of bridle 2
         BRIDFTOP(2) = (TETFOR(3) - (RBRIST(1,3)*(TETFOR(3)-TETFOR(2))
     &                / (RBRIST(1,3)-RBRIST(1,2))))
         BRIDFBOT(2) = ((RBRIST(1,3)*(RBRIST(2,3)-RBRIST(2,2))
```

```
     &              /(RBRIST(1,3)-RBRIST(1,2))) - RBRIST(2,3))

        VARA         =  RBRIST(1,3)*(TETFOR(3)-TETFOR(2))
     &              /  (RBRIST(1,3) - RBRIST  (1,2))

        VARATOP      = RBRIST  (1,3  )*(TETFOR(3)-TETFOR(2))

       VARATOP_R(K) = (RBRIST_R(1,3,K)*
     &                 (TETFOR  (3    )-TETFOR  (2    )))
     &               +  (RBRIST(1,3   )
     &               *  (TETFOR_R(N,3,K)-TETFOR_R(N,2,K)))

       VARATOP_ANG(K) = (RBRIST_ANG(1,3,K)*
     &                 (TETFOR  (3    )-TETFOR  (2    )))
     &               +  (RBRIST(1,3   )
     &               *  (TETFOR_ANG(N,3,K)-TETFOR_ANG(N,2,K)))

       VARATOP_POS(K) = (RBRIST_POS(1,3,K)*
     &                   (TETFOR  (3    )-TETFOR  (2    )))
     &                 +  (RBRIST(1,3   )
     &                 *  (TETFOR_POS(3,K)-TETFOR_POS(2,K)))

       VARATOP_ANGE(K) = (RBRIST_ANGE(1,3,K)*
     &                   (TETFOR  (3    )-TETFOR  (2    )))
     &                 +  (RBRIST(1,3   )
     &                 *  (TETFOR_ANGE(3,K)-TETFOR_ANGE(2,K)))

        VARABOT      = RBRIST  (1,3   ) - RBRIST  (1,2)

        VARABOT_R   (K) = RBRIST_R   (1,3,K) - RBRIST_R   (1,2,K)
        VARABOT_ANG (K) = RBRIST_ANG (1,3,K) - RBRIST_ANG (1,2,K)
        VARABOT_POS (K) = RBRIST_POS (1,3,K) - RBRIST_POS (1,2,K)
        VARABOT_ANGE(K) = RBRIST_ANGE(1,3,K) - RBRIST_ANGE(1,2,K)

        VARA_R(K)      = ((VARATOP_R(K)*VARABOT)-(VARATOP*VARABOT_R(K)))
     &              /    VARABOT**2
        VARA_ANG(K)    = ((VARATOP_ANG(K)*VARABOT)
     & -(VARATOP*VARABOT_ANG(K)))  /   VARABOT**2

        VARA_POS(K)     = ((VARATOP_POS(K)*VARABOT)-
     &  (VARATOP*VARABOT_POS(K)))  /   VARABOT**2

        VARA_ANGE(K)    = ((VARATOP_ANGE(K)*VARABOT)-
     &  (VARATOP*VARABOT_ANGE(K)))  /   VARABOT**2


        VARB            = (RBRIST(2,3)-RBRIST(2,2))
     &                  / (RBRIST(1,3)-RBRIST(1,2))

        VARBTOP          = RBRIST      (2,3   )-RBRIST      (2,2)
        VARBTOP_R    (K) = RBRIST_R    (2,3,K)-RBRIST_R    (2,2,K)
        VARBTOP_ANG  (K) = RBRIST_ANG  (2,3,K)-RBRIST_ANG  (2,2,K)
        VARBTOP_POS  (K) = RBRIST_POS  (2,3,K)-RBRIST_POS  (2,2,K)
        VARBTOP_ANGE(K)  = RBRIST_ANGE(2,3,K)-RBRIST_ANGE(2,2,K)


        VARBBOT          = RBRIST      (1,3   )-RBRIST      (1,2   )
        VARBBOT_R    (K) = RBRIST_R    (1,3,K)-RBRIST_R    (1,2,K)
        VARBBOT_ANG  (K) = RBRIST_ANG  (1,3,K)-RBRIST_ANG  (1,2,K)
        VARBBOT_POS  (K) = RBRIST_POS  (1,3,K)-RBRIST_POS  (1,2,K)
        VARBBOT_ANGE(K)  = RBRIST_ANGE(1,3,K)-RBRIST_ANGE(1,2,K)

        VARB_R(K)      = ((VARBTOP_R(K)*VARBBOT)-(VARBTOP*VARBBOT_R(K)))
     &              /    VARBBOT**2
        VARB_ANG(K)     = ((VARBTOP_ANG(K)*VARBBOT)-
     & (VARBTOP*VARBBOT_ANG(K)))    /   VARBBOT**2
```

```
          VARB_POS(K)       = ((VARBTOP_POS(K)*VARBBOT)-
     &    (VARBTOP*VARBBOT_POS(K)))    /    VARBBOT**2

          VARB_ANGE(K)       = ((VARBTOP_ANGE(K)*VARBBOT)-
     &    (VARBTOP*VARBBOT_ANGE(K)))    /    VARBBOT**2

          BRIDFTOP_R     (2,K)  = TETFOR_R    (N,3,K)  - VARA_R    (K)
          BRIDFTOP_ANG   (2,K)  = TETFOR_ANG  (N,3,K)  - VARA_ANG  (K)
          BRIDFTOP_POS   (2,K)  = TETFOR_POS  (   3,K)  - VARA_POS  (K)
          BRIDFTOP_ANGE(2,K)  = TETFOR_ANGE(   3,K)  - VARA_ANGE(K)

          BRIDFBOT_R(2,K)  = RBRIST_R(1,3,K)*VARB + RBRIST(1,3)*VARB_R(K)
     &                     - RBRIST_R(2,3,K)
          BRIDFBOT_ANG(2,K)  = RBRIST_ANG(1,3,K)*VARB
     & + RBRIST(1,3)*VARB_ANG(K)  - RBRIST_ANG(2,3,K)
          BRIDFBOT_POS(2,K)  = RBRIST_POS(1,3,K)*VARB +
     &    RBRIST(1,3)*VARB_POS(K)  - RBRIST_POS(2,3,K)
          BRIDFBOT_ANGE(2,K)  = RBRIST_ANGE(1,3,K)*VARB +
     &    RBRIST(1,3)*VARB_ANGE(K)  - RBRIST_ANGE(2,3,K)


        BRIDF_R(2,K)  = (BRIDFTOP_R(2,K)*BRIDFBOT   (2    )
     &                 -   BRIDFTOP   (2    )*BRIDFBOT_R(2,K))/BRIDFBOT(2)**2
        BRIDF_ANG(2,K)  = (BRIDFTOP_ANG(2,K)*BRIDFBOT   (2    )
     &               -   BRIDFTOP   (2    )*BRIDFBOT_ANG(2,K))/BRIDFBOT(2)**2

        BRIDF_POS(2,K)  = (BRIDFTOP_POS(2,K)*BRIDFBOT   (2    )
     &                 -   BRIDFTOP(2    )*BRIDFBOT_POS(2,K))/BRIDFBOT(2)**2
        BRIDF_ANGE(2,K)  = (BRIDFTOP_ANGE(2,K)*BRIDFBOT   (2    )
     &                 -   BRIDFTOP(2    )*BRIDFBOT_ANGE(2,K))/BRIDFBOT(2)**2

c--- derivatives of bridle 1
        BRIDFTOP(1)    = -(TETFOR(2)  + (RBRIST(2,2)  *BRIDF(2)))
        BRIDFBOT(1)    = RBRIST(1,2)

        BRIDFTOP_R(1,K)  = -(TETFOR_R(N,2,K)+(RBRIST_R(2,2,K)*BRIDF(2))
     &                     +   (RBRIST(2,2)*BRIDF_R(2,K)))
        BRIDFTOP_ANG(1,K)  = -(TETFOR_ANG(N,2,K)+(RBRIST_ANG(2,2,K)
     & *BRIDF(2))
     &                     +   (RBRIST(2,2)*BRIDF_ANG(2,K)))
        BRIDFTOP_POS(1,K)  = -(TETFOR_POS(2,K)+
     & (RBRIST_POS(2,2,K)*BRIDF(2))  + (RBRIST(2,2)*BRIDF_POS(2,K)))
        BRIDFTOP_ANGE(1,K)  = -(TETFOR_ANGE(2,K)+
     & (RBRIST_ANGE(2,2,K)*BRIDF(2))  + (RBRIST(2,2)*BRIDF_ANGE(2,K)))

        BRIDFBOT_R     (1,K)  =   RBRIST_R    (1,2,K)
        BRIDFBOT_ANG   (1,K)  =   RBRIST_ANG  (1,2,K)
        BRIDFBOT_POS   (1,K)  =   RBRIST_POS  (1,2,K)
        BRIDFBOT_ANGE(1,K)  =   RBRIST_ANGE(1,2,K)

        BRIDF_R(1,K)  = (BRIDFTOP_R(1,K)*BRIDFBOT   (1    )
     &                -  BRIDFTOP    (1    )*BRIDFBOT_R(1,K))/BRIDFBOT(1)**2
        BRIDF_ANG(1,K)  = (BRIDFTOP_ANG(1,K)*BRIDFBOT   (1    )
     &                -  BRIDFTOP    (1    )*BRIDFBOT_ANG(1,K))/BRIDFBOT(1)**2
        BRIDF_POS(1,K)  = (BRIDFTOP_POS(1,K)*BRIDFBOT   (1    )
     &                -  BRIDFTOP    (1    )*BRIDFBOT_POS(1,K))/BRIDFBOT(1)**2
        BRIDF_ANGE(1,K)  = (BRIDFTOP_ANGE(1,K)*BRIDFBOT   (1    )
     &                -  BRIDFTOP    (1    )*BRIDFBOT_ANGE(1,K))/BRIDFBOT(1)**2

        ENDDO

         DO K=1,3
          DO L=1,3
              BRIDFOR1_R(N,K,L)  = BRIDF_R(1,L)  *  RBRIST(1,K)  +
     & BRIDF(1)  *  RBRIST_R(1,K,L)
```

```
            BRIDFOR1_ANG(N,K,L) = BRIDF_ANG(1,L) * RBRIST(1,K) +
     & BRIDF(1) * RBRIST_ANG(1,K,L)
            BRIDFOR1_POS(N,K,L) = BRIDF_POS(1,L) * RBRIST(1,K) +
     & BRIDF(1) * RBRIST_POS(1,K,L)
            BRIDFOR1_ANGE(N,K,L) = BRIDF_ANGE(1,L) * RBRIST(1,K) +
     & BRIDF(1) * RBRIST_ANGE(1,K,L)

            BRIDFOR2_R(N,K,L) = BRIDF_R(2,L) * RBRIST(2,K) +
     & BRIDF(2) * RBRIST_R(2,K,L)

            BRIDFOR2_ANG(N,K,L) = BRIDF_ANG(2,L) * RBRIST(2,K) +
     & BRIDF(2) * RBRIST_ANG(2,K,L)
            BRIDFOR2_POS(N,K,L) = BRIDF_POS(2,L) * RBRIST(2,K) +
     & BRIDF(2) * RBRIST_POS(2,K,L)
            BRIDFOR2_ANGE(N,K,L) = BRIDF_ANGE(2,L) * RBRIST(2,K) +
     & BRIDF(2) * RBRIST_ANGE(2,K,L)

          ENDDO
          ENDDO

c-----------------------------------------------------------------------
c     set the tether force derivaives in an appropriate way
c-----------------------------------------------------------------------

      DO K=1,3
        DO L = 1, 18
          F_Q(K,L) = 0.
        ENDDO
        DO L = 1, 6
          F_QT(K,L) = 0.
        ENDDO
        DO L = 1, NRHS
          F_GL(K,L) = 0.
          F_GLT(K,L) = 0.
        ENDDO
        DO L = 1, NFRP
          F_FRP(K,L) = 0.
        ENDDO
      ENDDO

      DO K =1, 3
        F(K)                = (-1)*BRIDFOR(N,K)  !Fb()

        IF (N.EQ.1) THEN
          F_Q(K,1)            = (-1)*BRIDFOR1_R(N,K,1)  !x
          F_Q(K,2)            = (-1)*BRIDFOR1_R(N,K,2)  !y
          F_Q(K,3)            = (-1)*BRIDFOR1_R(N,K,3)  !z

          F_Q(K+3,1)          = (-1)*BRIDFOR1_ANG(N,K,1)  !phi
          F_Q(K+3,2)          = (-1)*BRIDFOR1_ANG(N,K,2)  !theta
          F_Q(K+3,3)          = (-1)*BRIDFOR1_ANG(N,K,3)  !psi

          F_GL(K,LPOS(1))     = (-1)*BRIDFOR1_POS(N,K,1)  !X
          F_GL(K,LPOS(2))     = (-1)*BRIDFOR1_POS(N,K,2)  !Y
          F_GL(K,LPOS(3))     = (-1)*BRIDFOR1_POS(N,K,3)  !Z

          F_GL(K,LBANK)       = (-1)*BRIDFOR1_ANGE(N,K,1)  !PHI
          F_GL(K,LELEV)       = (-1)*BRIDFOR1_ANGE(N,K,2)  !THETA
          F_GL(K,LHEAD)       = (-1)*BRIDFOR1_ANGE(N,K,3)  !PSI

        ELSE
          F_Q(K,1)            = (-1)*BRIDFOR2_R(N,K,1)  !x
          F_Q(K,2)            = (-1)*BRIDFOR2_R(N,K,2)  !y
          F_Q(K,3)            = (-1)*BRIDFOR2_R(N,K,3)  !z

          F_Q(K+3,1)          = (-1)*BRIDFOR2_ANG(N,K,1)  !phi
          F_Q(K+3,2)          = (-1)*BRIDFOR2_ANG(N,K,2)  !theta
```

```
            F_Q(K+3,3)          = (-1)*BRIDFOR2_ANG(N,K,3)  !psi

            F_GL(K,LPOS(1))     = (-1)*BRIDFOR2_POS(N,K,1)  !X
            F_GL(K,LPOS(2))     = (-1)*BRIDFOR2_POS(N,K,2)  !Y
            F_GL(K,LPOS(3))     = (-1)*BRIDFOR2_POS(N,K,3)  !Z

            F_GL(K,LBANK)       = (-1)*BRIDFOR2_ANGE(N,K,1)  !PHI
            F_GL(K,LELEV)       = (-1)*BRIDFOR2_ANGE(N,K,2)  !THETA
            F_GL(K,LHEAD)       = (-1)*BRIDFOR2_ANGE(N,K,3)  !PSI
          ENDIF

        ENDDO

C---- set moment vector  DRP x FW  and derivatives
c     these are assumed to be
        DO K=1, 3

          IC = ICRS(K)
          JC = JCRS(K)

          M(K) =           DRPBR(N,IC)*F(JC)
     &                   - DRPBR(N,JC)*F(IC)

          TETMNT(K) = M(K)

          DO L=1, 18
            M_Q(K,L)     = DRPBR(N,IC)*F_Q(JC,L)
     &                   - DRPBR(N,JC)*F_Q(IC,L)
          ENDDO
          DO L=1, 6
            M_QT(K,L)    = DRPBR(N,IC)*F_QT(JC,L)
     &                   - DRPBR(N,JC)*F_QT(IC,L)
          ENDDO
          DO L=1, NRHS
            M_GL(K,L)    = DRPBR(N,IC)*F_GL(JC,L)
     &                   - DRPBR(N,JC)*F_GL(IC,L)
C
            M_GLT(K,L)   = DRPBR(N,IC)*F_GLT(JC,L)
     &                   - DRPBR(N,JC)*F_GLT(IC,L)
          ENDDO
          DO L=1, NFRP
            M_FRP(K,L)   = DRPBR(N,IC)*F_FRP(JC,L)
     &                   - DRPBR(N,JC)*F_FRP(IC,L)
          ENDDO
          DO L=1, 3
            M_Q(K,L+3)   = M_Q(K,L+3)
     &                   + DRPBR_ANG(N,IC,L)*F(JC)
     &                   - DRPBR_ANG(N,JC,L)*F(IC)
          ENDDO
        ENDDO


         IEQ = IEB(N)
         I   = IB(N)

C---- add residual and Jacobian changes to appropriate slots
        DO K = 1, 3
C------ set row-major indexed arrays for calling EQNADD
          F(K) = (-1)*BRIDFOR(N,K)

          IF(IEQ.EQ.IFRST(IS)-1) THEN
           KEQF = KEQ0(K+9,IS)
           KEQM = KEQ0(K+6,IS)
           DO L = 1, 18
             FK_QO(L) = F_Q(K,L)
             FK_QP(L) = 0.
             MK_QO(L) = M_Q(K,L)
```

```
              MK_QP(L) = 0.

            ENDDO
            DO L = 1, 6
              FK_QTO(L) = F_QT(K,L)
              FK_QTP(L) = 0.
              MK_QTO(L) = M_QT(K,L)
              MK_QTP(L) = 0.

            ENDDO
          ELSE
            KEQF = KEQ(K+9,IEQ)
            KEQM = KEQ(K+6,IEQ)
            DO L = 1, 18
              FK_QO(L) = 0.
              FK_QP(L) = F_Q(K,L)
              MK_QO(L) = 0.
              MK_QP(L) = M_Q(K,L)

            ENDDO
            DO L = 1, 6
              FK_QTO(L) = 0.
              FK_QTP(L) = F_QT(K,L)
              MK_QTO(L) = 0.
              MK_QTP(L) = M_QT(K,L)

            ENDDO
          ENDIF
C
          DO L = 1, NRHS
            FK_GL(L) = F_GL(K,L)
            MK_GL(L) = M_GL(K,L)


            FK_GLT(L) = F_GLT(K,L)
            MK_GLT(L) = M_GLT(K,L)
          ENDDO
          DO L = 1, NFRP
            FK_FRP(L) = F_FRP(K,L)
            MK_FRP(L) = M_FRP(K,L)
          ENDDO

          CALL EQNADD(K+9,IEQ, KEQF, F(K), FK_QO, FK_QP,
     &                                      FK_QTO,FK_QTP,
     &                                      FK_GL  ,FK_GLT, FK_FRP)
          CALL EQNADD(K+6,IEQ, KEQM, M(K), MK_QO, MK_QP,
     &                                      MK_QTO,MK_QTP,
     &                                      MK_GL  ,MK_GLT,MK_FRP)

        ENDDO

C---- also add to reaction-force accumulators
        CALL FMDEL(NRHS,NFRP, 1.0, Q(1,I,IPNT),
     &            F, F_Q, F_QT, F_GL(1,1), F_GLT(1,1), F_FRP(1,1),
     &            M, M_Q, M_QT, M_GL(1,1), M_GLT(1,1), M_FRP(1,1),
     & RFORCE,RFOR_Q(1,1,I),RFOR_UT(1,1,I),RFOR_GL(1,1),RFOR_GLT(1,1),
     &                                                  RFOR_FRP(1,1),
     & RMOMNT,RMOM_Q(1,1,I),RMOM_UT(1,1,I),RMOM_GL(1,1),RMOM_GLT(1,1),
     &                                                  RMOM_FRP(1,1))

C---- if this is also a ground point, also add to ground-force accumulators
        DO KG=1, NGROU
          IF(IS.EQ.ISGROU(KG) .AND. IEQ.EQ.IGROU(KG)) THEN
           CALL FMDEL(NRHS,NFRP, 1.0, Q(1,I,IPNT),
     &              F, F_Q, F_QT, F_GL(1,1), F_GLT(1,1), F_FRP(1,1),
     &              M, M_Q, M_QT, M_GL(1,1), M_GLT(1,1), M_FRP(1,1),
     & GFORCE,GFOR_Q(1,1,I),GFOR_UT(1,1,I),GFOR_GL(1,1),GFOR_GLT(1,1),
```

```
     &                                                           GFOR_FRP(1,1),
     & GMOMNT,GMOM_Q(1,1,I),GMOM_UT(1,1,I),GMOM_GL(1,1),GMOM_GLT(1,1),
     &                                                           GMOM_FRP(1,1))
          ENDIF
      ENDDO



      ENDDO  !N

        ENDIF

 200  CONTINUE

      RETURN
      END ! SETTET
```

/Aswing/src/sloads.f

C.2 Tether-bridle attachment subroutine

```
      SUBROUTINE SETBRI(RBRI, LBRID, IS,ANG, POS, RTBA, RE0TET,
     & RBRI_ANG, RBRI_R, RTBA_ANG, RTBA_R, RTBA_POS,
     & RTBA_ANGE, DRPBR_ANGE)

      INCLUDE 'ASWING.INC'


      REAL  ANG(3), POS(3), RCIRC(3), DCIRC(3),
     & RETET(3), TE(3,3), TE_A(3,3,3), Y_R(3), Y_ANG(3),
     & Z1TOP_ANG(2,3), Z1TOP_R(2,3), Z1BOT_ANG(2,3), Z1BOT_R(2,3),
     & Z2TOP_ANG(2,3), Z2TOP_R(2,3), Z2BOT_ANG(2,3), Z2BOT_R(2,3),
     & RCIRC2TOP_ANG(2,3),  RCIRC2TOP_R(2,3),  RCIRC2BOT_ANG(2,3),
     & RCIRC2BOT_R(2,3), RCIRC2_ANG(2,3), RCIRC2_R(2,3), X_ANG(3),
     & X_R(3), ZRT_ANG(2,3), ZRT_R(2,3), Z1_ANG(2,3), Z1_R(2,3),
     & Z2_ANG(2,3), Z2_R(2,3), RCIRC_ANG(2,3,3), RCIRC_R(2,3,3),
     & RCIRCMIN_ANG(2,3,3), RCIRCMIN_R(2,3,3), RE0TET(3)

      REAL DOTDCIRC, MINDOTCIRC, ZRT, Z_1, Z_2, X_1, X_2, Z1TOP, Z2TOP,
     &  RCIRC2TOP, RCIRC2BOT, RBRI_ANG(2,3,3), RBRI_R(2,3,3),
     & TEMPRBRI(2,3), TETR(3,3), RCIRCMINTEMP(3),TEMPRBRI_R(2,3,3),
     & TEMPRBRI_ANG(2,3,3), Z1BOT, Z2BOT, TEMPX_R(3), TEMPY_R(3),
     & TEMPZ_R(3), TEMPX_ANG(3), TEMPY_ANG(3), TEMPZ_ANG(3)

      INTEGER M, NBRI

      REAL TheTerm

      REAL DISTB1B2, RSSIC, DISTB1CC, DISTB2CC, COEFA, COEFB, COEFC,
     & COEFD, RETETS(3), LRESCC, CIRCCENT(3), DISTRESCC(3)

      REAL DISTB1B2_ANG(2,3), DISTB1B2_R(2,3), RSSICTOP, RSSICBOT,
     & RSSICTOP_ANG(2,3), RSSICTOP_R(2,3),  RSSICBOT_ANG(2,3),
     & RSSICBOT_R(2,3),RSSIC_ANG(2,3), DISTB1CC_ANG(2,3),
     & DISTB1CC_R(2,3), DISTB2CC_ANG(2,3), DISTB2CC_R(2,3),
     & CIRCCENTTOP(3), CIRCCENTBOT(3), CIRCCENTTOP_ANG(2,3,3),
     & CIRCCENTTOP_R(2,3,3), CIRCCENTBOT_ANG(2,3),
     & CIRCCENTBOT_R(2,3), CIRCCENT_ANG(2,3,3), CIRCCENT_R(2,3,3),
     & COEF(4), COEF_ANG(2,4,3), COEF_R(2,4,3), RETETSTOP(3),
     & RETETSBOT, RETETSTOP_ANG(2,3,3), RETETSTOP_R(2,3,3),
     & RETETSBOT_ANG(2,3), RETETSBOT_R(2,3), RETETS_ANG(2,3,3),
     & RETETS_R(2,3,3), DISTRESCC_ANG(2,3,3), DISTRESCC_R(2,3,3),
     & LRESCC_ANG(2,3), LRESCC_R(2,3), RTBA_ANG(2,3,3), RTBA_R(2,3,3),
     & RSSIC_R(2,3)

      REAL DISTB1B2_POS(2,3), RSSICTOP_POS(2,3), RSSICBOT_POS(2,3),
     & RSSIC_POS(2,3), DISTB1CC_POS(2,3), DISTB2CC_POS(2,3),
     & CIRCCENTTOP_POS(2,3,3), CIRCCENTBOT_POS(2,3),
     & CIRCCENT_POS(2,3,3), COEF_POS(2,4,3), RETETSTOP_POS(2,3,3),
     & RETETSBOT_POS(2,3), RETETS_POS(2,3,3), DISTRESCC_POS(2,3,3),
     & LRESCC_POS(2,3), RTBA_POS(2,3,3), RBRI_POS(2,3,3),
     & RETET_POS(2,3,3)

      REAL DISTB1B2_ANGE(2,3), RSSICTOP_ANGE(2,3), RSSICBOT_ANGE(2,3),
     & RSSIC_ANGE(2,3), DISTB1CC_ANGE(2,3), DISTB2CC_ANGE(2,3),
     & CIRCCENTTOP_ANGE(2,3,3), CIRCCENTBOT_ANGE(2,3),
     & CIRCCENT_ANGE(2,3,3), COEF_ANGE(2,4,3), RETETSTOP_ANGE(2,3,3),
     & RETETSBOT_ANGE(2,3), RETETS_ANGE(2,3,3), DISTRESCC_ANGE(2,3,3),
     & LRESCC_ANGE(2,3), RTBA_ANGE(2,3,3), RBRI_ANGE(2,3,3),
     & RETET_ANGE(2,3,3), TETR_ANGE(3,3,3)

       REAL DRPBR_ANGE(2,3,3)

      REAL TEMPRETET(3), TEMPRE0TET(3), TEMPPOS(3)

```

```
c--- determine Earth transformation tensor and derivative
      CALL ROTENS3(ANG,TE,TE_A)

c---- set the RBRI
      DO L = 1,2
         DO K =1,3
            TEMPRE0TET(K) = RE0TET(K)
            TEMPPOS   (K) = POS   (K)

         ENDDO
      ENDDO

c--- set the transpose of the Earth transformation tensor
      DO K =1,3
        DO L =1,3
          TETR(K,L) = TE(L,K)
        ENDDO
      ENDDO


      DO J=1,3
         DO L=1,3
           DO K=1,3
             TETR_ANGE(K,L,J) = TE_A(L,K,J)
           ENDDO
         ENDDO
      ENDDO

c--- set POS and RE0TET in aircraft coordinates
      DO N=1,2
        DO K =1,3
           POS(K) = TEMPPOS(1) * TE(1,K) + TEMPPOS(2) * TE(2,K)
     &              + TEMPPOS(3) * TE(3,K)

           RE0TET(K) = TEMPRE0TET(1)*TE(1,K) + TEMPRE0TET(2)*TE(2,K)
     &                 + TEMPRE0TET(3)*TE(3,K)

          DO L=1,3
            RBRI_POS  (N,K,L) = 0
            RETET_POS(N,K,L) = -TETR(L,K)

            RBRI_ANGE  (N,K,L) = 0

            RETET_ANGE(N,K,L) =
     & (((TEMPRE0TET(1)-TEMPPOS(1))*TE_A(1,K,L))+
     &  ((TEMPRE0TET(2)-TEMPPOS(2))*TE_A(2,K,L))+
     &  ((TEMPRE0TET(3)-TEMPPOS(3))*TE_A(3,K,L)))

          ENDDO
        ENDDO
      ENDDO


      DO N=1,2
        DO M =1,3
          DO K=1,3

             TEMPRBRI_R   (N,M,K) = RBRI_R   (N,M,K)
             TEMPRBRI_ANG(N,M,K) = RBRI_ANG(N,M,K)

          ENDDO
        ENDDO
      ENDDO


      DO KP = 1, NPYLO
```

```
          IF(KPTYPE(KP).EQ.3) THEN
             TEMPRETET(1)   = QPYLO(4,KP) - TEMPPOS(1)
             TEMPRETET(2)   = QPYLO(5,KP) - TEMPPOS(2)
             TEMPRETET(3)   = QPYLO(6,KP) - TEMPPOS(3)
          ENDIF
        ENDDO

        DO K=1,3
             RETET(K) = TEMPRETET(1)*TE(1,K) + TEMPRETET(2)*TE(2,K)
       &             + TEMPRETET(3)*TE(3,K)
        ENDDO

        DO N=1,2
          DO M =1,3
            DO L=1,3
              RBRI_R   (N,M,L) = TEMPRBRI_R   (N,M,L)
              RBRI_ANG(N,M,L) = TEMPRBRI_ANG(N,M,L)
            ENDDO
          ENDDO

        IF (N.EQ.1) THEN
           DO M=1,3
              DO L =1,3
                 RBRI_ANG(2,M,L) = 0
                 RBRI_R   (2,M,L) = 0
             ENDDO
           ENDDO
        ELSE
           DO M=1,3
              DO L =1,3
                 RBRI_ANG(1,M,L) = 0
                 RBRI_R   (1,M,L) = 0
              ENDDO
           ENDDO
        ENDIF


c---- distance between both bridle points and the derivatives
        DISTB1B2 = SQRT((RBRI(1,1) - RBRI(2,1))**2
       &               +(RBRI(1,2) - RBRI(2,2))**2
       &               +(RBRI(1,3) - RBRI(2,3))**2)

        DO K=1,3
        DISTB1B2_ANG(N,K) =
       & (((RBRI(1,1) - RBRI(2,1))*(RBRI_ANG(1,1,K) - RBRI_ANG(2,1,K))) +
       & ((RBRI(1,2) - RBRI(2,2))*(RBRI_ANG(1,2,K) - RBRI_ANG(2,2,K)))  +
       & ((RBRI(1,3) - RBRI(2,3))*(RBRI_ANG(1,3,K) - RBRI_ANG(2,3,K)))) /
       & DISTB1B2

        DISTB1B2_R(N,K) =
       & (((RBRI(1,1) - RBRI(2,1))*(RBRI_R(1,1,K) - RBRI_R(2,1,K))) +
       & ((RBRI(1,2) - RBRI(2,2))*(RBRI_R(1,2,K) - RBRI_R(2,2,K)))  +
       & ((RBRI(1,3) - RBRI(2,3))*(RBRI_R(1,3,K) - RBRI_R(2,3,K)))) /
       & DISTB1B2

        DISTB1B2_POS(N,K) =
       & (((RBRI(1,1) - RBRI(2,1))*(RBRI_POS(1,1,K) - RBRI_POS(2,1,K))) +
       & ((RBRI(1,2) - RBRI(2,2))*(RBRI_POS(1,2,K) - RBRI_POS(2,2,K)))  +
       & ((RBRI(1,3) - RBRI(2,3))*(RBRI_POS(1,3,K) - RBRI_POS(2,3,K)))) /
       & DISTB1B2

        DISTB1B2_ANGE(N,K) =
       & (((RBRI(1,1) - RBRI(2,1))*(RBRI_ANGE(1,1,K)-RBRI_ANGE(2,1,K))) +
       & ((RBRI(1,2) - RBRI(2,2))*(RBRI_ANGE(1,2,K)-RBRI_ANGE(2,2,K)))  +
       & ((RBRI(1,3) - RBRI(2,3))*(RBRI_ANGE(1,3,K)-RBRI_ANGE(2,3,K)))) /
       & DISTB1B2
```

```fortran
      ENDDO

c---- radius of the sphere-sphere intersection circle
       RSSIC =  SQRT((LBRID(1) + LBRID(2) + DISTB1B2) *
     &               (LBRID(1) + LBRID(2) - DISTB1B2) *
     &               (LBRID(1) - LBRID(2) + DISTB1B2) *
     &               (LBRID(2) - LBRID(1) + DISTB1B2)) / (2*DISTB1B2)

       RSSICTOP = SQRT((LBRID(1) + LBRID(2) + DISTB1B2) *
     &               (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &               (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &               (LBRID(2) - LBRID(1) + DISTB1B2))

       RSSICBOT = 2*DISTB1B2

       DO K=1,3
          RSSICTOP_ANG(N,K) = ((DISTB1B2_ANG(N,K)              *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))  -
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                     (DISTB1B2_ANG(N,K))                *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))  +
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                     (DISTB1B2_ANG(N,K))                *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))  +
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                     (DISTB1B2_ANG(N,K))))/(2*RSSICTOP)

          RSSICTOP_R(N,K) = ((DISTB1B2_R(N,K)                  *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))  -
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                     (DISTB1B2_R(N,K))                  *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))  +
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                     (DISTB1B2_R(N,K))                  *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))  +
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                     (DISTB1B2_R(N,K))))/(2*RSSICTOP)

          RSSICTOP_POS(N,K) = ((DISTB1B2_POS(N,K)                *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)     *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)     *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))    -
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)     *
     &                     (DISTB1B2_POS(N,K))                  *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)     *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))    +
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)     *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)     *
     &                     (DISTB1B2_POS(N,K))                  *
     &                     (LBRID(2) - LBRID(1) + DISTB1B2))    +
     &                    ((LBRID(1) + LBRID(2) + DISTB1B2)     *
     &                     (LBRID(1) + LBRID(2) - DISTB1B2)     *
     &                     (LBRID(1) - LBRID(2) + DISTB1B2)     *
     &                     (DISTB1B2_POS(N,K))))/(2*RSSICTOP)
```

```
         RSSICTOP_ANGE(N,K) = ((DISTB1B2_ANGE(N,K)            *
     &                        (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                        (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                        (LBRID(2) - LBRID(1) + DISTB1B2))  -
     &                       ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                        (DISTB1B2_ANGE(N,K))                *
     &                        (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                        (LBRID(2) - LBRID(1) + DISTB1B2))  +
     &                       ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                        (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                        (DISTB1B2_ANGE(N,K))                *
     &                        (LBRID(2) - LBRID(1) + DISTB1B2))  +
     &                       ((LBRID(1) + LBRID(2) + DISTB1B2)   *
     &                        (LBRID(1) + LBRID(2) - DISTB1B2)   *
     &                        (LBRID(1) - LBRID(2) + DISTB1B2)   *
     &                        (DISTB1B2_ANGE(N,K))))/(2*RSSICTOP)

         RSSICBOT_ANG  (N,K) = 2*DISTB1B2_ANG  (N,K)
         RSSICBOT_R    (N,K) = 2*DISTB1B2_R    (N,K)
         RSSICBOT_POS  (N,K) = 2*DISTB1B2_POS  (N,K)
         RSSICBOT_ANGE(N,K) = 2*DISTB1B2_ANGE(N,K)

         RSSIC_ANG(N,K) = ((RSSICTOP_ANG(N,K)*RSSICBOT)  -
     &                        (RSSICBOT_ANG(N,K)*RSSICTOP)) / RSSICBOT**2

         RSSIC_R   (N,K) = ((RSSICTOP_R   (N,K)*RSSICBOT)  -
     &                        (RSSICBOT_R   (N,K)*RSSICTOP)) / RSSICBOT**2

         RSSIC_POS(N,K) = ((RSSICTOP_POS(N,K)*RSSICBOT)  -
     &                        (RSSICBOT_POS(N,K)*RSSICTOP)) / RSSICBOT**2

         RSSIC_ANGE(N,K) = ((RSSICTOP_ANGE(N,K)*RSSICBOT)  -
     &                        (RSSICBOT_ANGE(N,K)*RSSICTOP)) / RSSICBOT**2

      ENDDO


c---- distance between the bridle points and the center of the sphere-
c     sphere interesectoin circle
      DISTB1CC = SQRT(LBRID(1)**2 - RSSIC**2)
      DISTB2CC = SQRT(LBRID(2)**2 - RSSIC**2)

      DO K=1,3
         DISTB1CC_ANG(N,K) = (-1)*RSSIC*RSSIC_ANG(N,K)/DISTB1CC
         DISTB2CC_ANG(N,K) = (-1)*RSSIC*RSSIC_ANG(N,K)/DISTB2CC

         DISTB1CC_POS(N,K) = (-1)*RSSIC*RSSIC_POS(N,K)/DISTB1CC
         DISTB2CC_POS(N,K) = (-1)*RSSIC*RSSIC_POS(N,K)/DISTB2CC

         DISTB1CC_ANGE(N,K) = (-1)*RSSIC*RSSIC_ANGE(N,K)/DISTB1CC
         DISTB2CC_ANGE(N,K) = (-1)*RSSIC*RSSIC_ANGE(N,K)/DISTB2CC

         DISTB1CC_R   (N,K) = (-1)*RSSIC*RSSIC_R   (N,K)/DISTB1CC
         DISTB2CC_R   (N,K) = (-1)*RSSIC*RSSIC_R   (N,K)/DISTB2CC

      ENDDO

c---- center of the intersection circle
      DO K =1,3
         CIRCCENT(K) = RBRI(1,K) + (((DISTB1CC/DISTB1B2)*
     &                  (RBRI(2,K)-RBRI(1,K))))

         CIRCCENTTOP(K) = ((DISTB1CC*(RBRI(2,K)-RBRI(1,K))))

         CIRCCENTBOT(K) = DISTB1B2

```

```
          DO L=1,3
            CIRCCENTTOP_ANG(N,K,L) =
     &          (DISTB1CC_ANG(N,L)*(RBRI     (2,K   )-RBRI     (1,K   ))) +
     &          (DISTB1CC          *(RBRI_ANG(2,K,L)-RBRI_ANG(1,K,L)))

            CIRCCENTTOP_POS(N,K,L) =
     &          (DISTB1CC_POS(N,L)*(RBRI     (2,K   )-RBRI     (1,K   ))) +
     &          (DISTB1CC          *(RBRI_POS(2,K,L)-RBRI_POS(1,K,L)))

            CIRCCENTTOP_ANGE(N,K,L) =
     &          (DISTB1CC_ANGE(N,L)*(RBRI     (2,K   )-RBRI     (1,K   ))) +
     &          (DISTB1CC          *(RBRI_ANGE(2,K,L)-RBRI_ANGE(1,K,L)))

            CIRCCENTBOT_ANG(N,L) = DISTB1B2_ANG(N,L)

            CIRCCENTBOT_POS(N,L) = DISTB1B2_POS(N,L)

            CIRCCENTBOT_ANGE(N,L) = DISTB1B2_ANGE(N,L)

            CIRCCENT_ANG(N,K,L)= RBRI_ANG(1,K,L) +
     &         ((CIRCCENTTOP_ANG(N,K,L) * CIRCCENTBOT(K))
     &      -   (CIRCCENTBOT_ANG(N,L    ) * CIRCCENTTOP(K)))
     &      / CIRCCENTBOT(K)**2

            CIRCCENT_POS(N,K,L)= RBRI_POS(1,K,L) +
     &         ((CIRCCENTTOP_POS(N,K,L) * CIRCCENTBOT(K))
     &      -   (CIRCCENTBOT_POS(N,L    ) * CIRCCENTTOP(K)))
     &      / CIRCCENTBOT(K)**2

            CIRCCENT_ANGE(N,K,L)= RBRI_ANGE(1,K,L) +
     &         ((CIRCCENTTOP_ANGE(N,K,L) * CIRCCENTBOT(K))
     &      -   (CIRCCENTBOT_ANGE(N,L    ) * CIRCCENTTOP(K)))
     &      / CIRCCENTBOT(K)**2

            CIRCCENTTOP_R(N,K,L) =
     &          (DISTB1CC_R(N,L)*(RBRI   (2,K   )-RBRI   (1,K   ))) +
     &          (DISTB1CC        *(RBRI_R(2,K,L)-RBRI_R(1,K,L)))

            CIRCCENTBOT_R(N,L) = DISTB1B2_R(N,L)

            CIRCCENT_R(N,K,L) = RBRI_R(1,K,L) +
     &            ((CIRCCENTTOP_R(N,K,L)*CIRCCENTBOT(K))
     &          - (CIRCCENTBOT_R(N,L    )*CIRCCENTTOP(K)))
     &          / CIRCCENTBOT(K)**2


          ENDDO
        ENDDO


c---- project the tether ground attachment point into the plane of
c     intersection circle
        COEF(1) = (-2)*(RBRI(1,1)-RBRI(2,1))
        COEF(2) = (-2)*(RBRI(1,2)-RBRI(2,2))
        COEF(3) = (-2)*(RBRI(1,3)-RBRI(2,3))
        COEF(4) = RBRI(1,1)**2 - RBRI(2,1)**2 + RBRI(1,2)**2
     &          - RBRI(2,2)**2 + RBRI(1,3)**2 - RBRI(2,3)**2
     &          - LBRID(1  )**2 + LBRID(2  )**2

        DO L =1,3
          RETETS(L) = RETET(L) - (COEF(L) *
     &     (COEF(1)*RETET(1)+COEF(2)*RETET(2)+COEF(3)*RETET(3)+COEF(4))
     &     /(COEF(1)**2+COEF(2)**2+COEF(3)**2))


          DO K=1,3
            COEF_ANG (N,L,K) = (-2)*(RBRI_ANG (1,L,K)-RBRI_ANG (2,L,K))
```

```
            COEF_POS  (N,L,K) = (-2)*(RBRI_POS  (1,L,K)-RBRI_POS  (2,L,K))
            COEF_ANGE(N,L,K) = (-2)*(RBRI_ANGE(1,L,K)-RBRI_ANGE(2,L,K))
            COEF_R    (N,L,K) = (-2)*(RBRI_R    (1,L,K)-RBRI_R    (2,L,K))

            COEF_ANG(N,4,K) =
     &        RBRI(1,1)*2*RBRI_ANG(1,1,K) - RBRI(2,1)*2*RBRI_ANG(2,1,K)
     &      + RBRI(1,2)*2*RBRI_ANG(1,2,K) - RBRI(2,2)*2*RBRI_ANG(2,2,K)
     &      + RBRI(1,3)*2*RBRI_ANG(1,3,K) - RBRI(2,3)*2*RBRI_ANG(2,3,K)

            COEF_POS(N,4,K) =
     &        RBRI(1,1)*2*RBRI_POS(1,1,K) - RBRI(2,1)*2*RBRI_POS(2,1,K)
     &      + RBRI(1,2)*2*RBRI_POS(1,2,K) - RBRI(2,2)*2*RBRI_POS(2,2,K)
     &      + RBRI(1,3)*2*RBRI_POS(1,3,K) - RBRI(2,3)*2*RBRI_POS(2,3,K)

            COEF_ANGE(N,4,K) =
     &        RBRI(1,1)*2*RBRI_ANGE(1,1,K)-RBRI(2,1)*2*RBRI_ANGE(2,1,K)
     &      + RBRI(1,2)*2*RBRI_ANGE(1,2,K)-RBRI(2,2)*2*RBRI_ANGE(2,2,K)
     &      + RBRI(1,3)*2*RBRI_ANGE(1,3,K)-RBRI(2,3)*2*RBRI_ANGE(2,3,K)

            COEF_R(N,4,K) =
     &        RBRI(1,1)*2*RBRI_R(1,1,K) - RBRI(2,1)*2*RBRI_R(2,1,K)
     &      + RBRI(1,2)*2*RBRI_R(1,2,K) - RBRI(2,2)*2*RBRI_R(2,2,K)
     &      + RBRI(1,3)*2*RBRI_R(1,3,K) - RBRI(2,3)*2*RBRI_R(2,3,K)


            RETETSTOP(L) = (-1)*COEF(L)*(COEF(1)*RETET(1)+COEF(2)
     &                *RETET(2) +COEF(3)*RETET(3)+COEF(4))

            RETETSBOT = COEF(1)**2+COEF(2)**2+COEF(3)**2
           ENDDO
         ENDDO

      DO L =1,3
        DO K=1,3

            RETETSTOP_ANG(N,L,K) =((-1)*COEF_ANG(N,L,K)*((COEF(1)
     &         * RETET(1))
     &         + (COEF(2)*RETET(2))+(COEF(3)*RETET(3))+COEF(4)))
     &         - (COEF(L)*((COEF_ANG(N,1,K)*RETET(1))
     &                   + (COEF_ANG(N,2,K)*RETET(2))
     &                   + (COEF_ANG(N,3,K)*RETET(3))+COEF_ANG(N,4,K)))


            RETETSBOT_ANG(N,K) = COEF(1)*2*COEF_ANG(N,1,K)
     &                         + COEF(2)*2*COEF_ANG(N,2,K)
     &                         + COEF(3)*2*COEF_ANG(N,3,K)

            RETETS_ANG(N,L,K) = ((RETETSTOP_ANG(N,L,K) * RETETSBOT   )
     &                         - (RETETSBOT_ANG(N,K    ) * RETETSTOP(L)))
     &                         / RETETSBOT**2

            RETETSTOP_POS(N,L,K) =((-1)*COEF_POS(N,L,K)*((COEF(1)
     &         * RETET(1))
     &         + (COEF(2)*RETET(2))+(COEF(3)*RETET(3))+COEF(4)))
     &         - (COEF(L)*((COEF_POS(N,1,K)*RETET(1))
     &                   + (COEF_POS(N,2,K)*RETET(2))
     &                   + (COEF_POS(N,3,K)*RETET(3))+COEF_POS(N,4,K)
     &                   + (COEF(1)*RETET_POS(N,1,K))
     &                   + (COEF(2)*RETET_POS(N,2,K))
     &                   + (COEF(3)*RETET_POS(N,3,K))))

            RETETSBOT_POS(N,K) = COEF(1)*2*COEF_POS(N,1,K)
     &                         + COEF(2)*2*COEF_POS(N,2,K)
     &                         + COEF(3)*2*COEF_POS(N,3,K)

            RETETS_POS(N,L,K) = RETET_POS(N,L,K)
     &                         + (((RETETSTOP_POS(N,L,K) * RETETSBOT   )
```

```
     &                               - (RETETSBOT_POS(N,K   ) * RETETSTOP(L)))
     &                               / RETETSBOT**2)


              RETETSTOP_ANGE(N,L,K)  =((-1)*COEF_ANGE(N,L,K)*((COEF(1)
     &            * RETET(1))
     &            + (COEF(2)*RETET(2))+(COEF(3)*RETET(3))+COEF(4)))
     &            - (COEF(L)*((COEF_ANGE(N,1,K)*RETET(1))
     &                        + (COEF_ANGE(N,2,K)*RETET(2))
     &                        + (COEF_ANGE(N,3,K)*RETET(3))+COEF_ANGE(N,4,K)
     &                        + (COEF(1)*RETET_ANGE(N,1,K))
     &                        + (COEF(2)*RETET_ANGE(N,2,K))
     &                        + (COEF(3)*RETET_ANGE(N,3,K))))

              RETETSBOT_ANGE(N,K)  = COEF(1)*2*COEF_ANGE(N,1,K)
     &                               +  COEF(2)*2*COEF_ANGE(N,2,K)
     &                               +  COEF(3)*2*COEF_ANGE(N,3,K)

              RETETS_ANGE(N,L,K)  = RETET_ANGE(N,L,K)
     &                             + (((RETETSTOP_ANGE(N,L,K) * RETETSBOT   )
     &                             - (RETETSBOT_ANGE(N,K   ) * RETETSTOP(L)))
     &                             / RETETSBOT**2)


              RETETSTOP_R(N,L,K)  =((-1)*COEF_R(N,L,K)*((COEF(1)*RETET(1))
     &            + (COEF(2)*RETET(2))+(COEF(3)*RETET(3))+COEF(4)))
     &            - (COEF(L)*((COEF_R(N,1,K)*RETET(1))
     &                        + (COEF_R(N,2,K)*RETET(2))
     &                        + (COEF_R(N,3,K)*RETET(3))+COEF_R(N,4,K)))

              RETETSBOT_R(N,K)= COEF(1)*2*COEF_R(N,1,K)
     &                        + COEF(2)*2*COEF_R(N,2,K)
     &                        + COEF(3)*2*COEF_R(N,3,K)

              RETETS_R(N,L,K)  = ((RETETSTOP_R(N,L,K) * RETETSBOT   )
     &                         - (RETETSBOT_R(N,K   ) * RETETSTOP(L)))
     &                         / RETETSBOT**2

          ENDDO !K
       ENDDO !L


c---- vector from the projected tether attachment point location to the
c     center of the sphere-sphere intersection circle
       DO K =1,3
         DISTRESCC(K)  = RETETS(K)  - CIRCCENT(K)

         DO L  =1,3

         DISTRESCC_ANG  (N,K,L)  = RETETS_ANG  (N,K,L)-CIRCCENT_ANG  (N,K,L)
         DISTRESCC_POS  (N,K,L)  = RETETS_POS  (N,K,L)-CIRCCENT_POS  (N,K,L)
         DISTRESCC_ANGE(N,K,L)  = RETETS_ANGE(N,K,L)-CIRCCENT_ANGE(N,K,L)
         DISTRESCC_R     (N,K,L)  = RETETS_R     (N,K,L)-CIRCCENT_R     (N,K,L)

         ENDDO

       ENDDO

       LRESCC  = SQRT(DISTRESCC(1)**2+DISTRESCC(2)**2+ DISTRESCC(3)**2)

       DO K=1,3
          LRESCC_ANG(N,K)  = (DISTRESCC(1)*DISTRESCC_ANG(N,1,K)
     &                       +  DISTRESCC(2)*DISTRESCC_ANG(N,2,K)
     &                       +  DISTRESCC(3)*DISTRESCC_ANG(N,3,K)) / LRESCC
```

```fortran
            LRESCC_POS(N,K)  = (DISTRESCC(1)*DISTRESCC_POS(N,1,K)
     &                        + DISTRESCC(2)*DISTRESCC_POS(N,2,K)
     &                        + DISTRESCC(3)*DISTRESCC_POS(N,3,K)) / LRESCC

            LRESCC_ANGE(N,K)  = (DISTRESCC(1)*DISTRESCC_ANGE(N,1,K)
     &                         + DISTRESCC(2)*DISTRESCC_ANGE(N,2,K)
     &                         + DISTRESCC(3)*DISTRESCC_ANGE(N,3,K))/LRESCC

            LRESCC_R(N,K)    = (DISTRESCC(1)*DISTRESCC_R    (N,1,K)
     &                        + DISTRESCC(2)*DISTRESCC_R    (N,2,K)
     &                        + DISTRESCC(3)*DISTRESCC_R    (N,3,K)) /LRESCC
         ENDDO

c--- determine the location of the tether bridle attachment point
         DO K=1,3
           RTBA(K) = CIRCCENT(K) + ((RSSIC/LRESCC)*DISTRESCC(K))


           DO L =1,3
              RTBA_ANG(N,K,L) = CIRCCENT_ANG(N,K,L)
     &        + (((((RSSIC_ANG(N,L)*DISTRESCC(K))
     &        +  (RSSIC* DISTRESCC_ANG(N,K,L)))*LRESCC)
     &        - (LRESCC_ANG(N,L) * RSSIC*DISTRESCC(K))) / LRESCC**2)

              RTBA_POS(N,K,L) = CIRCCENT_POS(N,K,L)
     &        + (((((RSSIC_POS(N,L)*DISTRESCC(K))
     &        +  (RSSIC* DISTRESCC_POS(N,K,L)))*LRESCC)
     &        - (LRESCC_POS(N,L) * RSSIC*DISTRESCC(K))) / LRESCC**2)

              RTBA_ANGE(N,K,L) = CIRCCENT_ANGE(N,K,L)
     &        + (((((RSSIC_ANGE(N,L)*DISTRESCC(K))
     &        +  (RSSIC* DISTRESCC_ANGE(N,K,L)))*LRESCC)
     &        - (LRESCC_ANGE(N,L) * RSSIC*DISTRESCC(K))) / LRESCC**2)

              RTBA_R(N,K,L) = CIRCCENT_R(N,K,L)
     &        + (((((RSSIC_R(N,L)*DISTRESCC(K))
     &        +  (RSSIC* DISTRESCC_R(N,K,L)))*LRESCC)
     &        - (LRESCC_R(N,L) * RSSIC*DISTRESCC(K))) / LRESCC**2)

           ENDDO
         ENDDO

         ENDDO

         RETURN
         END
```

/Aswing/src/sloads.f

C.3 Aerodynamic and gravitational tether force subroutine

```
      SUBROUTINE SETWGTTET(IS,IPNT,TETMASS, CDA, TE,TEMPRE0TET, TEMPPOS,
     & RTBA_ANG, RTBA, TETTFOR)
      INCLUDE 'ASWING.INC'
C------------------------------------------------------------------
C     Imposes applied loads and moments due to tether mass.
C     The loads are applied only to beam IS at operating point IPNT.
C
C     This version applies inertial-reaction, gravity, aero drag loads:
C      -         _  _               -
C      F  =  m (g - a) +  0.5 rho V V CDA
C      -        _   -
C      M  =  dr x F
C------------------------------------------------------------------
      DIMENSION XYZ(3), ANG(3), UVW(3), OMG(3), UVWT(3), OMGT(3),
     &          T0(3,3), T(3,3), T_ANG(3,3,3),
     &          TNET(3,3), TNET_ANG(3,3,3)
      DIMENSION VEL(3), VAC(3), ROT(3), RAC(3)
      DIMENSION DRP(3), DRP_ANG(3,3), XYZP(3),
     &          HAP(3), HAP_ANG(3,3)
C
      DIMENSION VG(3), VG_POS(3,3), POS(3),
     &          VG_HEAD(3),VG_ELEV(3),VG_BANK(3),
     &          VG_XYZP(3,3)
C
      DIMENSION DV(3), DV_DRP(3,3), DV_OMG(3,3), DV_ROT(3,3),
     &          DA(3), DA_DRP(3,3), DA_OMG(3,3), DA_ROT(3,3),
     &                                DA_OMGT(3,3),DA_RAC(3,3)
C
      DIMENSION VR(3), VR_XYZ(3,3), VR_UVW(3,3), VR_ROT(3,3),
     &          AR(3), AR_XYZ(3,3), AR_UVW(3,3), AR_ROT(3,3),
     &                                AR_UVWT(3,3),AR_RAC(3,3),
     &          VE(3),
     &          VR_Q(3,18),VR_GL(3,0:NGLX),
     &          AR_Q(3,18),AR_GL(3,0:NGLX),AR_QT(3,6),AR_GLT(3,0:NGLX),
     &          VG_Q(3,18),VG_GL(3,0:NGLX),VG_FRP(3,0:NFRLX),
     &          VE_Q(3,18),VE_GL(3,0:NGLX),VE_FRP(3,0:NFRLX)
C
      DIMENSION GV(3), GV_ELEV(3), GV_BANK(3)
C
      DIMENSION VF(3,4),
     &          VF_XYZ(3,3,4),
     &          VF_POS(3,3,4),
     &          VF_HEAD(3,4),
     &          VF_ELEV(3,4),
     &          VF_BANK(3,4),
     &          VF_AK(3,3,4),
     &          VF_AL(3,3,4)
C
      DIMENSION F_VE(3)
      DIMENSION F(3), F_Q(3,18), F_GL(3,0:NGLX), F_FRP(3,0:NFRLX),
     &          M(3), M_Q(3,18), M_GL(3,0:NGLX), M_FRP(3,0:NFRLX),
     &                F_QT(3,6), F_GLT(3,0:NGLX),
     &                M_QT(3,6), M_GLT(3,0:NGLX),
     &       FK_QO(18)  ,FK_QP(18)   ,FK_GL(NGLX), FK_GLT(NGLX),
     &       MK_QO(18)  ,MK_QP(18)   ,MK_GL(NGLX), MK_GLT(NGLX),
     &       FK_QTO(6)  ,FK_QTP(6)   ,FK_FRP(NFRLX),
     &       MK_QTO(6)  ,MK_QTP(6)   ,MK_FRP(NFRLX)
C
      DIMENSION ICRS(3), JCRS(3), TETTFOR(3)

      DATA ICRS / 2, 3, 1 / , JCRS / 3, 1, 2 /

      REAL TETMASS, TE(3,3), TEMPRE0TET(3), TEMPPOS(3), RTBA_ANG(3,3)
C
```

```
      RHO  = PARAM(KPDENS,IPNT)
      VSO  = PARAM(KPVSOU,IPNT)
C
      HEAD  = PARAM(KPHEAD,IPNT)
      ELEV  = PARAM(KPELEV,IPNT)
      BANK  = PARAM(KPBANK,IPNT)

      DO K=1, 3
        VAC(K) = PARAM(KPVAC(K),IPNT)
        RAC(K) = PARAM(KPRAC(K),IPNT)
        VEL(K) = PARAM(KPVEL(K),IPNT)
        ROT(K) = PARAM(KPROT(K),IPNT)
        POS(K) = PARAM(KPPOS(K),IPNT)
      ENDDO
C

      GEE  = PARAM(KPGRAV,IPNT)
C
C---- set gravity vector x,y,z components GV(.)
      CALL GVCALC(GEE, ELEV,BANK, HEAD, GV,GV_ELEV,GV_BANK, GV_HEAD,
     & GRAVDIR)
C
C
C==== add loads due to point weights

      DO 200 KP=1, NPYLO
       IF(KPTYPE(KP).NE.3   .OR.
     &    ISPYLO(KP).NE.IS       ) GO TO 200
C
C---- interval where pylon is attached
      IEQ = IPYLO(KP)

C---- node to which pylon is effectively attached
      I = MIN( MAX(IEQ,IFRST(IS)) , ILAST(IS) )
C

C---- set local Euler angles and transformation tensor for undeformed state
      ANG(1) = Q0(4,I)
      ANG(2) = Q0(5,I)
      ANG(3) = Q0(6,I)
      CALL ROTENS(ANG, T0,T_ANG, KBTYPE(IS))
C
C---- set local Euler angles and transformation tensor
      ANG(1) = Q(4,I,IPNT)
      ANG(2) = Q(5,I,IPNT)
      ANG(3) = Q(6,I,IPNT)
      CALL ROTENS(ANG, T  ,T_ANG, KBTYPE(IS))
C
C          t
C---- set T To matrix
      DO K = 1, 3
        DO L = 1, 3
          TNET(K,L) = T(1,K)*T0(1,L)
     &              + T(2,K)*T0(2,L)
     &              + T(3,K)*T0(3,L)
          DO J = 1, 3
            TNET_ANG(K,L,J) = T_ANG(1,K,J)*T0(1,L)
     &                      + T_ANG(2,K,J)*T0(2,L)
     &                      + T_ANG(3,K,J)*T0(3,L)
          ENDDO
        ENDDO
      ENDDO
C
C---- set pylon offset vector dr_p
      DO K = 1, 3

        DRP(K) = 0
```

```
          DO L = 1, 3
            DRP_ANG(K,L) = RTBA_ANG(K,L)/2
          ENDDO
        ENDDO

C
C---- set beam position r
        XYZ(1) = Q(1,I,IPNT)
        XYZ(2) = Q(2,I,IPNT)
        XYZ(3) = Q(3,I,IPNT)

C---- set beam velocity u
        UVW(1) = Q(13,I,IPNT)
        UVW(2) = Q(14,I,IPNT)
        UVW(3) = Q(15,I,IPNT)
C
C---- set beam rotational rate w
        OMG(1) = Q(16,I,IPNT)
        OMG(2) = Q(17,I,IPNT)
        OMG(3) = Q(18,I,IPNT)
C                            .
C---- set beam acceleration u
        UVWT(1) = UDOT(1,I)
        UVWT(2) = UDOT(2,I)
        UVWT(3) = UDOT(3,I)
C                                    .
C---- set beam angular acceleration w
        OMGT(1) = UDOT(4,I)
        OMGT(2) = UDOT(5,I)
        OMGT(3) = UDOT(6,I)
C
C---- set beam's velocities VR and accelerations AR
        CALL VACALC(XYZ,     UVW,     ROT,     UVWT,     RAC,
     &      VR, VR_XYZ, VR_UVW, VR_ROT,
     &      AR, AR_XYZ, AR_UVW, AR_ROT, AR_UVWT,  AR_RAC )
C
C---- set additional velocity and acceleration due to DRP offset
        CALL VDCALC( DRP,     OMG,     ROT,     OMGT,     RAC,
     &       DV, DV_DRP, DV_OMG, DV_ROT,
     &       DA, DA_DRP, DA_OMG, DA_ROT, DA_OMGT, DA_RAC )
C
C---- set local gust velocity
        XYZP(1) = XYZ(1) + DRP(1)
        XYZP(2) = XYZ(2) + DRP(2)
        XYZP(3) = XYZ(3) + DRP(3)
        CALL VGUST(XYZP,    POS,    HEAD,    ELEV,    BANK,
     &      VG,VG_XYZP,VG_POS,VG_HEAD,VG_ELEV,VG_BANK )
C
C---- set sensitivities wrt local and global variables
        DO K=1, 3
          DO L = 1, 18
            VR_Q(K,L) = 0.
            AR_Q(K,L) = 0.
            VG_Q(K,L) = 0.
          ENDDO
          DO L = 1, 6
            AR_QT(K,L) = 0.
          ENDDO
          DO L=0, NRHS
            VR_GL(K,L) = 0.
            AR_GL(K,L) = 0.
            VG_GL(K,L) = 0.
            AR_GLT(K,L) = 0.
          ENDDO
          DO L = 0, NFRP
            VG_FRP(K,L) = 0.
```

```
          ENDDO
C
          VR(K)          = VR(K)        + DV(K)
          AR(K)          = AR(K)        + DA(K)
          DO L=1, 3
            VR_Q(K,L)      = VR_XYZ(K,L)
            VR_Q(K,L+12)   = VR_UVW(K,L)
            VR_Q(K,L+15)   =                DV_OMG(K,L)
            VR_Q(K,L+3)    =                DV_DRP(K,1)*DRP_ANG(1,L)
     &                                    + DV_DRP(K,2)*DRP_ANG(2,L)
     &                                    + DV_DRP(K,3)*DRP_ANG(3,L)

            AR_Q(K,L)      = AR_XYZ(K,L)
            AR_Q(K,L+12)   = AR_UVW(K,L)
            AR_Q(K,L+15)   =                DA_OMG(K,L)
            AR_Q(K,L+3)    =                DA_DRP(K,1)*DRP_ANG(1,L)
     &                                    + DA_DRP(K,2)*DRP_ANG(2,L)
     &                                    + DA_DRP(K,3)*DRP_ANG(3,L)
C
            VG_Q(K,L)      = VG_XYZP(K,L)
            VG_Q(K,L+3)    = VG_XYZP(K,1)*DRP_ANG(1,L)
     &                     + VG_XYZP(K,2)*DRP_ANG(2,L)
     &                     + VG_XYZP(K,3)*DRP_ANG(3,L)
C
            AR_QT(K,L)     = AR_UVWT(K,L)
            AR_QT(K,L+3)   =                DA_OMGT(K,L)
          ENDDO
          DO L=1, 3
            VR_GL(K,LROT(L))  = VR_ROT(K,L)  + DV_ROT(K,L)
C
            AR_GL(K,LROT(L))  = AR_ROT(K,L)  + DA_ROT(K,L)
            AR_GL(K,LRAC(L))  = AR_RAC(K,L)  + DA_RAC(K,L)
C
            VG_GL(K,LPOS(L))  = VG_POS(K,L)
C
cc            AR_GLT(K,LRAC(L))  = AR_RAC(K,L)  + DA_RAC(K,L)
          ENDDO
          VG_GL(K,LHEAD)  = VG_HEAD(K)
          VG_GL(K,LELEV)  = VG_ELEV(K)
          VG_GL(K,LBANK)  = VG_BANK(K)
        ENDDO
C
C---- set local frequency-gust unit velocity
        IF(LFGUST) THEN
          DO KG = 1, NFGUST
            CALL VFREQ(XYZ,    POS,    HEAD,    ELEV,    BANK,
     &                  AKGUST(1,KG),
     &                  ALGUST(1,KG),
     &                  VFGUST(1,1,KG),
     &        VF,
     &        VF_XYZ,
     &        VF_POS,
     &        VF_HEAD,
     &        VF_ELEV,
     &        VF_BANK,
     &        VF_AK,
     &        VF_AL  )
            DO K = 1, 3
              VG_FRP(K,LGUS1F(KG))  = VG_FRP(K,LGUS1F(KG))  + VF(K,1)
              VG_FRP(K,LGUS2F(KG))  = VG_FRP(K,LGUS2F(KG))  + VF(K,2)
              VG_FRP(K,LGUS3F(KG))  = VG_FRP(K,LGUS3F(KG))  + VF(K,3)
              VG_FRP(K,LGUS4F(KG))  = VG_FRP(K,LGUS4F(KG))  + VF(K,4)
            ENDDO
          ENDDO
        ENDIF
C

```

```
      PMASS = TETMASS/GEEW
C                                     _        _ _
C---- set weight force vector   F = m(g-a)  +  applied-force
      DO K=1, 3
        F(K)          = PMASS*(GV(K)       -AR(K)       - VAC(K))
        DO L=1, NRHS
          F_GL(K,L)  = PMASS*(             -AR_GL(K,L))
          F_GLT(K,L)= PMASS*(              -AR_GLT(K,L))
        ENDDO
        DO L=1, 18
          F_Q(K,L)   = PMASS*(             -AR_Q(K,L)  )
        ENDDO
        DO L=1, 6
          F_QT(K,L)  = PMASS*(             -AR_QT(K,L))
        ENDDO
C
        F_GL(K,LELEV)    = F_GL(K,LELEV)    + PMASS*GV_ELEV(K)
        F_GL(K,LBANK)    = F_GL(K,LBANK)    + PMASS*GV_BANK(K)
        F_GL(K,LVAC(K))  = F_GL(K,LVAC(K))  - PMASS
C
        DO L=1, NFRP
          F_FRP(K,L) = 0.
        ENDDO
      ENDDO

       DO K=1,3
          TETTFOR(K) = TETTFOR(K) - F(K)
       ENDDO


C                                 _
C---- set weight moment vector  DW x F  and precession moment
      DO K=1, 3
        IC = ICRS(K)
        JC = JCRS(K)
        M(K) = 0
        DO L=1, NRHS
          M_GL(K,L) = 0
          M_GLT(K,L)= 0
        ENDDO
        DO L=1, 18
          M_Q(K,L) = 0
        ENDDO
        DO L=1, 6
          M_QT(K,L)= 0
        ENDDO
C
        M_GL(K,LROT(JC)) = 0
        M_GL(K,LROT(IC)) = 0
        M_Q  (K,JC+15)   = 0
        M_Q  (K,IC+15)   = 0
        DO L=1, 3
          M_Q(K,L+3) = 0
        ENDDO
C
        DO L=1, NFRP
          M_FRP(K,L) = 0.
        ENDDO
      ENDDO
C
C---- add residual and Jacobian changes to appropriate slots,
      DO K = 1, 3
C------ set row-major indexed arrays for calling EQNADD
        IF(IEQ.EQ.IFRST(IS)-1) THEN
         KEQF = KEQ0(K+9,IS)
         KEQM = KEQ0(K+6,IS)
         DO L = 1, 18
```

```
            FK_QO(L) = 0.
            FK_QP(L) = F_Q(K,L)
            MK_QO(L) = 0.
            MK_QP(L) = M_Q(K,L)
          ENDDO
          DO L = 1, 6
            FK_QTO(L) = 0.
            FK_QTP(L) = F_QT(K,L)
            MK_QTO(L) = 0.
            MK_QTP(L) = M_QT(K,L)
          ENDDO
         ELSE
          KEQF = KEQ(K+9,IEQ)
          KEQM = KEQ(K+6,IEQ)
          DO L = 1, 18
            FK_QO(L) = F_Q(K,L)
            FK_QP(L) = 0.
            MK_QO(L) = M_Q(K,L)
            MK_QP(L) = 0.
          ENDDO
          DO L = 1, 6
            FK_QTO(L) = F_QT(K,L)
            FK_QTP(L) = 0.
            MK_QTO(L) = M_QT(K,L)
            MK_QTP(L) = 0.
          ENDDO
         ENDIF
C
         DO L = 1, NRHS
           FK_GL(L) = F_GL(K,L)
           MK_GL(L) = M_GL(K,L)
C
           FK_GLT(L) = F_GLT(K,L)
           MK_GLT(L) = M_GLT(K,L)
         ENDDO
         DO L = 1, NFRP
           FK_FRP(L) = F_FRP(K,L)
           MK_FRP(L) = M_FRP(K,L)
         ENDDO

       ENDDO

C=================================================================
C==== add on aero drag on point mass
C=================================================================

C---- skip aero force calculations if point mass has zero drag area
      IF(CDA.EQ.0.0) GO TO 200
C
C---- set local effective freestream velocity vector VE in body axes
      DO K=1, 3
        VE(K)         = VG(K)      - VR(K)    - VEL(K) + VIP(K,KP)
     &                                                 + WIP(K,KP)
     &                                                 + VEP(K,KP)

        DO L=1, 18
          VE_Q(K,L)   = VG_Q(K,L)   - VR_Q(K,L)
        ENDDO
        DO L=1, NRHS
          VE_GL(K,L)  = VG_GL(K,L)  - VR_GL(K,L)       + VEP_GL(K,L,KP)
        ENDDO
        DO L=1, NFRP
          VE_FRP(K,L)= VG_FRP(K,L)
        ENDDO
C
        VE_GL(K,LVEL(K)) = VE_GL(K,LVEL(K)) - 1.0
C
```

```
          DO N = 1, NNTOT
            VE_GL(K,LAN(N)) = VE_GL(K,LAN(N))   + VIP_AN(K,N,KP)
          ENDDO
C
          DO L = 1, 3
            VE_GL(K,LVEL(L)) = VE_GL(K,LVEL(L))  + VIP_AL(K,KP)*ALW_VEL(L)
     &                                          + VIP_BE(K,KP)*BEW_VEL(L)
     &                                          + WIP_VI(K,KP)*VIW_VEL(L)
     &                                          + WIP_AL(K,KP)*ALW_VEL(L)
     &                                          + WIP_BE(K,KP)*BEW_VEL(L)
     &                                          + VEP_AL(K,KP)*ALW_VEL(L)
     &                                          + VEP_BE(K,KP)*BEW_VEL(L)
          ENDDO
        ENDDO
C
        VSQ = VE(1)**2 + VE(2)**2 + VE(3)**2
C
C---- avoid numerical 0/0 error below for stationary cases (for V = 0)
        IF(VSQ .EQ. 0.0) VSQ = 0.000001
C
        DO K=1, 3
          F(K)       = 0.5*RHO*SQRT(VSQ)  *VE(K)      *CDA
          DO L=1, 3
            F_VE(L)  = 0.5*RHO*VE(L)/SQRT(VSQ)*VE(K)*CDA
          ENDDO
          F_VE(K)    = 0.5*RHO*SQRT(VSQ)               *CDA  +  F_VE(K)
C
C
          DO L=1, 18
            F_Q(K,L)  = F_VE(1)*VE_Q(1,L)
     &                + F_VE(2)*VE_Q(2,L)
     &                + F_VE(3)*VE_Q(3,L)
          ENDDO
          DO L=1, 6
            F_QT(K,L)  = 0.
          ENDDO
          DO L=1, NRHS
            F_GL(K,L)  = F_VE(1)*VE_GL(1,L)
     &                 + F_VE(2)*VE_GL(2,L)
     &                 + F_VE(3)*VE_GL(3,L)
            F_GLT(K,L)  = 0.
          ENDDO
          DO L=1, NFRP
            F_FRP(K,L)= F_VE(1)*VE_FRP(1,L)
     &                + F_VE(2)*VE_FRP(2,L)
     &                + F_VE(3)*VE_FRP(3,L)
          ENDDO
        ENDDO

C
        DO K=1, 3
          IC = ICRS(K)
          JC = JCRS(K)
          M(K) =          0
C
          DO L=1, 18
            M_Q(K,L)  = 0
          ENDDO
          DO L=1, 6
            M_QT(K,L)= 0
          ENDDO
          DO L=1, 3
            M_Q(K,L+3)  = 0
          ENDDO
          DO L=1, NRHS
            M_GL(K,L)   = 0
C
```

```
            M_GLT(K,L) = 0
          ENDDO
          DO L=1, NFRP
            M_FRP(K,L) = 0
          ENDDO
C
        ENDDO
C
C---- add residual and Jacobian changes to appropriate slots
        DO K = 1, 3
C------ set row-major indexed arrays for calling EQNADD
          IF(IEQ.EQ.IFRST(IS)-1) THEN
           KEQF = KEQ0(K+9,IS)
           KEQM = KEQ0(K+6,IS)
           DO L = 1, 18
             FK_QO(L) = 0.
             FK_QP(L) = F_Q(K,L)
             MK_QO(L) = 0.
             MK_QP(L) = M_Q(K,L)
           ENDDO
           DO L = 1, 6
             FK_QTO(L) = 0.
             FK_QTP(L) = F_QT(K,L)
             MK_QTO(L) = 0.
             MK_QTP(L) = M_QT(K,L)
           ENDDO
          ELSE
           KEQF = KEQ(K+9,IEQ)
           KEQM = KEQ(K+6,IEQ)
           DO L = 1, 18
             FK_QO(L) = F_Q(K,L)
             FK_QP(L) = 0.
             MK_QO(L) = M_Q(K,L)
             MK_QP(L) = 0.
           ENDDO
           DO L = 1, 6
             FK_QTO(L) = F_QT(K,L)
             FK_QTP(L) = 0.
             MK_QTO(L) = M_QT(K,L)
             MK_QTP(L) = 0.
           ENDDO
          ENDIF
C
          DO L = 1, NRHS
            FK_GL(L) = F_GL(K,L)
            MK_GL(L) = M_GL(K,L)
C
            FK_GLT(L) = F_GLT(K,L)
            MK_GLT(L) = M_GLT(K,L)
          ENDDO
          DO L = 1, NFRP
            FK_FRP(L) = F_FRP(K,L)
            MK_FRP(L) = M_FRP(K,L)
          ENDDO


          CALL EQNADD(K+9,IEQ, KEQF, F(K), FK_QO, FK_QP,
     &                                     FK_QTO,FK_QTP,
     &                                     FK_GL  ,FK_GLT,FK_FRP)
          CALL EQNADD(K+6,IEQ, KEQM, M(K), MK_QO, MK_QP,
     &                                     MK_QTO,MK_QTP,
     &                                     MK_GL  ,MK_GLT,MK_FRP)
        ENDDO



        CALL FMDEL(NRHS,NFRP, 1.0, Q(1,I,IPNT),
```

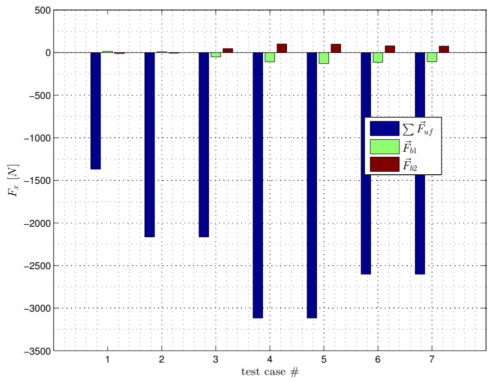
```
     &            F,  F_Q,  F_QT,  F_GL(1,1),  F_GLT(1,1),  F_FRP(1,1),
     &            M,  M_Q,  M_QT,  M_GL(1,1),  M_GLT(1,1),  M_FRP(1,1),
     & AFORCE,AFOR_Q(1,1,I),AFOR_UT(1,1,I),AFOR_GL(1,1),AFOR_GLT(1,1),
     &                                                  AFOR_FRP(1,1),
     & AMOMNT,AMOM_Q(1,1,I),AMOM_UT(1,1,I),AMOM_GL(1,1),AMOM_GLT(1,1),
     &                                                  AMOM_FRP(1,1))
C
C---- if this is also a ground point, also add to ground-force accumulators
      DO KG=1, NGROU
         IF(IS.EQ.ISGROU(KG)  .AND.  IEQ.EQ.IGROU(KG)) THEN
          CALL FMDEL(NRHS,NFRP,  1.0,  Q(1,I,IPNT),
     &            F,  F_Q,  F_QT,  F_GL(1,1),  F_GLT(1,1),  F_FRP(1,1),
     &            M,  M_Q,  M_QT,  M_GL(1,1),  M_GLT(1,1),  M_FRP(1,1),
     & GFORCE,GFOR_Q(1,1,I),GFOR_UT(1,1,I),GFOR_GL(1,1),GFOR_GLT(1,1),
     &                                                  GFOR_FRP(1,1),
     & GMOMNT,GMOM_Q(1,1,I),GMOM_UT(1,1,I),GMOM_GL(1,1),GMOM_GLT(1,1),
     &                                                  GMOM_FRP(1,1))
          ENDIF
      ENDDO
C
 200  CONTINUE
C
      RETURN
      END ! SETWGTTET
```

/Aswing/src/sloads.f

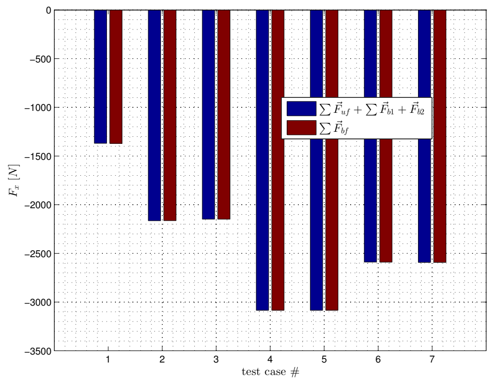
Appendix D

Verification plots

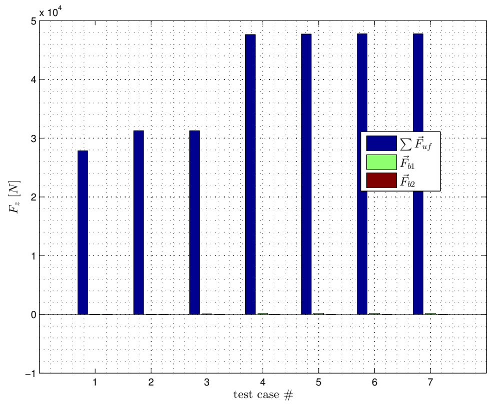
D.1 Force and moment equilibrium



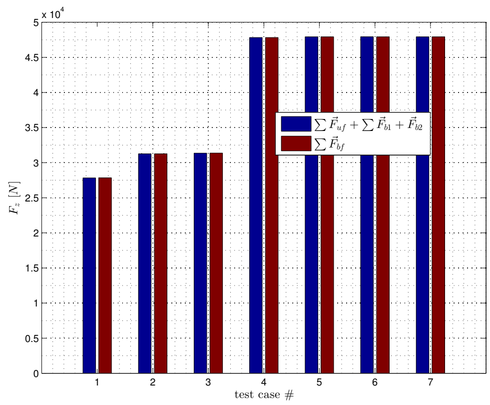
(a) Force equilibrium x-direction



(b) ASWING vs analytical, forces in x-direction



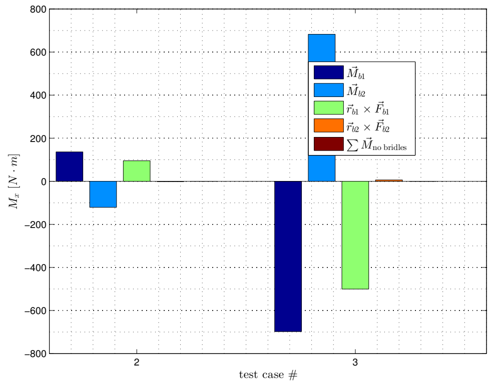
(c) Force equilibrium z-direction



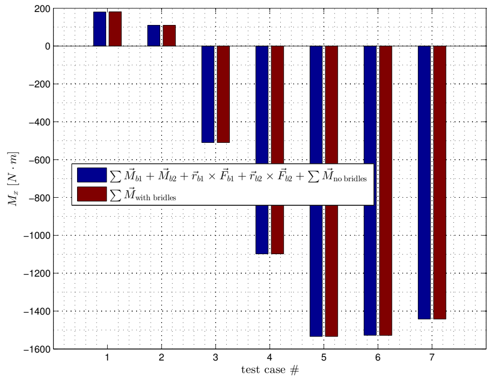
(d) ASWING vs analytical, forces in z-direction

Figure D.1: Comparison sum of forces in x and z-direction for the unbridled and bridled case

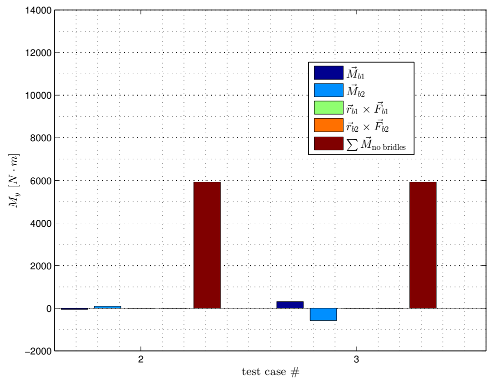
D.1.1 Moment additions to the Newton system



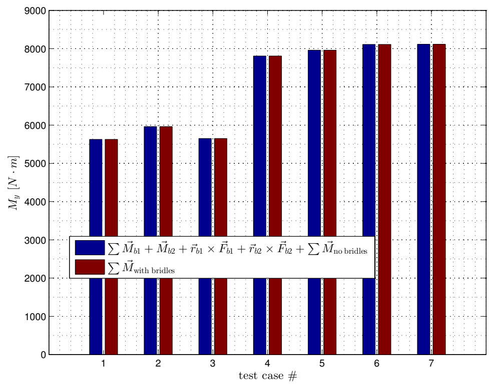
(a) Moment equilibrium x-axis



(b) ASWING vs analytical, moments about x-axis



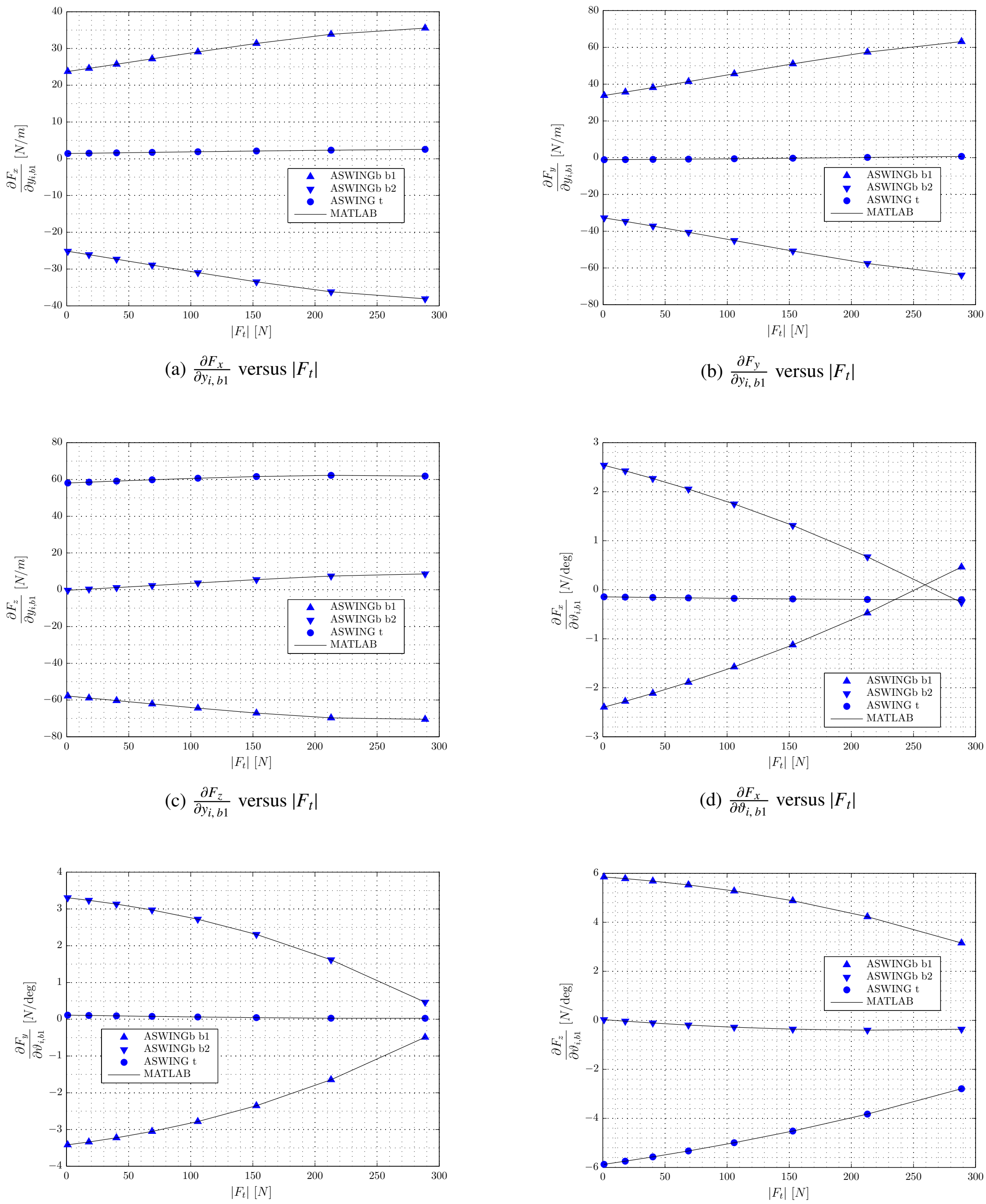
(c) Moment equilibrium y-axis



(d) ASWING vs analytical, moments about y-axis

Figure D.2: Comparison sum of moments about x and y-axis for the unbridled and bridled case

D.2 Tether and bridle force Jacobians



(a) $\frac{\partial F_x}{\partial y_{i,\,b1}}$ versus $|F_t|$

(b) $\frac{\partial F_y}{\partial y_{i,\,b1}}$ versus $|F_t|$

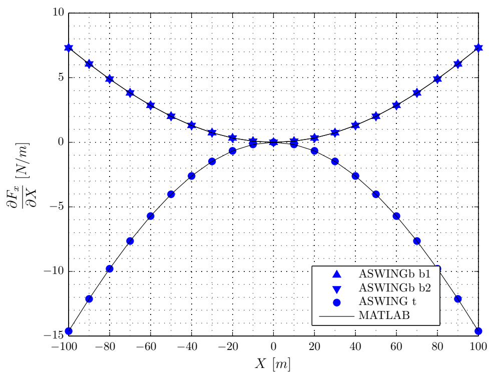
(c) $\frac{\partial F_z}{\partial y_{i,\,b1}}$ versus $|F_t|$

(d) $\frac{\partial F_x}{\partial \vartheta_{i,\,b1}}$ versus $|F_t|$

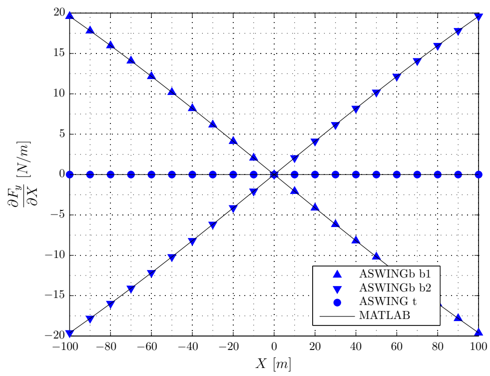
(e) $\frac{\partial F_y}{\partial \vartheta_{i,\,b1}}$ versus $|F_t|$

(f) $\frac{\partial F_z}{\partial \vartheta_{i,\,b1}}$ versus $|F_t|$

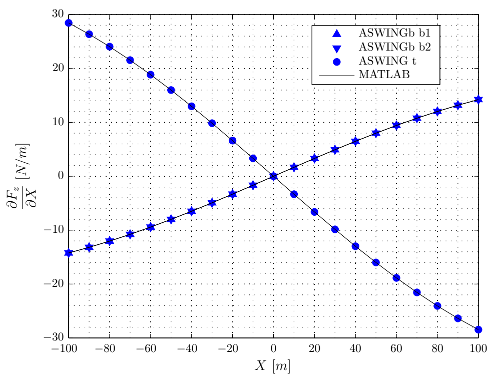
Figure D.3: Verification bridle force Jacobian entries for local beam coordinates



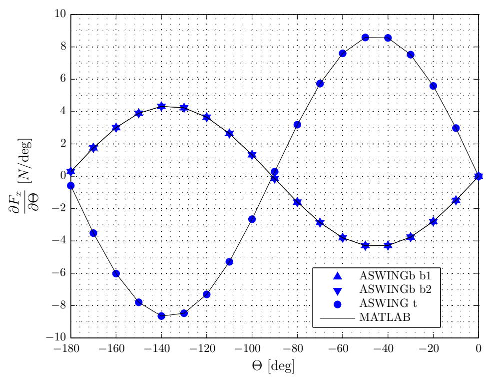
(a) $\frac{\partial F_x}{\partial X}$ versus $|F_t|$



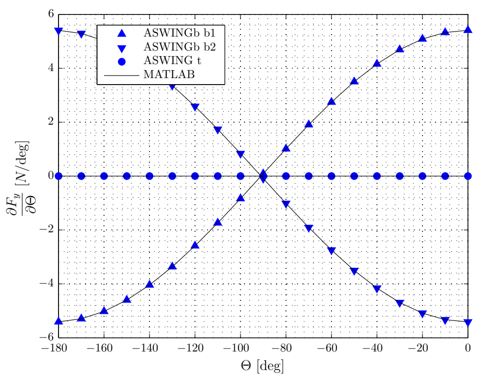
(b) $\frac{\partial F_y}{\partial X}$ versus $|F_t|$



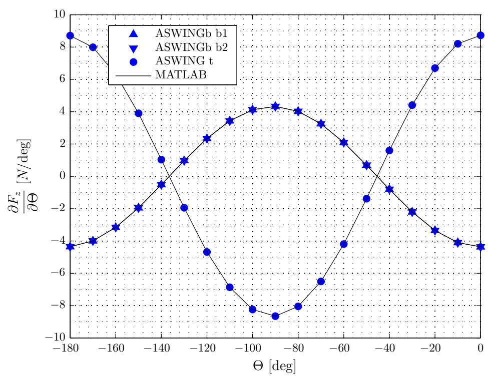
(c) $\frac{\partial F_z}{\partial X}$ versus $|F_t|$



(d) $\frac{\partial F_x}{\partial \Theta}$ versus $|F_t|$



(e) $\frac{\partial F_y}{\partial \Theta}$ versus $|F_t|$



(f) $\frac{\partial F_z}{\partial \Theta}$ versus $|F_t|$

Figure D.4: Verification bridle force Jacobian entries for Earth coordinates

Appendix E

Wind tunnel test schedule

Table E.1: Test schedule

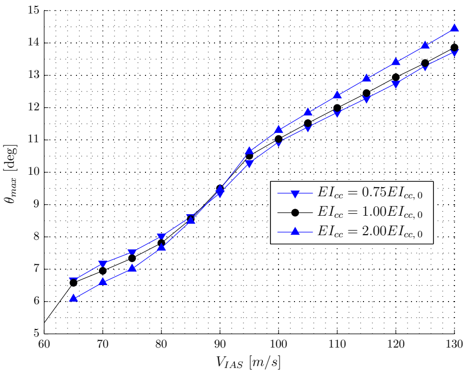
run#	$F_{t,\text{ini}}$ $[N]$	β_t [deg]	x_b $[cm]$	y_b
1	25.0	0.0	1.0	half-way span
2	50.0	0.0	1.0	half-way span
3	75.0	0.0	1.0	half-way span
4	25.0	30.0	1.0	half-way span
5	50.0	30.0	1.0	half-way span
6	75.0	30.0	1.0	half-way span
7	25.0	0.0	9.0	half-way span
8	50.0	0.0	9.0	half-way span
9	75.0	0.0	9.0	half-way span
10	25.0	30.0	9.0	half-way span
11	50.0	30.0	9.0	half-way span
12	75.0	30.0	9.0	half-way span
13	25.0	0.0	17.5	half-way span
14	50.0	0.0	17.5	half-way span
15	75.0	0.0	17.5	half-way span
16	25.0	30.0	17.5	half-way span
17	50.0	30.0	17.5	half-way span
18	75.0	30.0	17.5	half-way span
19	25.0	0.0	22.5	half-way span
20	50.0	0.0	22.5	half-way span
21	75.0	0.0	22.5	half-way span
22	25.0	30.0	22.5	half-way span
23	50.0	30.0	22.5	half-way span

24	25.0	0.0	1.0	half-way span
25	50.0	0.0	1.0	half-way span
26	75.0	0.0	1.0	half-way span
27	25.0	30.0	1.0	wing-tips
28	50.0	30.0	1.0	wing-tips
29	75.0	30.0	1.0	wing-tips
30	100.0	30.0	1.0	wing-tips
31	25.0	0.0	9.0	wing-tips
32	50.0	0.0	9.0	wing-tips
33	75.0	0.0	9.0	wing-tips
34	25.0	30.0	9.0	wing-tips
35	50.0	30.0	9.0	wing-tips
36	75.0	30.0	9.0	wing-tips
37	100.0	30.0	1.0	wing-tips
38	25.0	0.0	13.0	wing-tips
39	50.0	0.0	13.0	wing-tips
40	75.0	0.0	13.0	wing-tips
41	25.0	30.0	13.0	wing-tips
42	50.0	30.0	13.0	wing-tips
43	75.0	30.0	13.0	wing-tips

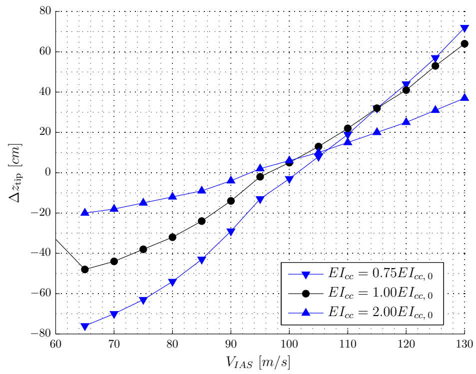
Appendix F

Results M600 aero-elastic analysis

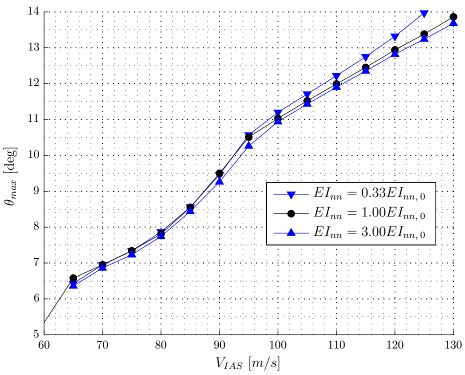
F.1 Results M600 torsional divergence analysis



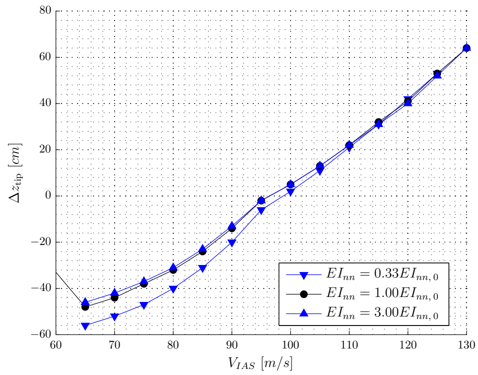
(a) EI_{cc} effect on twist angle



(b) EI_{cc} effect on tip deflection



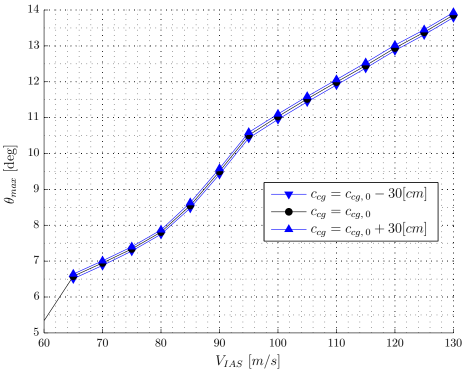
(c) EI_{nn} effect on twist angle



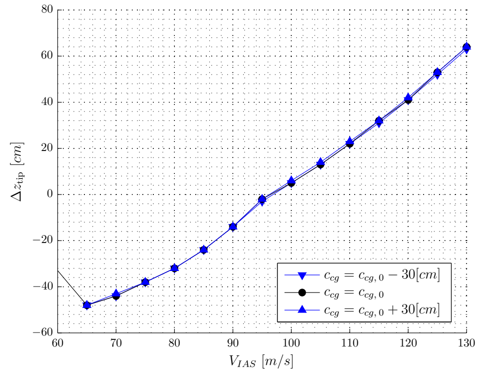
(d) EI_{nn} effect on tip deflection

Figure F.1: Figure will continue on next page

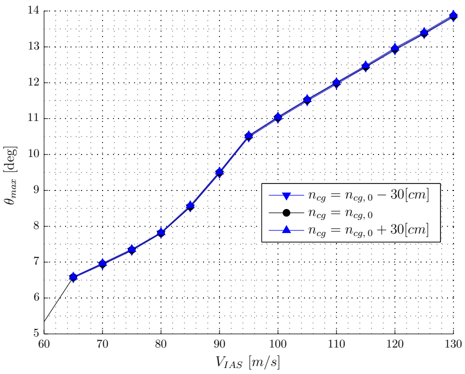
F.2 Results M600 control reversal and effectiveness analysis



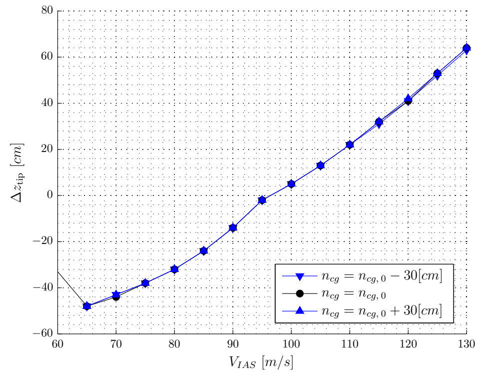
(e) c_{cg} effect on twist angle



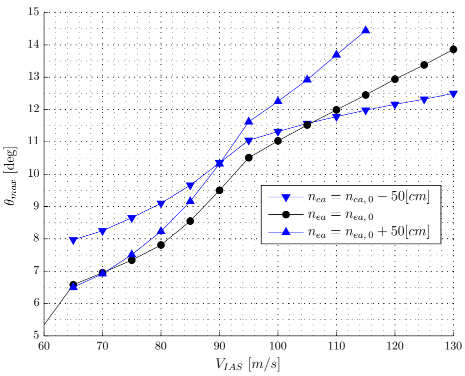
(f) c_{cg} effect on tip deflection



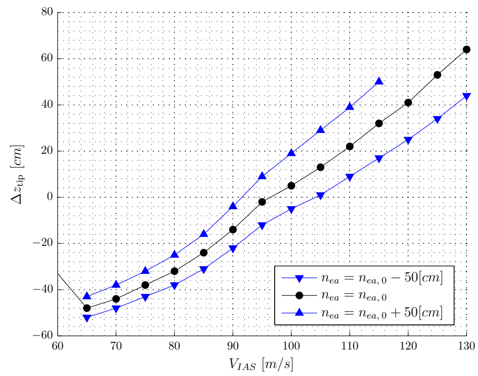
(g) N_{cg} effect on twist angle



(h) n_{cg} effect on tip deflection



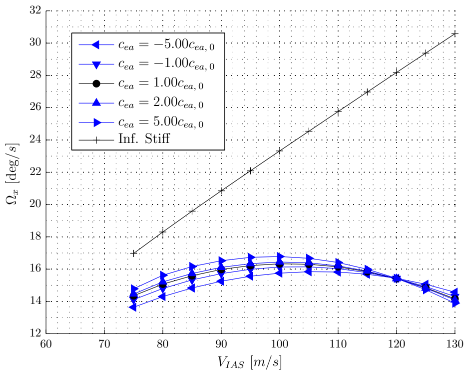
(i) n_{ea} effect on twist angle



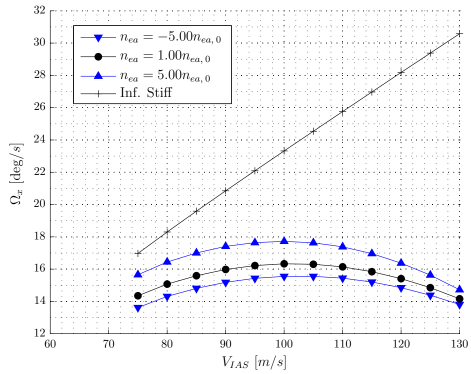
(j) n_{ea} effect on tip deflection

Figure F.1: Continued Figure: Other main wing parameters' effects on divergence

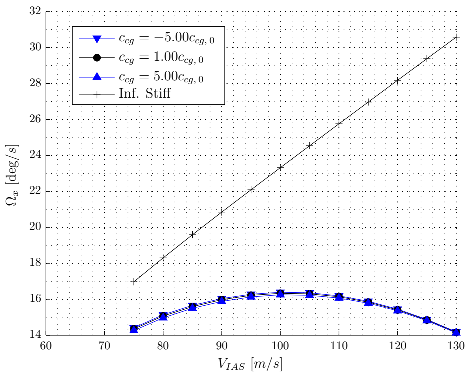
F.3 Results M600 flutter analysis



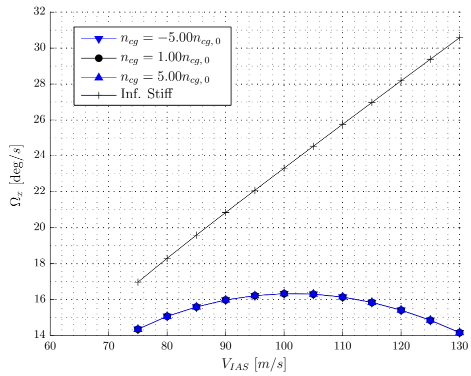
(a) Ω_x vs V, varying c_{ea}



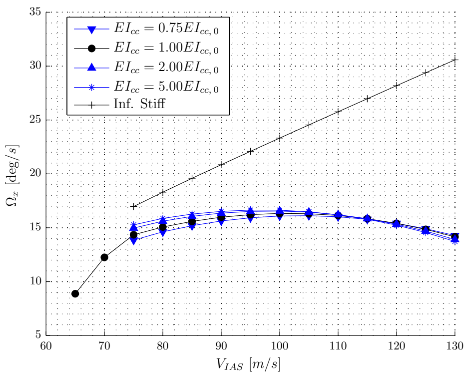
(b) Ω_x vs V, varying n_{ea}



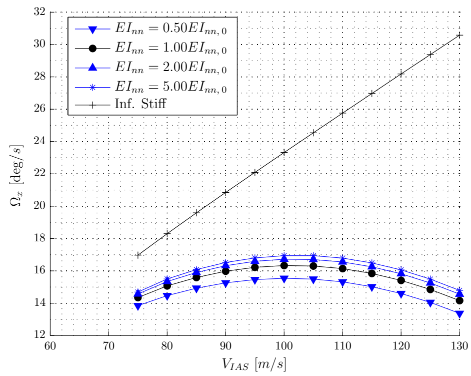
(c) Ω_x vs V, varying c_{cg}



(d) Ω_x vs V, varying n_{cg}



(e) Ω_x vs V, varying EI_{cc}



(f) Ω_x vs V, varying EI_{nn}

Figure F.2: Figure will continue on next page

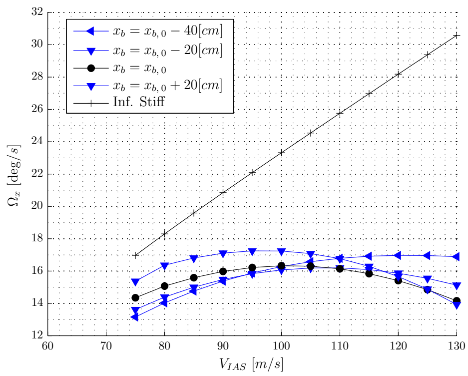
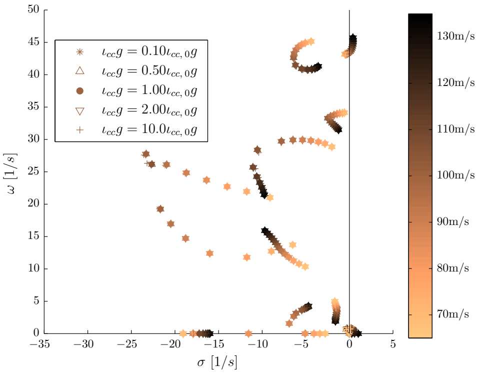
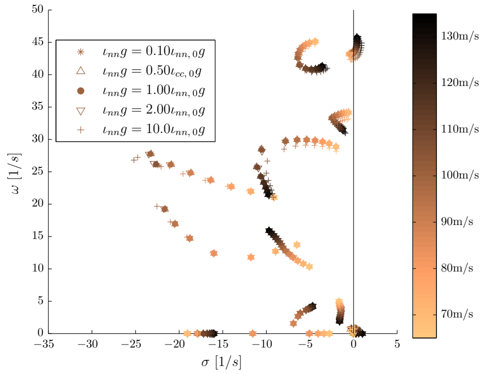


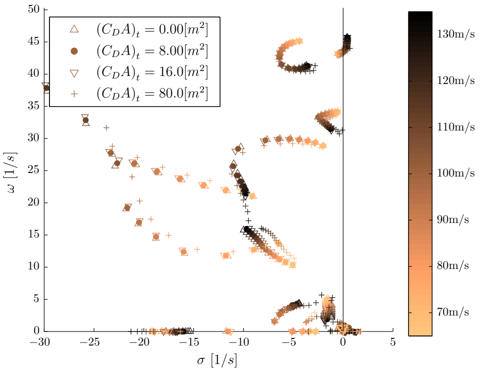
Figure F.2: Continued Figure: Other main wing parameters' effects on control effectiveness



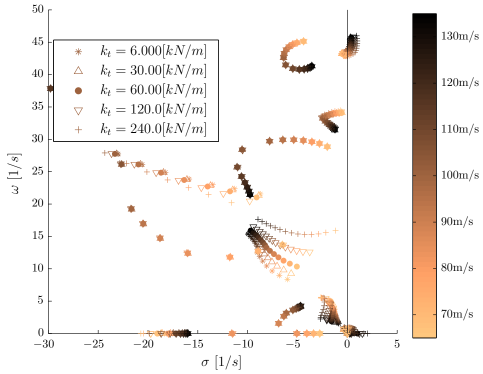
(a) Root locus with varying ι_{cc}



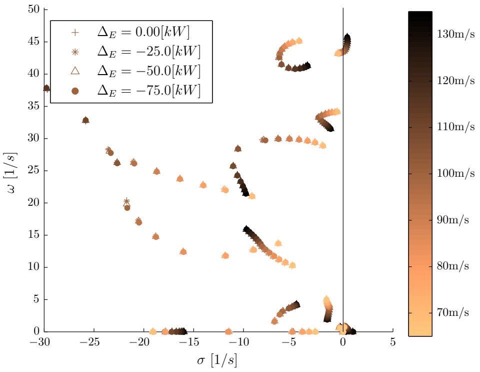
(b) Root locus with varying ι_{nn}



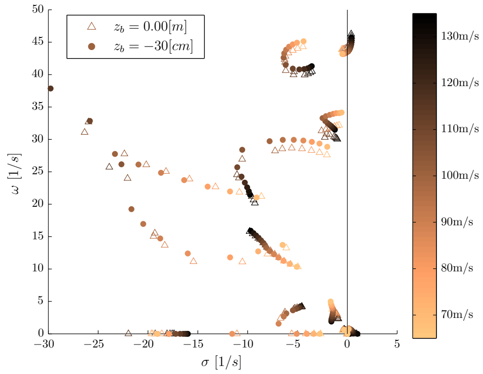
(c) Root locus with varying $(C_DA)_t$



(d) Root locus with varying k_t



(e) Root locus with varying Δ_E



(f) Root locus with varying z_b

Figure F.3: Other parameter effect on flutter; subscript $()_0$ denotes the benchmark run

Bibliography

Aerodynamics Research Group, . Low turbulence wind tunnel. URL http://www.lr.tudelft.nl/en/organisation/departments/aerodynamics-wind-energy-flight-performance-and-propulsion/facilities/low-speed-tunnels/low-turbulence-wind-tunnel/. Accessed 15 June 2013.

Agten, M.S. Aerodynamic analysis of wings in airborne wind energy applications. Master's thesis, Delft University of Technology, 2012.

Ahrens, U.; Diehl, M., and Schmehl, R. *Airborne Wind Energy*. Springer, Dordrecht, the Netherlands, 2013.

AircraftSpruce, . URL http://www.aircraftspruce.com/catalog/cmpages/bikevlar.php. Accessed 9 April 2013.

Archer, C.L. An introduction to meteorology for airborne wind energy. pages 81–94, booktitle =. Springer, Dordrecht, the Netherlands, 2013.

Argatov, I.; Rautakorpi, P., and Silvennoinen, R. Estimation of the mechanical energy output of the kite wind generator. *Renewable Energy*, 34(6):1525–1532, 2009.

Bisplinghoff, R.L.; Ashley, H., and Halfman, R.L. *Aeroelasticity*. Dover Publications, Inc., 1996.

Bontekoe, E. How to launch and retrieve a tethered aircraft. Master's thesis, Delft University of Technology, 2010.

Breukels, J. *An engineering methodology for kite design*. PhD thesis, Delft University of Technology, 2010.

Diehl, M. Airborne wind energy: Basic concepts and physical foundations. In Ahrens, U.; Diehl, M., and Schmehl, R., editors, *Airborne Wind Energy*, pages 3–22. Springer, Dordrecht, the Netherlands, 2013.

Drela, M. Integrated simulation model for preliminary aerodynamic, structural and control-law design of aircraft. In *proceedings of 40th AIAA, SDM Conference*, St. Louis,MO, 12-15 April 1999. doi:10.2514/6.1999-1394.

Drela, M. *ASWING 5.81 Technical Description - Unsteady Extension*, 2008a. URL http://web.mit.edu/drela/Public/web/aswing/asw_uns_theory.pdf. Accessed 3 November 2012.

Drela, M. *ASWING 5.82 User Guide*, 2008b. URL http://web.mit.edu/drela/Public/web/aswing/aswing_doc.txt. Accessed 15 November 2012.

Drela, M. *ASWING 5.86 Technical Description - Steady Formulation*, 2009. URL http://web.mit.edu/drela/Public/web/aswing/asw_theory.pdf. Accessed 10 November 2012.

Dunbar, B. Dfrc - research - facilities - research aircraft integration facility - ground vibration testing. URL http://www.nasa.gov/centers/dryden/research/Facilities/RAIF/vib.html#.Uo-LlTyJDAQ. Accessed at 12 November 2013.

FlyingFoam, . URL https://www.flyingfoam.com/content/choosing-right-foam. Accessed 5 April 2013.

Hepperle, M. Javafoil analysis of airfoils, 2006. URL http://www.mh-aerotools.de/airfoils/javafoil.htm. Accessed 2 April 2013.

Hulshoff, S.J. AE4930 Aeroelasticity. Reader, 2011. Delft University of Technology.

International Energy Agency, . World energy outlook 2009 fact sheet. URL http://www.worldenergyoutlook.org/media/weowebsite/2009/fact_sheets_WEO_2009.pdf. Accessed 15 May 2013.

Jensen, K. W7 aeroelastic analysis. Makani Power internal document, 2010.

Leuthold, R. Java-based surf-kite flight model for application to a kite-surfing simulator. Internship Report, 2013.

Loyd, M.L. Crosswind kite power. *Journal of Energy*, 4(3):106–111, 1980. doi:10.2514/3.48021.

Makani Power, . Makani history, 2012a. URL http://www.makanipower.com/company-history/. Accessed 10 January 2013.

Makani Power, . Makanipower how does it work?, 2012b. URL http://www.makanipower.com/how-does-it-work/. Accessed 10 January 2013.

Makani Power, . Makani power why airborne wind power?, 2012c. URL http://www.makanipower.com/why-airborne-wind/. Accessed 10 January 2013.

Meijaard, J.P.; Ockels, W.J., and Schwab, A.L. Modelling of the dynamic behaviour of a laddermill, a novel concept to exploit wind energy. In *proceedings of 3rd International Symposium on Cable Dynamics*, pages 229–234, Trondheim, Norway, 16-18 August 1999.

MSC.Nastran Version 68 Aeroelastic Analysis Users Guide. MSC. Software Corporation, 2004.

Noom, M.N. Theoretical analysis of mechanical power generaton by pumping cycle kite power systems. Master's thesis, Delft University of Technogoly, 2013.

NorthwestFoam.com, , 2007. URL http://www.northwestfoam.com/fact-specs.htm. Accessed 2 April 2013.

Ockels, W.J. Laddermill a novel concept to exploit the energy in the airspace. *Aircraft Design*, 4(2-3):81–97, 2001. doi:10.1016/S1369-8869(01)00002-7.

Ockels, W.J.; Lansdorp, B.; Breukels, J., and Spierenburg, G. The laddermill: work in progress. In *proceedings of European Wind Energy Conference, London*, European Wind Energy Conference & Exhibition, London, UK, 22-25 November 2004.

Peery, D.J. and Azar, J.J. *Aircraft Structures*. McGraw-Hill Book Company, New York, NY, 2nd edition, 1982.

Pocock, G. *The Aeropleustic Art or Navigation in the Air by the use of Kites, or Buoyant Sails*. W. Wilson, London, UK, 1827.

Richardson, D. The fundamental principles of composite material stiffness predictions, 2013. URL http://www.swcompositesgateway.co.uk/Property-Prediction.pdf. Accessed 8 April 2013.

Shahriar, S. and Erkan, T. When will fossil fuel reserves be diminished? *Energy Policy*, 37(1):181–189, 2009.

U.S. Department of Energy, . U.s. energy information administration. URL http://www.eia.gov/cfapps/ipdbproject/IEDIndex3.cfm?tid=2&pid=2&aid=2. Accessed 15 August 2013.

van der Vlugt, R.; Peschel, J., and Schmehl, R. Design and experimental characterization of a pumping kite power system. In Ahrens, U.; Diehl, M., and Schmehl, R., editors, *Airborne Wind Energy*, pages 403–436. Springer, Dordrecht, the Netherlands, 2013.

Van Garrel, A. *Development of a wind turbine aerodynamics simulation module*. ECN, 2003.

Vander Lind, D. M600 airfoils. Makani Power internal document, 2013a.

Vander Lind, D. M600 specifications. Makani Power internal document, October 2013b.

Vander Lind, D. Analysis and flight test validation of high performance airborne wind turbines. In Ahrens, U.; Diehl, M., and Schmehl, R., editors, *Airborne Wind Energy*, pages 473–490. Springer, Dordrecht, the Netherlands, 2013c.

Williams, P.; Lansdorp, B., and Ockels, W. Optiimal crosswind towing and power generation with tethered kites. *AIAA Journal of Guidance, Control, and Dynamics*, 31(1):81–93, 2008. doi:10.

Yung, Y.C. *An introduction to the theory of aeroelasticity*. Dover Publication, Inc., 31 East 2nd Street, Mineola, N.Y., 2002.

Zillmann, U. and Hach, S. Financing strategies for airborne wind energy. In Ahrens, U.; Diehl, M., and Schmehl, R., editors, *Airborne Wind Energy*, pages 117–137. Springer, Dordrecht, the Netherlands, 2013.