

UNIVERSITÀ DEGLI STUDI DI NAPOLI
“FEDERICO II”

THESIS

Application
of the Vortex Particle Method
to an UAV tiltrotor

Supervisors

Prof. Carlo de NICOLA
Ph.D. Paolo CACCAVALE

Candidate

Davide CANDELA

*A thesis submitted in fulfillment of the requirements
for the Master's degree in Aerospace Engineering*

in

March 2015

This page was intentionally left blank

Abstract

For an aerial vehicle, either manned or unmanned, a crucial aspect in the pre-prototyping phase of its design consists on the evaluation of the aircraft performance along the whole flight envelope, and so in all the different possible situations verifiable during the mission. This dataset is obviously important to understand the behaviour of the vehicle and to evaluate its conformity with the customer requirements. Moreover, these data are also checked by the regulator authority as a milestone on the certification process. For these reasons, it is clear that for a turboprop aircraft the inclusion of the propeller effect in the calculations is of central relevance since the rotational motion of the blades and of the nearby fluid strongly affects the vehicle performances, stability and controllability. However the flow field around a rotary wing is so complex and its resolution so computationally expensive that a direct numerical simulation based on computational fluid dynamics is unfeasible even employing parallel computing in combination with a partial and simplified geometry. This complexity is even greater in unsteady situations, such as those regarding a tiltrotor which switches from a helicopter like configuration to an airplane like one. From these considerations it emerges that an alternative way of modelling the effects of the propeller on the airplane aerodynamics and overall performances has to be found. In particular, in the present work the solution to this problem has been identified in the Vortex Particle Method (VPM): a technique based on the solution of the potential equations, hence valid in the assumptions of incompressible, inviscid and irrotational flow. This method is therefore suitable for a small vehicle which operates in the low-subsonic range of speeds and, unlike the actuator disk model, allows to take into account the peculiar characteristics of the employed propeller. Moreover, the relatively low impact of the VPM on the computational resources makes it particularly appropriate for unsteady multibody simulations which involve a relative motion among the different geometries employed. The particular problem analyzed in this work belongs to this category: the conversion phase of a surveillance tiltrotor UAV has been studied, considering both the moving propeller and the fixed wing which hosts its nacelle.

Keywords: potential flow, vortons, Vortex Particle Method, propeller, tiltrotor, panel methods, unsteady rotordynamics, propeller-wing interaction



Contents

Contents	iv
List of Figures	vii
List of Tables	x
List of Source Codes	xi
Notation	xii
List of Symbols	xiv
Acronyms	xvii
Introduction	xix
Thesis objectives and outline	xix
Background	xxi
Historical remarks	xxii
Tiltrotor: a critical analysis	xxvi
I Theory	1
Overview	2
1 Fluid-dynamics Equations	3
1.1 Navier-Stokes Equations	3
1.2 Euler's Equations	4
1.3 Vorticity Equation	5

1.4	Potential Flows	6
2	Panel Methods	22
2.1	Introduction	22
2.2	Boundary Conditions	23
2.3	Wake treatments	25
2.4	Numerical procedure	26
2.5	Unsteady panel methods	30
2.6	Trefftz technique	33
2.7	Vortex Particle Method	34
3	Propeller	40
3.1	Generalities and definitions	40
3.2	Main theories	42
II	Analysis & Simulation	46
	Overview	47
4	E-Pteron	48
4.1	Overall design	48
4.2	Fore propellers	50
5	Panel Method Solver	59
5.1	Generalities	59
5.2	Input Data Files	60
5.3	Output Data Files	78
5.4	Test Case	83
5.5	Motion law test	87
6	Simulation	92
6.1	Geometry discretization	92
6.2	Propeller performance estimation	94
6.3	Tilting rotor	97
6.4	Propeller-wing interaction	98
	Conclusions and future developments	108
6.5	Conclusions	108
6.6	Future developments	109
	Appendices	110
A	Points cloud treatments	111
B	e-Pteron in Digital Datcom+	118

CONTENTS

C An Example of PaMS DATAIN File	121
D An Example of Motion Routine	123
E PaMS DATAIN file for a tiltrotor	125
Bibliography	128
Webography	132
Index of Names and Citations	133
Index of Vehicles	135
Index	136

List of Figures

1	V-22 Osprey in different flight conditions	xxii
2	Bell XV-15 TRRA	xxiv
3	Chronology of tiltrotor evolution	xxv
4	NASA Large Civil TiltRotor concept	xxvi
5	Evolution of helicopter cruise speed	xxvii
6	Tiltrotor productivity ratio	xxvii
7	Rotor forward flight performance in terms of efficiency	xxviii
8	Tiltrotor and Helicopter productivity ratio	xxix
9	Payload as a function of range	xxix
10	Specific Productivity	xxx
11	Unfueled Range	xxx
1.1	Scheme summarizing the main fluid-dynamics models	9
1.2	Possible solutions for the flow over an airfoil	11
1.3	Fluid domain and nomenclature	13
1.4	The velocity potential near a solid boundary	14
1.5	Flow-field induced by a free stream	16
1.6	Flow-field induced by a point source/sink	17
1.7	Point doublet	18
1.8	Flow-field induced by a point vortex	19
1.9	Singularity distribution over a segment $\overline{x_1x_2}$	20
1.10	Doublet-Vortex equivalence	21
2.1	Leakage flow	27
2.2	Approximation of the body surface by panel elements	27
2.3	Influence of panel k on point P	29
2.4	Steady panel method procedure	30
2.5	Unsteady panel method procedure	31

LIST OF FIGURES

2.6	Trefftz plane	33
2.7	Vortex tube and its corresponding vorton	35
2.8	Gaussian core function	36
2.9	Example of vorton wake model	38
2.10	PaMS vorton wake model	38
2.11	Panel and vorton wake	39
3.1	Working conditions of a blade element	41
3.2	Propeller characteristic curves	43
3.3	Working conditions of a blade element according to the BEMT	45
4.1	E-Pteron: three views	50
4.2	E-Pteron: isometric view of the wing	51
4.3	E-Pteron: wing characteristics	51
4.4	E-Pteron: isometric view of the canard	52
4.5	E-Pteron: canard characteristics	52
4.6	Geometrical characteristics of the adopted blade	54
4.7	Isometric view of the designed propeller	54
4.8	Propeller performances	55
4.9	Propeller dimensional performances for different angular speeds	56
4.10	Propeller dimensional performances in forward flight	57
4.11	Minimum angular speed for forward and ascent flight	58
5.1	PaMS governing equations	60
5.2	Example of discretization in PaMS	61
5.3	Explanation for the <code>ang</code> keyword	71
5.4	Criticality at the wake boundaries	72
5.5	Core functions employed in Panel Method Solver (PaMS)	75
5.6	Explanation of the motion law applied in listing 5.9	78
5.7	PaMS in action	79
5.8	PaMS output tree	80
5.9	Reference frames in the <code>DATAOUTgeom</code> file	81
5.10	Experimental apparatus for a hovering test case	84
5.11	Axial velocity for a simple hovering test case and FFW	85
5.12	Axial velocity for a simple hovering test case and NRW	86
5.13	Thrust coefficient in hovering	87
5.14	Axial velocity for climb and descent with FFW	88
5.15	Time-event correlation for the tilting rotor case	89
5.16	Path of the rotor center in the tilting rotor case	89
5.17	Pivot position history for the tilting rotor case	90
5.18	Velocity history for the tilting rotor case	90
5.19	Rotor tip trajectory for the tilting rotor case	90
5.20	Average velocity for a preliminary tilting rotor case	91
6.1	E-Pteron fore propeller discretization	93

LIST OF FIGURES

6.2	E-Pteron canard discretization	94
6.3	Thrust evolution through time at 1250 rpm for different free-stream velocities	95
6.4	Propeller performances estimated by means of different techniques	96
6.5	Percentage error on thrust for PaMS	97
6.6	Time-event correlation for the tiltrotor simulation	98
6.7	Force history for the isolated tilting propeller	99
6.8	Reference frames operations for the tiltrotor	101
6.9	Assembly mesh for the tiltrotor	101
6.10	Streamlines for the tiltrotor case	103
6.11	Force history for the tiltrotor configuration	103
6.12	Pressure distribution for the tiltrotor case	104
6.13	Pressure distribution over the canard at $t = 0,72$ s	105
6.14	Pressure distribution for the tiltrotor case at $t = 6,00$ s	107
A.1	E-Pteron wing: original points cloud	113
A.2	Influence areas	114
A.3	E-Pteron wing: treated points cloud with detected LEs, TEs and focal line	114
A.4	E-Pteron: wing characteristics	115
B.1	E-Pteron geometry produced by listing B.1	120



List of Tables

4.1	E-Pteron: characteristics	49
4.2	E-Pteron: wing and canard characteristics	49
4.3	E-Pteron: design parameters for the two fore propellers	53
4.4	E-Pteron: fore propellers geometrical characteristics	54
5.1	Summary of the RFF keyword usage	66
5.2	Accepted values for the bctype keyword	74
5.3	Accepted values for the coremod keyword	75
5.4	Experimental conditions and results for the hovering test case	84
6.1	Estimation of the drag acting on the canard due to the propeller	106



List of Source Codes

5.1	Example of <i>geo</i> file in PaMS	61
5.2	Typical DATAIN file for PaMS - Far-Field conditions	62
5.3	Typical DATAIN file for PaMS - Unsteadiness management	63
5.4	Typical DATAIN file for PaMS - convergence check and solution speed up	64
5.5	Typical DATAIN file for PaMS - Body and Motion-Related Parameters	66
5.6	Typical DATAIN file for PaMS - Output Handling	68
5.7	Typical DATAIN file for PaMS - Geometry and Closure	69
5.8	Typical DATAIN file for PaMS - Body Motion	76
5.9	Extract of motion routine for PaMS	77
6.1	Propeller body motion, extracted from the DATAIN file	99
6.2	Canard body motion, extracted from the DATAIN file	100
A.1	Points cloud treatment algorithm	112
B.1	Digital Datcom+ input for E-Pteron	118
C.1	Typical DATAIN file for PaMS	121
D.1	Example of motion routine for PaMS	123
E.1	PaMS DATAIN file for a wing & canard tiltrotor configuration	125

Notation

Before reading this document it is favorable to introduce some notations employed hereafter in this thesis. At this purpose it has to be underlined that in the following dissertation the conventional index notation has been used, aiming to readability and simplicity. In this context the so called *Einstein summation convention* applies, so that repeated indices are implicitly summed over. For coherence with notation, the components of the position and velocity vectors are denoted by

$$\begin{aligned}\mathbf{x} &= (x, y, z) = (x_1, x_2, x_3), \\ \mathbf{V} &= (u, v, w) = (V_1, V_2, V_3),\end{aligned}$$

respectively. In these statements the vector notation has been pointed out, too. In fact a vectorial quantity is denoted by means of a bold character, whereas, in a similar fashion, the tensors are both bolded and marked with an underscore (e.g. $\underline{\boldsymbol{\tau}}$).

In addition, in order to write the N-S equations in their differential form it is useful to introduce the classical definition of *substantial derivative* (also called *material derivative*). This quantity is typically denoted – after *Gibbs* – by means of the symbol

$$\frac{Df}{Dt} = \frac{\partial f}{\partial t} + \mathbf{V} \cdot \nabla f, \quad (1)$$

where ∇ is the *nabla* operator and f is a generic field quantity (either scalar or vectorial). Physically, the substantial derivative can be seen as the rate of change of f as experienced by an observer that is moving along a streamline. In fact, such an observer would see variations of f due to both the stationary time-rate-of-change of the property (as stated by the term $\frac{\partial f}{\partial t}$) and the the actual motion of the observer himself (measured by the so called *convective derivative*, $\mathbf{V} \cdot \nabla f$). Furthermore, if f is a vectorial quantity, according to the index notation, the equation (1) can be also written as

$$\frac{Df_i}{Dt} = \frac{\partial f_i}{\partial t} + V_j \frac{\partial f_i}{\partial x_j}.$$

NOTATION

—

List of Symbols

Dimensionless Numbers and Coefficients

Symbol	Description	Definition
C_P	Power coefficient	$C_P = \frac{P}{\rho n^3 D^5}$
C_P	Pressure coefficient	$C_P = \frac{p - p_\infty}{\frac{1}{2} \rho_\infty V_\infty^2}$
C_Q	Torque coefficient	$C_Q = \frac{Q}{\rho n^2 D^5}$
C_T	Thrust coefficient	$C_T = \frac{T}{\rho n^2 D^4}$
J	Advance Ratio	$J = \frac{V_\infty}{nD}$
M	Mach number	$M = \frac{V}{a}$
Re	Reynolds number	$Re = \frac{\rho V L}{\mu}$
T_c	Thrust coefficient	$T_c = \frac{T}{\frac{1}{2} \rho (\Omega R^2)^2}$

Greek Symbols

Symbol	Description	Units
α	Vorton strength	m^3/s
ϵ	Twist angle	deg
η	Efficiency	—

LIST OF SYMBOLS

Γ	Circulation of a vector field, Vortex intensity, Dihedral angle	m^2/s , m^2/s , deg
γ	Vortex intensity per unit length/area/volume	m/s
κ	<i>Fourier's</i> coefficient of heat conductivity	$\text{W}/\text{m}\cdot\text{K}$
Λ	Sweep angle	deg
λ	Taper ratio, c_{tip}/c_{root}	—
μ	Dynamic viscosity, Doublet strength	$\text{Pa}\cdot\text{s}$, m^4/s
μ_v	Bulk viscosity	$\text{Pa}\cdot\text{s}$
Ω	Angular speed	rad/s
ω	Vorticity vector	$1/\text{s}$
ϕ	Velocity potential	m^2/s
ψ	Stream function	m^2/s
ρ	Density	kg/m^3
σ	Source strength, Vorton core radius, Solidity	m^3/s , m, —
$\underline{\tau}$	Free shear tensor	Pa
$\underline{\tau}_d$	Deviatoric portion of the free shear tensor	Pa
θ	Pitch angle	deg
ξ_ψ	Core-function	—

Roman Symbols

Symbol	Description	Units
a	Laplacian sound speed	m/s
c	Chord	m
D	Drag, Rotor diameter	N, m
e	Internal energy	J/kg
f	Generic function	—
f_b	Body force	N/kg
L	Lift, Characteristic length	N, m

LIST OF SYMBOLS

\mathbf{n}	Unit vector locally normal to a surface	m
N	Number of blades	—
p	Pressure	Pa
\bar{r}	Dimensionless radial coordinate r/R	m
\mathbf{r}, r	Position vector, radial coordinate	m, m
R	Gas constant, Rotor radius	J/kg·K, m
T	Temperature, Thrust	K, N
t	Time	s
$\underline{\mathbf{U}}$	Unit tensor	—
(u, v, w)	Velocity vector components	m/s
\mathbf{V}	Velocity vector	m/s
\mathbf{x}	Position vector	m
(x, y, z)	Position vector components	m
\bar{y}	Dimensionless spanwise coordinate, $2y/b$	—

Subscripts and Superscripts

Symbol Description

b	Body
f	Focal line
∞	Far-field conditions
l	Lower side
LE	Leading Edge
t	Tilting phase
TE	Trailing Edge
u	Upper side
w	Wake



Acronyms

BC	Boundary Condition
BEMT	Blade Element Momentum Theory
CFD	Computational Fluid Dynamics
FAA	Federal Aviation Administration
FFW	Fully Free Wake
JVX	Joint Vertical Experimental
LCTR	Large Civil TiltRotor
LE	Leading Edge
MSL	Mean Sea Level
NACA	National Advisory Committee for Aeronautics
NASA	National Aeronautics and Space Administration
NRW	No Roll-up Wake
N-S	Navier-Stokes
PaMS	Panel Method Solver
PDE	Partial Differential Equation
QTR	Quad TiltRotor
RHS	Right Hand Side

ACRONYMS

S/L	Sea Level
SOR	Successive Over Relaxation
STOL	Short Take Off and Landing
TE	Trailing Edge
TRRA	TiltRotor Research Aircraft
UAV	Unmanned Aerial Vehicle
VPM	Vortex Particle Method
VTOL	Vertical Take Off and Landing
VUAV	Vertical Take Off and Landing Unmanned Aerial Vehicle



Introduction

“The moment you doubt whether you can fly,
you cease for ever to be able to do it.”

— J.M. Barrie, *Peter Pan*, 1902

Thesis objectives and outline

According to the current trends, due to the huge exploitation of the skies and the consequent rise of environmental issues, the future of both civil and military aviation is expected to be more eco-friendly and less fuel-demanding. For these reasons the aerospace engineering is reconsidering the propeller propulsion since it is more efficient than the one provided by jets and moreover it can be combined with electrical engines, reducing thus the emissions even more. Whether this perspective is joint with the increasing air traffic volume and hence with the need to expand the sky capabilities, then the idea of a propeller-driven vehicle able to vertically take off like a helicopter (*i.e.* without requiring long runways) and to efficiently fly like an airplane spreads out. The potentiality of such an aircraft, known as tiltrotor, are so wide that in the last fifty years huge investments have been done on this concept. Nevertheless, as witnessed by the long testing times required for the actual vehicles, it is easy to understand that the conversion phase of a tiltrotor is extremely breakable so that flight campaigns should be preceded by a complete set of both wind tunnel tests and numerical simulations. However, it is clear that the facilities needed to collect reliable wind tunnel data are not available to small companies and, moreover, their employment is expensive and not always affordable for the whole duration of the tests. This is especially true in the early stages of the aircraft design, when the configuration has to be changed in order to explore several possibilities and hence to find an optimum condition. For these reasons the numerical simulation appears to be the most reasonable and doable way to design and develop a tiltrotor.

Even though currently the numerical fluid dynamics is mainly governed by the CFD, this technique is not always the best choice to approach a fluid dynamic problem. In fact, despite its generality, especially for complex problems, the CFD may require long times and high performance cluster hardware to compute the results. In particular, for a propeller-driven aircraft the flow field around the vehicle is governed by a strong swirl and by a thigh mutual interaction between the vortex structures detached from the rotor and the airplane surfaces. These characteristics make the solution of the flow field around a rotating propeller almost unfeasible through the CFD even exploiting the most powerful commercial computers, so that it may be needed to enforce some simplifying assumptions, for example employing the actuator disk model. However, despite allowing an estimation of the the performances of a generic rotor, this last solution does not take into account the geometry of the particular propeller adopted, so that the details of the flow field near the blades remain unknown. For these reasons, if the propeller under investigation has diameter and angular speed such that the resulting flow field can be assumed to be both incompressible and inviscid, then one could employ a potential method rather than perform a CFD analysis.

In this context, the objective of this thesis is to prove the effectiveness of the so called Vortex Particle Method applied to some rotary wing problems related to a small surveillance tiltrotor Unmanned Aerial Vehicle (UAV). In particular, on the bases of some known data concerning the considered aircraft, a suitable propeller has been designed targeting the thrust required to perform a balanced vertical flight and its performances have been estimated by means of both the classical Blade Element Momentum Theory (BEMT) and the VPM, validating hence the latter method through a good agreement with the predictions of the BEMT. The thrust data evaluated in this way have been employed to find out some ranges of the flight envelope, which appeared to be consistent with the available data, proving thus the forcefulness of the designed propeller. Finally, a rough tilting rotor configuration, composed by a propeller and a wing, has been analyzed, carrying out the time evolution of the produced thrust and evaluating the forces acting on the wing due to the flow induced by the rotating propeller.

Despite the academic purpose in fulfillment of the Master's degree in aerospace engineering, the present work is intended as a baseline for the development of a real tiltrotor UAV. In particular, the knowhow acquired during the writing of this volume should lead to the optimization of the control laws responsible for the power supply to the two tilting rotors equipping the vehicle. At this purpose, it could be useful to extend the code employed furnishing it with the capability to solve the dynamics equations, too.

In order to lead the reader toward the simulation of a rough tiltrotor configuration made by a tilting propeller and a fixed wing, the present volume has been organized in two different parts:

- *Part I* – Theory;
- *Part II* – Analysis & Simulation.

The first part introduces the main theoretical topics required for a full understanding of the topics dealt in the subsequent chapters of this volume. In particular, great attention has been paid to the theory lying behind the potential flows and their numerical solution through the so called panel methods. Nonetheless a brief introduction to the propeller nomenclature and classical

theories has been written, so that the reader is endowed at least with some general guidelines that may both help him to understand the numerical data reported hereafter and stimulate him to increase his knowledge through the bibliography at the end of this volume.

After the dutiful introduction carried out in Part I, the core of the work is reported in Part II, where the vehicle under investigation is presented, its tilting propellers are designed and analyzed and, finally, the wing & tilting propeller simulation is carried out. Furthermore, among these results, a whole chapter has been dedicated to the software employed: PaMS. In effect, a short guide to the correct usage of this solver has been written, and a couple of test cases have been presented aside with the source codes on which they are based.

Background

Today the aviation world is mainly populated by two different engineering concepts whose way of counteracting the gravitational force represents two extreme cases: the airplane and the helicopter. In fact, whereas the first one exploits a fixed wing to provide the lift needed in each condition of the flight envelope, the latter accomplishes the same task employing a big diameter rotary wing. Obviously, as it is easy to think about, these two kind of vehicles are related to several trade-offs, resulting in different performances and in a limited application scope. For instance one of the main virtues of a helicopter consists on its ability to take off vertically: for this reason a helicopter is said to be a Vertical Take Off and Landing (VTOL) vehicle. However, by contrast, this feature becomes a limitation on the vehicle maximum speed due to the presence of reverse flow phenomena, ultimately resulting in the stall of the rotor. On the other hand an airplane has better cruise performances but lacks the controllability and the possibility to hover that instead are the strength of a rotary wing aircraft. In the light of these considerations it appears clear that a mission could be suitable either for an airplane or for a helicopter: for example the first one could be successfully employed in long range missions where the short flight time is a key point, whereas the second one may be used to deliver a cargo in a location characterized by low accessibility.

Despite being the two most common kind of lifting aerial vehicles, the aircraft and the helicopter are not the only flying machines which belong to this category. In fact some intermediate and bridging configurations between the fixed wing and the rotary one do exist: in this scenario an important role is played by the so called *tiltrotor*. A tiltrotor is an hybrid aircraft which combines the VTOL capabilities of the helicopter with the relatively high cruise speed, range and efficiency characterizing a fixed wing airplane. For this reason the flight envelope of a tiltrotor stems from the overlapping of those of a plane and of a helicopter, therefore increasing the mission flexibility. This vehicle indeed generates lift and propulsion by way of one or more powered rotors (sometimes called *proprotors*) mounted on rotating engine pods or nacelles usually located at the ends of a fixed wing. For vertical flight, the rotors are angled so the plane of rotation is horizontal, lifting the way a helicopter rotor does. As the aircraft gains speed, the rotors are progressively tilted forward, with the plane of rotation eventually becoming vertical (see figure 1). In this mode the wing provides the lift, and the rotor provides thrust as a propeller. In this configuration the rotor operates nearly in axial flux, therefore the asymmetrical flow (and hence lift) conditions which affect a conventional helicopter rotor at high speeds are not verified: it is possible to realize higher

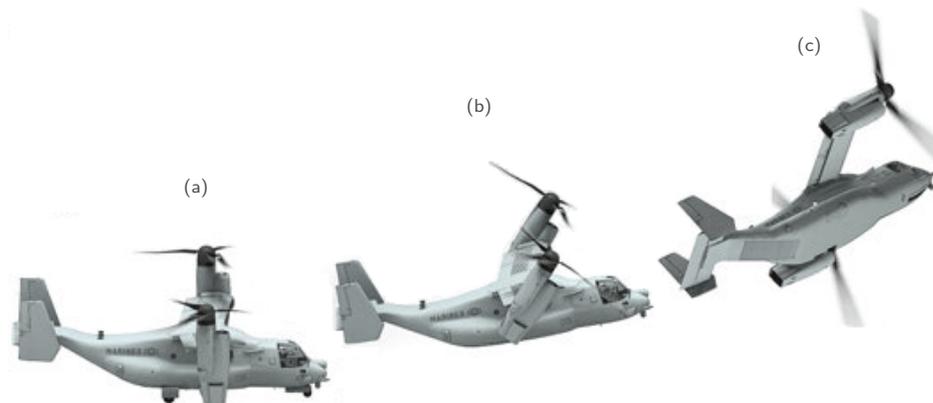


Figure 1: V-22 Osprey in different flight conditions: (a) hovering; (b) tilting; (c) cruise

speeds. Moreover, since in a tiltrotor the blades can be designed to optimize the horizontal flight rather than the vertical one, this aircraft can be less expensive in terms of fuel consumption with respect to the helicopter. This same concept can be seen over another light, too: with the same tank capacity, the range of a tiltrotor could be greater than that of a helicopter.

Historical remarks

Although the first successful flight operated by a tiltrotor is dated 1954, the history of this concept is far more ancient (see timeline in figure 3 on page xxv). The first decades of the XX century were dominated by a slight growth of the aviation industry: during those years numerous flying machines were devised, among which there were the first VTOL concepts. One of these was developed in the U.S. by *Henry Berliner* in the early 1920s. This design resembled a fixed-wing biplane of the period, except that it had a large diameter fixed-pitch propeller mounted on a vertical shaft near the tip of each wing. For forward flight, the shafts would be tilted forward. Even though the propellers were not able to assume the conventional airplane configuration, the Berliner helicopter employed a rotor arrangement used on current tiltrotor aircraft. Ten years later, in September 1930, the convertiplane idea was patented by *George Lehberger*, but he did not further develop the concept: the vectored thrust VTOL aircraft required many technology advancements before it would become a practical aircraft type.

After other attempts performed during World War II both by the Germans (in the form of the *Focke-Achgelis FA-269* trail-rotor convertiplane) and by the Americans (by means of *Platt-LePage XR-1A* lateral twin-rotor helicopter), the next significant appearance of the tiltrotor occurred in early 1947 when the *Transcendental Aircraft Corporation* initiated work that led to the development of the *Transcendental Model 1-G* tiltrotor aircraft. This single-seated aircraft is the first tiltrotor to have flown: it was the July 6, 1954. The *Transcendental Model 1-G*, however, met an unfortunate end. After successfully completing more than 100 flights in a period of just over one year, including partial conversions to the airplane mode, the prototype crashed on July 20, 1955. In spite of its success, the *Transcendental* project ceased in 1957 due to the

withdrawal of funds by the U.S. Air Force.

The death of the *Transcendental* was, however, the starting point of a milestone in the tiltrotor evolution: the *Bell XV-3*. In fact the funds taken away from the *Transcendental* were deployed on this project. The purpose of the XV-3 was to prove that a convertiplane could be practical, stable and controllable. However the first hovering tests hardly proved these features: since the first helicopter-like flights in 1955, the XV-3 appeared to suffer a rotor dynamic instability which resulted in strong vibrational phenomena. It occurred three years to the *Bell* to solve this issue, but finally, on December 18, 1958, the XV-3 achieved to be the first aircraft to complete a dynamically stable full conversion to the airplane mode.

The success of the *Bell* pushed the U.S. Army and the National Aeronautics and Space Administration (NASA) to form a joint venture aimed at the further developing of the tiltrotor technology. In 1971 this collaboration brought the TiltRotor Research Aircraft (TRRA) Project Office to life, whose prime contractor was – once again – the *Bell Helicopter Textron Inc.*. During this collaboration the advantage of multiple or joint participation became clear: in this way it was possible to maintain the continuation of project funding even when one agency was experiencing a temporary funding shortfall. The result of the Project Office was the *Bell XV-15* TRRA (see figure 2), a twin-engine tiltrotor research aircraft. Two aircrafts were built to prove the tiltrotor design and explore the operational flight envelope for military and civil applications.

Since its first test flight on April 23, 1979, the XV-15 rapidly improved: three months later, on July 24, the flight envelope expansion activities led to the first full in-flight conversion from helicopter to airplane mode. During that initial airplane mode flight lasting about 40 minutes, the crew evaluated climbs, descents, turns, accelerations, and decelerations and reached an airspeed of 160 kts. The remarkable achievements, both technical and operational, of the XV-15 TRRA were directly responsible for the introduction of the world's first military and civil tiltrotor aircraft. Without the technology validation and the demonstrations provided by the TRRA, it would not have been possible for the leaders of industry and the Government to be confident enough to launch these new aircraft production programs.

At the beginning of 1980s, in the U.S. the Department of Defence was seeking a multi-purpose aerial vehicle able to perform missions related to different branches of the armed forces. In particular, among these missions there were: Marine vertical assault, Navy rescue and logistics, Air Force long-range special operations, Army medical evacuation, long-range combat logistics support and combat air assault support. With a single technical approach identified that could satisfy the requirements, advocacy of the multiservice tiltrotor aircraft to the Congress and to the administration was initiated: the Joint Vertical Experimental (JVX) program took place. The TRRA experience suggested both that a joint venture was needed in order to reduce the financial risk and that the *Bell* Company owned the state of the art know-how in the field. As a consequence two previously concurrent industries begun to work side by side: *Bell* and *Boeing*. The outcome of this collaboration was the V-22 Osprey, whose first flight was successfully performed on March 19, 1989. Despite the initial enthusiasm, in the early stage of his life, the V-22 proved not to be highly reliable. In fact two of five full scale prototypes crashed and were destroyed, nevertheless the accidents did not compromise the project survival. Also due to these troubles the V-22 Osprey struggled to become operative. In fact this machine joined the U.S. Navy in 2007 - eighteen years later its first test flight at Bell's Flight Research Center at Arlington, Texas.

The market expansion of the tiltrotor to the civil aviation (started in June 1981 with the first



Figure 2: *Bell XV-15 TRRA*

public demonstration of the XV-15 TRRA capabilities at the Paris le Bourget Air Show) begun to concretize in 1996, when the two prime contractors of the V-22 Osprey – *Bell* and *Boeing* – renewed their partnership with the goal to develop a civil tiltrotor aircraft. This program was intended to produce a civil tiltrotor aimed to become part of the private regional transportation market, focusing on offshore oil and gas operators. However in 1998 it was announced that *Boeing* had pulled out from the project and that it was going to be replaced by the anglo-italian *AugustaWestland*, a *Finmeccanica* company. The product of this partnership, the BA609, achieved to fly on March 3, 2003, and to perform a full conversion to the airplane mode on July 22, 2005. In September 2009 the fatherhood of the project changed again. In fact, since the *Bell* was dissatisfied with the commercial prospects, *AugustaWestland* ruled it out to speed the project up. As a consequence the BA609 was redesignated as AW609. Actually this aircraft has performed about 800 hours of flying test, and *AugustaWestland* is targeting the Federal Aviation Administration (FAA) civil certification in 2017 [AugustaWestland, 2014] with deliveries set to begin immediately after.

Even though over the last sixty years the tiltrotor design has been an improvement of the initial know-how developed within the XV-3 program, it does not lack alternative configurations and applications. Among these it is possible to cite the Quad TiltRotor (QTR), a four-rotor derivative of the V-22 developed jointly by *Bell Helicopters* and *Boeing*. This ambitious concept – in development since the 1999 – is intended to have a cargo capacity roughly equivalent to the C-130 HERCULES, to carry ninety passengers and to cruise at 250 kts. An example of non conventional employment of the tiltrotor technology is, instead, represented by the *Bell Helicopters'* *Eagle Eye* Vertical Take Off and Landing Unmanned Aerial Vehicle (VUAV), currently waiting to take service by U.S. Coast Guard [Beshears and Peterson, 2004].

17/12/1903 ... ●	Wright Flyer I first flight.
1922-1925 ... ●	Henry Berliner Helicopters.
Sep 1930 ... ●	George Lehberger's patent.
1937-1939 ... ●	Baynes Heliplane.
1942 ... ●	Focke-Achgelis FA-269 trail-rotor convertiplane.
1947 ... ●	<i>Transcendental Model</i> 1-G development begins.
06/03/1954 ... ●	<i>Transcendental Model</i> 1-G first flight.
15/02/1955 ... ●	Haviland Platt's patent.
20/07/1955 ... ●	<i>Transcendental Model</i> 1-G crash.
11/08/1955 ... ●	initial hover trial of the XV-3.
1956 ... ●	York convertiplane.
25/10/1956 ... ●	number 1 XV-3 crashes due to instability.
18/07/1957 ... ●	beginning of the investigations of the dynamic behaviour of the XV-3.
18/12/1958 ... ●	the XV-3 completes a full conversion.
1971 ... ●	birth of the TRRA Project Office.
23/04/1979 ... ●	XV-15 first flight.
24/07/1979 ... ●	XV-15 performs a full conversion.
1981 ... ●	birth of the J VX program.
Jun 1981 ... ●	the XV-15 makes a public demonstration at the Paris Air Show.
19/03/1989 ... ●	V-22 Osprey first flight.
1993 ... ●	starting of the <i>Eagle Eye</i> program.
Nov 1996 ... ●	<i>Bell</i> and <i>Boeing</i> announce the beginning of the civil tiltrotor program.
1998 ... ●	<i>AugustaWestland</i> replaces <i>Boeing</i> in the civil tiltrotor program.
1999 ... ●	beginning of the QTR program.
Mar 2000 ... ●	beginning of V-22 Osprey training for the Marine Corps.
2002 ... ●	U.S. Coast Guard orders the <i>Eagle Eye</i> VUAV.
03/03/2003 ... ●	BA609 first test flight.
2007 ... ●	the V-22 Osprey become operative in the U.S. Navy.
2009 ... ●	<i>AugustaWestland</i> becomes the only owner of the civil tiltrotor project.

Figure 3: Chronology of tiltrotor evolution



Figure 4: NASA Large Civil TiltRotor concept [Acree, 2012]

Tiltrotor: a critical analysis

Although a first list of *pro et contra* of the tiltrotor concept has already been carried out in the Background section, a deeper, quantitative analysis of this issue is required in order to build a cost-benefit scenario and to find out the market segment to whom the product could be attractive [see Leishman, 2007].

The peculiarity of a tiltrotor to combine VTOL capabilities with relatively high cruise speeds has been already pointed out. At the first sight this integration could seem to be the answer to the requests of the military world, asking for it in order to perform their missions more effectively. However this meeting of characteristics – which are somehow concurrent – is restricted by costs, whose form is of technological development, manufacture and operation of the aircraft. Howbeit another aspect of this trade-off is the possibility offered by VTOL or Short Take Off and Landing (STOL) aircrafts to remove or at least limit the dependency on runways. This last point is one of the main reasons which are pushing the NASA to develop the so called Large Civil TiltRotor (LCTR). In fact the purpose of the agency is to improve the capacity of the airspace system as a whole without requiring construction of new runways or expansion of airport boundaries [Acree, 2012].

The figure 5 underlines the speed advantage of the tiltrotor over the VTOL niche ruler: the helicopter. In particular, it could be observed that the gain in speed for a helicopter is becoming slower and slower year by year. This decreasing trend states that, from this point of view, the helicopters are reaching a technological barrier which cannot be broken without employing more innovative technologies or concepts. On the other hand the same graph shows that a tiltrotor such as the *Bell-Boeing V-22 Osprey* (that can be considered the actual *state of the art* in this field) demonstrated cruise speeds up to 250 kts: that is to say 35% ÷ 50% faster than a modern helicopter. Even though this could seem a good result, other aspects have to be taken into account. At this purpose the data in figure 6 show that the productivity ratio of the tiltrotor at this speed only gives a gain of 25% over the helicopter at a radius of action of 200 nm. Moreover, technology developments already in the pipeline mean that future helicopters will likely cruise at speeds approaching 190 kts, and on this basis only a 9% increase in speed productivity would be obtained over the helicopter.

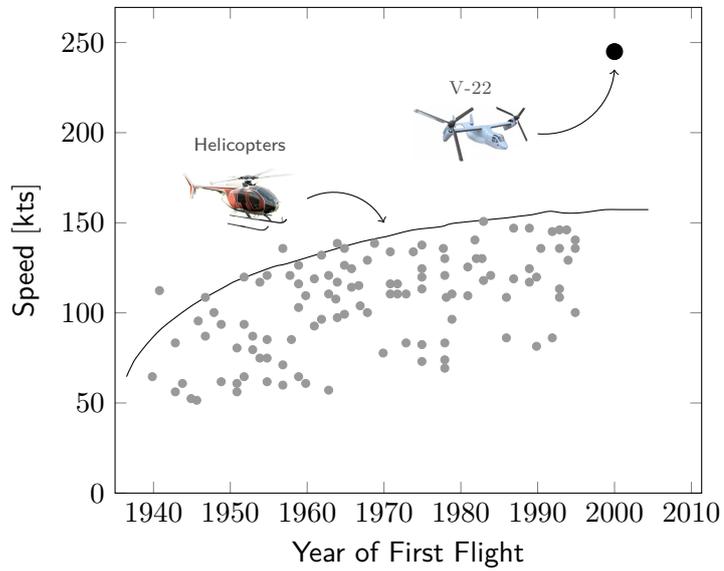


Figure 5: Evolution of helicopter cruise speed [Leishman, 2007]

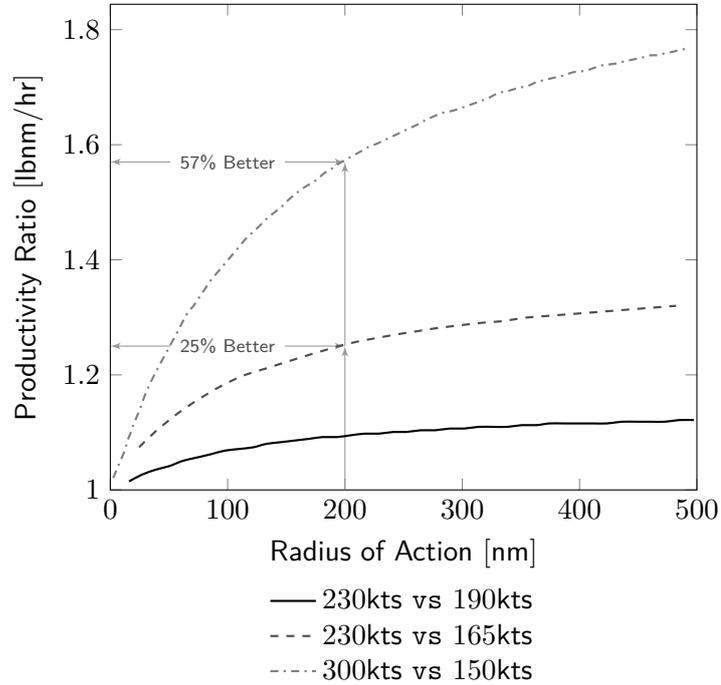


Figure 6: Tiltrotor productivity ratio [Leishman, 2007]

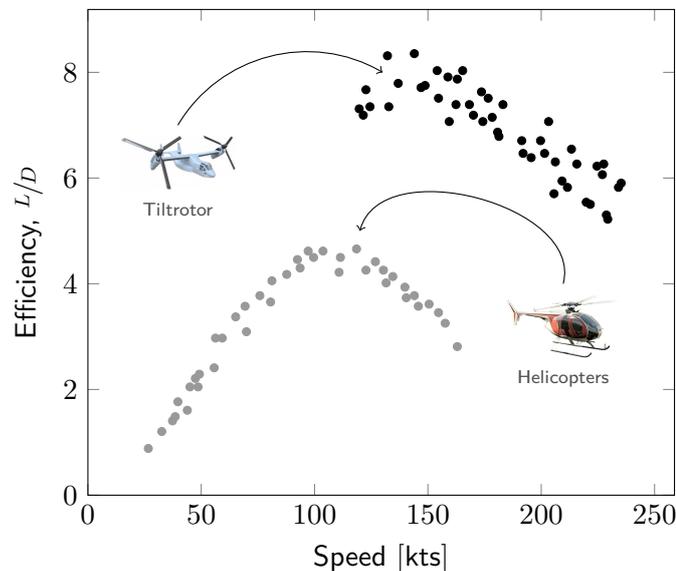


Figure 7: Rotor forward flight performance in terms of efficiency [Johnson, 2013]

Obviously the cruise speed is not the only relevant factor to judge the goodness of a transport vehicle. In fact other important aspects to an operator are, for instance, range and payload, aside with the transport efficiency. Moreover for a rotorcraft concept, a hover time requirement could also be specified, resulting in a trade-off between efficient hover and flight. At this purpose it is necessary to specify that a tiltrotor, being a hybrid concept, is not as aerodynamically efficient as a helicopter in hovering flight or as efficient as an airplane in forward flight. This issue is depicted in figure 7, where the aerodynamic efficiency (*i.e.* the Lift over Drag ratio) in forward flight is reported for both the helicopter and the tiltrotor (in this case the data are referred to a *Bell XV-15*).

About the transportable payload, figure 8 shows that a tiltrotor can carry only about half the payload of a transport helicopter such as the *Sikorsky CH-53E*, irrespective of the range. Furthermore the same picture points out that the *V-22 Osprey* experiences a considerable loss in performance when operating at high altitudes (*i.e.* in low density conditions), and that this loss is even greater than that relative to a medium weight utility helicopter such as the *UH-60 Blackhawk*. In fact, despite being able to carry a heavier payload at Mean Sea Level (MSL), the *V-22* is outperformed by the *Blackhawk* in *hot and high* conditions. So it appears clear that the tiltrotor is not going to replace the helicopter as a load carrier. Additionally, due to the fact that the tiltrotor – like the helicopter – is not as aerodynamically efficient as an airplane, it has to carry a great deal of fuel to cover longer distances. Since this amount of fuel erodes the payload which is possible to carry, maximum range and maximum payload form a trade-off pair. This perspective is highlighted in figure 9, which shows that from this point of view the *V-22 Osprey* has almost the same performance as the average helicopter. Note that a helicopter can be designed to carry a significant useful load over short distances by trading against fuel capacity, or can carry a smaller payload over higher ranges by using long-range fuel tanks. For this reason

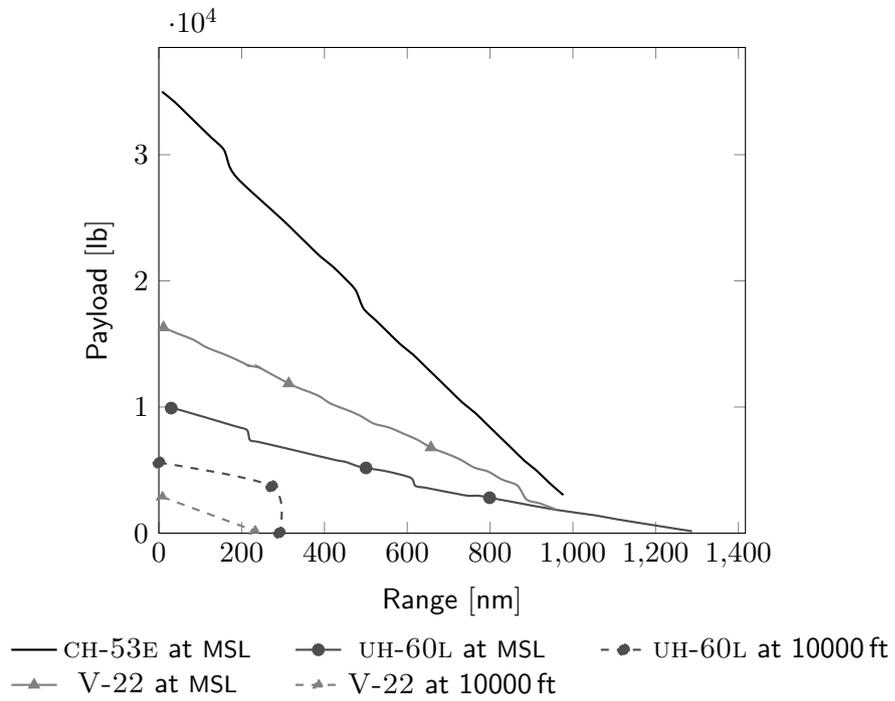


Figure 8: Tiltrotor and Helicopter productivity ratio [Leishman, 2007]

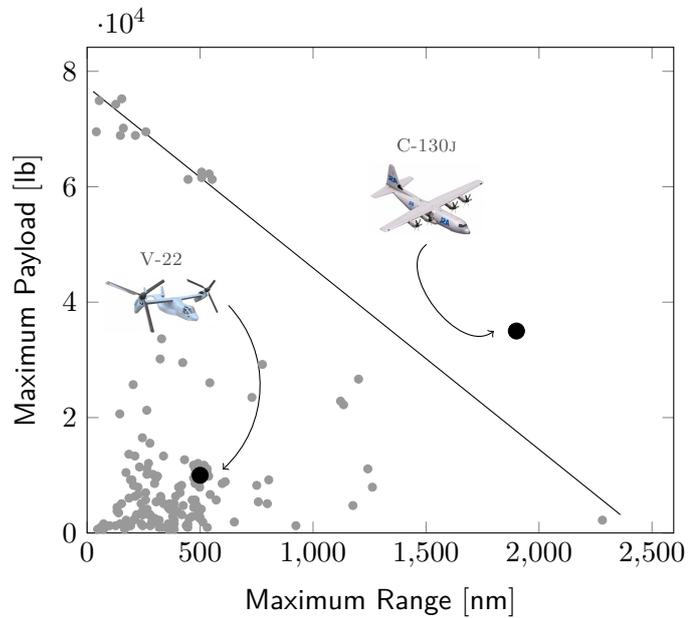


Figure 9: Payload as a function of range [Leishman, 2007]

on the graph depicted in figure 9 extreme points – characterized either by low range and high payload or, *vice versa* by high range and poor payload capabilities – are present. These points are aligned along a bounding line representing the technological limit of the conventional helicopters. On the graph under examination it is also represented the point related to a carrier aircraft such as the *Lockheed C-130J*: its position is above the helicopter limiting line, meaning that cargo planes are almost peerless in high range transportation of heavy payloads.

Both the factors already analyzed (*i.e.* payload carried and speed of transportation) can be summarized and merged in just one parameter: the *specific productivity*. This quantity is defined as the product of the maximum payload transported (over a given distance) and the speed of transportation, normalised by the maximum takeoff weight of the aircraft; *i.e.*

$$\text{specific productivity} = \frac{\text{maximum payload} \cdot \text{speed}}{\text{maximum takeoff weight}}.$$

In figure 10 are reported specific productivity data in terms of a 200 nm stage distance. As it could be noted, the trends appear to be asymptotic, confirming that helicopters are now reaching aerodynamic and other technical barriers limiting their productivity. But the data given in figure 10 also suggest that the tiltrotor (at least in its actual form) is not the answer to improving vertical-lift productivity. In fact the ability of the *V-22 Osprey* to cruise up to 50% faster than a modern helicopter is nearly all offset in terms of its specific productivity by both its relatively lower payload capability and its relatively higher empty weight compared to a helicopter. As a consequence, the *V-22* is again comparable with the average helicopter and it is outperformed by the classical fixed wing aircrafts.

Aside to the specific productivity ratio, it is possible to define another parameter, exploiting another metric and so taking into account the fuel consumption: the *range specific transport efficiency*. This quantity is defined as the ratio of the payload weight transported to the fuel weight consumed for a specific transport range, *i.e.*

$$\text{range specific transport efficiency} = \frac{\text{payload weight}}{\text{fuel weight}}.$$

Since the range specific transport efficiency involves the fuel burned to accomplish the mission, it is a better measure of unit payload transported per unit of resource expended than the specific productivity. The values assumed by this parameter for a variety of aerial vehicles are reported in figure 11. This picture shows that for a helicopter, the higher is the range, the lower is the efficiency (*i.e.* at longer flight ranges the helicopter becomes a much less productive aircraft concept). This behaviour is due to the poor aerodynamic efficiency of the helicopter (less than a third of the cruise efficiency of an airplane) and the need for it to carry a great amount of fuel to reach longer ranges. Also of significance is that current tiltrotors do not exceed the capabilities of the helicopter for the typical transportation missions. In fact both the specific productivity and the transport efficiency of the *V-22 Osprey* are no better than those of a helicopter.

In conclusion, from the analysis performed in this section it appears clear that the tiltrotor does not represent the new incarnation of the helicopter, neither it is the future of the cargo transportation. In particular it has been pointed out that even though this hybrid concept has

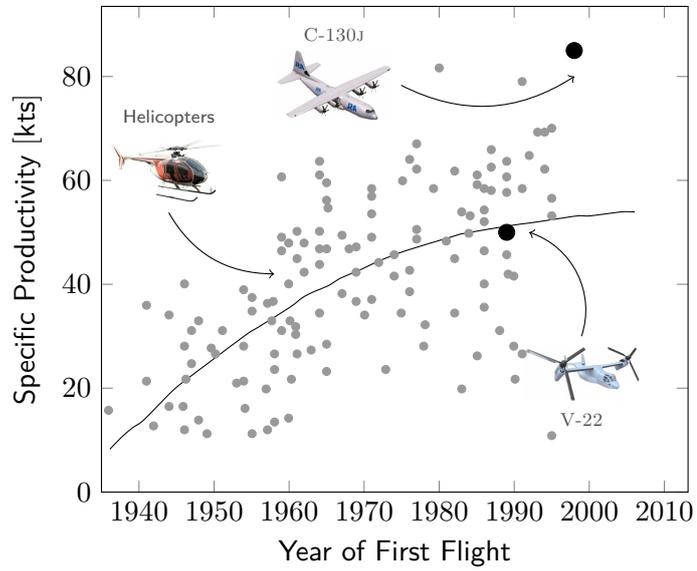


Figure 10: Specific Productivity [Leishman, 2007]

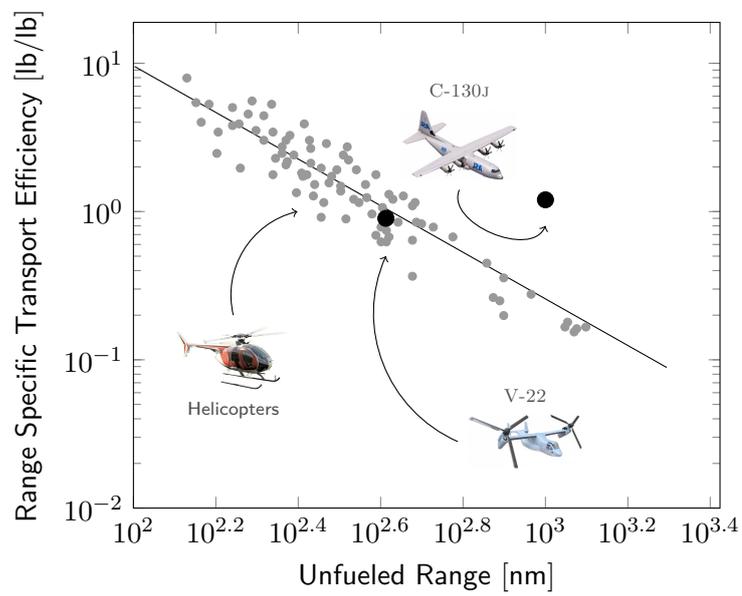


Figure 11: Unfueled Range [Leishman, 2007]

poor performance as payload carrier, especially in *high and hot* conditions, it has its niche in both low and medium range transportation at relatively high cruise speed. Moreover, this peculiarity could be exploited to move regional transportation air traffic off of the main runways, resulting in the possibility to employ those runways for use by greater numbers of larger and longer-range aircraft. From this perspective the future of the tiltrotor seems to be, at least potentially, related to that of the regional civil aviation, as witnessed by the LCTR project. Therefore in this scenario helicopters, airplanes and tiltrotors live side by side.

Part I
Theory



Overview

“There is nothing so practical as a good theory.”

— *K. Lewin, 1951*

This first part reports all the theoretical basis needed for a full understanding of the methods applied in the rest of the thesis. It should be pointed out that the wideness of the arguments here dealt with is so high that just the key concepts are developed in the following pages, whereas for a deeper insight the author suggests to the interested reader to refer to the bibliography at the end of this volume.

In particular, the main three matters of interest are the classic potential flow theory, the solution of the potential problem through the panel method and the propeller theory. For this reason the current part is composed of three different chapters:

- even though they are beyond the scope of this thesis, in section 1.1, chapter 1 introduces the Navier-Stokes equations, whose simplified forms are presented in sections 1.2 and 1.4, where the Euler's equations and the potential flow problem are respectively dealt. Moreover, the general solution to this latter issue is described in section 1.4.6, whereas its classical basic solutions – whose importance is related to the linearity of the Laplace's equation – are listed in section 1.4.7.
- The numerical solution of the potential problem by means of the panel methods is described in chapter 2. In sections 2.4 and 2.5, particular attention has been paid to the numerical procedure, for both steady and unsteady situations. Finally, the vortex element and, more in general, the Vortex Particle Methods are introduced in section 2.7;
- in chapter 3 an overview of the theory lying on the basis of the propellers analysis has been briefly introduced, with the aim to make the reader comfortable with the notation applied in the subsequent chapters. In particular, the general definitions related to the propellers and the original blade element theory are dealt with in section 3.1. Finally, the momentum theory and the Blade Element Momentum Theory have been introduced in sections 3.2.1 and 3.2.2, respectively.

1 Fluid-dynamics Equations

“ Science is a differential equation.
Religion is a boundary condition. ”

— A. Turing

The goal of this chapter is to define the mathematical problem – in terms of both fluid-dynamics differential equations and Boundary Conditions (BCs) – of low-speed aerodynamics, whose solution will be dealt with in chapter 2. In particular, even though the reader's attention will be focused on potential flows, a quick recall of the general Navier-Stokes (N-S) equations appears to be dutiful to the author [for a full reference see Monti and Savino, 2003].

1.1 Navier-Stokes Equations

The motion of a viscous, compressible, unsteady and heat-conductive fluid is described by the well known Navier-Stokes (N-S) equations. In particular, for a single phase fluid, these nonlinear Partial Differential Equations (PDEs) form a coupled system composed of three main equations stating the balance of mass, momentum and energy. The first N-S equation to be introduced is the mass balance, often referred to as the *continuity equation*. This equation can be written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho V_i)}{\partial x_i} = 0, \quad (1.1)$$

where $\rho(\mathbf{x}, t)$ is the density field. It should be noted that for incompressible flows (*i.e.* $\rho = \text{const}$) the equation (1.1) states that the velocity vector has zero divergence (*i.e.* it is *solenoidal*)

$$\nabla \cdot \mathbf{V} = \frac{\partial V_i}{\partial x_i} = 0. \quad (1.2)$$

The second N-S equation concerns the balance of momentum. It can be written as

$$\frac{\partial (\rho V_j)}{\partial t} + \frac{\partial (\rho V_i V_j)}{\partial x_i} = -\frac{\partial p}{\partial x_j} + \frac{\partial \tau_{dji}}{\partial x_i} + \rho f_{bj}, \quad (1.3)$$

where $p(\mathbf{x}, t)$ is the pressure, $\mathbf{f}_b(\mathbf{x}, t)$ is an external body force (for instance due to gravity or electromagnetic actions) per unit mass exerted on the fluid and $\underline{\tau}_d$ is the deviatoric portion of the shear stress tensor. In fact, if \underline{U} is the unit tensor, it is $\underline{\tau} = -p\underline{U} + \underline{\tau}_d$ and

$$\underline{\tau}_d = \mu_v (\nabla \cdot \mathbf{V}) \underline{U} + 2\mu (\nabla \mathbf{V})_0^s,$$

being μ_v and μ the bulk and dynamic viscosity coefficient, respectively.

The last equation of interest in the scope of this thesis is the energy balance, obtained by the equation (1.3) dot multiplied by \mathbf{V} . This step yields the

$$\begin{aligned} \frac{\partial}{\partial t} \left(\rho e + \frac{1}{2} \rho V^2 + p \right) + \frac{\partial}{\partial x_i} \left[\left(\rho e + \frac{1}{2} \rho V^2 + p \right) V_i \right] = \\ = \frac{\partial p}{\partial t} + \frac{\partial}{\partial x_i} \left(\tau_{im} V_m + \kappa \frac{\partial T}{\partial x_i} \right) + \rho f_i V_i, \end{aligned} \quad (1.4)$$

where $e(\mathbf{x}, t)$ is the internal energy of the fluid, κ is the *Fourier's* coefficient of heat conductivity and $T(\mathbf{x}, t)$ is the temperature.

In addition to the equations (1.1), (1.3) and (1.4), a fluid-dependent relation, whose general expression is on the form

$$f(\rho, p, T) = 0, \quad (1.5)$$

should be taken into account. For instance, for a perfect gas the equation (1.5) is represented by the well known relation

$$p = \rho R T. \quad (1.6)$$

1.2 Euler's Equations

For inviscid, thermally non conductive and chemically inert flows, the N-S system reduces to the so called Euler equations

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho V_i)}{\partial x_i} = 0; \quad (1.7)$$

$$\frac{\partial (\rho V_j)}{\partial t} + \frac{\partial (\rho V_i V_j)}{\partial x_i} + \frac{\partial p}{\partial x_j} = \rho f_{bj}; \quad (1.8)$$

$$\frac{\partial}{\partial t} \left(\rho e + \frac{1}{2} \rho V^2 \right) + \frac{\partial}{\partial x_i} \left[\left(\rho e + \frac{1}{2} \rho V^2 + p \right) V_i \right] = 0. \quad (1.9)$$

In practice, the adopted simplifying assumptions imply that, in equations (1.3) and (1.4), both the deviatoric stress tensor $\underline{\tau}_d$ and the Fourier's transport coefficient κ are equal to zero: this approximation is as much verified as greater is the Reynolds number Re , so that ideally the equations (1.7) to (1.9) hold in the limit $Re \rightarrow \infty$.

The fluid-dynamic model based on the Euler's equations includes the possibility to successfully simulate both rotational flows and flow fields characterized by the presence of a discontinuity surface (e.g. a shock wave or a vortex sheet). These properties make the Euler's equations suitable to model a wide range of engineering flows, such as a turbofan exhaust or the interaction between the wake produced by a propeller and a wing.

1.3 Vorticity Equation

An alternative formulation of the incompressible form of the Euler's momentum equation (1.8) can be obtained in terms of the vorticity vector

$$\boldsymbol{\omega} = \nabla \times \mathbf{V} ,$$

namely,

$$\frac{D\boldsymbol{\omega}}{Dt} = \frac{\partial \boldsymbol{\omega}}{\partial t} + \mathbf{V} \cdot \nabla \boldsymbol{\omega} = \boldsymbol{\omega} \cdot \nabla \mathbf{V} . \quad (1.10)$$

For motions in two dimensions $\boldsymbol{\omega} = \omega \mathbf{k}$, where \mathbf{k} is the unit vector normal to the 2D plane, hence the Right Hand Side (RHS) of equation (1.10) is zero and thus the vorticity is conserved on the trajectories of \mathbf{V} and remains bounded for all time. On the other hand, three-dimensional flows are, in general, characterized by a nonzero value for $\boldsymbol{\omega} \cdot \nabla \mathbf{V}$, so that the vorticity may be generated by means of the so called *vortex tube stretching* mechanism. For this reason the last term of equation (1.10) is named *vortex stretching* and represents how the strength and magnitude of the vorticity change as they are exposed to velocity gradients in the fluid field.

The vorticity evolution equation may be viewed as a differential form of the Helmholtz and Kelvin's vortex theorems [Tabak, 2002, web]. To better understand this point, consider the time evolution of a vortex tube: according to Helmholtz's theorem the tube itself moves with the fluid and, from Kelvin's theorem, its strength does not change with time. Therefore, vorticity should amplify proportionally to the area of a cross-section of the tube. Nevertheless, being the flow incompressible by assumption, the volume between any two sections of the tube has to remain constant, so that any shrinking of the cross-sectional area must be accompanied by a longitudinal stretching.

Furthermore, it should be pointed out that for inviscid flows (*i.e.* in the scope of applicability of equation (1.10)) the vorticity that is created by the body is convected along with the flow at a rate infinitely greater than the one associated to viscous diffusion across the flow. In fact the viscous form of the vorticity equation includes a term proportional to the reciprocal of the Reynolds number. The ultimate implication of this fact is that the vorticity is confined in the thin region occupied by the wake.

1.4 Potential Flows

For low-speed aerodynamic problems the presence of the *boundary layer* allows to efficaciously consider the flow external to it as incompressible, inviscid and irrotational, thus leading to a deep simplification of the equations lying behind the problem.

1.4.1 The Helmholtz Decomposition

Before introducing the *potential flows* it is favourable to spend some words about the so called *Helmholtz decomposition*, also referred to as the *Helmholtz theorem* [von Helmholtz, 1858]. This fundamental result, obtained by *Hermann Ludwig Ferdinand von Helmholtz* in 1958, is based on the idea that the motion of a volume element of a continuous fluid media in \mathbb{R}^3 consists of:

1. expansion or contraction in three orthogonal directions,
2. rotation about an instantaneous axis,
3. translation.

In particular, due to its irrotationality, the portion relative either to an expansion or a contraction can be represented as the gradient of a scalar potential function. Similarly, the rotation can be represented as the curl of a vector potential function since it is incompressible. However, on the other hand, the translation is both incompressible and irrotational, so that it can be indeed represented as either the gradient of a scalar potential, or the curl of a vector potential. A third option is to employ a representation based on a separate harmonic component. Therefore, as a consequence, according to Helmholtz any vector field defined into \mathbb{R}^3 and subjected to suitable asymptotic conditions at infinity can be written as the sum of two different parts: the first one represents the rotational degree of freedom and hence it is incompressible and can be expressed as the curl of a vector potential function; the second part is related to both translation and compression/expansion and can be therefore written as the gradient of a scalar potential function. Since these scalar and vector potentials can be computed from the divergence and the curl of the vector field, a vector field is uniquely defined whenever both its divergence and curl are known. This dissertation can be formalized as follows [Bhatia et al., 2013]:

Theorem (Helmholtz decomposition) *The motion of a fluid $\mathbf{V}(\mathbf{x})$ in an infinite space ($\mathbf{x} \in \mathbb{R}^3$) such that it vanishes at infinity is determinate when we know the values of $\theta(\mathbf{x})$ and $\omega(\mathbf{x})$, where*

$$\begin{aligned}\theta(\mathbf{x}) &= \nabla \cdot \mathbf{V}(\mathbf{x}) && \text{(Divergence) ,} \\ \omega(\mathbf{x}) &= \nabla \times \mathbf{V}(\mathbf{x}) && \text{(Curl) .}\end{aligned}$$

On the other hand, if the motion of the fluid is limited to a simply connected region $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega$, it is determinate if $\theta(\mathbf{x})$ and $\omega(\mathbf{x})$ and the value of the flow normal to the boundary, $V_n(\mathbf{x}) = \mathbf{V}(\mathbf{x}) \cdot \mathbf{n}$ for $\mathbf{x} \in \partial\Omega$, are known.

This formulation of the original Helmholtz idea admits an inverse problem, consisting in a decomposition of a vector field into its divergence-free (incompressible) and curl-free (irrotational) components. In the dedicated literature this inverse problem is also known as the Helmholtz decomposition. This problem can be formally exposed as follows [Bhatia et al., 2013]:

Theorem (Helmholtz decomposition – Inverse) *Every smooth field $\mathbf{V} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ defined in a simply connected region with a single boundary surface face can be expressed as a sum of the gradient of a scalar potential and the curl of a vector potential, i.e.*

$$\mathbf{V} = \nabla\phi + \nabla \times \boldsymbol{\psi} , \quad (1.11)$$

where the scalar potential ϕ and the vector one $\boldsymbol{\psi}$ are evaluated from θ and $\boldsymbol{\omega}$, respectively, as

$$\phi(\mathbf{x}) = -\frac{1}{4\pi} \int \frac{\theta(\mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|} d\mathbf{x}' , \quad (1.12)$$

$$\boldsymbol{\psi}(\mathbf{x}) = \frac{1}{4\pi} \int \frac{\boldsymbol{\omega}(\mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|} d\mathbf{x}' , \quad (1.13)$$

where $\frac{1}{4\pi|\mathbf{x} - \mathbf{x}'|}$ is the Green's function.

In particular, by definition, $\nabla\phi$ is curl-free (i.e. $\nabla \times (\nabla\phi) = 0$) and $\nabla \times \boldsymbol{\psi}$ is divergence-free (i.e. $\nabla \cdot (\nabla \times \boldsymbol{\psi}) = 0$). Moreover, it should be noted that the vanishing condition at infinity is required in order to have converging integrals in equations (1.12) and (1.13) and that the potentials ϕ and $\boldsymbol{\psi}$ are unique up to a constant, meaning that the decomposition is unique [Blumenthal, 1905].

1.4.2 Laplace and Poisson's equations

It is known from the vector calculus [see Jeffrey and Zwillinger, 2007] that the curl of the gradient of any scalar field Φ is always the zero vector and, similarly, that it is equal to zero the divergence of the curl of any vector field \mathbf{F} , too. In formulae

$$\nabla \times (\nabla\Phi) = 0 , \quad (1.14)$$

$$\nabla \cdot (\nabla \times \mathbf{F}) = 0 . \quad (1.15)$$

Moreover, standing the vectorial identity

$$\nabla \times (\nabla \times \mathbf{F}) = \nabla (\nabla \cdot \mathbf{F}) - \nabla^2 \mathbf{F} ,$$

it follows that if both the vector potential $\boldsymbol{\psi}$ and the field \mathbf{V} are divergence-free, then equation (1.11) yields

$$\nabla^2 \phi = 0 , \quad (1.16)$$

$$-\nabla^2 \boldsymbol{\psi} = \nabla \times \mathbf{V} , \quad (1.17)$$

which, each in order, are known as the *Laplace* and *Poisson's* equations, whose solutions are expressed by equations (1.12) and (1.13), respectively.

Consider the following line integral in a simply connected region of \mathbb{R}^3 , along the line C

$$\int_C \mathbf{V} \cdot d\ell = \int_C u dx + v dy + w dz .$$

If the flow is irrotational in this region, then $u dx + v dy + w dz$ is an exact differential of a potential ϕ that is independent of the integration path C and is a function of the location of the point $P(x, y, z)$ [Kreyzig, 2010]. Whether P_0 is an arbitrary reference point, that is to say

$$\phi(x, y, z) = \int_{P_0}^P u dx + v dy + w dz ,$$

where ϕ is called the *velocity potential* and it is defined so that

$$\mathbf{V} = \nabla\phi , \tag{1.18}$$

which, in cartesian coordinates, can be written as

$$u = \frac{\partial\phi}{\partial x} \quad v = \frac{\partial\phi}{\partial y} \quad w = \frac{\partial\phi}{\partial z} .$$

In other words, the irrotationality is a necessary and sufficient condition for the existence of the velocity potential, so that it allows to employ the Helmholtz decomposition to the velocity vector field \mathbf{V} , obtaining a result formally analogue to the one stated by equation (1.11). In this case the vector potential for the flow field ψ is the *stream function*, which is related to the mass conservation by the fact that the vectorial identity reported in equation (1.15) implies the self-satisfaction of the continuity equation (1.2) for the divergence-free portion of the velocity field (*i.e.* $\nabla \cdot \mathbf{V}_\psi = \nabla \cdot (\nabla \times \psi) = 0$). From the physical point of view, this result suggests that the *iso- ψ* lines are coincident with the streamlines and that the difference between two streamlines represents the volumetric flow rate between them. Furthermore, it should be noted that whereas the stream function can be defined for both rotational and irrotational flows, the velocity potential holds only in the latter case, when the three components of the main vectorial unknown can be substituted by a single scalar.

On the other hand, substituting the whole expression (1.11) of the velocity into the incompressible continuity equation (1.2) written for an irrotational fluid, one obtains the Laplace equation (1.16), in fact

$$\nabla \cdot \mathbf{V} = \nabla \cdot (\nabla\phi + \nabla \times \psi) = \nabla^2\phi = 0 .$$

This is a linear elliptic differential equation which results in a boundary-value problem, whose solutions are *harmonic functions*. Therefore, the superposition of the effects can be employed and the velocity field can be obtained from the scalar potential ϕ : no relation with the pressure has been employed so far, since the Euler's momentum equation has not been introduced yet in this context.

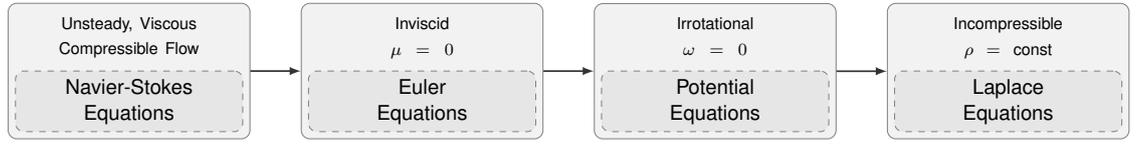


Figure 1.1: Scheme summarizing the main fluid-dynamics models

Finally, in order to help the reader to fix in mind the hypothesis chain which lead to the Laplace equation starting from the Navier-Stokes system , a summarizing scheme is reported in figure 1.1.

1.4.3 Unsteady Bernoulli equation

The relation between the velocity field and the pressure distribution (required to compute the aerodynamic forces acting on a body) is provided by the Euler momentum equation (1.8), which, under the hypothesis of incompressible flow, becomes

$$\frac{D\mathbf{V}}{Dt} = \frac{\partial \mathbf{V}}{\partial t} + \mathbf{V} \cdot \nabla \mathbf{V} = \mathbf{f}_b - \frac{\nabla p}{\rho}.$$

Moreover, standing the vectorial identity [see Jeffrey and Zwillinger, 2007]

$$\nabla \left(\frac{V^2}{2} \right) = \mathbf{V} \cdot \nabla \mathbf{V} + \mathbf{V} \times (\nabla \times \mathbf{V}) = \mathbf{V} \cdot \nabla \mathbf{V} + \mathbf{V} \times \boldsymbol{\omega},$$

it follows that

$$\frac{D\mathbf{V}}{Dt} = \frac{\partial \mathbf{V}}{\partial t} + \nabla \left(\frac{V^2}{2} \right) - \mathbf{V} \times \boldsymbol{\omega} = \mathbf{f}_b - \frac{\nabla p}{\rho},$$

thus, for an irrotational flow (*i.e.* $\boldsymbol{\omega} = 0$)

$$\frac{\partial \mathbf{V}}{\partial t} = \frac{\partial}{\partial t} \nabla \phi = \nabla \left(\frac{\partial \phi}{\partial t} \right) = \mathbf{f}_b - \nabla \left(\frac{V^2}{2} \right) - \frac{\nabla p}{\rho}. \quad (1.19)$$

If, furthermore, the body force \mathbf{f}_b is conservative with a potential E – *i.e.* if it does exist a scalar field such that $\mathbf{f}_b = -\nabla E$ as in the case of the gravitational force (for which $E = -gz$) – then equation (1.19) yields

$$\nabla \left(\frac{\partial \phi}{\partial t} + \frac{V^2}{2} + \frac{p}{\rho} + E \right) = 0, \quad (1.20)$$

that is to say to the unsteady Bernoulli equation for inviscid, incompressible and irrotational flows

$$\frac{\partial \phi}{\partial t} + \frac{V^2}{2} + \frac{p}{\rho} + E = C(t),$$

where $C(t)$ is a time dependent integration variable. A more useful form of this famous equation can be obtained evaluating it in two different points of the flow field at the same time. In particular, it is convenient to chose one of these two points as located at the infinity, and characterized by the following reference conditions: $E_\infty = 0$, $\phi_\infty = \text{const}$ and $V_\infty = 0$. Under these assumptions the pressure p at any point of the flow field and at any time, can be evaluated by means of the

$$\frac{p - p_\infty}{\rho} = \frac{\partial \phi}{\partial t} + \frac{V^2}{2} + E. \quad (1.21)$$

It should be noticed that for steady, incompressible and rotational fluids equation (1.20) holds (obviously with the time derivative set to zero), too, if the integration parameter $C(t)$ is allowed to vary from a streamline to another one. In fact, the cross product $\mathbf{V} \times \boldsymbol{\omega}$ is locally normal to the streamline $d\ell$, so that their dot product vanishes along the streamline itself. As a consequence equation (1.20) can be employed for steady rotational flows only if applied along one streamline.

1.4.4 Boundary Conditions

The problem under analysis, composed of the Laplace and Poisson's equations (1.16) and (1.17), as stated in section 1.4.2, results to be unclosed, meaning that the solution is neither unique nor physically consistent with the particular geometric and free-stream conditions. This is due to the fact that a differential problem requires an appropriate number of *Boundary Conditions (BCs)* to be in closed form.

These BCs provide information about the particular geometry of the body, the flow conditions at an ideally non-finite distance away from the body itself and, for lifting bodies, the smoothness of the solution at the Trailing Edge (TE) of the wing (*i.e.* the *Kutta condition*). Moreover, for unsteady situation – which involve, for example, local time variation of the velocity vector, deformations of the body or time evolution of the wake geometry – the intrinsic dependence of the BC itself upon time has to be taken into account.

Free-stream

Since the domain boundaries are located far away from the body, it could be assumed that here there is no disturbance on the flow field due to the presence of the body itself. In other words on the far-field boundaries the velocity is everywhere coincident with its undisturbed value \mathbf{V}_∞ , *i.e.*

$$\lim_{|x| \rightarrow \infty} \mathbf{V} = \mathbf{V}_\infty. \quad (1.22)$$

Wall

For the general viscous case – formalized by means of the N-S equations (1.1), (1.3) and (1.4) – the closure condition imposed at the body wall is typically characterized by the so called *no-slip condition*, which imposes the relative velocity between the wall and the nearby fluid to be zero due to the action of the viscosity μ . However the problem of interest is inviscid, so that the no-slip condition cannot be employed. On the contrary of viscous flows, in this case just one wall

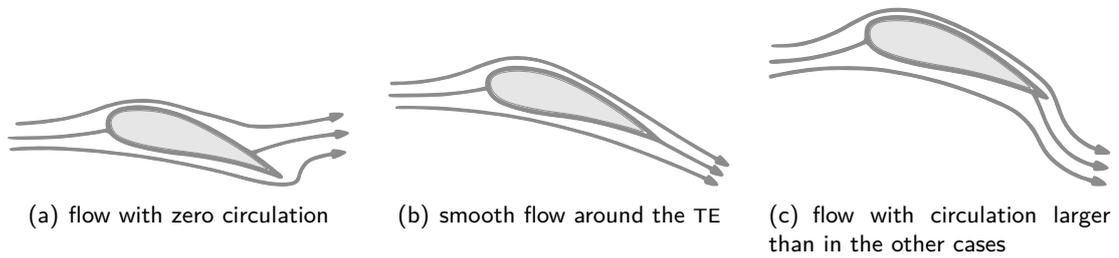


Figure 1.2: Possible solutions for the flow over an airfoil [Katz and Plotkin, 1991]

condition should be formulated, so that the component V_n of the velocity locally normal to the body surface is prescribed. In general one could deal with two different situations:

- the wall is non-porous, so that the flow is – point by point – tangent to the solid surface;
- the wall is porous and transpiration phenomena do occur, meaning that V_n is not zero and it is equal to the transpiration velocity.

Therefore, defining $\mathbf{n}(\mathbf{x}, t)$ the unit vector locally normal to the surface in the point $P(\mathbf{x})$ at the time t and V_{tr} the transpiration velocity, in general the wall BC can be written as

$$V_n = \mathbf{V} \cdot \mathbf{n} = V_{tr}, \quad (1.23)$$

where $V_{tr} = 0$ for non-porous surfaces.

Kutta condition

Imposing both the wall boundary condition and the free stream one results in a problem whose solution is not unique since an arbitrary value can be assumed for the circulation Γ . This phenomenon has been depicted in figure 1.2, where three different possible solutions to the same problem concerning a two-dimensional flow around an airfoil are shown: in case (a) the circulation may be equal to zero, in case (b) it is such that the flow at the TE is parallel to its bisector line and, finally, in case (c) the circulation is larger and the flow turns downward near the TE (as it can be achieved, for instance, by means of blowing).

In the early 1900s *Martin Wilhelm Kutta* showed [Kutta, 1902] that the physical solution is the one represented in figure 1.2(b). Thus the Kutta condition states that

the flow leaves the sharp Trailing Edge of an airfoil smoothly and the velocity there is finite.

Moreover, since the TE angle is finite, in order to satisfy the Kutta condition, the normal component of the velocity, from both sides of the airfoil, has to vanish. For a continuous velocity this is possible only if the TE is a stagnation point. Hence this condition can be expressed through the

$$\Delta p_{TE} = 0 . \quad (1.24)$$

Obviously, for unsteady flows a time dependent Kutta condition has to be employed. In this case particular attention should be paid to the wing-wake relationship: in fact, holding the *Helmholtz vortex theorem* (stating that the time rate of change of circulation around a closed curve consisting of the same fluid elements is zero [Katz and Plotkin, 1991]), any increasing in bound vorticity on the wing must be balanced by an equivalent increase in vorticity in the wake. This is possible only if the rate of change of wake vorticity is equal to the one related to the wing and if its rotation is in the opposite direction: *i.e.*

$$\frac{d}{dt} \Gamma_{span} \Big|_{wing} = - \frac{d}{dt} \Gamma_{span} \Big|_{wake} , \quad (1.25)$$

where Γ denotes the circulation.

1.4.5 Statement of the potential flow problem

Summarizing what have already been said in sections 1.4.2 and 1.4.4, in this paragraph a formalization of the potential flow problem is presented.

Considering an incompressible and irrotational flow, the continuity equation reduces to

$$\nabla^2 \phi = 0 ,$$

and the velocity component normal to every solid non-porous surface should be zero, so that in a body fixed reference frame

$$\nabla \phi \cdot \mathbf{n} = 0 .$$

Also, far from the body the disturbances due to its presence into the flow-field have to vanish, so that

$$\lim_{|\mathbf{x}| \rightarrow \infty} \nabla \phi = \mathbf{V}_\infty .$$

1.4.6 General solution

Typically, in aerospace engineering applications, the fluid volume \mathcal{V} of interest contains a solid body (whose boundaries are denoted by S_b) with additional far-field boundaries (S_∞) that may define an outer flow problem such as the study of the aerodynamic forces acting on a wing. This general situation is depicted in figure 1.3, where the notation employed hereafter is pointed out, too. It should be clear that the BCs stated into equations (1.22) and (1.23) apply to S_b and S_∞ , respectively. Moreover, the local normal \mathbf{n} to the boundaries of the volume \mathcal{V} is defined so that it always points outwards with respect to the region of interest (*e.g.* the one included between S_∞ and S_b).

The sought general solution to the potential flow problem formulated in section 1.4.5 is based onto the *Green's identities* [for a complete reference see Kellogg, 2010]. In particular, if Φ_1 and

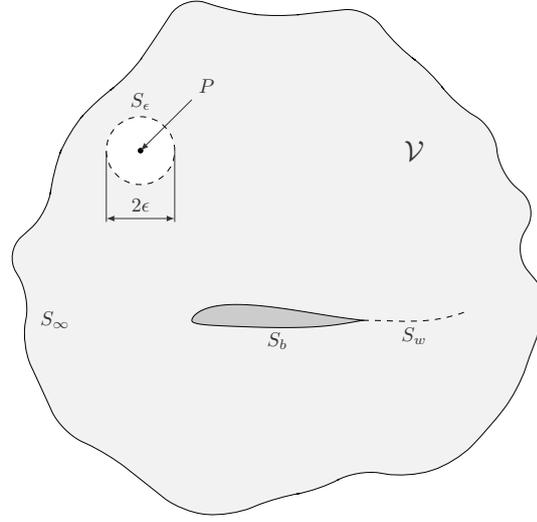


Figure 1.3: Fluid domain and nomenclature

Φ_2 are two scalar functions defined in \mathcal{V} and here continuously differentiable with continuous partial derivatives of second order, then the *Green's second identity* can be written as

$$\iint_S (\Phi_1 \nabla \Phi_2 - \Phi_2 \nabla \Phi_1) \cdot \mathbf{n} \, dS = \iiint_{\mathcal{V}} (\Phi_1 \nabla^2 \Phi_2 - \Phi_2 \nabla^2 \Phi_1) \, d\mathcal{V}, \quad (1.26)$$

where the surface integral is taken over all the boundaries S , including a wake surface S_w across which a discontinuity in either the velocity potential or the velocity itself may occur. In formula

$$S = S_\infty \cup S_b \cup S_w.$$

Whether in equation (1.26) one considers both Φ_1 and Φ_2 to be *harmonic functions* (i.e. they satisfy the Laplace's equation (1.16)) such that $\Phi_1 = 1/r$ and $\Phi_2 = \phi$ – where $r = |\mathbf{r} = (x, y, z)|$ is the distance of a point P and ϕ is the potential of the flow of interest in \mathcal{V} – when $P \notin \mathcal{V}$ the equation (1.26) itself becomes

$$\iint_S \left(\frac{1}{r} \nabla \phi - \phi \nabla \frac{1}{r} \right) \cdot \mathbf{n} \, dS = 0, \quad P \notin \mathcal{V}.$$

Nevertheless, of particular interest is the case when $P \in \mathcal{V}$. The point P can be tush considered as enveloped in a small sphere of radius ϵ , $\mathcal{S}(P, r)$, as shown in figure 1.3. From the harmonic behaviour of both Φ_1 and Φ_2 in $\mathcal{V} \setminus \mathcal{S}(P, r)$, equation (1.26) yields

$$\phi(P) = \frac{1}{4\pi} \iint_S \left(\frac{1}{r} \nabla \phi - \phi \nabla \frac{1}{r} \right) \cdot \mathbf{n} \, dS = 0, \quad P \in \mathcal{V}, \quad (1.27)$$

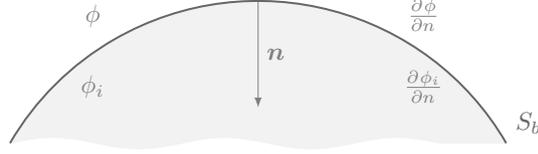


Figure 1.4: The velocity potential near a solid boundary

which is known as *Green's third identity*. This result gives the value of the potential ϕ in any point P belonging to the fluid domain \mathcal{V} in terms of the values assumed by ϕ itself and by $\nabla\phi \cdot \mathbf{n} = \frac{\partial\phi}{\partial n}$ on the boundaries S .

In general, an *internal potential* ϕ_i can be defined, meaning that the flow is located inside the boundaries S_b , so that the point $P \in \mathcal{V}$ is exterior to the region of interest. Furthermore, assuming the wake surface S_w to be thin, so that the quantity $\frac{\partial\phi}{\partial n}$ is continuous across it, and defining the far-field potential ϕ_∞ as

$$\phi_\infty(P) = \frac{1}{4\pi} \iint_{S_\infty} \left(\frac{1}{r} \nabla\phi - \phi \nabla \frac{1}{r} \right) \cdot \mathbf{n} \, dS = 0 ,$$

the equation (1.27) can be modified in order to include also the effects of the internal potential, obtaining

$$\phi(P) = \frac{1}{4\pi} \iint_{S_b} \left[\frac{1}{r} \nabla(\phi - \phi_i) - (\phi - \phi_i) \nabla \frac{1}{r} \right] \cdot \mathbf{n} \, dS - \frac{1}{4\pi} \iint_{S_w} \left(\phi \mathbf{n} \cdot \nabla \frac{1}{r} \right) \, dS + \phi_\infty(P) . \quad (1.28)$$

Consider now a segment of the boundaries S_b , as shown in figure 1.4. Defining the differences between the internal and external potentials and between their normal derivatives respectively as

$$-\mu = \phi - \phi_i , \quad (1.29)$$

$$-\sigma = \frac{\partial\phi}{\partial n} - \frac{\partial\phi_i}{\partial n} , \quad (1.30)$$

equation (1.28) becomes

$$\phi(P) = -\frac{1}{4\pi} \iint_{S_b} \left[\sigma \frac{1}{r} - \mu \frac{\partial}{\partial n} \left(\frac{1}{r} \right) \right] \, dS + \frac{1}{4\pi} \iint_{S_w} \left[\mu \frac{\partial}{\partial n} \left(\frac{1}{r} \right) \right] \, dS + \phi_\infty(P) , \quad (1.31)$$

where, each in order, μ and σ are called the *doublet* and the *source* singular solutions of the Laplace's equation since they are both singular as r approaches zero. On the other hand, in the limit $r \rightarrow \infty$ these quantities vanish, so that the free-stream BC (1.22) is automatically fulfilled.

According to equation (1.31), the flow-field may be modelled by means of a singularity superposition, so that, in order to determine the velocity potential ϕ in the region \mathcal{V} , the strength of

the distribution of sources and doublets on the boundaries has to be evaluated. Moreover, it does not exist a unique possible combination of source and doublet for a particular problem, therefore it is possible, for instance, to set

$$\frac{\partial \phi_i}{\partial n} = \frac{\partial \phi}{\partial n}, \quad \forall P \in \mathcal{V}$$

leading to a pure doublets distribution (according to equation (1.30)). On the other hand, equation (1.29) states that if

$$\phi_i = \phi, \quad \forall P \in \mathcal{V}$$

than the resulting singularity distribution is made of sources only.

1.4.7 Basic solutions

The linearity of the Laplace's equation allows to employ the superposition of the effects, so that, as stated by equation (1.31), the flow-field can be thought of as it is generated by several basic solutions distributed along the domain boundaries, whose strength is obtained enforcing the BCs. For this reason it is favourable to analyze the main basic solutions which characterize the problem under examination: the *free stream*, the *point source* (with its dual entity, the *point sink*), the *point doublet* and the *vortex*. Hence the following paragraphs are dedicated to an insight view of these singularities.

Free stream

The simplest basic solution to the Laplace's equation is the free stream, so called because it represents a uniform flow-field, expressed by the formula

$$\mathbf{V}_\infty = \left(\frac{\partial \phi}{\partial x}, \frac{\partial \phi}{\partial y}, \frac{\partial \phi}{\partial z} \right) = (u_\infty, v_\infty, w_\infty)$$

and depicted in figure 1.5, where both the iso- ψ lines (*i.e.* the streamlines) and the equipotential ones (*i.e.* iso- ϕ) are reported. In particular, the potential ϕ related to this basic solution is

$$\phi_\infty(x, y, z) = u_\infty x + v_\infty y + w_\infty z = V_{\infty i} x_i. \quad (1.32)$$

The point source/sink

The *source* (*sink*) has already been presented into equations (1.30) and (1.31). This element represents a concentric motion directed away (towards) the point where the singularity itself is located and, if placed at the origin of a spherical reference frame, it is characterized by the potential

$$\phi(P) = -\frac{\sigma}{4\pi r}, \quad (1.33)$$

where r is the distance of a point P from the origin. The velocity induced by this element in the surrounding space can be found through the equation (1.18), where the ∇ operator has to be

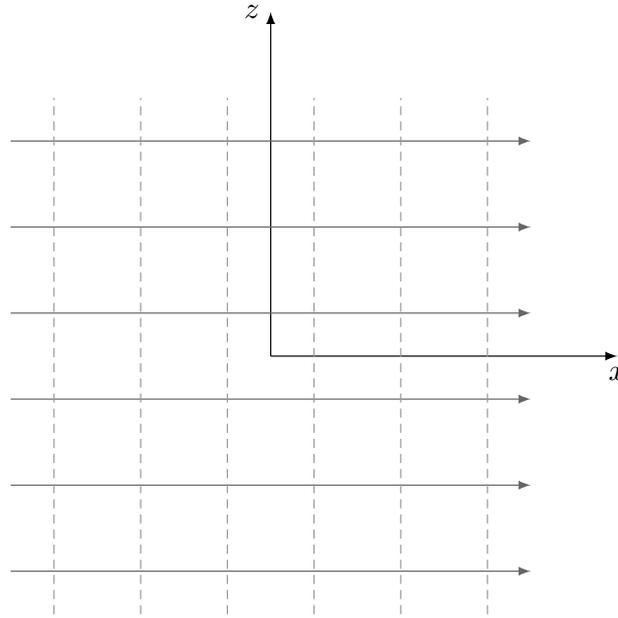


Figure 1.5: Flow-field induced by a free stream

evaluated in spherical coordinates [see Jeffrey and Zwillinger, 2007]. As shown in figure 1.6, the solution obtained in this way is a vector field with just the radial component, *i.e.*

$$\mathbf{V} = (V_r, V_\theta, V_\phi) = \left(\frac{\partial \phi}{\partial r}, 0, 0 \right) = \left(\frac{\sigma}{4\pi r^2}, 0, 0 \right).$$

So the velocity decays as $1/r^2$ and it is singular in $r = 0$, as it has already been said in section 1.4.6.

Considering the origin-located source of figure 1.6(a), it can be observed that σ represents the volumetric flow rate through a spherical surface whose radius is r . Therefore, the positive σ is the volumetric rate at which the fluid is introduced at the source, whereas the negative σ is the rate at which the fluid is flowing towards the sink. Moreover, the basic point solution here shown can be integrated over a line, a surface or a volume in order to create more complex singularity elements. In these cases σ is the source strength per unit length, surface or volume, respectively.

The point doublet

Aside to the source, equation (1.31) contained also terms related to the singularity usually referred to as *doublet*, whose potential is expressed by

$$\phi = \frac{\mu}{4\pi} \mathbf{n} \cdot \nabla \frac{1}{r}. \quad (1.34)$$

It should be noticed that, being for elements of unit strength

$$\phi_\mu = -\frac{\partial \phi_\sigma}{\partial n},$$

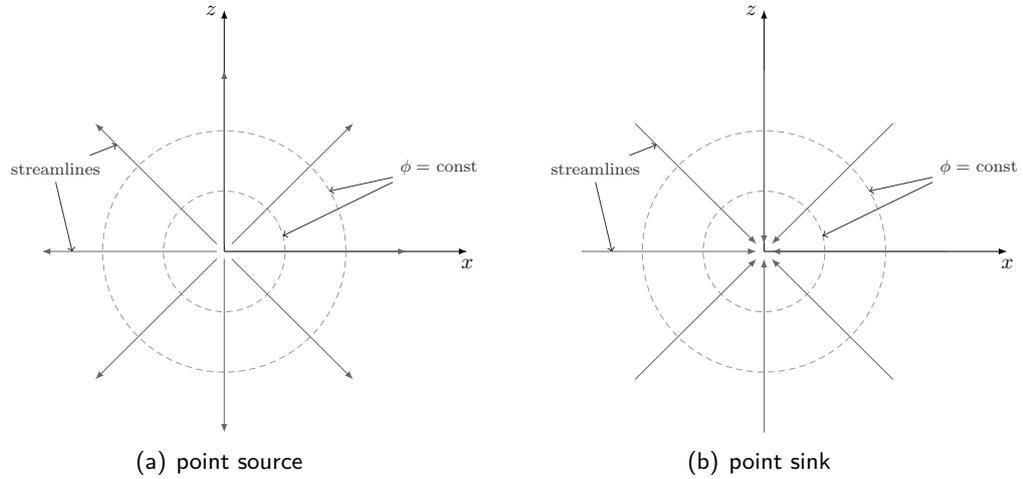


Figure 1.6: Flow-field induced by a point source/sink

a relationship between the doublet and the source does exist. This concept is clarified, on the physical ground, by figure 1.7(a), where a sink (located at the origin of the adopted reference frame) and a source of the same intensity σ are aligned along the x axis and separated by a distance ℓ . This scheme leads to the genesis of a doublet if $\ell \rightarrow 0$ and $\sigma \rightarrow \infty$, so that the product $\ell\sigma$ is finite and such that $\ell\sigma \rightarrow \mu$.

Moreover, it can be proved [Katz and Plotkin, 1991] that the velocity field induced by a doublet is not characterized by a radial symmetry, but, on the other hand, it has a directional behaviour. This can be evinced by figure 1.7(b), where a doublet – whose axis is coincident with the x direction – is shown. In particular it is called the *doublet axis* the direction where both the originating sink and source lie. For this reason it is possible to consider a the quantity μ as a vector: $\boldsymbol{\mu} = \mu \mathbf{e}_\mu$, where \mathbf{e}_μ is the unit vector in the doublet direction.

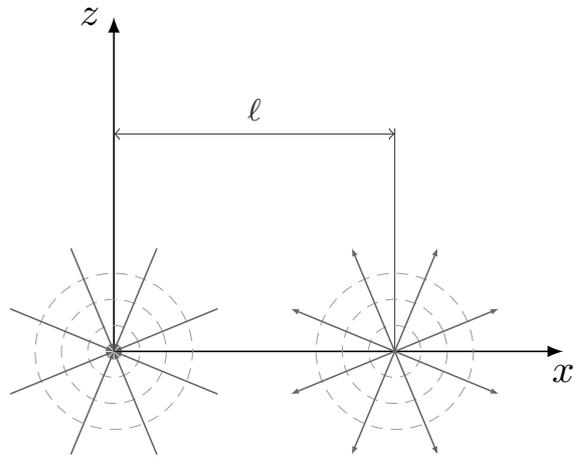
Vortex

Even though the equation (1.31) suggests that the general solution to the Laplace equation is only given by a superposition of sources and doublets, other basic solutions to the potential flow problem do exist. In particular, it could be convenient to define a singularity characterized by a purely circular motion (*i.e.* dual with respect to the source): the *vortex*.

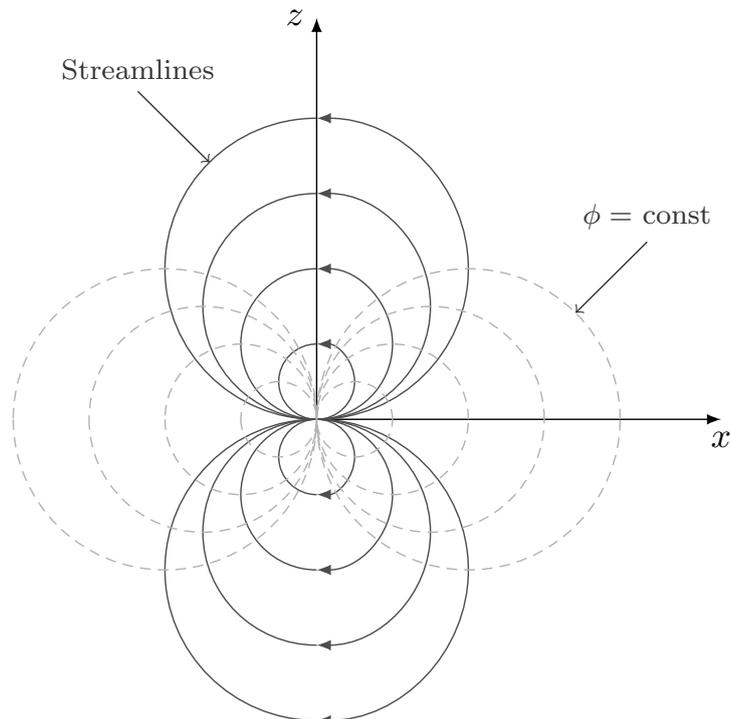
As shown in figure 1.8 a vortex is characterized by the solely presence of the tangential velocity component

$$V_\theta = -\frac{\Gamma}{2\pi r} ,$$

where Γ is the circulation evaluated along a generic path which surrounds the vortex itself. On the other hand, the velocity potential for a vortex at the origin (obtained integrating the induced velocity \mathbf{V}) is



(a) Doublet genesis



(b) Flow-field induced by a point doublet

Figure 1.7: Point doublet

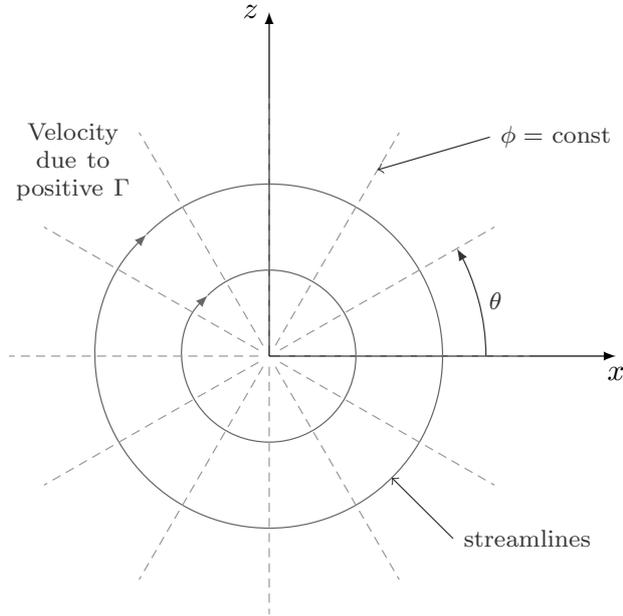


Figure 1.8: Flow-field induced by a point vortex

$$\phi = -\frac{\Gamma}{2\pi}\theta. \quad (1.35)$$

This result shows that the velocity potential of a vortex is multivalued, depending on the round of revolutions performed around the vortex point. Hence a vortex is characterized by vorticity concentrated at a zero area point, but with a finite circulation.

Line distributions of the basic solutions

In section 1.4.6 it has been shown that the solution of a potential flow-field around an arbitrary body can be obtained by distributing basic singularity entities over the boundaries S_b , so that it is favourable to analyze the behaviour of these singularity distributions along, for the sake of simplicity, a segment lying along the x axis and whose limits are denoted by x_1 and x_2 , such those shown in figure 1.9.

Consider at first a source distribution whose strength per unit length is denoted by $\sigma(x)$, as the one depicted in figure 1.9(a). Since – as discussed in section 1.4.7 – each source element emits flow in all directions, it is easy to understand that the resulting velocity will be away from the surface and hence a discontinuity in the w component, evaluated at $z = 0$ (*i.e.* along the considered $\overline{x_1x_2}$ segment), is generated. For this reason a pure-source distribution is suitable to model a symmetric flow-field. In particular, denoting with the plus sign the upper side of the segment and with a minus the lower one, it could be proved [Katz and Plotkin, 1991] that

$$w(x, 0\pm) = \pm \frac{\sigma(x)}{2} \quad (1.36)$$

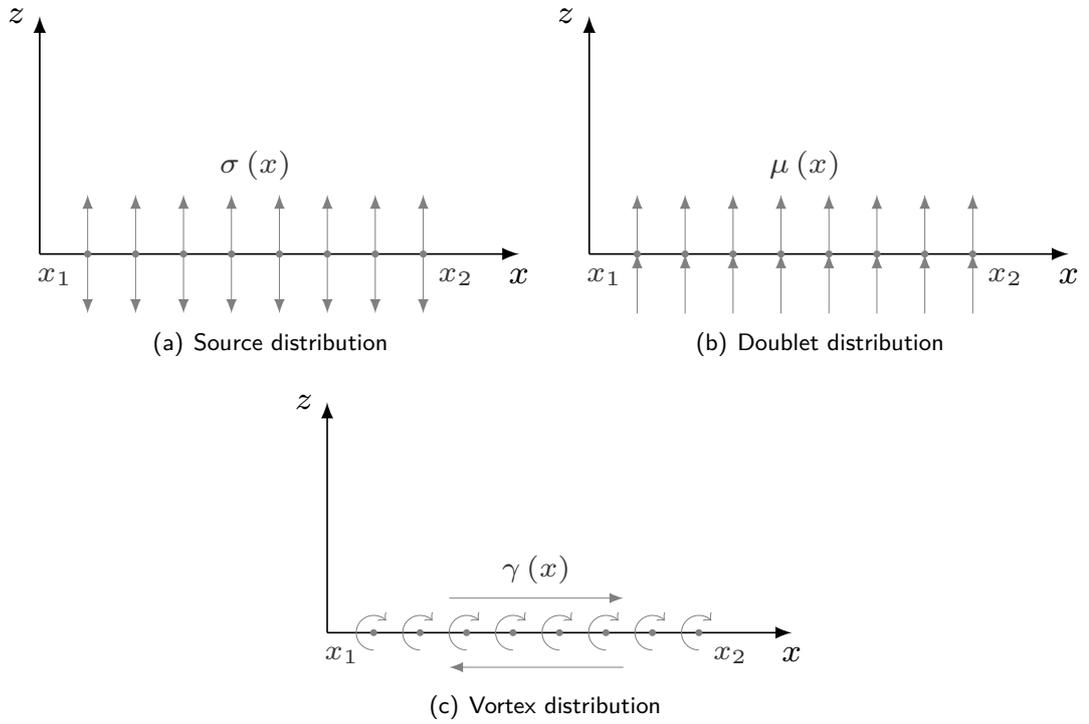


Figure 1.9: Singularity distribution over a segment $\overline{x_1x_2}$ [Katz and Plotkin, 1991]

and hence the discontinuity value amounts to

$$\Delta w = w^+ - w^- = \sigma(x) . \quad (1.37)$$

Figure 1.9(b) reports a scheme concerning the flow field near a line distribution of doublets, whose intensity per unit length is denoted by $\mu(x)$. Similarly to the source case, the doublet is characterized by a discontinuity in the potential Φ

$$\Phi(x, 0\pm) = \mp \frac{\mu(x)}{2} \quad (1.38)$$

which implies a discontinuous tangential velocity

$$u(x, 0\pm) = \frac{\partial \Phi}{\partial x} \Big|_{(x, 0\pm)} = \mp \frac{1}{2} \frac{\partial \mu(x)}{\partial x} . \quad (1.39)$$

Moreover, the circulation evaluated along a path surrounding the segment $\overline{x_1x}$ is equal to the jump in potential, *i.e.* $\Gamma(x) = -\mu(x) = \Delta\Phi(x)$.

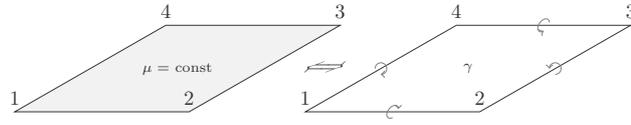


Figure 1.10: Doublet-Vortex equivalence

Finally, in figure 1.9(c) a uniform vortex sheet of intensity per unit length $\gamma(x)$ is represented. This image makes it clear that the current case is characterized by a discontinuity in the u component of the velocity on the segment $\overline{x_1x_2}$, amounting to

$$u(x, 0\pm) = \left. \frac{\partial \Phi}{\partial x} \right|_{(x, 0\pm)} = \pm \frac{\gamma(x)}{2}. \quad (1.40)$$

Furthermore in the present case $\Gamma(x) = \Delta \Phi(x)$. As a consequence, being $\Gamma = -\mu$, similar flow conditions can be modelled by either a vortex or a doublet distribution. In fact, comparing equations (1.39) and (1.40) it could be noticed that a vortex distribution is equivalent to a doublet one if

$$\gamma(x) = -\frac{d\mu(x)}{dx}. \quad (1.41)$$

Finally, due to its importance in the numerical applications, it should be pointed out that this equivalence can be extended to a three-dimensional case, where – as depicted in figure 1.10 – it holds between a vortex distribution and a doublet one of higher order (*i.e.* a dipole panel is equivalent to a vortex ring lying along its boundaries).

2 Panel Methods

“When a flow is both frictionless and irrotational, pleasant things happen.”

— F.M. White, *Fluid Mechanics*

In chapter 1 it has been pointed out that an inviscid, incompressible and irrotational flow field, either steady or unsteady, can be modelled by means of source and doublet entities located at the boundaries of the computational domain. In particular the unknown of this problem results to be the strength of the singularities employed. This approach allows to face engineering problems otherwise impossible to analyze through exact analytical methods. Numerical techniques based onto the Laplace's equation are also convenient – in terms of required computational resources – with respect to those which solve the fluid-dynamics equations in the whole flow-field (e.g. finite-difference methods). This advantage is due to the already cited peculiarity of the methods under examination to deal with surface singularity elements only. Nevertheless the limitations due to the simplifying assumptions employed should be taken in mind, since the introduction of other techniques is related to the need to treat more complex (e.g. viscous or compressible) situations.

2.1 Introduction

Consider a fluid domain surrounding, in general, a three-dimensional lifting body and let the flow be incompressible, inviscid and irrotational everywhere but in a thin surface – representing the wake – located in the aft part of each lifting body (*i.e.* for all the bodies for which the Kutta condition needs to be enforced), where all the vorticity is localized.

Given the linearity of the Laplace's equation, according to the

$$\Phi(\mathbf{x}) = \phi_{\infty}(\mathbf{x}) + \phi(\mathbf{x}) ,$$

it is always possible to decompose the total potential Φ into two contributes: the first one is the far-field potential, denoted by ϕ_∞ and given by equation (1.32), and already introduced in section 1.4.6; instead the second one is the perturbation potential ϕ induced by the presence in the field of the body. Therefore, in order to fulfill the free-stream BC (1.22), this latter contribution to Φ has to vanish as $r = |\mathbf{x}| \rightarrow \infty$, *i.e.*

$$\lim_{|\mathbf{x}| \rightarrow \infty} \nabla \phi = 0 \iff \lim_{|\mathbf{x}| \rightarrow \infty} \nabla \Phi = \nabla \phi_\infty . \quad (2.1)$$

The flow of interest is in the region \mathcal{V} outer from the body. Here the Laplace's equation for the total potential $\Phi(\mathbf{x})$ holds, so that

$$\nabla^2 \Phi = \nabla^2 \phi_\infty + \nabla^2 \phi = 0 . \quad (2.2)$$

The general solution to this problem, given by equation (1.31), can be achieved by means of a superposition of source and doublet entities placed at the body boundaries S_b . As discussed in section 1.4.4, since the same flow-field can be given by several singularity distributions, this formulation has not a unique solution until other closure conditions are enforced. Therefore an arbitrary choice should be made according to the particular problem under examination. In fact, for example a lifting problem requires the employment of antisymmetric elements, such as doublets and vortices, whereas, on the other hand, sources and sinks could be exploited to emulate some thickness-related effects [Katz and Plotkin, 1991]. At this stage the uniqueness of the solution can be achieved enforcing both the Boundary Conditions and the Kutta condition. Finally, as a consequence of the presence of a rotational wake modelled through either vortex or doublet elements, the equation (1.31) can be modified as follows

$$\Phi(\mathbf{x}) = \frac{1}{4\pi} \iint_{S_b \cup S_w} \mu \mathbf{n} \cdot \nabla \frac{1}{r} dS - \frac{1}{4\pi} \iint_{S_b} \sigma \frac{1}{r} dS + \phi_\infty , \quad (2.3)$$

where the unit vector \mathbf{n} points in the direction of the potential jump μ which is normal to S_b and positive towards the exterior of \mathcal{V} .

2.2 Boundary Conditions

The BCs suitable for the problem into analysis have already been discussed in section 1.4.4, however here a more detailed insight into this argument is carried out.

In general, for non-porous media, the wall BC stated in equation (1.23) can be fulfilled in two different ways. In fact either a zero normal velocity on S_b can be specified or it can be imposed the value assumed by Φ on these boundaries. The first, direct, approach is said to be a *Neumann BC*, whereas the second one is called the *Dirichlet problem* and it is characterized by an indirect fulfillment of the impermeability condition (1.23). Obviously a combination of these two methods can be employed, obtaining a *mixed BCs problem*.

2.2.1 Neumann Boundary Condition

In terms of the free-stream and perturbation potential, the equation (1.23) can be rewritten as

$$\nabla\Phi \cdot \mathbf{n} = \nabla(\phi + \phi_\infty) \cdot \mathbf{n} = 0 \quad (2.4)$$

and consequently the equation (2.3) leads to

$$\left\{ \frac{1}{4\pi} \iint_{S_b \cup S_w} \mu \nabla \left[\frac{\partial}{\partial n} \left(\frac{1}{r} \right) \right] dS - \frac{1}{4\pi} \iint_{S_b} \sigma \nabla \left(\frac{1}{r} \right) dS + \phi_\infty \right\} \cdot \mathbf{n} = 0. \quad (2.5)$$

This equation lies on the basis of the numerical solution of the Neumann's problem and it should be enforced for every point which belong to the boundary surface S_b . Obviously this cannot be achieved in a real problem, so that, in practice, equation (2.5) could be specified just in a limited amount of points (called either *collocation* or *control points*), resulting tush in a set of algebraic equations.

It should be noticed that, as a consequence of the *Green's first identity* [Kellogg, 2010], if – as stated by the BC (2.4) – it is $\frac{\partial\Phi}{\partial n} = 0$ for each point of a closed-loop boundary (e.g. S_b for an airfoil) which does not contain any singularity, then

$$\Phi_i = \text{const}. \quad (2.6)$$

Therefore the Neumann's problem is equivalent to a Dirichlet's one concerning the internal potential Φ_i .

2.2.2 Dirichlet Boundary Condition

As opposite to the Neumann's BC, the Dirichlet's problem consists in imposing the value assumed by the singularity distribution over S_b , and it is tush an indirect method to fulfill the wall BC. In this case equation (2.4) is satisfied employing the results stated into equation (2.6). In fact, this formula allows to write the equation (2.4) as

$$\Phi_i = (\phi + \phi_\infty)_i = \frac{1}{4\pi} \iint_{S_b \cup S_w} \mu \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS - \frac{1}{4\pi} \iint_{S_b} \sigma \frac{1}{r} dS + \phi_\infty = \text{const}, \quad (2.7)$$

which is a milestone in the resolution of problems characterized by the indirect BC. This equation can be related to different kinds of solutions. For instance, it is possible to set $\Phi_i = 0$, so that the equation (2.7) can be solved over the surface S_b . However, sice the resulting singularity distribution obtained through this method includes the free-stream potential ϕ_∞ , this approach may be characterized by elevated values of the singularities strength and hence by numerical issues. On the other hand, the inner potential could be set to $\Phi_i = (\phi + \phi_\infty)_i = \phi_\infty$, that is equivalent to specify equation (2.7) for the perturbation ϕ only in a reference frame where $\phi_\infty = 0$ (i.e. fixed with respect to the ground). In this latter case equation (2.7) becomes

$$\frac{1}{4\pi} \iint_{S_b \cup S_w} \mu \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS - \frac{1}{4\pi} \iint_{S_b} \sigma \frac{1}{r} dS = 0, \quad (2.8)$$

which holds when

$$\sigma = \mathbf{n} \cdot \mathbf{V}_\infty \quad (2.9)$$

where \mathbf{n} is directed towards the inner side of the surface. This result can be proved considering that

$$\frac{\partial \Phi}{\partial n} = \frac{\partial \phi}{\partial n} + \frac{\partial \phi_\infty}{\partial n} = \frac{\partial \phi}{\partial n} + \mathbf{n} \cdot \nabla \phi_\infty = 0 \quad \Rightarrow \quad \frac{\partial \phi}{\partial n} = -\mathbf{n} \cdot \mathbf{V}_\infty$$

and recalling the definition (1.30). In fact, being

$$-\sigma = \frac{\partial \Phi}{\partial n} - \frac{\partial \Phi_i}{\partial n} = \frac{\partial \phi}{\partial n} - \frac{\partial \phi_i}{\partial n}$$

and since $\Phi_i = \phi_\infty \Rightarrow \phi_i = 0$, then it is $\frac{\partial \phi_i}{\partial n} = 0$, too, and hence equation (2.9) holds.

2.3 Wake treatments

In three-dimensional lifting problems, the need for a *wake model* is justified by the Helmholtz vortex theorems. In fact, as remarked in section 1.4.4, this result states that a vortex line cannot start or end in a fluid, meaning that the bound vorticity has to be continued into a wake located downstream with respect to the wing TE. Moreover, in order for the wing to have a spanwise circulation Γ , a discontinuity in the velocity potential at the TE is required, being – as shown in section 1.4.7 –

$$\Gamma = \Phi_u - \Phi_l = \Delta \Phi ,$$

where the subscripts u and l refer to two points located above and under the wing TE, respectively.

Furthermore, as pointed out in section 1.4.4, a wake model is required to achieve the closure of a lifting problem (*i.e.* to fix one value of the circulation Γ), specifying both the wake strength at the TE and its shape and location.

2.3.1 Wake strength

In order to determine the wake strength at the TE of a wing, the two-dimensional Kutta condition could be imposed along the 3D TE itself. In the present case, the condition stated in equation (1.24) can be formulated in terms of the vortex specific intensity $\gamma(x)$, imposing that

$$\gamma_{TE} = 0 . \quad (2.10)$$

From this formula, holding the equation (1.41), it stems that

$$\mu_{TE} = \text{const} = \mu_w \quad \Longleftrightarrow \quad \mu_u - \mu_l = \mu_w$$

or, in terms of circulation

$$\Gamma_u - \Gamma_l = \Gamma_w ,$$

that is to say that μ is constant in the wake and it is equal to the value assumed on the TE.

2.3.2 Wake shape

Whereas in a two-dimensional problem, due to the absence of a trailing vortex segment for the wake, it is sufficient to specify the location where the Kutta condition is met, in a three-dimensional case a more complete closure condition is required. Moreover, in this latter case, the solution is more affected by the wake.

In order to find out the wake shape, it is necessary to assume that the wake itself (assumed to be thin) is not a solid body and hence it does not produce any lift. In general, the aerodynamic force $\Delta \mathbf{F}$ generated by a vortex sheet whose specific intensity is denoted by γ is given by the *Kutta-Joukowski Theorem*

$$\Delta \mathbf{F} = \rho \mathbf{V} \times \gamma . \quad (2.11)$$

Therefore, imposing $\Delta \mathbf{F} = 0$ into equation (2.11) and assuming $\gamma \neq 0$, the condition for the wake geometry is carried out

$$\mathbf{V} \times \gamma_w = 0 \quad \Longleftrightarrow \quad \gamma_w \parallel \mathbf{V} . \quad (2.12)$$

Recalling that a vortex distribution is equivalent to a doublet one of higher order (see section 1.4.7), equation (2.12) can be rewritten as

$$\mathbf{V} \times \nabla \mu_w = 0 \quad \Longleftrightarrow \quad \mu_w = \text{const} . \quad (2.13)$$

2.4 Numerical procedure

The numerical method employed to solve the potential flow problem can be obtained if both body and wake geometries are discretized into several (quadrilateral or triangular) portions, called *panels*, whose shape can be expressed, in general, by a function f such that $z = f(x, y)$. However, for simplicity, f is usually in the form of a piecewise polynomial, whose order is related to the wanted approximation level of the real geometry. In particular, due to the increasing computational cost, typically the most employed polynomials are up to the second order, *i.e.*

$$\begin{aligned} z &= a_0 + b_1 x + b_2 y , \\ z &= a_0 + b_1 x + b_2 y + c_1 x^2 + c_2 xy + c_3 y^2 , \end{aligned}$$

where the coefficients a , b and c are constants. However, since in general the real geometry is curved and characterized by two different principal radii, the discretization could rise some issues, such as the one depicted in figure 2.1, where a *leakage flow* does occur. This problem may result in difficulties in specifying the BC, since the leakage between adjacent panels could compromise the satisfaction of the zero normal flow across the boundary requirement.

Aside with the geometry discretization into panel, a similar process is required for the singularity distribution. This task is typically accomplished by means of polynomial approximations either of the first or the second order, too. It should be underlined that the approximation order employed for both the geometry and the singularities should be the same, otherwise the resulting

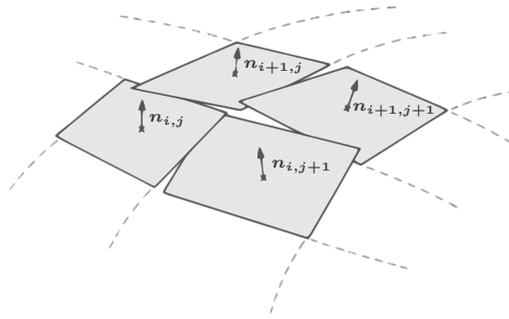


Figure 2.1: Leakage flow [Katz and Plotkin, 1991]

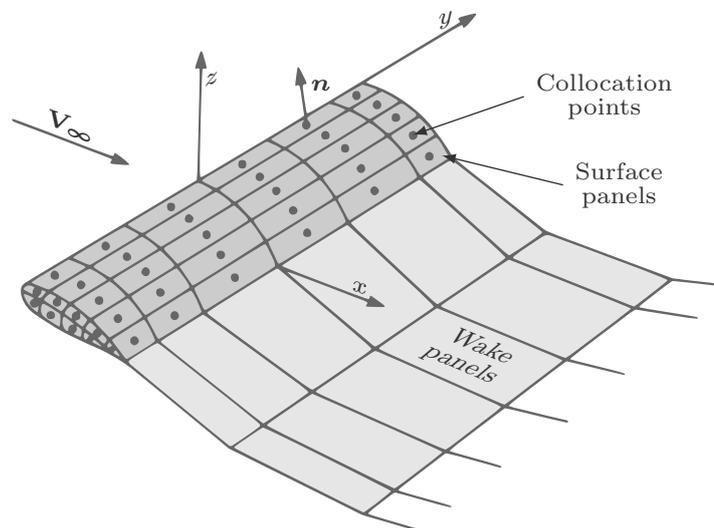


Figure 2.2: Approximation of the body surface by panel elements [Katz and Plotkin, 1991]

approximation will be determined by the lowest order one, so that no gain in accuracy is obtained despite the greater computational effort required.

Assuming that – as shown in figure 2.2 – the geometry has been discretized into N body surface panels and N_w wake ones, the Boundary Condition (either Dirichlet or Neumann) can be enforced at the collocation point of each panel. In particular, if a Dirichlet's BC is imposed at the boundaries, the collocation points should not be located on the panel surface, but inside the body (*i.e.* at a point under the center of the panel), where $\Phi_i = \phi_{\infty}$.

Consider now a potential Dirichlet's problem, so that equation (2.8) holds. If the discretization shown in figure 2.2 is applied, then this equation becomes

$$\sum_{k=1}^N \frac{1}{4\pi} \iint_{S_b^k} \mu \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS + \sum_{l=1}^{N_w} \frac{1}{4\pi} \iint_{S_w^l} \mu \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS - \sum_{k=1}^N \frac{1}{4\pi} \iint_{S_b^k} \sigma \frac{1}{r} dS = 0, \quad (2.14)$$

where the integrals are taken over each panel, and for a unit singularity element (either σ or μ) they depend on the panel geometry only. Moreover, as illustrated in figure 2.3, these integrals represent the influence of a particular panel on the considered point P . Denoting with $1, \dots, 4$ the four angles of a quadrilateral panel and assuming constant strength elements, this influence can be written as

$$\begin{aligned} \frac{1}{4\pi} \iint_{1,2,3,4} \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS \Big|_k &= C_k, \\ -\frac{1}{4\pi} \iint_{1,2,3,4} \frac{1}{r} dS \Big|_k &= B_k, \end{aligned}$$

where k denotes the k^{th} panel and B_k and C_k are the *influence coefficients*: in general, they depend on the relative positions of the points P and $1, \dots, 4$ only. Employing these results, for every internal point, the BC expressed by equation (2.14) yields

$$\sum_{k=1}^N C_k \mu_k + \sum_{l=1}^{N_w} C_l \mu_l + \sum_{k=1}^N B_k \sigma_k = 0. \quad (2.15)$$

If the strength of the sources has been selected according to equation (2.9), since the influence coefficients B_k can be computed, the third term in equation (2.15) is known and hence it can be moved to the RHS of the formula. Moreover, as shown in section 2.3.1, imposing the Kutta condition at the TE, the strength of the wake doublets μ_w can be written as a linear combination of the intensities of the unknown surface doublets μ_k adjacent to the considered wake panel. Therefore the influence of the generic wake element can be expressed as

$$\mu_w = \mu_u - \mu_l \quad \Rightarrow \quad C_w \mu_w = C_w (\mu_u - \mu_l),$$

which can be substituted into equation (2.15), leading to

$$\sum_{k=1}^N A_k \mu_k = - \sum_{k=1}^N B_k \sigma_k, \quad \begin{cases} A_k = C_k & \text{if the panel is not at the TE} \\ A_k = C_k \pm C_w & \text{if the panel is at the TE} \end{cases}. \quad (2.16)$$

Evaluating the equation (2.16) at each of the N collocation points results in a system of N linear equations in the N unknown μ_k . In fact, equation (2.16) can be reformulated as

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_N \end{pmatrix} = - \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1N} \\ b_{21} & b_{22} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ b_{N1} & b_{N2} & \cdots & b_{NN} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_N \end{pmatrix}, \quad (2.17)$$

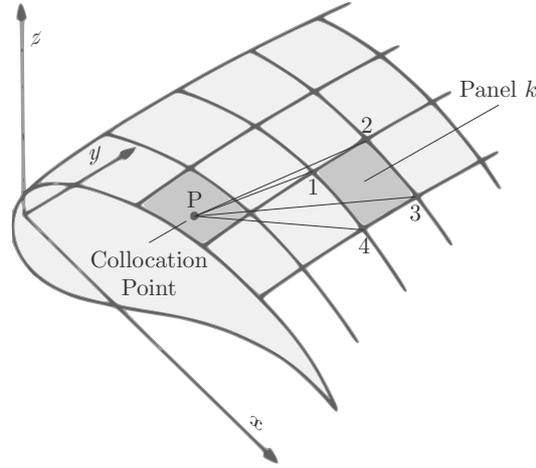


Figure 2.3: Influence of panel k on point P [Katz and Plotkin, 1991]

where the RHS is known since, as said before, equation (2.9) holds. Furthermore, it should be pointed out that employing the decomposition of the total potential Φ into a free-stream contribution ϕ_∞ and a perturbation one ϕ , jointly with the choice $\Phi_i = \phi_\infty$ (see section 2.2.2), results into a relatively small μ distribution, and hence the problem (2.17) is expect to be stable. However, as it is obvious, the goodness of the solution to the problem is strictly related to the solution of the system (2.17).

Once the system (2.17) has been solved for the unknown μ_k , the perturbation velocities on the k^{th} panel can be computed in a local reference frame (P, l, m, n) as

$$\begin{aligned} v_l &= -\frac{\partial \mu}{\partial n} , \\ v_m &= -\frac{\partial \mu}{\partial m} , \\ v_n &= -\sigma , \end{aligned}$$

where the normal component is compatible with the assumption (2.9). It should be noticed that v_n is, in general, different from zero: this is due to the fact that the impermeability condition applies to the total velocity $\mathbf{V} = \mathbf{V}_\infty + \mathbf{v}$ and not just on its perturbation contribution \mathbf{v} . At this stage the pressure coefficient on the k^{th} panel can be evaluated by means of its incompressible definition, *i.e.*

$$C_{P_k} = 1 - \left(\frac{V_k}{V_{\text{ref}}} \right)^2 .$$

Moreover, the contribution of a single panel to the total aerodynamic force is

$$\Delta C_{F_k} = -\frac{C_{P_k} \Delta S_k}{S} ,$$

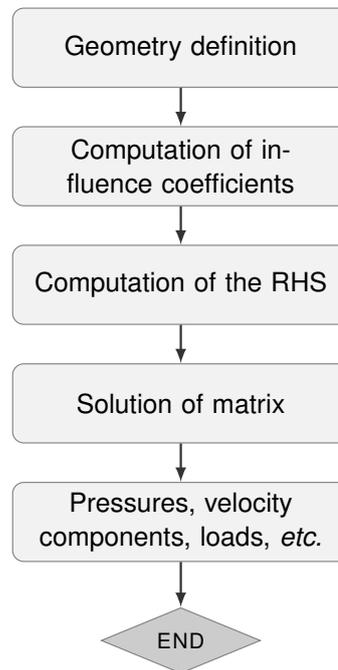


Figure 2.4: Numerical procedure for the solution of the steady surface singularity distribution problem [Katz and Plotkin, 1991]

where S_k is the panel area and S is the reference one.

Finally, the numerical procedure aimed at the resolution of the steady surface singularity distribution problem described in this section has been summarized in figure 2.4.

2.5 Unsteady panel methods

Even though the method shown in section 2.4 holds for steady flows only, with slight modifications to the BCs and employing the Bernoulli equation in the form (1.21), it can be generalized to the unsteady case, too.

As depicted in figure 2.5, in unsteady panel methods the solution of the potential flow is embedded into a time stepping loop, starting at $t = 0$. Each time step of this loop is characterized by the fact that a new row of wake panels is shed, as their strength has been evaluated enforcing the Kutta condition.

As in section 2.4, consider a distribution of constant strength sources and doublets and a Dirichlet BC, so that equation (2.15) is verified for every time step. If two different reference frames are taken into account, one of which (denoted by (O, X, Y, Z)) is inertial, whereas the other one (denoted by (o, x, y, z)) is integral with respect to the body (*i.e.* in general, it is in relative motion with respect to (O, X, Y, Z)), the zero velocity condition at the body becomes

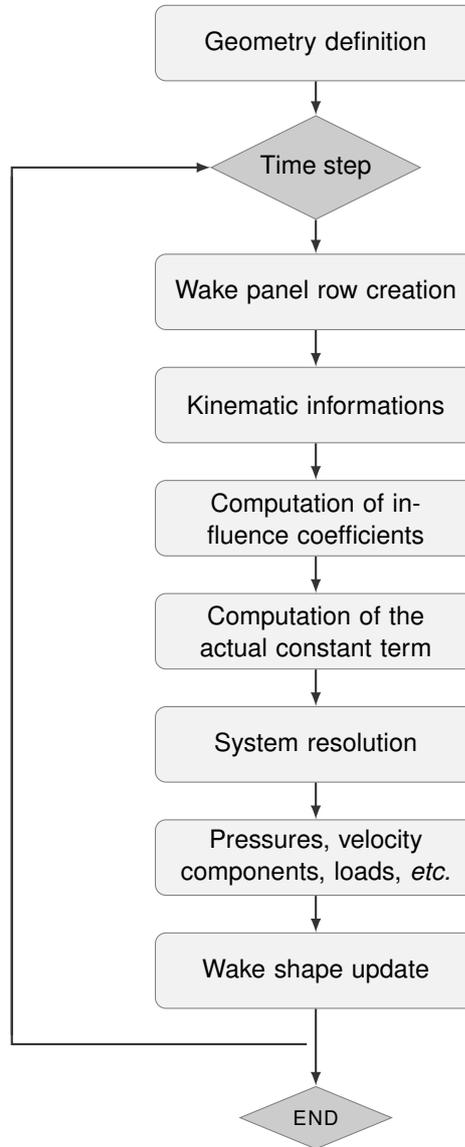


Figure 2.5: Numerical procedure for the solution of the unsteady surface singularity distribution problem [De Nicola, 2013]

$$\frac{\partial \phi}{\partial n} = (\mathbf{V}_0 + \mathbf{V}_r + \boldsymbol{\Omega} \times \mathbf{r}) \cdot \mathbf{n} = 0 ,$$

where $\mathbf{r} = (x, y, z)$ is the position vector of a point as seen by an observer integral with the moving coordinate system and \mathbf{V}_0 , \mathbf{V}_r and $\boldsymbol{\Omega}$ are the velocity of the origin of the body-fixed reference frame, the relative velocity and the angular one, respectively. Enforcing the BC, this equation yields

$$\sigma = -\mathbf{n} \cdot (\mathbf{V}_0 + \mathbf{V}_r + \boldsymbol{\Omega} \times \mathbf{r}) . \quad (2.18)$$

Supposing that at the time $t = 0$ the two reference frames introduced above are coincident, the calculation starts at the time $t = \Delta t$, where Δt is the duration of a time step. That is to say that the first row of wake panels is generated near the TE and their strength can be therefore evaluated by means of the Kutta condition. Once the first row of wake panels has been shed, similar developments to those applied in section 2.4 can be carried out, so that the Dirichlet BC can be written in the form (2.16). Moreover, it should be pointed out that, typically, the characteristic length of the wake panels adjacent to the TE amounts to $0.2 \div 0.3$ times the space travelled by the TE itself in a time Δt (*i.e.* $V_\infty \Delta t$) [Katz and Plotkin, 1991].

At $t = 2\Delta t$ the second time step begins and a new wake panel row should be created. In order to achieve this task, since a relative motion between the two different coordinate systems here taken into account has occurred, the row generated in the previous time step should be considered fixed with respect to the inertial frame (O, X, Y, Z). In fact, in this way a gap between the actual TE and the existing wake panels is created and can be filled shedding the new row that, being adjacent to the TE, can be evaluated by means of the Kutta condition. In this scenario the influence of the pre-existing wake panels on the new ones has to be considered, so that in the unsteady case equation (2.16) holds for the first time step only, whereas for the following times the BC at the collocation point of the i^{th} panel becomes

$$\sum_{k=1}^N A_{ik} \mu_k + \sum_{l=1}^{M_w} C_{il} \mu_l + \sum_{k=1}^N B_{ik} \sigma_k \quad t = n\Delta t, \forall n \geq 2 , \quad (2.19)$$

where M_w is the number of wake panels that influences the BC and thus it does not include the ones belonging to the actual row (*i.e.* the row that is being generated). Since, according to equation (2.18), the knowledge of the kinematics of the body implies the one of the source distribution strength, both the second and the third term in equation (2.19) are known and hence the system is in closed form.

Obviously this procedure has to be iterated until the final simulation time is reached. However, it should be noticed that, if the body geometry remains unchanged in time, the inverse matrix can be evaluated just once.

When the calculation has been carried out and the strength of the unknown doublets has been evaluated, the pressure coefficient C_P can be found out through the formula

$$C_P = 1 - \left(\frac{V}{V_{ref}} \right)^2 - \frac{2}{V_{ref}^2} \frac{\partial \phi}{\partial t} ,$$

where V is the local velocity, $\mathbf{V}_{ref} = (\mathbf{V}_0 + \boldsymbol{\Omega} \times \mathbf{r})$ and $\frac{\partial \phi}{\partial t} = \frac{\partial \mu}{\partial t}$ if $\phi_i = 0$. In a similar fashion, the aerodynamic force acting on an element of area ΔS_k is

$$\Delta \mathbf{F}_k = -C_{P,k} \left(\frac{1}{2} \rho V_{ref}^2 \right)_k \Delta S_k \mathbf{n}_k .$$

Finally, at each time step the wake roll-up can be evaluated through the induced velocity in the *corner points* of the wake panels in the inertial reference frame. It should be pointed

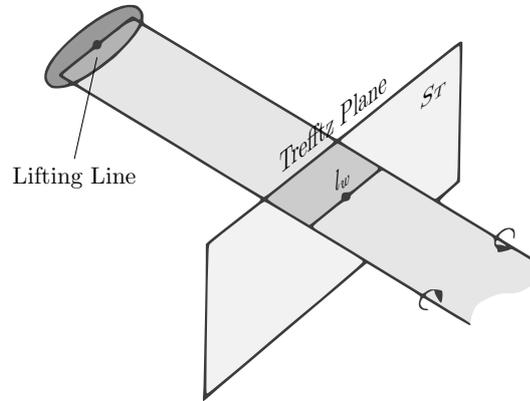


Figure 2.6: Trefftz plane [Katz and Plotkin, 1991]

out that the wake can be either flexible or rigid (or a combination of both): in the first case the previous considerations about the roll-up applies, whereas in the latter one the problem is linearized imposing the wake geometry (*i.e.* the flow field cannot influence the wake).

2.6 Trefftz technique

Even though the aerodynamic forces could be evaluated through a direct integration of the pressure over the panels, this approach may require a great number of panels and hence may result in elevated computational times, partially mining one the greatest advantages of the methods based onto the Laplace's equation. Nevertheless, this approach is not the only one that can be exploited to achieve the goal: in fact both the lift and the induced drag can efficaciously and efficiently be evaluated by means of an indirect analysis conducted into the *Trefftz plane* S_T (*i.e.* the plane located at an infinite distance downstream of the body and perpendicular to its wake), depicted in figure 2.6.

Under the assumptions of incompressible and inviscid flow, whose vorticity is only enclosed in the thin wake, the integral form of the momentum equation is

$$\mathbf{F} = \iint_S \rho \mathbf{V} (\mathbf{V} \cdot \mathbf{n}) dS + \iint_S \frac{1}{2} \rho V^2 \mathbf{n} dS . \quad (2.20)$$

If the reference frame is characterized by an unperturbed wind-parallel x axis (*i.e.* $\mathbf{V} = (V_\infty + u, v, w)$) and if the control volume is large enough so that the perturbation velocity components will vanish everywhere but on the wake, then equation (2.20) yields [Katz and Plotkin, 1991]

$$D_i = \frac{\rho}{2} \iint_{S_T} (v^2 + w^2) dS = -\frac{\rho}{2} \int_{l_w} \Delta \phi \frac{\partial \phi}{\partial n} dl = -\frac{\rho}{2} \int_{l_w} \Gamma V_n dl , \quad (2.21)$$

where, due to the chosen coordinate system, the induced drag D_i is the x component of the total aerodynamic force. Moreover, in equation (2.21), the wake has been projected along a path

parallel to the flow, onto the Trefftz plane, as shown in figure 2.6. Here Γ is the circulation evaluated on the l_w path shown in figure 2.6 and V_n is the component of the velocity induced by the wake orthogonal to the wake itself. Furthermore, the Trefftz analysis may be led in a near-field plane [Kroo, 2001], considering the doublets located nearby the wing TE and thus overcoming some far-field-related issues such as the need to compute the velocities over a very large area. In this context, according to the *lifting line* theory, evaluating the downwash at the start of a streamwise trailing vortex wake one obtains half of the downwash at the Trefftz plane (because the wake extends infinitely in both directions from the Trefftz plane but only in one direction from its start). As a consequence the near-field technique differs from the classical far field method by a factor $1/2$, *i.e.*

$$D_i = \int_{wing} \rho \Gamma V_n d\ell .$$

Moreover, this technique allows to avoid the numerical issues that may arise whenever in the field there are some vortices located far away from the body. Furthermore, both for time dependent and independent calculations, it is difficult to obtain a regular shape of the wake at great distances from the body. In fact, since the wake is inherently fluctuating, it may be hardly handled even for a steady case.

Obviously, the same far-field technique employed to evaluate the induced drag, can be exploited in order to compute the lift generated by a body. In this case

$$L = \rho V_\infty \int_{l_w} \Delta\phi d\ell = \rho V_\infty \int_{l_w} \Gamma d\ell .$$

2.7 Vortex Particle Method

Hitherto the present dissertation has dealt with a panel-only representation of the wake shed behind an aerodynamic body, even though this choice is not the only one possible. In particular, this section introduces the concept of *vorton*: a three dimensional singularity point vortex whose clusters are equivalent to wake panels. This singularity may be employed in the so called Vortex Particle Methods, where the domain vorticity (lying only in the thin wake) is modelled by vortons, rather than – as usual in 3D problems – by means of either vortex sheets or doublet panels.

The vorton methods have been introduced in the computational aerodynamics in order to overcome one of the main limits inherent the panel methods in their classical formulation: the intersection of the wake with downstream body surfaces. In fact, whenever a doublets wake intersects with a downstream body (*e.g.* the wing portion located downstream with respect to a propeller), the physical meaning of the solution to the potential flow problem is irremediably lost. Even though actually it is possible to manage this kind of interference by means of wake panels, it is simple to understand that this task is easily achieved by means of vortex particles due to their nature of independent elements. Moreover the vorton methods allow to reduce the numerical instabilities related to discretized vortex sheets, do not require an excessively high density of vortex particle in order to accurately model the vortex sheet and do not require elevated computational times.

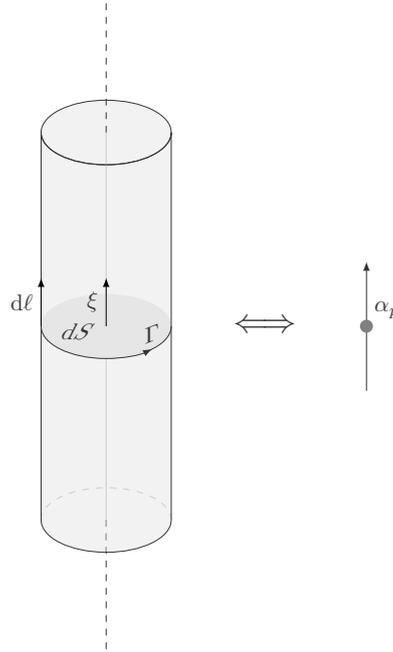


Figure 2.7: Vortex tube and its corresponding vorton [Caccavale, 2008, web]

2.7.1 Formulation

Consider a vorticity region of volume \mathcal{V} discretized in a set of cubes whose side length is denoted by h . With this decomposition a *vorton* (also known as *vortex particle* or *vortex stick*) can be simply defined as a three-dimensional point vortex which discretizes the vorticity field, *i.e.* [Winckelmans, 1989]

$$\boldsymbol{\omega}(\mathbf{x}, t) = \sum_P \boldsymbol{\omega}_p(t) h^3 \delta(\mathbf{x} - \mathbf{x}_p(t)) = \sum_P \boldsymbol{\alpha}_p(t) \delta(\mathbf{x} - \mathbf{x}_p(t)) , \quad (2.22)$$

where $\boldsymbol{\alpha}_p(\mathbf{x}, t)$ is the strength vector, given by the product between the vorticity and the volume of the element (*i.e.* $\boldsymbol{\alpha}_p = \boldsymbol{\omega}_p h^3$), \mathbf{x} is the position vector of the point of evaluation, \mathbf{x}_p is the one related to the location of the p^{th} vortex particle and $\delta(\mathbf{x})$ is the three-dimensional delta function. Therefore a vorton is a vector quantity identified by a position vector \mathbf{x} , a strength vector $\boldsymbol{\alpha}_p$ and eventually a *core radius* [Cottet and Koumoutsakos, 2000].

In figure 2.7 the equivalence between a vortex tube with constant cross-sectional area and a vorton is depicted. Obviously a single vortex tube may be discretized in several vortex particles and, consequently, a vorton may be thought as a small section of a vortex tube [Winckelmans and Leonard, 1993].

Furthermore, even though in equation (2.22) the velocity field $\mathbf{V}(\mathbf{x}, t)$ is clearly not given, its vorticity induced component \mathbf{V}_ψ (see section 1.4) can be evaluated taking the curl of a stream function $\psi(\mathbf{x}, t)$ which solves the Poisson's equation (1.17). In particular, it could be proved [see Winckelmans, 1989] that the vector potential may be written in the form

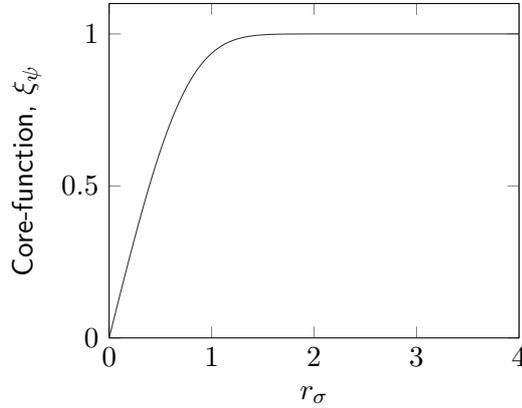


Figure 2.8: Gaussian core function [Strickland and Baty, 1994]

$$\psi(\mathbf{x}, t) = \frac{1}{4\pi} \sum_p \frac{\boldsymbol{\alpha}_p(t)}{|\mathbf{x} - \mathbf{x}_p(t)|}. \quad (2.23)$$

It appears clear that equation (2.23) is singular for $|\mathbf{x} - \mathbf{x}_p| = 0$ and hence a *core-function* ξ_ψ is required in order to regularize the vector potential when $|\mathbf{x} - \mathbf{x}_p| \rightarrow 0$ (i.e. it should be finite) [Chorin and Bernard, 1973]. Denoting by σ the *core radius* associated with the vorton and by r_σ the ratio between the absolute distance of the evaluation point from the vortex particle core and σ itself (i.e. $r_\sigma = |\mathbf{x} - \mathbf{x}_p(t)|/\sigma$), the core-function ξ_ψ is such that, when $r_\sigma < 1$, the vector potential ψ decays linearly to zero. Moreover, it should be noted that, in order to preserve the form (2.23) for ψ and being

$$\psi_\sigma(\mathbf{x}, t) = \frac{1}{4\pi} \sum_p \frac{\boldsymbol{\alpha}_p(t)}{|\mathbf{x} - \mathbf{x}_p(t)|} \xi_\psi = \sum_p \boldsymbol{\alpha}_p(t) \chi_\sigma(\mathbf{x} - \mathbf{x}_p(t)), \quad (2.24)$$

ξ_ψ has to approach unity for values of r_σ greater than one. A wide variety of core-functions do exist [Winckelmans and Leonard, 1993]. For example, among these there is the *Gaussian smoothing function* whose form is [Strickland and Baty, 1994]

$$\xi_\psi = 1 - e^{-qr_\sigma}, \quad q = 1.354 + 0.842 r_\sigma + 0.559 r_\sigma^2$$

and which, as shown in figure 2.8, approaches unity for about $r_\sigma \geq 1.5$.

As for the vector potential, the velocity results singular for $|\mathbf{x} - \mathbf{x}_p| = 0$, too, therefore a core-function ξ_V similar to the one already introduced for ψ is required.

A peculiar characteristic of the vortons is that these elements are governed by the vorticity evolution equation (1.10) [Strickland et al., 2002] and hence each vortex particle is convected by the local velocity and stretched by the local velocity gradient. In particular, in this context the vortex stretching term is [Willis, 2006]

$$\nabla(\nabla \times \psi(\mathbf{x})) = \frac{1}{4\pi} \sum_p \nabla \left[\nabla \left(\frac{1}{|\mathbf{x} - \mathbf{x}_p(t)|} \right) \times \boldsymbol{\alpha}_p(\mathbf{x}, t) \right].$$

At this point, it should be noted that the particle vorticity field (2.22) is not divergence-free [see Winckelmans, 1989]. This fact makes the method inconsistent in some sense, because a basis which is not divergence-free is employed to represent a vector field that should be solenoidal for all times. Similarly, since the Poisson's equation (1.17) is solved with ω generally not divergence-free, the vector potential (2.23) is also not generally solenoidal.

Of particular interest in the scope of this thesis is the conversion of a wake modelled by means of doublet panels into vortex particles, that is to say how to assign the strength to a vorton so that the vorticity region which it represents is equivalent to the one modelled through a common panel element. A possible conversion approach is made of three different steps [Willis, 2006]:

1. determine the equivalent vortex representation for each doublet panel to be converted to vortex particles;
2. establish the number of vortons to be emitted by each panel;
3. divide the panel into equal area segments and generate a vortex particle at the centroid of each of the segments.

Therefore, the vorton strength vector α_p can be evaluated by integrating the strength of the vortex line surrounding each panel area segment, *i.e.*

$$\alpha_p(\mathbf{x}, t) = \oint_{S_p} \Gamma(t) d\ell,$$

and hence, recalling that $\mathbf{V}_\psi = \nabla \times \psi$ and according to equation (2.23), the rotational velocity field may be calculated as a linear combination of a set of velocities induced by the vortons. A consequence of the velocity locally induced by a vortex particle is that the vortons themselves are characterized by a time evolution, consisting on a change in time of strength and position. These two effects can be respectively written, in a lagrangian representation, as [see Winckelmans, 1989]

$$\begin{aligned} \frac{d\mathbf{x}_p}{dt} &= \mathbf{V}_p(\mathbf{x}_p(t), t); \\ \frac{D\alpha_p}{Dt} &= \alpha_p(t) \cdot \nabla \mathbf{V}_p(\mathbf{x}_p(t), t). \end{aligned}$$

2.7.2 Vorton wake

The Vortex Particle Methods can be efficaciously employed to model the thin wake in three-dimensional, unsteady, inviscid and incompressible flows since they are easier to handle with respect to the classical panel methods. In fact it could be observed that, whereas, on the one hand, a doublets wake requires the evaluation of the induction of the flow field due to four segments of each quadrilateral panel, on the other hand the vorton methods are characterized by the need to compute, for every vortex particle, the induction due to only a point. Moreover, VPMs are extremely flexible: in fact a three-dimensional point vortex does not necessarily belong to the a specific wake panel for every time step.

Aside with the classical wake representation by means of dipole panels, it is possible to employ a VPM, so that, as shown in figure 2.9, the wake modelling is composed by both a doublet buffer

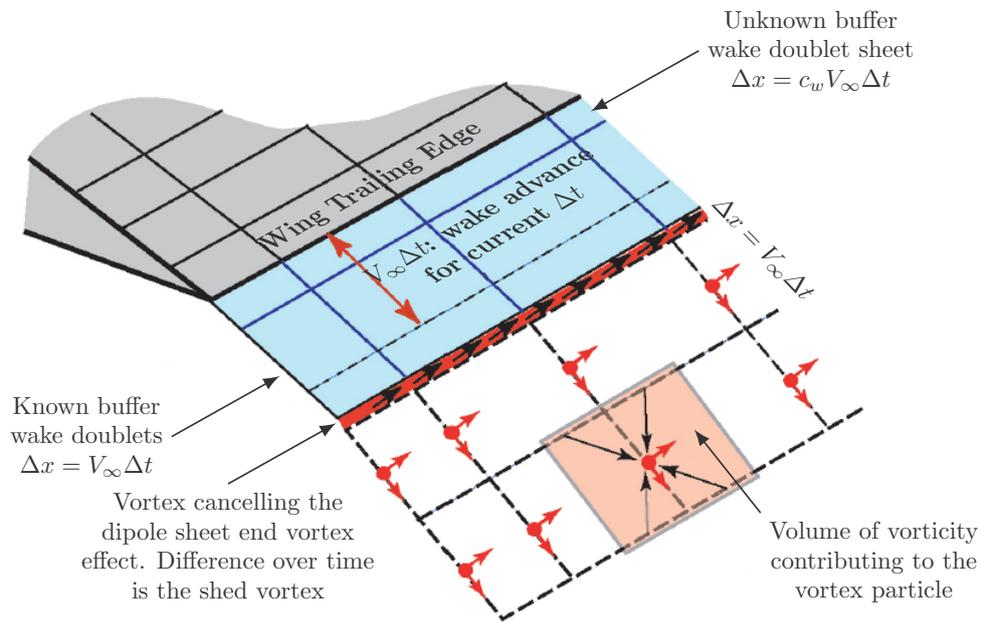


Figure 2.9: Example of vorton wake model [Willis et al., 2005]

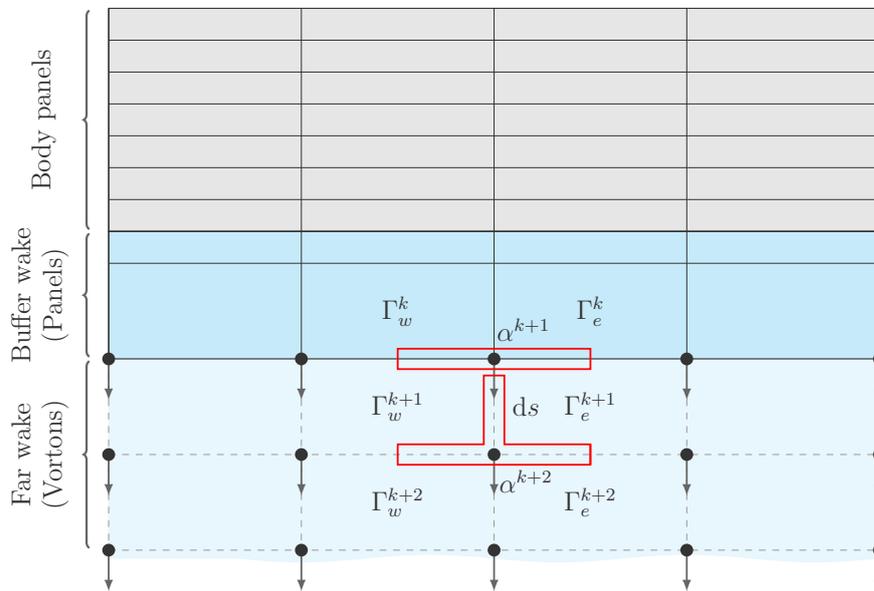


Figure 2.10: PaMS vorton wake model [Caccavale, 2008, web]. The area enclosed by the red line represents the integration zone employed to compute the vorton strength

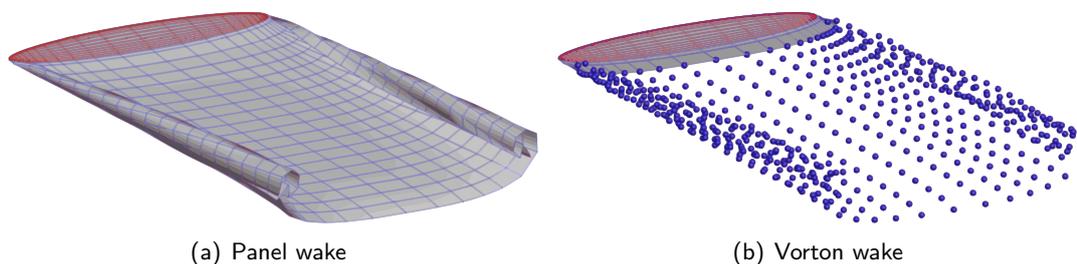


Figure 2.11: Panel and vorton wake [Caccavale, 2008, web]

sheet and vortex particles [Willis et al., 2005]. According to this perspective, the near wake (trailing every lifting surface) is formed by a doublet sheet which – in the spanwise direction – comprises the same amount of panels as the wing TE, whereas in the streamwise direction it is composed of at least two panel rows. Moreover, as stated in section 2.5, the row of wake panels closest to the TE has a strength determined by enforcing the potential Kutta condition and considering a characteristic length of $c_w V_\infty \Delta t$, where c_w is a constant typically in the range $0.2 \div 0.3$ [Katz and Plotkin, 1991] and Δt is the duration of a time step. On the other hand the second row of wake buffer panels has a length of $V_\infty \Delta t$ and a strength corresponding to the trailing edge potential jump evaluated at the previous time step. At this stage, it should be clear that the dipole buffer wake is required in order to achieve the closure of the potential problem through the imposition of the Kutta condition.

By contrast, the far wake region (*i.e.* the one located outside the buffer zone) is modelled through vortex particles, so that it is needed to convert the previous time step last row of wake panels into vortons. This task may be accomplished considering the doublet-vortex equivalence discussed in section 1.4.7, stating that the change in dipole strength along a surface in a given direction is equivalent to vorticity oriented in the surface tangential direction normal to the dipole gradient [Katz and Plotkin, 1991]. As shown in figure 1.10, for the particular case of constant doublet panels, the vortex analogue is a vortex ring around the perimeter of the given panel, implying that the strength of the vortex line segment between two adjacent constant strength dipole panels is merely the difference in their strengths. Therefore, the vorton is computed by integrating the strength of the vortex line segments between adjacent panels depending on the examined model.

It should be noticed that several strategies to convert a panel in vorton elements do exist: in this scenario the Willis' approach reported in figure 2.9 represents just an example, whereas, for instance, another way to accomplish the conversion is to locate the vortex particles on the four vertices of each quadrilateral panel, as shown in figure 2.10. In particular, this latter method is the one exploited by the solver employed in this work (described in chapter 5 on page 59).

3 Propeller

Since this thesis concerns the study of complex rotating objects as propellers with variable rotation axis, in order to provide to the reader all the instruments needed to achieve a full understanding of the subjects here dealt, it is dutiful to briefly introduce both the definitions and the theories lying on the bases of a rotary wing analysis. This is the purpose of the current chapter, where the **BEMT** is introduced. However the contents here reported are far from thorough, so that for complete argumentations the author recommends to refer to the bibliography [see for example Johnson, 1994, 2013; Tognaccini, 2011].

3.1 Generalities and definitions

Slicing a generic propeller blade of radius R at a distance $r_h < r < R$ from its hub (whose radius has been denoted with the symbol r_h) by means of a plane orthogonal to the propeller plane of rotation, one gets a wing section whose chord $c(r)$ forms an angle $\theta(r)$ – called either *blade pitch angle* or *feathering angle* – with the plane of rotation itself.

As it can be observed in figure 3.1, the relative motion between this blade element and the surrounding flow is given by a velocity vector $\Omega r = 2\pi nr$ which is due to the revolution of the propeller about its axis (and hence lays in the plane of rotation) and by a vector V_∞ which, on the other hand, is related to the forward translation of the rotor along its symmetry axis. Therefore the *effective velocity* V_e can be written as

$$V_e = \sqrt{(\Omega r)^2 + V_\infty^2}. \quad (3.1)$$

The vector V_e forms an angle ϕ – called the *inflow angle* – with $\Omega \times r$ (i.e. with the plane of rotation), whereas, the angle of incidence

$$\alpha = \theta - \phi = \theta - \arctan \frac{V_\infty}{2\pi nr} = \theta - \arctan \left(J \frac{D}{2\pi r} \right) \quad (3.2)$$

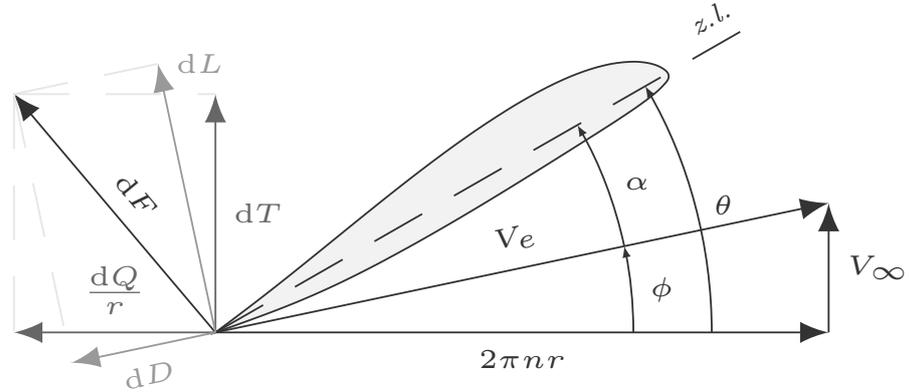


Figure 3.1: Working conditions of a blade element

is comprised between V_e and the airfoil *zero lift line* (z.l.). In equation (3.2) $D = 2R$ is the propeller diameter and the dimensionless quantity $J = V_\infty/nD$ is called *advance ratio* and is the distance advanced by the propeller in one revolution.

The airflow at the blade section produces elementary lift and drag forces, dL and dD , which, by definition, are normal to and parallel to the resultant velocity V_e , respectively. Similarly, if the elementary aerodynamic force $dF = \sqrt{dL^2 + dD^2}$ acting on the wing section is decomposed in its components parallel to and normal to the plane of rotation, each in order the elementary thrust dT and torque over r $\frac{dQ}{r}$ are found out

$$\begin{aligned} dT &= dL \cos \phi - dD \sin \phi , \\ \frac{dQ}{r} &= dL \sin \phi + dD \cos \phi . \end{aligned}$$

Finally, in this blade element context, denoting by N the number of blades, the overall forces acting on the propeller can be evaluated as N times the integral of these differential quantities over the working span of the blade, *i.e.*

$$\begin{aligned} T &= N \int_{r_h}^R \frac{dT}{dr} dr , \\ \frac{Q}{r} &= N \int_{r_h}^R \frac{1}{r} \frac{dQ}{dr} dr . \end{aligned}$$

As every aerodynamic force, the thrust T has to expressed through:

- a dynamic pressure $1/2\rho V_e^2$ proportional to $\rho n^2 D^2$;
- a reference surface, related to D^2 ;

- a dimensionless coefficient which depends on the angle of attack (and hence, through equation (3.2), on the advance ratio J) and on both the Reynolds and Mach's numbers.

Therefore, one could formally write

$$T = C_T \rho n^2 D^4$$

and, in a similar fashion

$$Q = C_Q \rho n^2 D^5 \quad \text{and} \quad P = C_P \rho n^3 D^5 ,$$

where Q is the torque and P the power. On the basis of these definitions, the *propeller efficiency* η can be introduced. This quantity is the ratio between the useful power output $V_\infty T$ and the one supplied to the propeller $P = \Omega Q$, *i.e.*

$$\eta = \frac{V_\infty T}{\Omega Q} = \frac{C_T}{C_P} J .$$

All the dimensionless coefficients here defined depend on the advance ratio J , and their typical behaviour is shown in figure 3.2 [Weick, 1931]. As J increases, the angles of attack of each blade element decrease. Since the most of the thrust T is produced by the sectional lift acting on the propeller airfoils and due to the fact that far away from the stalled regions the lift coefficient C_l is a linear function of α , the overall thrust T is a nearly linear decreasing function of J . Similarly, as the advance ratio increases, the coefficients C_P and C_Q decrease with a nearly quadratic law since they mostly depend on the sectional drag force and, for a wing section, $C_d \propto \alpha^2$.

As it can be seen in figure 3.1, the single blade element produce a positive elementary thrust dT when it generates an aerodynamic force dF which points toward the propeller advance direction. Instead, when dF lays on the plane of rotation (*i.e.* it is parallel to Ωr) the elementary thrust is equal to zero. For greater values of the advance ratio the component dT of the aerodynamic force dF becomes negative and the wing section is then acting as a brake. It should be noted that this dissertation holds for a single blade element only, even though a similar reasoning can be performed considering the integral values of the aerodynamic force F . Therefore, exploiting this concept of rotation of the F vector in the plane of figure 3.2, the fact that the thrust coefficient C_T equals zero at a lower advance ratio with respect to the power coefficient C_P can be justified: in effect, F lies on the plane of rotation at small positive values of the sectional angle of attack, whereas it is normal to the plane of rotation for negative α , *i.e.*, according with equation (3.2) for greater advance ratios J .

3.2 Main theories

Historically the detached propeller (*i.e.* the propeller isolated by its carrier body, *e.g.* an airplane) has been analyzed by means of two opposite viewpoints: the *momentum theory* and the *blade element theory*.

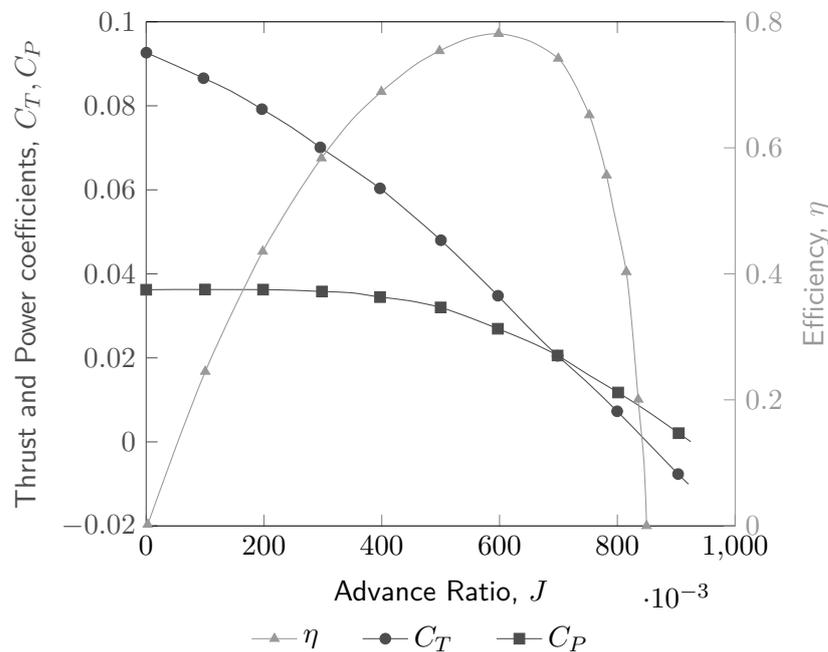


Figure 3.2: Propeller characteristic curves [Weick, 1931]

3.2.1 Momentum theory

The first viewpoint is the so called *propeller-slipstream theory* and was firstly developed by *Rankine* [Rankine, 1865] and then extended by *Robert Edmund Froude* [Froude, 1889]. This theory, also known as *momentum theory*, considers the momentum and energy of the fluid before and behind the propeller, applying the basic conservation laws of fluid mechanics to the rotor and fluid as a whole in order to estimate the rotor performance. In fact, the basic idea is that, since the rotor disk supports a thrust created by the action of the fluid on the blades, according to the Newton's law there must be an equal and opposite reaction of the rotor on the air. As a consequence, the bulk of air confined into the rotor wake is speeded up in the direction opposite to that of the thrust vector, so that the rotor has to supply an amount of kinetic energy. This energy constitutes the induced power loss of a rotary wing and corresponds to the induced drag of a fixed wing [Johnson, 1994].

In the momentum theory the propeller is modelled as an *actuator disk*: a circular surface of zero thickness located in the rotor plane, which is able to absorb the totality of the power produced by the engine and to dissipate it in order to generate a pressure jump across the disk itself, increasing hence the total head of the fluid. In the most general form of the theory [Betz, 1919, 1922], the actuator disk is able to carry not only a thrust, but also a torque, which imparts angular momentum to the fluid. However, even though the actuator disk is intended to simulate a propeller, it should be noticed that deep differences between this model and an actual rotor do exist. For instance, distributing the rotor blade loading over a disk is equivalent to considering an infinite number of blades [Johnson, 1994]. The detailed flow of an actuator disk is thus

very different from that of a real rotor with a limited number of blades. As a consequence, the actual induced power loss will be greater than that predicted by the theory due to the presence of nonuniform unsteady induced velocity. Therefore this model is mainly employed to obtain a first estimate of the wake-induced flow.

3.2.2 Blade Element Momentum Theory

The second viewpoint concerns the *blade element theory* already introduced in section 3.1 and firstly developed by *William Froude* [Froude, 1878]. This theory, unlike the Rankine's one, takes into account the geometry of the propeller blade, even though, in its original form, it did not consider the acceleration of the fluid incoming across the rotor disk.

It should be noticed that the blade element theory can be considered as an application of *Prandtl's lifting-line* to a rotary wing [Johnson, 1994]. In fact, as it has been implicitly assumed in section 3.1, in this context each blade element acts as a two-dimensional airfoil to produce aerodynamic forces, with the influence of the wake and the rest of the rotor contained in an induced angle of attack at the section. Moreover, for a rotor or propeller the vortices in the wake are trailed in helical paths rather than straight back as for fixed wings.

As said before, the blade element theory considers the phenomena on the propeller blade establishing the fact that the resistance of the blade to its direction of motion causes a loss of energy which can likewise be expressed in terms of a blade efficiency. In particular, this loss is of different nature with respect to that taken into account by the momentum theory, so that these contributes have somehow to be added and the two theories can therefore be joint into the so called Blade Element Momentum Theory (BEMT). In fact, it should be noted that, on the one hand the blade resistance affects the induced velocities in the slipstream and, on the other hand, the mutual effect of the blades on one another can best be expressed by the propeller slipstream theory. However, since for a well-designed propeller the blade drag is negligible with respect to the generated thrust, its effect on the induced velocities can generally be disregarded and hence the momentum theory can be considered as independent of the blade properties. Nevertheless, on the contrary, the results of the momentum theory are generally required for the perfect application of the blade element theory. In fact, in order to evaluate the phenomena that occur nearby the blades, the incoming velocity should be known. Even though, as depicted in figure 3.1, in the original theory this quantity was taken to be the free stream velocity V_∞ , in effect it is made by the superposition of V_∞ itself with a positive plus due to the motion imparted by the propeller, as predicted by the slipstream theory.

As it is simple to think about, the action exerted by the propeller over the surrounding fluid is composed not only by an induced axial velocity, but also by a rotational one which imparts a rotary motion to the slipstream. This additional motion implies a further loss in efficiency, especially nearby the portion of the rotor situated near the axis, where the torsional energy is relatively large. As the axial induced velocity, this rotational effect can be estimated by means of the general momentum theory formulated by Betz [Betz, 1919, 1922].

Finally, an overall Blade Element Momentum Theory sketch is reported in figure 3.3, where the axial and tangential induced velocities are denoted by $V_\infty a$ and $\Omega r a'$, respectively. Given that both these quantities are positive, a comparison with the results of figure 3.1 suggests that, if J is the same, the inflow angle ϕ is greater in the BEMT than in the original blade element theory.

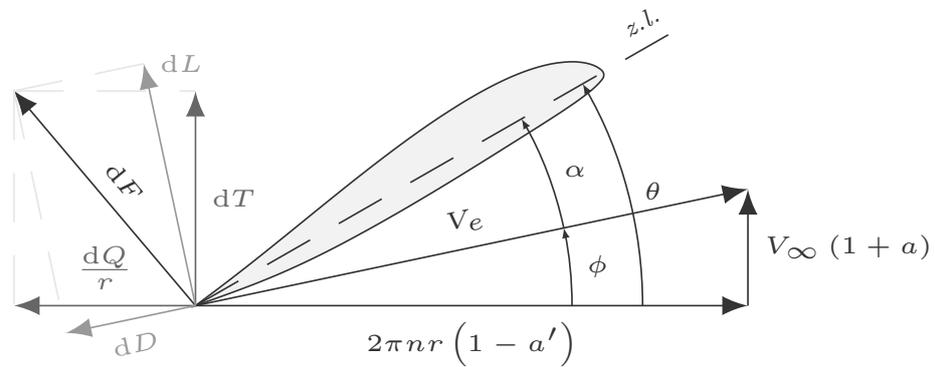


Figure 3.3: Working conditions of a blade element according to the BEMT

The fact that the BEMT takes into account the blade geometry makes it suitable not only for a detailed analysis of the performances of a particular propeller, but also to design a rotor targeting either the wanted thrust or a particular value of the available power. In both cases, this process requires in input some basic parameters, such as: the rotor diameter D , the number of blades N , the hub radius r_h , the number of revolution in the time unit n , the free stream velocity V_∞ , the operational altitude H and the spanwise airfoil distribution. This process has been carried out in section 4.2.1 on page 50 to design the propeller employed in the rest of the thesis.

Part II

Analysis & Simulation



Overview

This part leads the reader towards the simulation of a rough tiltrotor configuration – composed by a propeller and a wing – performed by means of the Vortex Particle Method. In order to achieve this task it has been dutiful to split the path in several stages, so that each argument could have been deeply treated and all the gap bridged at least with a few explanatory words. For this reason the current part, entitled *Analysis & Simulation*, is made of three different chapters:

- in section 4.1, chapter 4 firstly introduces the subject of the VPM analysis: the E-Pteron VUAV, which is obviously equipped with the tiltrotor technology. Here the overall design of the drone is presented and particular focus is given to its wing and canard. Albeit this vehicle is in the mock-up phase of his design cycle, its propeller has not been defined yet, so that in section 4.2.1 a design process has been performed and the performance of the resulting rotor have been estimated in section 4.2.2. Moreover, these evaluations have been used to assess the minimum rate of rotation required to the propeller to achieve both a forward flight and an ascent one;
- chapter 5 describes the software exploited to perform the calculations, pointing out its strengths and limitations and analyzing all its features through an insight of both the input and the output data files (see sections 5.2 and 5.3, respectively). Furthermore, in section 5.4.1 a test case has been reported, so that the numerical results could be compared to the experimental ones.
- Finally, in chapter 6 the geometries introduced in chapter 4 have been studied. In particular, in sections 6.1.1 and 6.1.2 both the propeller and the canard have been discretized in panel elements, in section 6.2 the lone rotor has been analyzed and its performance have been calculated and compared to those found out in section 4.2.2. As shown in sections 6.3 and 6.4, respectively, the last step consisted in the simulation of a bench test of the tilt phase both with and without the canard.

4 E-Pteron

In general, the design of an unmanned vehicle (especially of small dimensions) is deeply different from that related to a similar full-scale aircraft which exploits the same technologies and the same background ideas. This is due to the fact that a drone has to comply with regulations and safety levels less strict than those concerning a manned vehicle (*e.g.* under the noise production and safety perspectives). As a consequence, the key design parameters for a Unmanned Aerial Vehicle (UAV) are mission-oriented and strongly cost-dependent. For instance, in the particular case of a tiltrotor VUAV, such as the *Eagle Eye*, an issue of primary importance concerns the manoeuvrability: in fact, in a full-scale tiltrotor the control systems are governed by cyclic and collective actuators, resulting in a highly complex rotor hub and hence in great costs [Di Francesco et al., 2014], so that it may be useful to find out another solution for a drone.

4.1 Overall design

The aircraft studied in this thesis is E-Pteron: a three-propellers, electric-powered surveillance Vertical Take Off and Landing Unmanned Aerial Vehicle (VUAV) equipped with tiltrotor technology aimed to enlarge its range of applicability and to provide an effective control during the vertical flight phases of the flight envelope. This drone, able to be either remotely controlled or autonomous, has been funded by the Italian defence ministry and is currently being developed by the university of Naples "*Federico II*", the *second university of Naples* SUN and the *Caltec* consortium, responsible for the aerodynamic and structural studies, the developing of the flight control systems and the end to end design and manufacturing, respectively.

As it can be observed in figure 4.1, under the aerodynamic point of view, E-Pteron is characterized by two lifting surfaces in a canard configuration and by a total of three propellers: two fore-mounted tilting ones, located at the canard tips and one ducted immersed into the fuselage. In particular, the body-fan – composed by two counter rotating propellers – assists the tilting rotors when the drone is in helicopter mode, generating a thrust roughly amounting to the 90% of the aircraft weight thanks to an angular speed up to about 8000 rpm [Di Francesco et al., 2014]. On the other hand, the two tilting nacelles are equipped with fixed pitch propellers whose angular

Table 4.1: E-Pteron: characteristics [Di Francesco et al., 2014]

length	4,40 m
width	6,00 m
height	1,19 m
maximum takeoff weight	45,4 kg
payload	4,54 kg
operational range	92,6 km
loitering time	6 hr
cruise speed	18.00 m/s
cruise altitude	1524 m
dive speed	41.16 m/s

Table 4.2: E-Pteron: wing and canard characteristics

	b	S	\mathcal{R}	ϵ_0	Γ	Λ_{LE}	Λ_{TE}	Λ_f	λ
Wing	6,00 m	3.22 m ²	11.16	-2,88°	$\Gamma(\bar{y})$	$\Lambda_{LE}(\bar{y})$	$\Lambda_{TE}(\bar{y})$	$\Lambda_f(\bar{y})$	0.44
Canard	2,18 m	0.49 m ²	9.61	0,00°	0,00°	0,00°	-8,14°	-2,05°	0.56

speed is in the range 1500 ÷ 4000 rpm [Di Francesco et al., 2014] and whose role in hovering is mainly related to stability and control.

As suggested by the name E-Pteron itself (*e* stands for *electric*, whereas *pteron* comes from the Greek $\pi\tau\epsilon\rho\upsilon\nu$, meaning "wing") and as said before, this VUAV is fully electrical powered, as fuel cells and Lithium-Polymer batteries are hosted in the lifting fuselage, whose cross section is approximately elliptical.

E-Pteron is a tactical vehicle primarily intended for medium endurance and medium-short range patrolling missions, being characterized by a loitering time of about 6 hr and by an operational range of roughly 90 km, as summarized in table 4.1, where all the main performance parameters are reported. Moreover, the peculiar three-rotors design of the VUAV under examination, thanks to the crucial contribution played by the double counter rotating fan blades, allows for the optimization of the forward flight mode, so that a high aspect ratio is adopted for both the wing and the canard, as it can be observed both in the graphical representations in figures 4.2 and 4.4 and in the numerical data reported in table 4.2. In particular, as depicted in figure 4.3, the high-mounted wing is almost untwisted, characterized by a slight, positive and nonlinear dihedral distribution and by a nearly parabolic chord distribution along the spanwise direction which, due to the absence of Leading Edge (LE) sweep, results in a similar behaviour for the function $\Lambda_{TE}(\bar{y})$. On the other hand, as shown in figure 4.5 the canard geometry is simpler: no dihedral and no twist are present and a trapezoidal planform implies a linearly variable distribution of chords and a constant TE sweep angle.

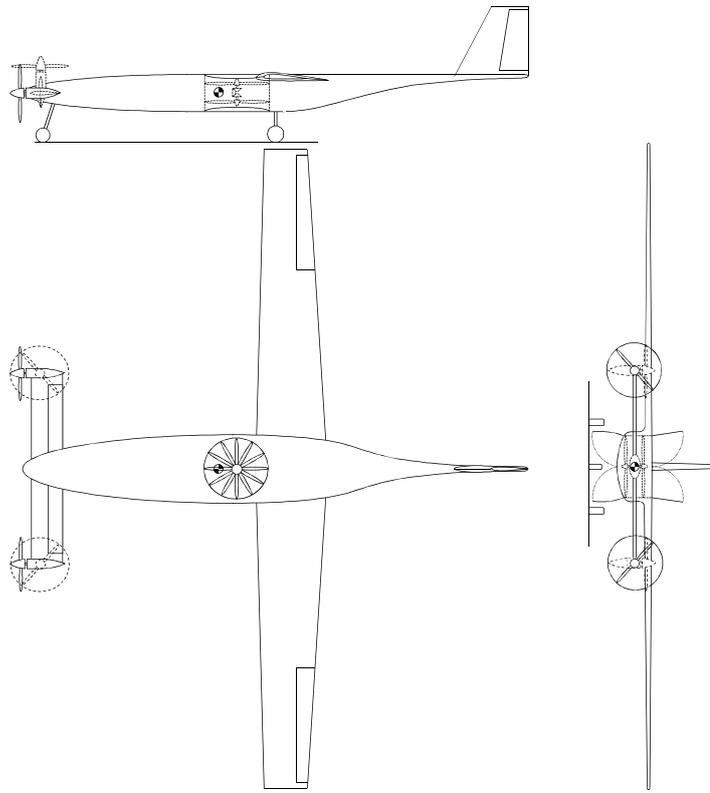


Figure 4.1: E-Pteron: three views [Di Francesco et al., 2014]

4.2 Fore propellers

Being this thesis focused on the study of the thrust evolution during the conversion phase performed by the two tilting rotors equipping the E-Pteron VUAV and the aerodynamical interaction between the propellers wakes and the canard surface, particular attention has been paid to the fore propellers.

4.2.1 Design

At the actual state of the work, only some general information regarding the propulsion unit were known, implying the need of a design process aimed at the achievement of a blade geometry suitable with the required performances. In particular, the only data available aside from those reported in table 4.1 concerned the rotor diameter, nominally amounting to 0,8 m. However, it was known that during the vertical flight, and in particular during the hover phase, the two ducted twin counter rotating rotors embedded into the fuselage are able to produce an overall thrust equal to the 90% of the aircraft weight. Therefore, in order to the hovering to be possible even for the fully loaded aircraft, at 0 m/s the thrust produced by both the two fore mounted rotors should amount to just the 10% of the maximum takeoff weight listed in table 4.1, so that, in hovering, each of the two propellers is supposed to generate a vertical thrust T_r amounting to

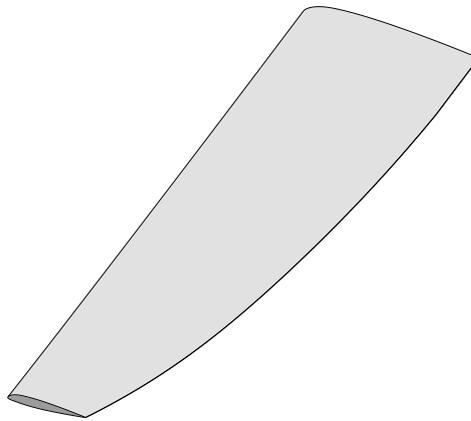


Figure 4.2: E-Pteron: isometric view of the wing

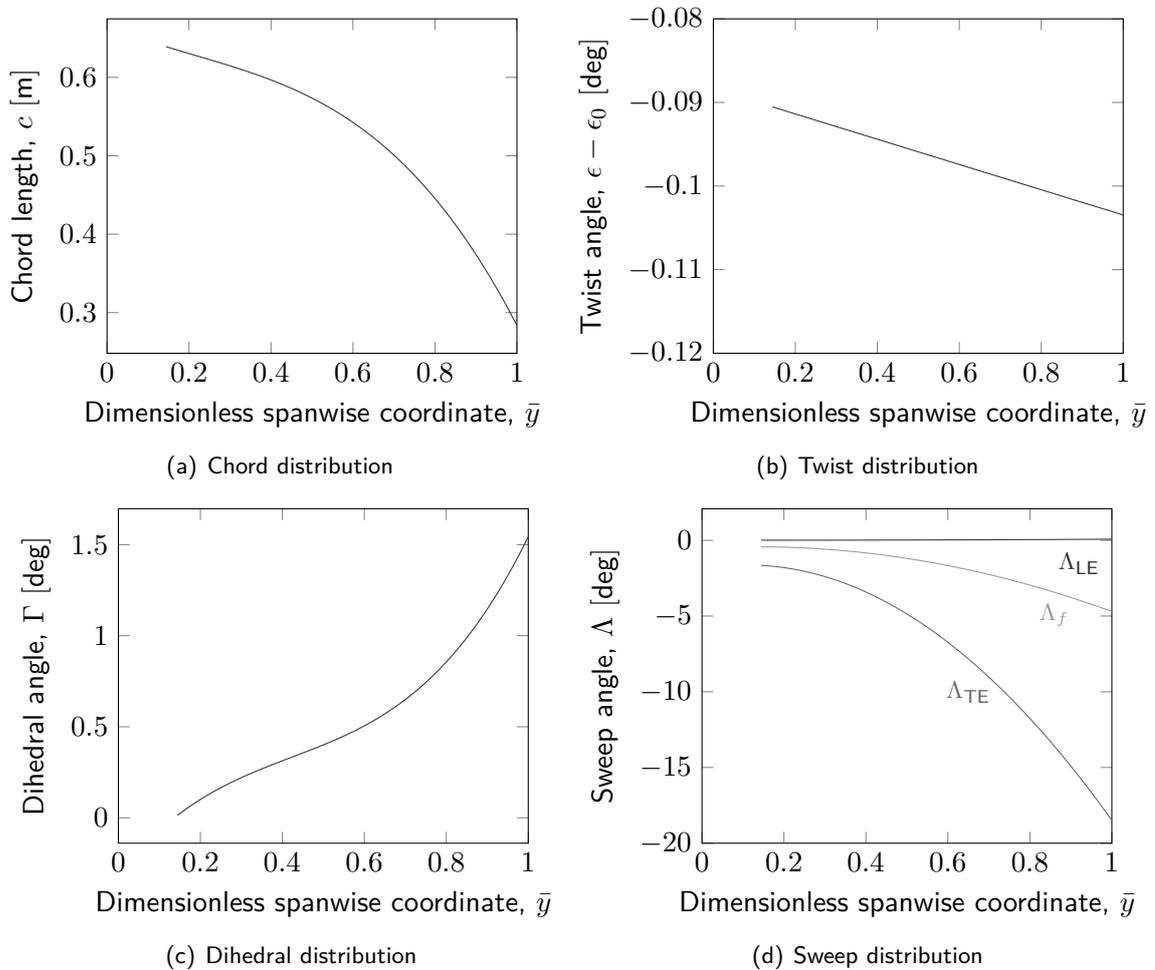


Figure 4.3: E-Pteron: wing characteristics

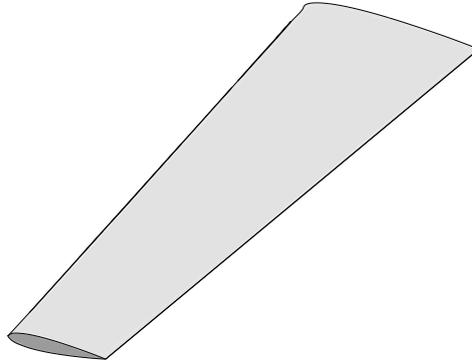


Figure 4.4: E-Pteron: isometric view of the canard

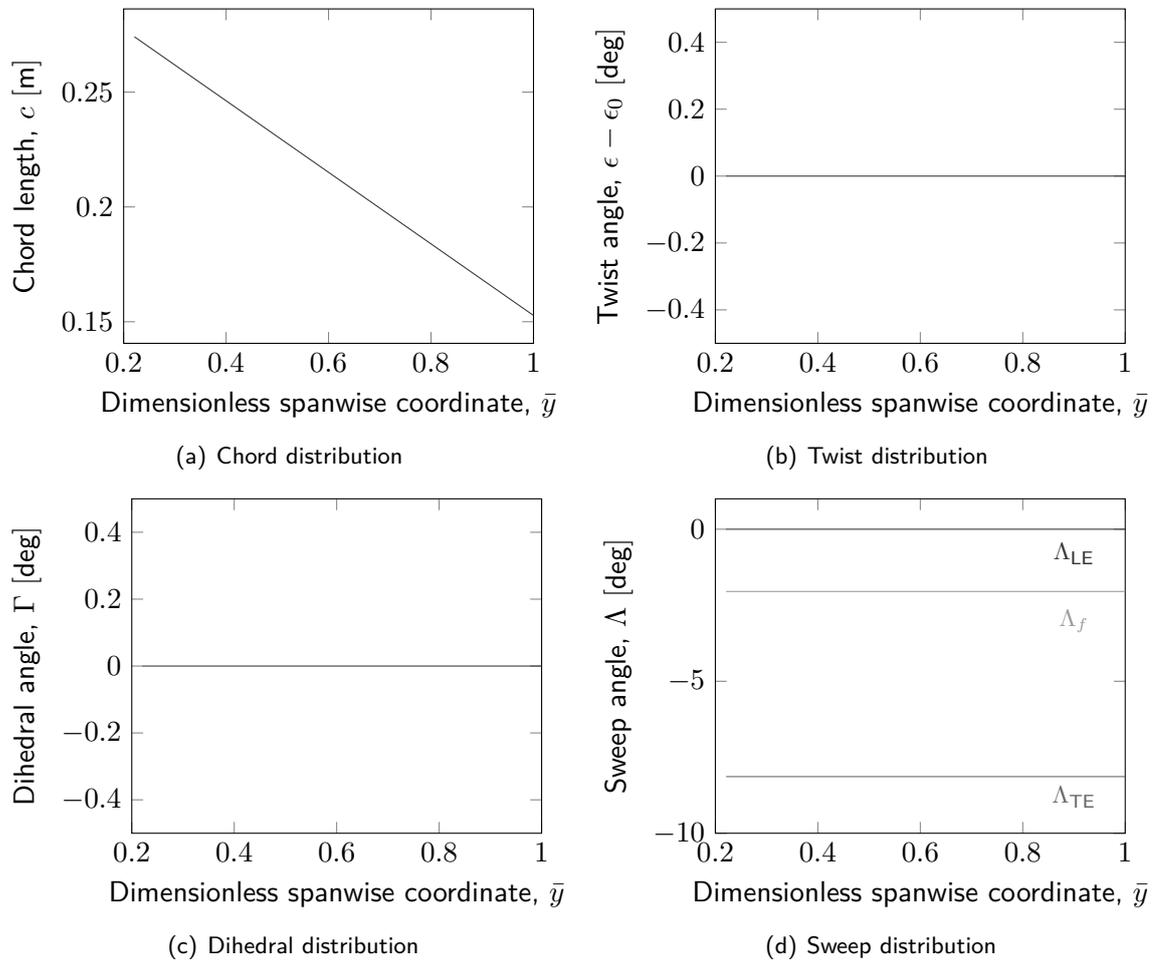


Figure 4.5: E-Pteron: canard characteristics

Table 4.3: E-Pteron: design parameters for the two fore propellers

R	r_{hub}	H	ρ	Airfoil	V_∞	n	J	C_l	T	N
0,40 m	0,04 m	S/L	1.225 kg/m ³	NACA 4412	1 m/s	3000 rpm	0.025	0.6	23 N	3

$$T_r = \frac{10\%T}{2} = \frac{1}{20}T \simeq \frac{1}{20}W = \frac{1}{20}45,4 \text{ kg}_f = 2,27 \text{ kg}_f = 22,28 \text{ N} .$$

Nevertheless, since the low values of the advance ratio J do represent a criticality for the BEMT (employed in this phase), the design airspeed has been shifted from the hovering 0 m/s to a slow ascent value of 1 m/s. Furthermore, this change in the design parameters implies the presence of a safety margin, since typically the function $T(V_\infty)$ is decreasing for increasing values of the airspeed V_∞ . Similarly, the computed value of the thrust T_r has been rounded to 23 N. Moreover, considering that the propellers angular speed is in the range 1500 ÷ 4000 rpm [Di Francesco et al., 2014], an average design speed of 3000 rpm has been assumed.

Taking into account the preliminary data here calculated, the design process has been carried out by means of the software *Xrotor* [Drela, 2001, web] at Sea Level (S/L) and adopting a constant C_l distribution along the blade. Regarding the number of blades N , this parameter has been set to three in agreement with a common choice in both commercial and military manned tiltrotors (e.g. *AugustaWestland AW609* or *Bell-Boeing V-22 Osprey*). A summary of the design parameters here adopted is reported in table 4.3.

The geometrical characteristics of the blade produced as *Xrotor* output are both listed in table 4.4 and plotted in figure 4.6 in terms of distribution of chords and twist angle, as functions of the radial coordinate r non dimensionalized with respect to the rotor radius R . Furthermore, a three-dimensional graphical representation of the designed propeller is shown in figure 4.7.

From figure 4.6(b) it should be noted that the twist distribution along the blade is nearly hyperbolic, according to an approximated optimum condition related to the Blade Element Momentum Theory [Tognaccini, 2011]. In fact, assuming a poorly loaded propeller with high solidity (i.e. $N \gg 1$) so that both the wake contraction and the radial flow nearby the blade tips can be neglected, and considering an inviscid flow (i.e. not taking into account the viscous losses), the inflow angle ϕ is given by

$$\tan \phi = \frac{V_\infty}{\Omega r} \frac{1 + a}{1 - a'} ,$$

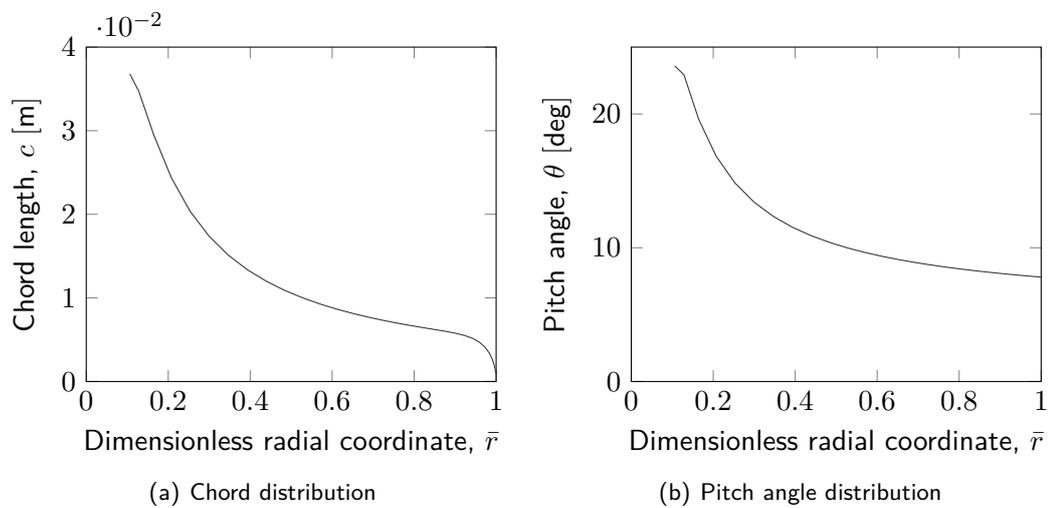
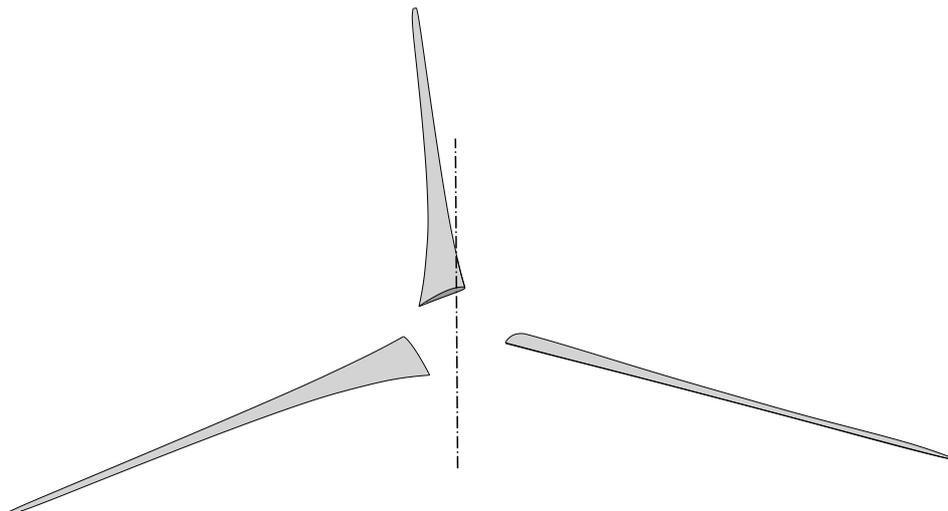
where a and a' are the axial and rotational inductions, respectively. For a slow propeller (i.e. for $J < 1$) as the one into analysis, both a and a' can be assumed to be constant along the wingspan [Tognaccini, 2011], hence $\phi \propto 1/r$. Moreover, if the blade is made of the same airfoil at each radial position, then at the design point the angle of attack α is nearly coincident with its ideal value α_i and thus it is also constant along the blade. As a consequence, for the distribution of the pitch angle $\theta(r)$, the

$$\theta = \alpha + \phi \simeq \alpha_i + \phi \propto \frac{1}{r}$$

holds and a hyperbolic behaviour has been found out.

Table 4.4: E-Pteron: fore propellers geometrical characteristics

		1	2	3	4	5	6	7	8	9	10
r	[m]	0.043	0.083	0.139	0.194	0.245	0.291	0.330	0.361	0.384	0.396
\bar{r}	[-]	0.106	0.207	0.347	0.485	0.614	0.728	0.826	0.903	0.959	0.991
c	[m]	0.037	0.024	0.015	0.011	0.009	0.007	0.006	0.006	0.005	0.003
θ	[deg]	23.592	16.836	12.339	10.386	9.347	8.725	8.329	8.071	7.911	7.826
σ	[-]	0.413	0.140	0.052	0.027	0.017	0.012	0.009	0.008	0.006	0.003

**Figure 4.6:** Geometrical characteristics of the adopted blade**Figure 4.7:** Isometric view of the designed propeller

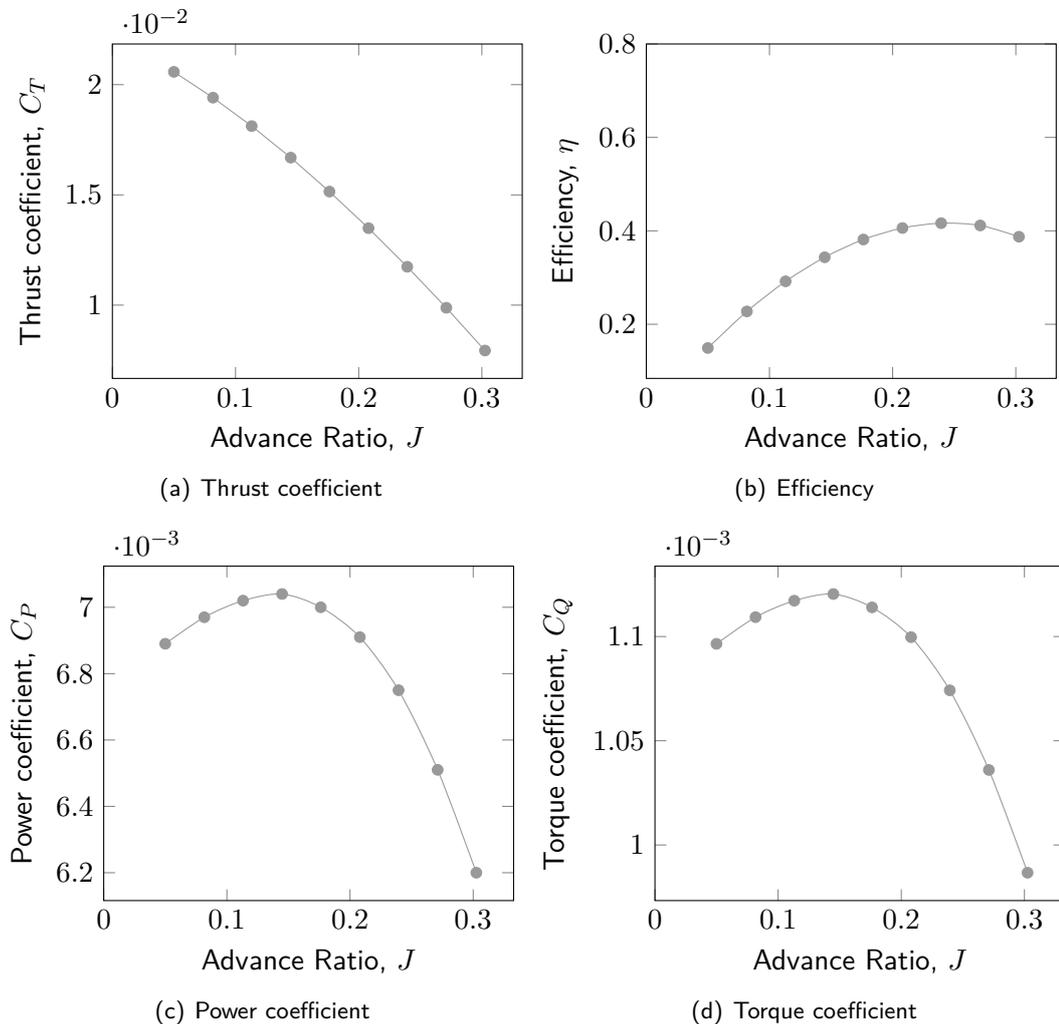


Figure 4.8: Propeller performances

4.2.2 Performance

The off-design performances of the propeller designed in section 4.2.1 have been estimated by means of the BEMT code *Xrotor*, covering a wide range of operational conditions in terms of both airspeed and rate of rotation (and hence of advance ratio J).

At first, this analysis led to the evaluation of the characteristic curves reported in figure 4.8, where the range of advance ratios on the x axis is lower bounded by the estimation of the sectional drag coefficient C_d and upper bounded by the operational range of the propeller. In fact, at higher J the rotor ceases to be a propulsion system and starts to behave like a brake, compromising its own purpose.

Apart from the dimensionless curves of figure 4.8, of particular interest to the aim of this work are the dimensional data reported in figure 4.9. Here the function $T(V_\infty)$ – evaluated on

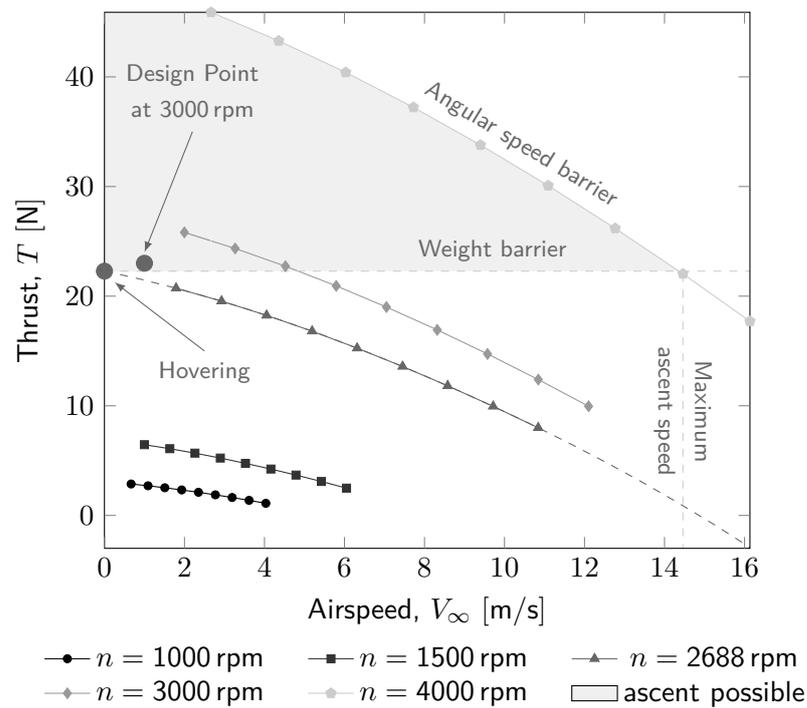


Figure 4.9: Propeller dimensional performances for different angular speeds. The data have been calculated with *Xrotor*

the basis of the characteristic curve 4.8(a) – is plotted for different discrete values of the angular speed compatible with the working range of the E-Pteron fore propellers. In particular, on this graph the design point of table 4.3 is also highlighted: it does not lie on the curve at 3000 rpm since the design is based on a standard airfoil, whereas the curves in figures 4.8 and 4.9 have been calculated considering the specifications of a NACA 4412 airfoil [extracted from Abbott and von Doenhoff, 1959]. Moreover, assuming that the maximum angular speed for the propellers into analysis amounts to 4000 rpm and neglecting the viscous effects, in figure 4.9 two boundaries which limit the flight envelope are also shown: an angular speed barrier and a weight one. The first of these limits is both related to the blade geometry and to structural and power issues which bound the propeller maximum rate of rotation and is such that the triads T - V_∞ - n which lies above this curve do not belong to the flight envelope, since they are not physically realizable. On the other hand, the weight barrier is the locus of the points where the weight is balanced by the thrust and hence where a constant speed ascent (*i.e.* the body forces are absent) is possible. For E-Pteron, this latter barrier depends only on the takeoff weight, being zero the fuel consumption due to the electrical nature of the power supply. Moreover, given that below the weight barrier the developed thrust is not enough to balance the gravity force and that the points on the right of the angular speed limit are not physical, a maximum ascent speed does exist at the intersection between these two curves: for the maximum takeoff weight this velocity amounts to 14.46 m/s. In a similar fashion, the intersection between the weight barrier and the zero-velocity line (*i.e.*

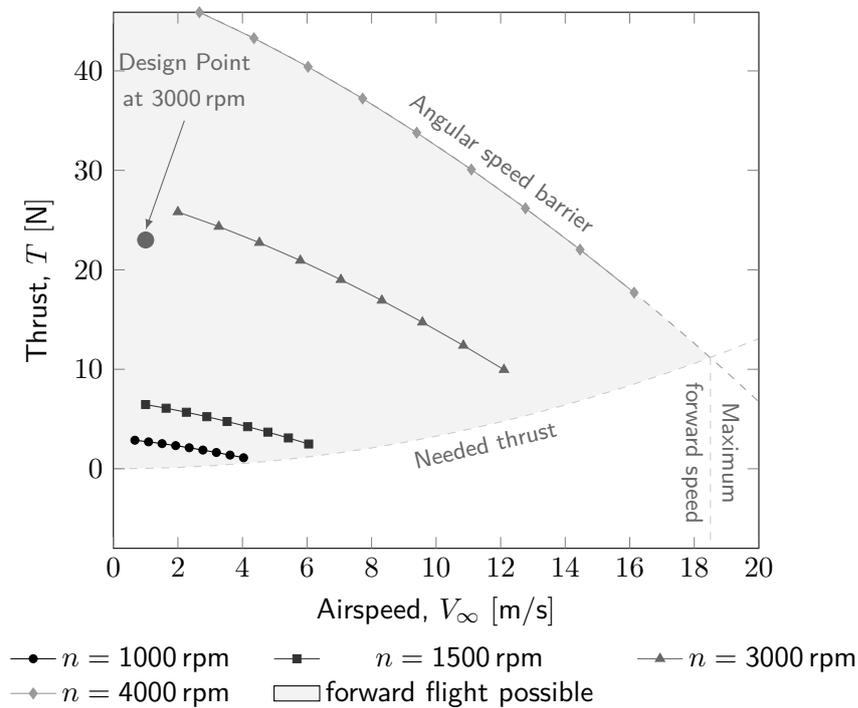


Figure 4.10: Propeller dimensional performances in forward flight. The data have been calculated with *Xrotor*

the y axis) identifies the only combination of thrust, airspeed and angular speed which makes the hovering possible. Obviously, the location of this point on the $V_\infty - T$ plane depends on the aircraft loading and for the maximum takeoff weight it is verified for $n = 2688$ rpm.

Similarly, in figure 4.10 the limitations concerning a balanced forward flight are reported. In this plot the already described angular speed barrier is sketched together with the curve representing the thrust needed to balance the aerodynamic drag of the whole vehicle. This curve holds for zero angle of attack (*i.e.* for horizontal fuselage) only and has been carried out by means of the software Digital DATCOM+ [Holy Cows, Inc., 2002, web] (for the full input file see listing B.1 on page 118), that has been exploited to obtain an estimation of the E-Pteron drag coefficient C_D for a zero thrust condition. It should be clear that all the points lying on the needed thrust curve are characterized by a perfect balance between the produced thrust and the current drag force, therefore the forward flight is sustainable only in the half-plane located on the upper side of this barrier. On the other hand, the triads $V_\infty - T - n$ belonging to the lower half-plane can be only related to a decelerating motion, since here the drag force overcomes the available thrust. A consequence of the simultaneous presence of this limitation and of that related to the maximum rate of rotation of the fore propellers is the existence of a maximum speed for the forward flight: with zero angle of incidence this velocity, given by the intersection between the two cited curves, amounts to 18.5 m/s and hence it is just slightly different (2.78%) from the declared cruise speed (*i.e.* 18 m/s). Furthermore, at this purpose it should be noticed that the calculations concerning

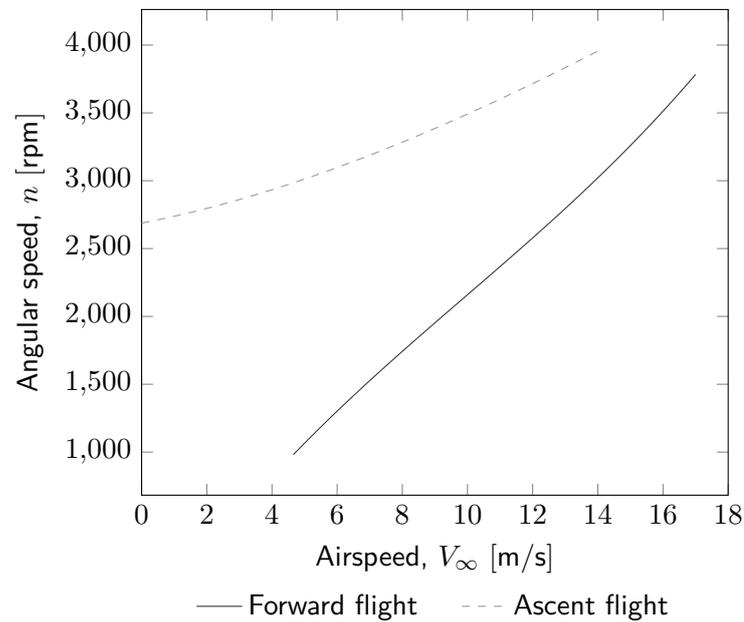


Figure 4.11: Minimum angular speed for forward and ascent flight at constant velocity. The data have been calculated with *Xrotor*

figure 4.10 have been performed assuming a S/L altitude, whereas the cruise speed is estimated at about 1500 m, where the air density is lower and hence where the needed thrust is expected to be lower, too.

Finally, in figure 4.11 the minimum number of revolutions required for both a constant speed forward flight and ascent flight is reported as a function of the free-stream velocity V_∞ . These curves have been calculated as the intersection of the functions $T(V_\infty, n)$ reported in both figures 4.9 and 4.10 with the weight barrier of figure 4.9 and the needed thrust curve of figure 4.10, respectively. As expected, these curves show that, in the same airspeed conditions, the vertical flight is always more costly than the horizontal one in terms of required rate of rotation of the two fore propellers and hence of needed power.

5 Panel Method Solver

“*There* is something irreversible about acquiring knowledge; and the simulation of the search for it differs in a most profound way from the reality. ”

— *J. Robert Oppenheimer,*
Physics in the Contemporary World, 1949

PaMS [Caccavale, 2008, web] is a 3D low-order unsteady Boundary Element Method for the solution of ideal flows (inviscid, incompressible and irrotational) with a numerical methodology based on the third *Green's Identity* and the *Morino's formulation*. The solver allows to perform both single and multibody simulations in either internal or external flow conditions. Furthermore this code is able to deal with bodies relative motion and with the body-wake aerodynamic interaction: these capabilities, joint with the short computational time required, make PaMS the ideal software solution to afford complex problems such as the conversion phase of a marching tiltrotor.

5.1 Generalities

PaMS has been developed by *Paolo Caccavale* [Caccavale, 2008, web] in FORTRAN 90 in order to have a powerful, flexible and almost costless analysis tool whose scope encompasses a variety of aeronautical and naval unsteady calculations. In fact, for instance, the possibility to deal with multibody or morphing geometries allows PaMS to work side by side with a structural analysis software with the common aim to solve a complex fluid-structure interaction problem. This is possible thanks to the modularity philosophy which lies behind the code, that can be integrated with external routines in order to achieve complex tasks such as, for example, the evaluation of the aerodynamic forces on an airplane during the execution of a particular manoeuver. Moreover PaMS is available both in a serial and in a – shared memory only – parallelized version (both of which exploit the dynamic memory allocation), so that the computational effort related to

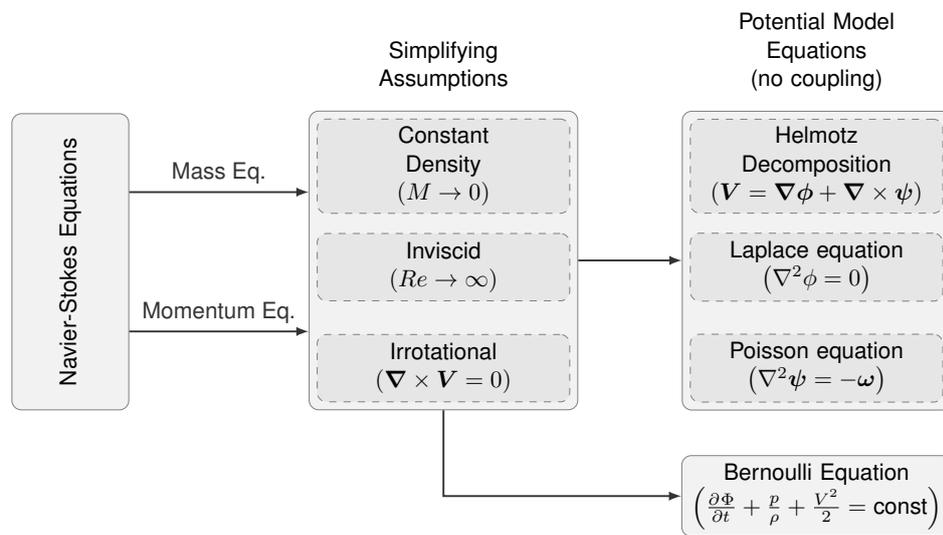


Figure 5.1: PaMS governing equations [Caccavale, 2008, web]

the composition of the equations system and to its solution can be distributed among different processing units hosted on a single machine. However not all the tasks accomplished by the software can be solved by means of parallel computing (which, when it is possible, is realized through the OPENMP API [OpenMP ARB, 1997, web]).

Finally, the physical and mathematical core of PaMS is unveiled in figure 5.1, where the simplifying hypothesis and the consequent problem are pointed out.

5.2 Input Data Files

To perform the requested calculations, PaMS requires at least two different input files: the *geo* file and the *DATAIN* one. The first one comes with geometry-related information and contains data regarding its discretization into panels, whereas the latter provides end-to-end data concerning almost all the perspectives of the analysis to be performed: from the free stream conditions to the convergence check, from the motion imposed to the output to be generated.

Even though these two input data files are the only ones required by the solver in order to compute the solution, sometimes a greater control over some motion-related parameters is needed. This task can be achieved exploiting the modularity of PaMS code, which can be extended with external routines providing additional features.

A detailed description of the input data files here introduced is given in sections 5.2.1 to 5.2.3.

5.2.1 The *geo* file

Information concerning both the geometry and the body discretization into panels are provided by means of at least one *geo* file, whose structure is divided in two different sections:

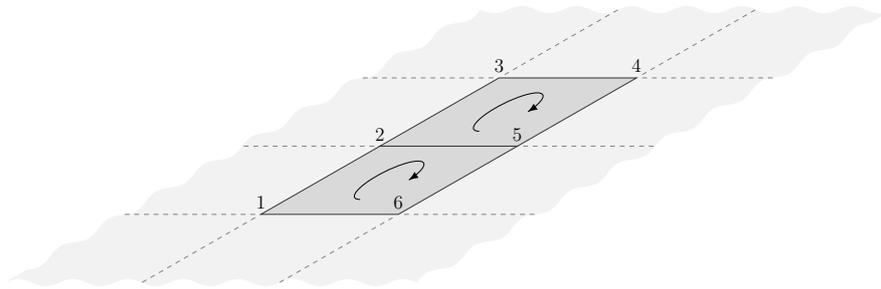


Figure 5.2: Example of discretization in PaMS

- **GRIDP** – this portion of the *geo* file is dedicated to the *grid points*, which describe the geometry through a set of three-dimensional cartesian coordinates x, y, z . In particular, the coordinates regarding each point – describing the geometry in its own reference frame – should be listed on different rows, employing a blank space as field separator and a newline character at the end of a set of a single x, y, z triad.

The points included in the GRIDP section do not require an explicit label to be referred, since each point can be addressed by means of its position into the *geo* file. That is to say that the first point of the list (*i.e.* the one located in the second row of the file) will be pointed by the number 1, the second one (*i.e.* that occupying the third row) by the number 2 and so on.

- **PANEL** – this section follows the GRIDP one and contains all the information needed to discretize into panels the geometry described in the first section of the *geo* file. In particular, in this file each panel is identified by its vertices, which have to be present in the grid points list. In this case a single vertex is addressed by its labelling number, as stated before. Moreover, since PaMS is able to deal with both triangular and quadrilateral panels, each row of the PANEL section should contain three or four space-separated grid point labels and should be terminated by a newline character.

For instance, in order to describe the geometry depicted in figure 5.2, a *geo* file like the one reported in listing 5.1 should be employed.

Listing 5.1: Example of *geo* file in PaMS related to figure 5.2

```

GRIDP
0.0 0.0 0.0
0.0 1.0 0.0
0.0 2.0 0.0
1.0 2.0 0.0
1.0 1.0 0.0
1.0 0.0 0.0
...
PANEL
    1      2      5      6
    2      3      4      5

```

...

Moreover, figure 5.2 and listing 5.1 also show that the panels listed in the *geo* file have to be characterized by a common orientation of the normal vectors, that is to say that the edges shared by two adjacent panels should be travelled in two opposite directions.

Furthermore, it should be noticed that the *geo* file (that, starting from a surface mesh created through any preprocessor, can be generated by an automated PaMS converter) does not contain any information about the TE panels (*i.e.* those which require the enforcement of the Kutta condition), since these closure requirements are demanded either to the DATAIN input file or, for complex cases, to a third input file.

5.2.2 The DATAIN File

In listing C.1 on page 121 a typical input file for PaMS is reported. It should be noticed that this file has to be named DATAIN in order to be recognized by the solver as a valid PaMS input file.

The file DATAIN is composed of different *namelists* starting with an ampersand symbol followed by a particular keyword and ending with the &END keyword. Furthermore this file is organized in more than one logical section, each of which encompasses several characteristic keywords, so that it is possible to distinguish among:

- far-field conditions;
- unsteadiness management;
- convergence check and solution speed up;
- body and motion-related parameters;
- output handling;
- geometry and closure;
- body motion.

The following paragraphs provide a detailed description of each element in the list, supplying information about their peculiar keywords.

Far-Field conditions

Listing 5.2: Typical DATAIN file for PaMS - Far-Field conditions

```
-----
&INFTY    velinf(1,1)=0.0    velinf(2,1)=0.0    velinf(3,1)=0.0    &END
&INFTY    omginf(1,1)=0.0    omginf(2,1)=0.0    omginf(3,1)=0.0    &END
&INFTY    density( 1)=1.225    stpress( 1)=0.0    soundsp( 1)=340.3    &END
-----
```

The first logical section by which the DATAIN file is composed of concerns the far-field conditions. For the sake of clarity the snippet of code into analysis is reported in listing 5.2. Here data about asymptotic values of velocities, density (`density`), gauge pressure (`stpress`) and sound speed (`soundsp`) are specified in SI units within the INFTY namelist.

In particular, the velocity (both translational – `velinf` – and rotational – `omginf`) is stored into a matrix-like structure whose dimensions are up to 3×2 . In fact each column of this data structure contains the components of the vectorial quantity along the three axes of the cartesian global reference frame related to the actual fluid phase. PaMS is indeed able to manage two different species in a wave motion multiphase (air-water) problem. These argumentations can be efficaciously expressed in mathematical notations, as

$$V_{\infty} = \begin{pmatrix} \text{velinf}(1,1) & \text{velinf}(1,2) \\ \text{velinf}(2,1) & \text{velinf}(2,2) \\ \text{velinf}(3,1) & \text{velinf}(3,2) \end{pmatrix} = \begin{pmatrix} u_1 & u_2 \\ v_1 & v_2 \\ w_1 & w_2 \end{pmatrix},$$

where the subscripts 1 and 2 refer to two different phases. Obviously this dissertation can be also extended to the other - scalar - parameters belonging to this section: in this case the matrix-like structure is reduced to a vector whose dimensions are up to 1×2 . Due to the single-phase nature of the test case into analysis, in listing 5.2 all the parameters of the INFTY namelist are characterized only by the phase index 1.

Moreover, it should be pointed out that, although nearly every input data is expected to be in SI units, some exceptions do exist. In particular, as all the other angular speeds reported in the DATAIN file, the free-stream rate of rotation `omginf` has to be expressed in rpm.

About the gauge pressure `stpress`, it should be underlined that this value is used just as a reference, therefore if it is set to zero (as in listing 5.2), the computed pressure has to be seen as the difference $P - P_{\infty}$ and could hence be negative wherever an expansion is located.

Unsteadiness Management

Listing 5.3: Typical DATAIN file for PaMS - Unsteadiness management

```
-----
&TIME      dt=0.003           tstart=0.0           tend=0.300           &END
&TIME      nsubi=2
-----
```

Since PaMS is able to perform unsteady analyses, some time-related parameters have to be set up by means of the TIME namelist:

- `dt` – the time step, measured in seconds;
- `tstart` – the initial time (in the current PaMS versions, this parameter has to be equal to zero);
- `tend` – the final time of the simulation, measured in seconds;

- `nsubi` – number of sub iterations: considering (not rigorously) a time step as an iterative process, every sub-loop located inside a time step itself could be named a sub-iteration. In particular this parameter concerns a refinement in the wake definition, since it specifies the number of iterations intended to achieve a better prediction of the shape of the first row of wake panels (*i.e.* the one nearest to the wing TE), so that the Kutta condition can be enforced in a more efficacious way. In fact, whether `nsubi` is set to one, then the problem is assumed to be linear and the first row of wake panels is totally defined by the first attempt performed. However, as seen in section 2.3, the wake geometry (and in particular the panel row used to enforce the Kutta condition) strongly affects the solution in terms of both lift and drag, so that a first order solution is not advised and the minimum recommended value for `nsubi` is two, as in listing 5.3.

Convergence check and solution speed up

Listing 5.4: Typical DATAIN file for PaMS - convergence check and solution speed up

```

-----
&INVERTER omegainv=0.0          convinv=1.E-6          maxiter=100          &END
&INVERTER maxblkiter=5         nblkmin=1             mfile='Y'            &END
-----
&CUTOFF   RFF(1)=1.E+9         RFF(2)=1.E+9          &END
-----

```

As shown in listing 5.4, the solution convergence and speed up issues are managed by means of the `INVERTER` and `CUTOFF` namelists, respectively. In particular, the following control parameters can be handled:

- `omegainv` – as many other solvers of elliptical problems, in order to solve an equations system in the form $\underline{A}\mathbf{x} = \mathbf{f}$, PaMS employs the iterative method known as Successive Over Relaxation (SOR), so that a scalar relaxation factor ω is required, according with the [see Hirsch, 2007; Meola and de Felice, 1996]

$$\mathbf{x}^{n+1} = -(\omega \underline{D} + \underline{L})^{-1} (\underline{A}_2 \mathbf{x}^n - \mathbf{f})$$

where $\underline{A}_2 = \underline{A} - (\omega \underline{D} + \underline{L})^{-1} (\underline{A} \mathbf{x}^n - \mathbf{f})$ and \underline{D} is a diagonal matrix whose nonzero elements are coincident with those occupying the same positions in \underline{A} and \underline{L} is a matrix which has as nonzero elements the ones located below the main diagonal of \underline{A} . In this context `omegainv` = ω and when `omegainv` = 0, since in this case there would be no iteration at all, the technique is switched to a *biconjugate gradient stabilized method* [van der Vorst, 1992]. In general, it could be proved [Meola and de Felice, 1996] that a necessary condition for the convergence of the SOR method requires $0 < \omega < 2$, so that `omegainv` should belong to this range. Moreover, since typically the optimum value for ω is in the range $[1, 2]$ [see for example Carré, 1961; Reid, 1996], whether the user wants to employ the SOR algorithm, in order to increase the performances in terms of convergence speed it is suggested that $1 < \text{omegainv} < 2$.

- `convinv` – value of the residual to the convergence: to contain the error propagation, this parameter should be not greater than 10^{-5} ;
- `maxiter` – the maximum number of iterations to be performed: the iterative process stops whether at the `maxiter` iteration the `convinv` threshold has not been reached.

In order to save RAM (but with a typical loss of performance), PaMS allows to solve the linear system by splitting its matrix into different blocks. This method can be user controlled by means of the the following keywords of the `INVERTER` namelist:

- `nblkmin` – minimum number of blocks in which the linear system is split up;
- `maxblkiter` – maximum number of iterations to be performed on each block;
- `mfile` – a char toggle which, if activated by assigning to it the 'Y' (Yes) value, makes PaMS to write the whole matrix to an auxiliary file. In this way, whenever it is requested, each block can be recalled without the need to be recomputed. Typically it is convenient to enable this option, because the reading/writing time is lower than the one needed to reevaluate all the matrix coefficients.

In this scenario each block is solved exploiting the information provided by the whole `INVERTER` namelist: the iterative process ends either whether the `convinv` requirement is met for all the blocks or if the `maxblkiter` limit is reached. Moreover, if the chosen `nblkmin` is not compatible with the available RAM (*i.e.* it is too low), PaMS automatically increases its value until the available memory is sufficient to carry out the calculation.

Finally, the `CUTOFF` namelist provides the keyword `RFF`, which is a vector of two elements:

- `RFF(1)` denotes the distance (expressed in meters) above which the point P is far enough from the center of the inducing panel k so that the panel itself can be considered as a point due to the induction strength decay;
- `RFF(2)` represents the distance (expressed in meters) above which the point P is far enough from the center of the inducing panel k so that the influence of this panel over P can be neglected. From these considerations it should be clear that $\text{RFF}(2) \geq \text{RFF}(1)$.

Therefore, denoting by r the modulus of the position vector of P with respect to the k^{th} panel, the role played by the parameter `RFF` is clarified in table 5.1.

Moreover it should be pointed out that the `RFF` parameter influences the time spent to set up the system matrix because, obviously, the calculation of the induction due to a point singularity is less costly (and hence faster) than that related to an equivalent panel. Finally, it should be needless to say that the gain in computational speed is even greater if the point has not to be considered at all.

Table 5.1: Summary of the RFF keyword usage

$r < \text{RFF}(1)$	k induces like a panel
$\text{RFF}(1) < r < \text{RFF}(2)$	k induces like a point
$r \geq \text{RFF}(2)$	the induction of k over P can be neglected

Body and Motion-related Parameters

Listing 5.5: Typical DATAIN file for PaMS - Body and Motion-Related Parameters

```

-----
&BODY      nbody =1          relmot='N'          symplane=0          &END
&BODY      cbdgap=0.0
-----
&MOTION    nsubimot=2      onlymot='N'          &END
-----

```

This section of the DATAIN file is employed to handle the multibody analyses performed by means of PaMS in terms of both number of bodies and their possible relative motion. At this purpose two different namelists are defined: BODY and MOTION. In particular the BODY namelist comes with the following keywords:

- `nbody` – an integer denoting the number of *geo* files to be imported into the current analysis. This parameter has to be at least equal to one and, obviously, the project folder has to contain a number of *geo* files equal (or greater) than the value assumed by `nbody`.
- `relmot` – a flag of char type that rules the relative motions among the panels. At first it should be noticed that this kind of relative motion is not verified for multibody analyses only, but for single body morphing geometries, too. In particular, this parameter rules the body related influence matrix updates: in fact, when `relmot='N'` this matrix is never recomputed (and other parameters concerning this issue are out ruled). Therefore, whether `nbody` has been set to one and if the geometry is not changing in time, than `relmot` has to be set to 'N' (No), otherwise it could be set to 'Y' (Yes);
- `symplane` – an integer taking into account the presence of symmetries in the problem under examination. If this is not the case, then `symplane` should be equal to zero, otherwise it may assume values in the range $1 \div 3$, accordingly with the only nonzero component of the normal to the plane of symmetry (that has to be coincident with one of the three reference planes). Whether, for instance, the symmetry plane has its unit vector towards the *y* axis (*i.e.* it is the *xz* plane and its normal vector is $(0, 1, 0)$), then the user should set `symplane=2`.

This symbology can be easily extended to a case where two different symmetry planes cohabits: in fact the `symplane` parameter accepts as input the combinations 12, 13, 23, too. At this stage it should be easy to understand that, for example, the expression `symplane=12` refers to a problem where both the *yz* and the *xz* planes are of symmetry and consequently

in this situation just a quarter of the model (hosted in two adjacent octants of the three-dimensional space) is simulated.

Finally, even though it should be clear, it is dutiful to emphasize that the symmetry here considered concerns the whole flow-field and not only the geometry. Therefore, in this context an aircraft with two co-rotating propellers in forward flight and a vehicle cruising with a lateral wind are two examples of non-symmetrical problems;

- `cbdgap` – since (through the *common body index* `icb` of the *Geometry and Closure* section of the DATAIN file) PaMS allows to consider more than one geometry (imported by means of different *geo* files) as one rigid entity, the gap between these sub-geometries has to be managed: here comes the `cbdgap` (*i.e. common body gap*) parameter. This global variable represents the minimum distance (expressed in meters) which can separate two sub-geometries sharing the same *common body index* without PaMS bridging the interested nodes. When `cbdgap=0.0` this check is skipped and the geometries are kept as separated despite their small reciprocal distances. At this stage the careful reader could have reckoned that the `cbdgap` keyword may result useful to deal with the fuselage-wing intersections;
- `nsubimot` – given that the simulation time step (ruled by the `dt` parameter of the TIME namelist) is not always suitable to handle the body motions, the keyword under examination allows to split it up in `nsubimot` sub-intervals. In other words, between two subsequent time steps, the relative positions of the different bodies is updated performing `nsubimot` sub-iterations. In particular, this parameter should be greater than unity whenever the simulation involves multiple rotations around different axes and if the simulation time step is relatively large. In fact in this case reducing `dt` by setting `nsubimot=1` results in a restrained positioning error, but on the other hand it implies an increased overall simulation time.

It is clear that when dealing with complex multibody analyses, whose solution is computed in a relatively long time, it may be expensive to reevaluate the whole solution due to an error occurred during the motion setup. For this reason, through the `onlymot` parameter of the MOTION namelist, PaMS is equipped with the possibility to only simulate the relative motion between the considered bodies, without resolving the flow-field and hence saving time to the user. In particular, as it is easy to think about, this parameter accepts the values 'Y' and 'N', respectively standing for *Yes* and *No* and implying each in order the simulation of the motion only and the solution of the problem. As a consequence for multibody analysis it is advised to firstly set `onlymot = 'Y'` and, whether the result is the one expected, to switch to the case with `onlymot = 'N'`.

Finally it should be pointed out that when in a multibody analysis two or more bodies permeate each other, PaMS automatically gets the rid of the panels internal to another body.

Output Handling

This section of the DATAIN file is used to manage the output produced by PaMS. Before going into the detail, it should be pointed out that the generated output is intended to be analyzed through the postprocessing software `tecplot 360EX` and it should be recalled that PaMS is able to

work side by side with external software packages like *Nastran* and *Patran*. For these reasons many of the output handling options provided by the FILEOUT namelist concern these applications:

- `fofreq` – File Out Frequency: a positive integer denoting the time step interval separating two subsequent output reports. In other words the output information are generated every `fofreq` time steps;
- `tecplotgp` – a flag of char type which, if 'Y' (Yes) enables the output reporting (in tecplot 360EX format) in the grid points;
- `tecplotpn` – similar to `tecplotgp`, this is a flag of char type which, if 'Y' (Yes) enables the output reporting (in tecplot 360EX format) in the panel centers;
- `tecplottm` – if 'Y' (Yes) all the output generated every `fofreq` time steps is merged in one tecplot 360EX file, otherwise multiple files are produced;
- `patran` and `patrantm` – similar to `tecplotgp` and `tecplottm`;
- `tecplot3d` and `tecplot3dtm` – these parameters follow the same logic of the previously analyzed ones. In particular `tecplot3d` allows to compute the solution in both internal and external points with respect to the bodies, according to the specifications of the OUT3D namelist;
- `sfo` – Single File Output: if enabled through the 'Y' (Yes) value, this option (available in Linux environment only, compresses all the output folders in one zipped archive, whose name (comprehensive of the `.tgz` extension) is specified by the `sfname` keyword.

Listing 5.6: Typical DATAIN file for PaMS - Output Handling

```
-----
&FILEOUT  fofreq    = 1                                &END
&FILEOUT  tecplotgp = 'Y'      tecplotpn = 'Y'      tecplottm = 'Y'  &END
&FILEOUT  tecpress  = 'N'                                tecpresstm = 'N' &END
&FILEOUT  tectreff  = 'Y'                                tectrefftm = 'Y' &END
&FILEOUT  patran    = 'N'                                patrantm   = 'N' &END
&FILEOUT  tecplot3d = 'N'                                tecplot3dtm = 'N' &END
&FILEOUT  sfo       = 'N'      sfname     = 'pams.tgz'  &END
-----
&OUT3D    out3dcen(1)=0.      out3dcen(2)=0.      out3dcen(3)=1.    &END
&OUT3D    out3ddim(1)=0.25    out3ddim(2)=4.      out3ddim(3)=0.25 &END
&OUT3D    ngpout3d(1)=50      ngpout3d(2)=1       ngpout3d(3)=50   &END
-----
```

As anticipated before, the OUT3D namelist is used, in combination with the `tecplot3d` keyword, to specify a set of external points where the solution is going to be computed. This namelist is characterized by the following parameters:

- `out3dcen` – a vector of three elements specifying the coordinates of the center of the volume where the solution is evaluated;

- `out3ddim` – a vector of three elements which rules the dimension of the computational volume. Each component denotes the length, expressed in meters, of one edge of the volume itself;
- `ngpout3d` – a vector of three elements used to discretize the computational volume in points. In particular every component of this vector specifies the number of points in which the correspondent volume edge is split up. Moreover, it should be noticed that up to three components of `ngpout3d` may be equal to one: if one 1 is present, then the volume is reduced to a plane; if there are two 1 then the volume degenerates into a rake and, finally, if all the components of `ngpout3d` are equal to one then the external evaluation zone results to be a point.

Geometry and Closure

Listing 5.7: Typical DATAIN file for PaMS - Geometry and Closure

```

-----
&GEOIN    geoname="rotor_8deg.geo"                icb=0                &END
&GEOIN    orient    =+1                orignum = 'N'        maxang=180.0        &END
&SCALE    sclfac(1)=1.0                sclfac(2)=1.0        sclfac(3)=1.0        &END
&KWAKE    ang=140.                    tmsfac=1.0           kwnumb   = 'N'        &END
&KWAKE    nkwt =400                    nkwtstrig = 0        nkwtspan =16         &END
&KWAKE    xanru = 0.                    yanru = 0.           zanru = 0.           &END
&BC       bctype(1)='D'                bctype(2)='R'        bctype(3)='P'        &END
&BC       bcnvel =0.0                    bcidou   = 'N'        &END
&BCLIM    corerad =0.2                    coremod  = 'R'        pt2wknd  =0.         &END
-----

```

In this part of the DATAIN file the closure conditions described in section 1.4.4 are applied. In order to achieve this goal, the geometry (and hence the *geo* file) has to be read by the solver first: this task is accomplished by means of the `geoname` keyword of the GEOIN namelist. Moreover, since PaMS comes with the possibility to handle multibody analyses, a *common body index* has to be defined for each body in order to distinguish among them. The `icb` keyword of the GEOIN namelist plays this role: if more than one body is present (*i.e.* if $nbody \geq 2$) the whole *Geometry and Closure* section of the DATAIN file has to be written $nbody$ times, with `icb` being different for every body. At this stage the `icb` can be seen as a label which can be employed, for example, when writing a routine for complex motions. As a consequence no particular rule to assign a value to the `icb` variable exists. However, as said talking about the `cbdgap` parameter of the BODY namelist, PaMS is able to handle more than one sub-geometry as one entity (*i.e.* no relative motion between the single parts of the same group is possible): this grouping option can be achieved by labelling the interested *Geometry and Closure* sections of the DATAIN file with a single common body index.

At this stage an orientation to the unit vectors normal to each panel should be provided. For ordinary geometries the PaMS converter is able to execute this operation in full automatic mode identifying the foremost panel along the x direction and hence distinguishing the interior and the exterior of the body. If PaMS is not able to carry out this process, then the same procedure

is applied to the y direction first, to the z one then and, eventually, the normal orientation is arbitrarily chosen. At the end of this process the converter generates the *geo* file which is ready to be read by PaMS. However sometimes the normals orientation obtained at this stage is not satisfactory. For example this may happen when the user is intended to perform an internal fluid analysis. In these situations, recalling that for thick bodies the normal should point toward the fluid and that for thin body all the normal vectors have to be coherent, the toggle *orient* of the GEOIN namelist could be set to -1 , so that the all normals are flipped. Otherwise *orient* should be $+1$.

Moreover, whenever the user is intended to perform a structural analysis on the basis of the PaMS calculated pressure distribution, it may be useful to keep the GRIDP and PANEL indices as they were defined in the preprocessor software. In fact, when the *orignumb* flag of the GEOIN namelist is set to 'N' (*No*) PaMS relabels both GRIDP and PANEL. On the other hand, imposing *orignumb* = 'Y' (standing for *Yes*), the solver expects as input another file, called *geofilename-on.geo*, where *geofilename* is the name of the current *geo* file and the suffix *on* is the acronym for *Original Numbering*. Obviously this file contains the original numbering for the nodes and the panels, and consequently this enumeration can be preserved in the output files, which is composed by two sections (whose meaning should be obvious) named GPNUM and PNNUM. For instance, whether the preprocessor exported the geometry discretization in a *Nastran*-compatible format, then the solver generated output can be directly imported in *Nastran* itself in order to perform a structural analysis.

In order to contain the error on the velocities nearby the sharp edges (*e.g.* at the wing-fuselage intersection), PaMS provides the *maxang* keyword of the GEOIN namelist. In effect, this parameter denotes the maximum angle (expressed in degrees of arc) included between the normals of two adjacent panels so as these can be considered as continuous. In particular *ang=0* denotes two panels lying on the same plane, whereas, on the other extreme, *maxang=180* means that the considered panels are totally overlapped. Therefore it is easy to understand that $0 \text{ deg} \leq \text{maxang} \leq 180 \text{ deg}$ and consequently there is no difference between acute and obtuse angles.

Finally, since PaMS works in meters, the geometry should be conveniently scaled in SI units through the keyword *sclfac* of the SCALE namelist. The scale factor is intended as a vector quantity, so that the corresponding parameter in the DATAIN file has three scalar components, all of which should be set to one when the imported geometry has already been represented in meters in the source CAD or meshing software. On the other hand, whether, for instance, the source geometry is expressed in millimeters, then all the three components of *sclfac* have to be equal to one thousandth (*i.e.* $\text{sclfac}(i) = 0.001, i = 1 \dots 3$). However this SI unit perspective is not the only field where the *sclfac* keyword may be useful. In fact, the vectorial nature of this parameter has been introduced to make it easy to perform comparative analyses using the same *geo* file and changing, for example, the wingspan or the wing thickness only.

In section 5.2.1 it has been pointed out that the *geo* file does not contain any information about the panels to which the Kutta condition should be applied. This gap, in fact, is bridged by means of the KWAKE namelist and, in particular, of its arguments *ang* e *kwnumb*. In effect, as shown in figure 5.3, for ordinary thick geometries PaMS automatically applies the Kutta condition to all the adjacent panels whose normals are separated by an angle greater to the one specified through the *ang* keyword (expressed in degrees of arc). Nonetheless, it should be clear that not all the sharp edges are TEs. For this reason, aside to the *ang* keyword, PaMS takes into account the

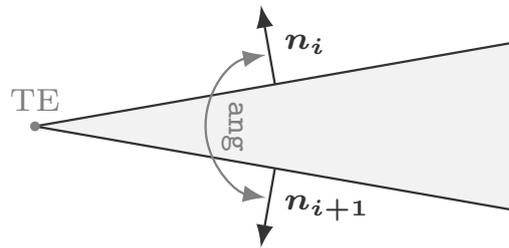


Figure 5.3: Explanation for the `ang` keyword

relative motion between every potential TE and the airflow. In this scenario the Kutta condition is applied where the previously described `ang` condition is met and where, at the same time, the bisector line to the TE forms with the flow an angle lower to 90 deg. For example, according to this procedure the LE angle of a supersonic airfoil is correctly not characterized by the enforcement of the Kutta condition despite being characterized by an angle between the normals greater than `ang`. In a similar fashion, for surface bodies the Kutta condition is applied to the free edges where the angle formed by the airflow and the tangent vector (starting from the panel center and pointing towards the TE) is lower than `ang`.

However uncommon bodies (such as a the portion of a propeller blade nearby the hub) could be characterized by sharp angles between two panels even though their common sides do not require the enforcement of the Kutta condition. In this case the automatic process performed by the software may be out ruled listing all the TE grid points in a third file, named `geofilename-te.geo` (where `geofilename` is the name of the `geo` file related to the body under examination) and loaded setting to Y (*i.e.* Yes) the `kwnumb` parameter. The first line of this file is occupied by string `TEGPN`, standing for Trailing Edge Grid Point Number, whereas all the others report the `gridp` indices of the TE points. Whether `kwnumb='Y'` PaMS only verifies that all the points listed in the `geofilename-te.geo` file belongs to a free edge of a thin body or to an edge of a thick body where the `ang` condition is also met.

In PaMS the control of the user over the wake can be extended by the `tmsfac` keyword of the `KWAKE` namelist. In particular, this parameter is a scale factor which rules the velocity with which the wake is convected downwards: if `tmsfac=1` the wake velocity is not altered, whether `tmsfac < 1` the wake development is slowed down and, finally, when `tmsfac > 1` it is speeded up. This fragile parameter may be useful whenever in the flow-field there are two or more bodies whose characteristic speeds are deeply different: for example this is the case of a wing-propeller analysis, where it could be useful to stretch the wing wake (setting `tmsfac > 1`) so that its development has the same speed of the propeller one.

One of the most fragile issues of an aerodynamics problem concerns the wake treatment, especially near its three free edges (being the fourth the TE, hence constrained to the wing). In fact, as shown in figure 5.4, whereas the wake singularity inductions in the inner portion of the wake are balanced due to their relative orientations (and hence even really strong singularities do not cause any trouble here), on the boundaries an unbalance exists and hence a criticality does rise. This is true especially far away from the body, whether a fully free wake model is adopted.

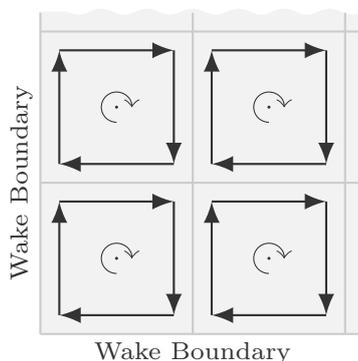


Figure 5.4: Criticality at the wake boundaries

For this reason (and in order to make the problem less costly) it may be convenient to manage some of the wake peculiar characteristics, such as their development in space, their modeling (flexible, rigid or hybrid) or the possibility to employ either panels or vortons. In PaMS these features can be handled means of the `nkwts`, `nkwtstrig` and `nkwtspan` keywords of the `KWAKE` namelist:

- `nkwts` – number of time steps characterized by wake panels (or vortons) shedding. If this variable is lower than the number of time steps to be simulated, the wake is cut off as described in section 2.5;
- `nkwtstrig` – number of rigid wake time steps. Denoting with ts the current time step, if the total time steps number exceeds `nkwtstrig`, then the wake is modeled as rigid for $ts \leq nkwtstrig$ and as flexible otherwise. In particular, when reached, the flexible model is extend to the already generated wake portion, too. This parameter can hence be exploited to generate a flexible wake starting from a rigid one, thus resulting in a more robust calculation. Finally, it should be clear that if the user wants to keep a rigid model for the whole simulation, than `nkwtstrig` should be greater than the total number of time steps.
- `nkwtspan` – number of wake time steps to be modelled through panels. This integer parameter has to be at least equal to one, although it is recommended $nkwtspan \geq 2$

It should be noticed that, since on small scales the flow-field is better described by a doublet panel rather than by a vortons cluster, the panel element is to be preferred to the vorton one near the body. Informally, this statement can be understood noting that, despite the presence of leakage flows (see section 2.4), a surface singularity distribution guarantees better results in terms of impermeability than those achievable by means of point singularities. Obviously, in order to make the enforcement of the Kutta condition possible, the TE has to border with panel elements. For these reasons the `nkwtspan` parameter has to be greater than one. Moreover, the total span of the wake panels should be of the same order of magnitude of the body one due to the fact that the wake shape strongly affects the value assumed by the aerodynamic force. On the other hand, the total number of wake time steps should be greater or equal than the number of wake time

steps modelled through panels, namely $n_{kwts} \geq n_{kwtspan}$. In particular, whether the inequality sign holds, then the remaining $n_{kwts} - n_{kwtspan}$ wake time steps are modelled by means of vortex particles.

Furthermore, it is dutiful to do a digression about the way PaMS limits the total amount of wake panel rows. As said in section 2.5, the wake panels shedding mechanism is performed in a fixed, inertial reference frame, whereas the body coordinate system – as seen by an observer integral with the inertial one – moves across the three-dimensional space. At each time step other than the first one the wake panels previously generated are motionless in the inertial space, so that a gap is generated between the wing TE and the row of wake panels nearest to it. This gap is then filled with others panels, so that it appears clear that the software only retains the first $n_{kwtspan}$ rows of wake panels, cutting off all the others. As a consequence, according to what has just been said regarding the wake creation process, the wake itself is formed by the $n_{kwtspan}$ panels nearest to the wing (and hence different at each time step).

PaMS is able to deal with both rigid and flexible wakes: in the first case the wake only rigidly translates due to the airflow, or, if the body is considered to be in motion, the wake keep staying in the same place where it has been generated, without aligning itself to the flow-field. On the other hand, due to the fact that it has not to produce load (*i.e.* it has to be parallel to the airflow) the flexible wake is totally governed by the local velocity. However the flexible wake is obviously sensible to other factors, such as the presence of obstacles or the induction due to other wakes. Aside to these two classical wake models, PaMS comes with the possibility to deal with an hybrid model lying midway between the flexible wake and the rigid one. This task is achieved acting on the self induction of the wake on itself, interpreted as the velocity induced by the wake elements on other elements belonging to the same wake. In this context, considering for example a propeller with zero free-stream velocity, if one takes into account the only component of the induction parallel to the rotor axis (*i.e.* along the expected wake propagation direction), the result is somehow similar to neglecting both the roll-up and the wake contraction/spreading. In the DATAIN file this effect is controlled by means of the three parameters `xanru`, `yanru`, `zanru` of the `KWAKE` namelist, which represents the three components (in a fixed reference frame – see section 5.2.2) of the vector normal to the plane where the self-induction of the wake is inhibited (*i.e.* of the propeller axis in the previous example). In particular, the Fully Free Wake (FFW) model is obtained by setting to zero the values assumed by all the three parameters here analyzed. This control over the wake may be extremely useful whenever the user is interested in a post start-up situation, that is to say when the wake significantly rolls up far away from the body, so that the influence of the phenomenon on the performances of the aerodynamic device under examination is negligible. The main justification to this statement lies in the fact that the roll-up phenomenon may result in some numerical issues and, in order to be evaluated, implies additional computational costs with respect to a No Roll-up Wake (NRW). As a consequence, it appears clear that a NRW could also be useful whenever one is intended to perform an analysis far away from the body, when a FFW may cause some troubles.

In the currently examined portion of the DATAIN input file information about the closure (*i.e.* about the BCs) are provided, too. In particular, the namelist concerning this issue is intuitively named `BC` and contains the following keywords:

- `bctype` – it is an array of three elements concerning both the body and the wake, as

Table 5.2: Accepted values for the `bctype` keyword

bctype	Accepted values	Meaning	Applied to
1	D	Dirichlet	Thick Body
	N	Neumann	Thin Body
	F	Free Surface	Air-Water Interface
2	R,F	Rigid, Flexible	Body
3	P,I	Permeable, Impermeable	Wake-Body Interaction

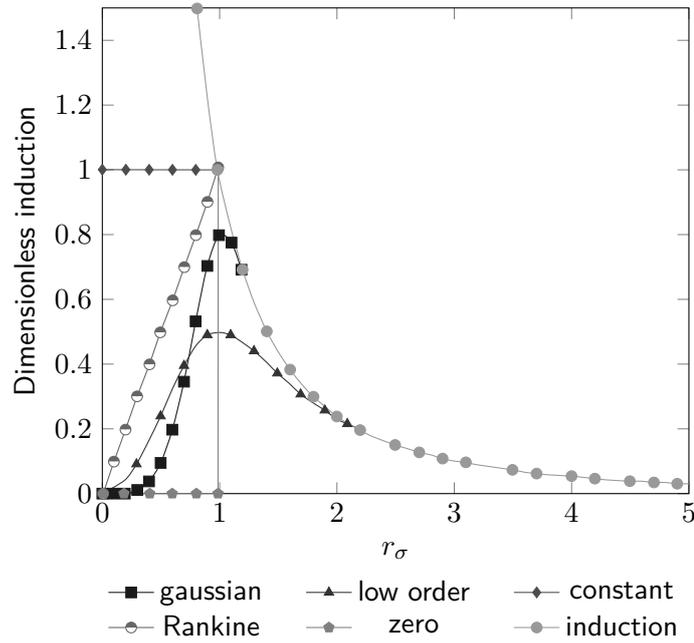
described in table 5.2. In particular, other than the classical Dirichlet and Neumann options (respectively for thick and thin bodies), the parameter `bctype(1)` can be used to model an air-water interface through the option 'F', standing for *free surface*. In this case the interface deformation is computed by the solver itself and does not require any additional input. On the other hand, this is not the case of a general morphing geometry, which, in order to be modeled, requires a *geo* file for each time step. In particular these *geo* files have to be characterized by the same sequence of grid points employed in the original discretization (*i.e.* only the coordinates of the grid points are altered) and could, for example, produced by a structural solver in a fluid-structure interaction context. In addition in this case the reference field of the `bctype` variable is the second one (*i.e.* `bctype(2)`), whose possible values are 'R' and 'F', respectively standing for *rigid* and *flexible*. Since the influence matrix concerning the body panels (without the wake) depends on geometrical information only, when `bctype(2)='R'`, PaMS computes it just ones, resulting in a performance boost. On the contrary, whether the current problem deals with morphing (and hence flexible) geometries, `bctype(2)` should be set to 'F' and the body related influence matrix is updated at every time step, whence the need for additional *geo* files. However, as explained regarding the *Body and Motion-related Parameters*, the `bctype(2)` parameter is out ruled by `relmot` one of the BODY namelist, so that for flexible bodies `relmot` should be set to 'Y'. Finally `bctype(3)` can assume the values 'P' (*i.e.* *Permeable*) and 'I' (*i.e.* *Impermeable*) and it is related to the wake-body intersections. In fact this variable handles the wake-body piercing checks, allowing to verify if a wake panel/vorton is located inside a thick body.

- `bcnvel` – transpiration velocity V_{tr} (see equation (1.23)) expressed in m/s. This value has obviously to be set to zero for non-porous media.
- `bcidou` – char flag which if set on 'Y' (Yes) makes the corresponding surface isopotential (*i.e.* it is modeled through doublets of the same intensity). This feature, often in combination with the transpiration velocity, can be useful whenever the user deals with internal flows problem characterized by the presence of an inlet/outlet where the flow rate and the zero tangential velocity conditions have to be enforced.

Finally, as described in section 2.7.1, in order to regularize the point vortex elements, the VPM requires the definition of both a core radius and a core-function. In the DATAIN file these quantities are specified by means of the keywords `corerad` and `coremod` of the BCLIM namelist, respectively. The first of these two parameters is a number to be expressed in meters, whereas

Table 5.3: Accepted values for the `coremod` keyword

Value	Meaning
Z	Zero
R	Rankine
G	Gauss
L	Low order
C	Constant

**Figure 5.5:** Core functions employed in PaMS

the second one is expected to be of the char type, according to table 5.3. Moreover, in figure 5.5 a graphical representation of all the core models available in PaMS is reported: here the direct effect of the core-functions on the dimensionless induction due to a single vorton is underlined, and particular attention should be paid to the different cutoff strategies applied to the asymptote located nearby the vortex particle core (*i.e.* the origin of the reference frame). Furthermore it should be noticed that an approach similar to that here described about the vortons holds for the panels located in the wake, too. Consequently in PaMS the `corerad` and `coremod` parameters are exploited in the modeling of both the vortex particles and the wake panels, even if some internal corrections are carried out in order to take into account the different nature of the two elements.

Body Motion

Listing 5.8: Typical DATAIN file for PaMS - Body Motion

```

-----
&GEOTRA  GT(1,1)=0.0          GT(1,2)=0.0          GT(1,3)=0.0          &END
&GEOTRA  GT(2,1)=0.0          GT(2,2)=0.0          GT(2,3)=0.0          &END
&GEOTRA  GT(3,1)=0.0          GT(3,2)=0.0          GT(3,3)=0.0          &END
&GEOROT  GR(1,1)=0.0          GR(1,2)=0.0          GR(1,3)=0.0          &END
&GEOROT  GR(2,1)=0.0          GR(2,2)=0.0          GR(2,3)=0.0          &END
&GEOROT  GR(3,1)=0.0          GR(3,2)=0.0          GR(3,3)=0.0          &END
&SYSTRA  ST(1,1)=0.0          ST(1,2)=0.0          ST(1,3)=0.0          &END
&SYSTRA  ST(2,1)=0.0          ST(2,2)=0.0          ST(2,3)=0.0          &END
&SYSTRA  ST(3,1)=0.0          ST(3,2)=0.0          ST(3,3)=0.0          &END
&SYSROT  SR(1,1)=0.0          SR(1,2)=0.0          SR(1,3)=0.0          &END
&SYSROT  SR(2,1)=0.0          SR(2,2)=0.0          SR(2,3)=1250.         &END
&SYSROT  SR(3,1)=0.0          SR(3,2)=0.0          SR(3,3)=0.0          &END
-----

```

In order to achieve complex trajectories in space, PaMS is characterized by the simultaneous presence of two different reference frames: an inertial global reference frame and one referred to the current body. The first coordinate system is fixed in both space and time, whereas the second one can have a relative motion with respect to the body and, obviously, to the fixed frame, too. In particular, the geometry data imported through the *geo* file are expressed in the body reference frame, which, without the user action, is initially coincident with the global coordinate system.

This reference frame logic is applied in listing 5.8, where a constant angular speed of the body with respect to its own frame is pointed out by the keyword `GR(2,3)=1250` into the `GEOROT` namelist. In particular, this keyword is characterized by the matrix-like structure represented into equation (5.1), where ϕ , θ and ψ are the Euler's angles related, in this case, to the orientation of the body with respect to its reference frame. Obviously a similar arrangement holds for the `GEOTRA`, `SYSTRA` and `SYSROT` namelists.

$$\mathbf{GR} = \begin{pmatrix} \mathbf{GR}(1,1) & \mathbf{GR}(1,2) & \mathbf{GR}(1,3) \\ \mathbf{GR}(2,1) & \mathbf{GR}(2,2) & \mathbf{GR}(2,3) \\ \mathbf{GR}(3,1) & \mathbf{GR}(3,2) & \mathbf{GR}(3,3) \end{pmatrix} = \begin{pmatrix} \phi & \theta & \psi \\ \dot{\phi} & \dot{\theta} & \dot{\psi} \\ \ddot{\phi} & \ddot{\theta} & \ddot{\psi} \end{pmatrix} \quad (5.1)$$

In particular, some exceptions to the SI unit standard adopted in PaMS can be pointed out with the help of equation (5.1). In effect, in the `DATAIN` file all the angles (e.g. `GR(1,1)`) have to be expressed in degrees of arc, all the angular speeds (e.g. `SR(2,3)`) are expected to be in rpm and all the angular accelerations (e.g. `GR(3,2)`) have to be in revolutions/min².

Summarizing, the namelists with the `GEO` prefix are related to a motion of the body in its own reference frame, whereas those starting with `SYS` concern the motion of the body coordinate system with respect to the fixed one. In this scenario, for instance, one could simulate the rotation of a propeller around its axis by means of a `GEOROT` and, at the same time, the motion of the rotor disk along a particular flight trajectory through a `SYS` operation. In fact, in this example the blades rotate around the propeller axis which is coincident with one of the main directions of the body reference frame and this last coordinates frame is moved along a specified path.

At this stage the reader could have noticed that – for simple non-rotational, uniform streams – two different ways to prescribe the motion law do exist: the first one concerns the INFTY namelist, whereas the second one is related to the actual dissertation. Obviously the difference between these two different approaches lies in the variety of motion laws that each method allows to employ.

5.2.3 Routine for Complex Motions

Listing 5.9: Extract of motion routine for PaMS

```
OMGTILT=(PI/2.)/.12
IF (TM.LE..048*3 ) TT=0.
IF (TM.GT..048*3.AND.TM.LE..048*3+.12) TT=TM-.048*3
IF (TM.GT..048*3+.12 ) TT=.12

GEOR(IB,2,3)=+1250.*(2.*PI/60.)*COS(OMGTILT*TT)
GEOR(IB,2,1)=-1250.*(2.*PI/60.)*SIN(OMGTILT*TT)

GEOR(IB,2,2)=0.
IF (TM.GT..048*3.AND.TM.LE..048*3+.12) GEOR(IB,2,2)=-OMGTILT

SYST(IB,2,3)=0.
SYST(IB,2,1)=0.
IF (TM.LE..048*2 ) SYST(IB,2,3)=+10.
IF (TM.GT..048*4+.12 ) SYST(IB,2,1)=-10.
```

In section 5.2.2 the possibility to prescribe arbitrary motion laws exploiting the coexistence of the body and inertial reference frames has been pointed out. However this method appears to suffer of some limitations whenever the complexity of the motion law increases. This is, for example, the case of a tilting rotor: in this situation, in fact, the superposition of rotations around multiple axes (the shaft one for the propeller and the wing-to-wing one for the nacelle) and of translations into a three dimensional environment make the simple motion instructions present in the DATAIN file insufficient. In this case, however, PaMS capabilities can be extended by means of a support routine whose role is to efficaciously implement the wanted motion into the current analysis.

A complete example of such a routine, suitable for a tilting rotor, is reported in listing D.1 on page 123, whereas in listing 5.9 only the core of this code is reported. Here it could be noticed that the rotation of the nacelle is provided by a GEOROT, whose velocity (OMGTILT) is evaluated imposing that a quarter of revolution has to be covered in the tilting time (amounting to 0,12 s). Moreover a conditional statement rules the time at which the conversion has to happen. In a similar fashion the translational velocities are imposed, too. However, actually a slight difference between the techniques employed to assign the rotational and translational velocities does exist: whether, on the one hand – as pointed out before – the rotation is provided acting on the relative position between the body and its own reference frame, on the other hand the translational motion is provided by means of a SYSTRA. Nevertheless, in both cases the syntax employed in listing 5.9 is the same, in fact both GEOR and SYST require three arguments: the first one is the common body

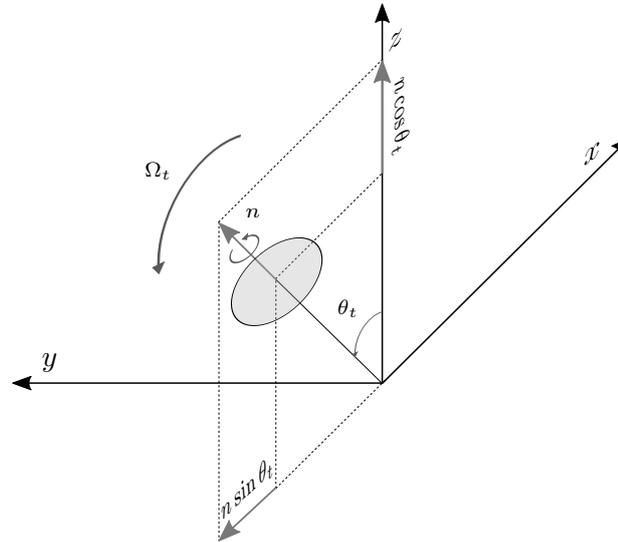


Figure 5.6: Explanation of the motion law applied in listing 5.9

index, the second one denotes a position, a velocity or an acceleration as described in section 5.2.2 and, finally, the latter refers to the axis along which the rotation/velocity/acceleration vector lies. For instance, according to this nomenclature, the code `SYST(IB,2,3)=+10` means that the body whose common body index `icb` is equal to the value stored in the variable `IB` has a reference frame which travels along the global Z axis with a velocity of 10 m/s. Obviously a general motion in a direction not lying along one of the three reference axes can be obtained considering the three components of the the vector of interest either in the body coordinate system or in the inertial one.

Finally, the particular motion law employed in listing 5.9 can be understood with the help of figure 5.6 and thinking about the definition of `GEOROT`. In fact, since in a `GEOROT` the body moves in its own reference frame, when the tilt occurs the revolution of the rotor around its axis is made of two different components which, in the body reference frame, are

$$n_x = -n \sin \theta_t \quad \text{and} \quad n_z = n \cos \theta_t ,$$

where n is expressed in rad/s and $\theta_t = \Omega_t t$ is the tilt angle, assumed to be zero when the rotor axis is vertical. In this context $n = 1250$ rpm and hence the quantity $2\pi/60$ is a conversion fraction from rpm to rad/s.

5.3 Output Data Files

When all the required input data files described in section 5.2 have been correctly written and stored in the same directory where the PaMS solver is located, the code can be invoked. The work-

```

-----
|           START           |
-----

> call ALLOCATIONGED
ela-t  0h: 0m: 0s: 0d
> call JOBDATA
ela-t  0h: 0m: 0s: 0d
> call GEOIN
ela-t  0h: 0m: 0s: 0d
> call NEIGHBOR
ela-t  0h: 0m: 0s: 0d
> call BPDAT
ela-t  0h: 0m: 0s: 0d
> call ALLOCATIONKW
ela-t  0h: 0m: 0s: 0d
> call ALLOCATIONMAT
-----

```

Figure 5.7: PaMS in action

ing software appears like in figure 5.7 and, once the analysis is complete, it produces a subdirectory called PaMS-w3.OBS@year-month-day@hour.minute.second into the current path.

This new folder – as shown by the tree in figure 5.8 – is populated by several subdirectories, related to the output settings chosen into the DATAIN file, as described in section 5.2.2. In general, aside to the output directories, the DATAIN folder is created: here all the input files are stored and, whether the PaMS executable file is named pams.exe, a copy of it is placed in this path, too. Moreover a DATAOUT folder is created, containing a REPORT file: it is a log file where all the task performed during the analysis (*i.e.* all the on screen information similar to those represented in figure 5.7) are reported. This folder contains other two key files: the DATAOUTforce and the DATAOUTgeom.

The first one is characterized by the following variables list

```

VARIABLES="BD"    "TS"    "T"    "Fx"    "Fy"    "Fz"
             "Fx(U)" "Fy(U)" "Fz(U)" "Mx"    "My"    "Mz"
             "L(Tx)" "L(Ty)" "L(Tx)" "Di(Tx)" "Di(Tx)" "Di(Tx)".

```

Here information about the components of both the forces (F , [N]) and the moments (M , [Nm]) acting on each body (BD) and evaluated through a pressure integral are reported as functions of both the time step index (TS) and the time itself (T , [s]). Moreover both the unsteady components of the pressure integral ($F(U)$, [N]) and the steady components of the lift (L , [N]) and of aerodynamic drag (Di , [N]), evaluated by means of a *near-field Trefftz* technique, are listed.

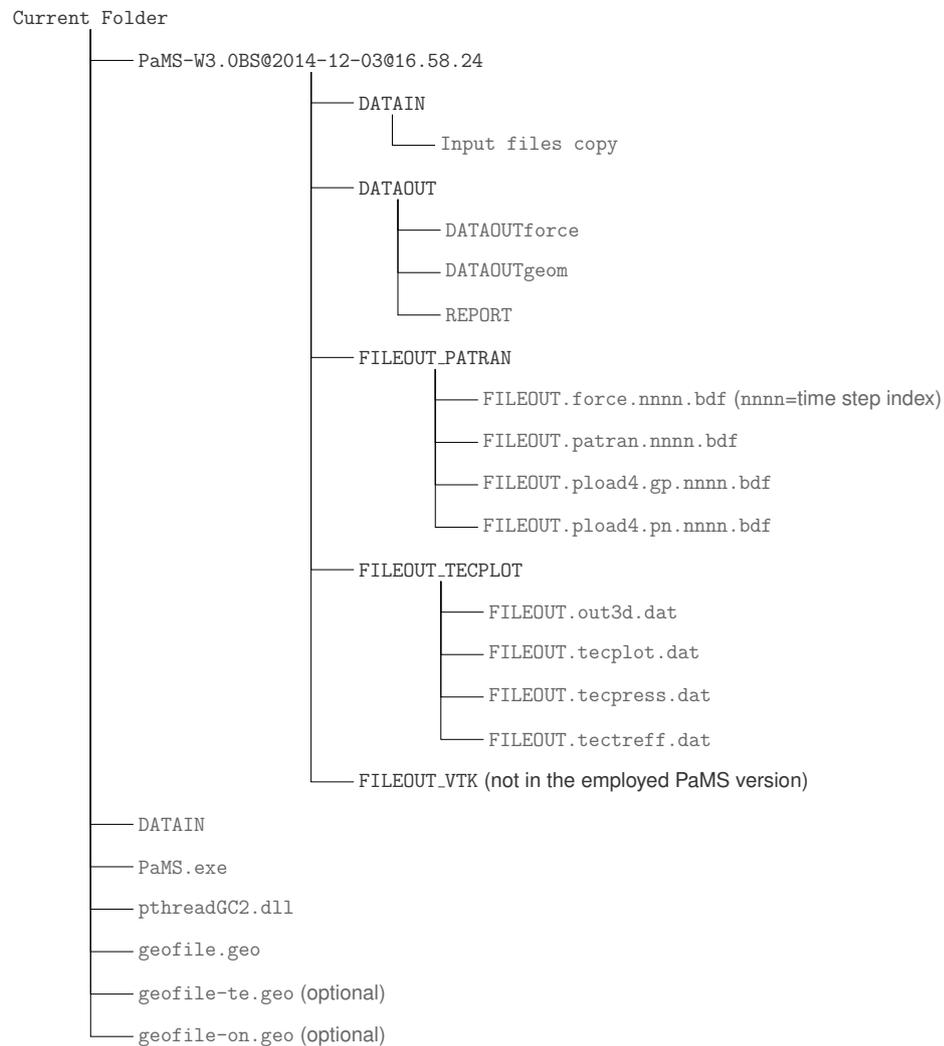


Figure 5.8: PaMS output tree. The string `nnnn` denotes the time step index

The `DATAOUTgeom` file is structured in a similar fashion, however here the kinematics of the case into study is reported. For this file the variable list is

```
VARIABLES="BD"      "TS"      "T"      "Xg"      "Yg"      "Zg"
           "xXang" "xYang" "xZang" "yXang" "yYang" "yZang"
           "zXang" "zYang" "zZang" "A"      "Ax"      "Ay"      "Az",
```

so that the centroid – *i.e.* the origin of the body frame – coordinates (X_g , Y_g , Z_g , [m]), all the angles between two different axes – belonging to both the body reference frame and to the global one – ($xXang$, $xYang$ *etc.*, [deg]), the wetted area (A , [m²]) and the area projections on the coordinate planes (A_x , A_y , A_z , [m²]) are listed for each body. In particular, denoting with an uppercase letter the names of the global coordinate system and with a lowercase one those of the

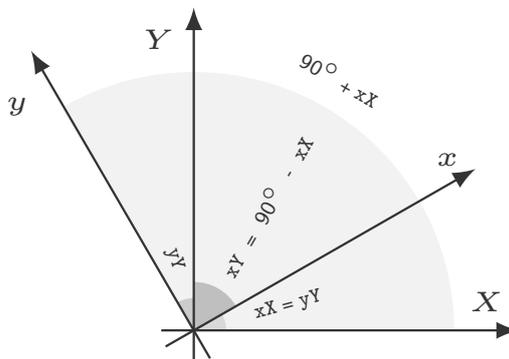


Figure 5.9: PaMS – reference frames in the DATAOUTgeom file

body reference frame, the logic of the angles reported into the file is clarified by the figure 5.9, where a simple single-rotation (around the $z \equiv Z$ axis) case is depicted.

Even though it has already been said above, it could be useful to point out that for multibody analyses both the DATAOUTforce and the DATAOUTgeom files provide information on all the bodies listed in the DATAIN file through the value assumed by the BD parameter. In particular for each time step a total of $nbody$ data rows are reported, with the BD index in the range $1 \div nbody$, so that the user can easily read the results produced by PaMS.

Apart from the DATAOUT folder, in the scope of this thesis another relevant subdirectory is the one named FILEOUT_TECPLOT, containing up to four bridging files between PaMS and the commercial postprocessing software tecplot 360EX: FILEOUT.tecplot, FILEOUT.tectreff, FILEOUT.out3d and FILEOUT.tecpress.

The variable list of the first file is

```
VARIABLES="X"   "Y"   "Z"
           "U"   "V"   "W"   "Vm"
           "M"   "P"   "rho"
           "Fx"  "Fy"  "Fz"  "Fm"
           "TSi" "PTi",
```

where X , Y and Z are the coordinates of the evaluation points ([m]), V_m is the modulus of the velocity vector $\mathbf{V} = (U, V, W)$ and, as all the other velocities, is expressed in meters. Similarly the string F_m ([N]) represents the modulus of the aerodynamic force $\mathbf{F} = (F_x, F_y, F_z)$. As those just examined, the variable names M , P ([Pa]), ρ ([kg/m³]) should be self-explanatory since, each in order, represent the Mach number, the static pressure and the flow density. Among all the standard variables here described, the FILEOUT.tecplot file allows to work with two peculiar quantities, called TS_i and PT_i and representing the *time step index* and the *grid point* one, respectively. In particular, all the vortons characterized by a common PS_i belongs to the same *vortex line*. Moreover it should be pointed out that, unlike the panel methods, tecplot 360EX requires node based data, so that the data reported in the FILEOUT.tecplot are obtained from the original dataset evaluated by PaMS through a least square approach. Furthermore it should be

recalled that, depending on what the settings employed in the *Output handling* of the DATAIN file, the evaluation points can be coincident with the grid points, the panel centers or both.

On the other hand, the FILEOUT.tectreff file is generated for each lifting body (*i.e.* for each body where a *Kutta condition* is imposed). The variable list related to this file is

```
VARIABLES="X" "Y" "Z" "DX" "DY" "DZ" "C" "S"
          "cCl(P X)" "cCl(P Y)" "cCl(P Z)"
          "cCl(PuX)" "cCl(PuY)" "cCl(PuZ)"
          "cCl(TsX)" "cCl(TsY)" "cCl(TsZ)".
```

In particular, the reference point (whose coordinates are X , Y and Z , [m]) is located at the center of the trailing edge of each segment, whose length is DX , DY and DZ ([m]). Furthermore, this segment is referred to a wing section whose chord and reference surface are C ([m]) and S ([m²]), respectively. On the other hand, the remaining variables are related to the aerodynamic load: $cCl(P)$ represents the load evaluated by means of a pressure integral, $cCl(Pu)$ are the unsteady portions of the load, whereas, finally, $cCl(Ts)$ are the three steady components or the load estimated with a *Trefftz's* technique (based on the value of the wake doublet nearby the trailing edge).

The FILEOUT.out3d file is similar to the FILEOUT.tecplot, lying the only differences in the evaluation points and in the lack of the force components among the output variables. In fact, in the FILEOUT.out3d file the evaluation points belong to the external zone defined in the DATAIN file by means of the OUT3D namelist, whereas the force is inherently related to the presence of a surface and hence cannot be calculated in a point located in the interior of the fluid domain.

Information about the panels are contained in the FILEOUT.tecpress, whose variables are

```
VARIABLES="X" "Y" "Z" "Nx" "Ny" "Nz"
          "A" "P" "rho" "Fx" "Fy" "Fz",
```

where each panel has its center located in the point (X, Y, Z) , a normal unit vector whose components are N_x , N_y and N_z and an area A ([m²]). This file may be useful to perform integral calculations of both the pressures and the forces acting on a body.

Whether the user made the *Patran* output active by setting *patran='Y'* in the DATAIN file, the FILEOUT_PATRAN folder in the output three is filled with four different files, each of which is generated for every time step:

- FILEOUT.patran.nnnn.bdf – can be employed to visualize in *Patran* both the body and the pressure distribution;
- FILEOUT.pload4.gp.nnnn.bdf and FILEOUT.pload4.pn.nnnn.bdf – contain only pressure data in *Nastran* pload4 format, respectively evaluated in the grid points and in the panel centers;
- FILEOUT.force.nnnn.bdf – comes with force data referred to the nodes and expressed in the *Nastran* force format.

It should be noted that in the previous list the `nnnn` variable is a time step counter and that information about the geometry are provided in the `.patran.ts` file only: this is due to the fact that *Nastran* and *Patran* allow to import load data related to a geometry stored in another source file. Moreover, it is obvious that, if the user enabled the one-file-output option just four different files will be created: in this case the time step will be directly selectable in *Nastran*.

Finally, PaMS allows to export the output in the *ParaView* compatible VTK format, too. However this option, which produces results similar to that related to the tecplot 360EX output, is not documented here because it is beyond the scope of this thesis.

5.4 Test Case

To prove the ability of PaMS to efficaciously deal with unsteady situations such as those related to the flow field induced by a rotary wing, here a simple test case is reported: it concerns a rotor in hovering and it has been employed to validate the code through a comparison with experimental results.

5.4.1 Rotor Hovering

The present case of study is based on some experimental measurements performed on a hovering rotor at the NASA *Ames Research Center* [see Caradonna and Tung, 1981]. The rotor – shown in figure 5.10 – is characterized by two cantilever-mounted, manually adjustable blades with half degree precone. These blades used a NACA 0012 profile and were untwisted and untapered. An aspect ratio of 6 was chosen in order to maximize Reynolds Number.

This test case [extracted by Caccavale, 2008] has been analyzed employing both a Fully Free Wake (FFW) model and a No Roll-up Wake (NRW) one. This difference has been set up acting on the `xanru`, `yanru` and `zanru` keywords of the KWAKE namelist in the DATAIN file. In fact, as discussed in section 5.2.2, the FFW model is obtained whenever all these three parameters have zero values. On the other hand, in this case the NRW is achieved setting `zanru` to one and keeping to zero the remaining components of the normal to the plane where the self-induction of the wake is inhibited. For more details regarding the DATAIN file see appendix C, where the full input file in the case of FFW is reported.

A memorandum of the data concerning the experimental conditions and the subsequent results of the test at *Ames Research Center* is reported in table 5.4. The simulation has been carried out for a total time of 0,6s, so that, according to the angular speed reported in table 5.4, roughly twelve revolutions are completed by the blade. The resulting time evolution of the vortons wake is shown in figures 5.11 and 5.12 for the two cases of both FFW and NRW, respectively. Here the axial component (w) of the velocity is emphasized, so that in the first case the roll-up phenomenon can be greatly appreciated through the presence of zones characterized by a positive axial velocity (*i.e.* directed upwards, as it can be observed in figures 5.11(e) and 5.11(f)). Furthermore a direct comparison between the two cases taken into account and the static experimental value of the thrust coefficient is reported in figure 5.13. Here one could observe that the percentage error on the thrust coefficient evaluated after the full analysis time of 0,6s is about the 6% in the case of FFW and nearly the 8% for the NRW model.

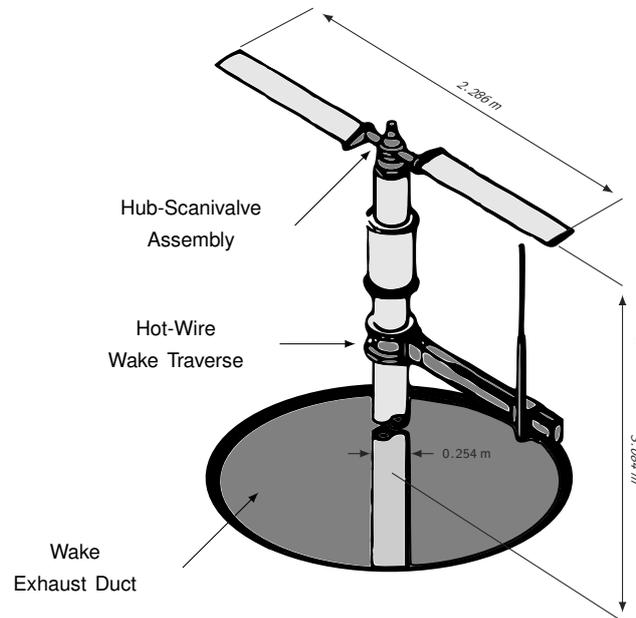


Figure 5.10: Experimental apparatus for a hovering test case [Caradonna and Tung, 1981]

Table 5.4: Experimental conditions and results for the hovering test case

M_{tip}	Re_{tip}	Pitch angle, θ_0	n [rpm]	static T_C
0.46	$2.5 \cdot 10^6$	8 deg	1250	0.0046

Slight modifications to the DATAIN file could lead to the analysis of cases characterized by a translational motion too, such as, for instance, an axial descent or climb. In fact these cases differ to the basic hovering one reported in appendix C just for the value assumed by the $ST(2,3)$ parameter (*i.e.* the translational velocity of the body reference frame with respect to the global one, measured along the z axis). Prove of that is given in figure 5.14, where both a climb and a descent case at $V = \pm 1$ m/s have been depicted at the final analysis time of 0,6s in the FFW case. It appears clear that the wake is extremely sensible to the vertical velocity, so that it results to be elongated or contracted for the climb and descent cases, respectively. As a consequence, in an unsteady situation, during the descent phase the rotor disk collides with its own wake until the thrust generated is enough to pull down the wake itself. For this reason this kind of calculation is subjected to numerical errors and should be dealt with caution.

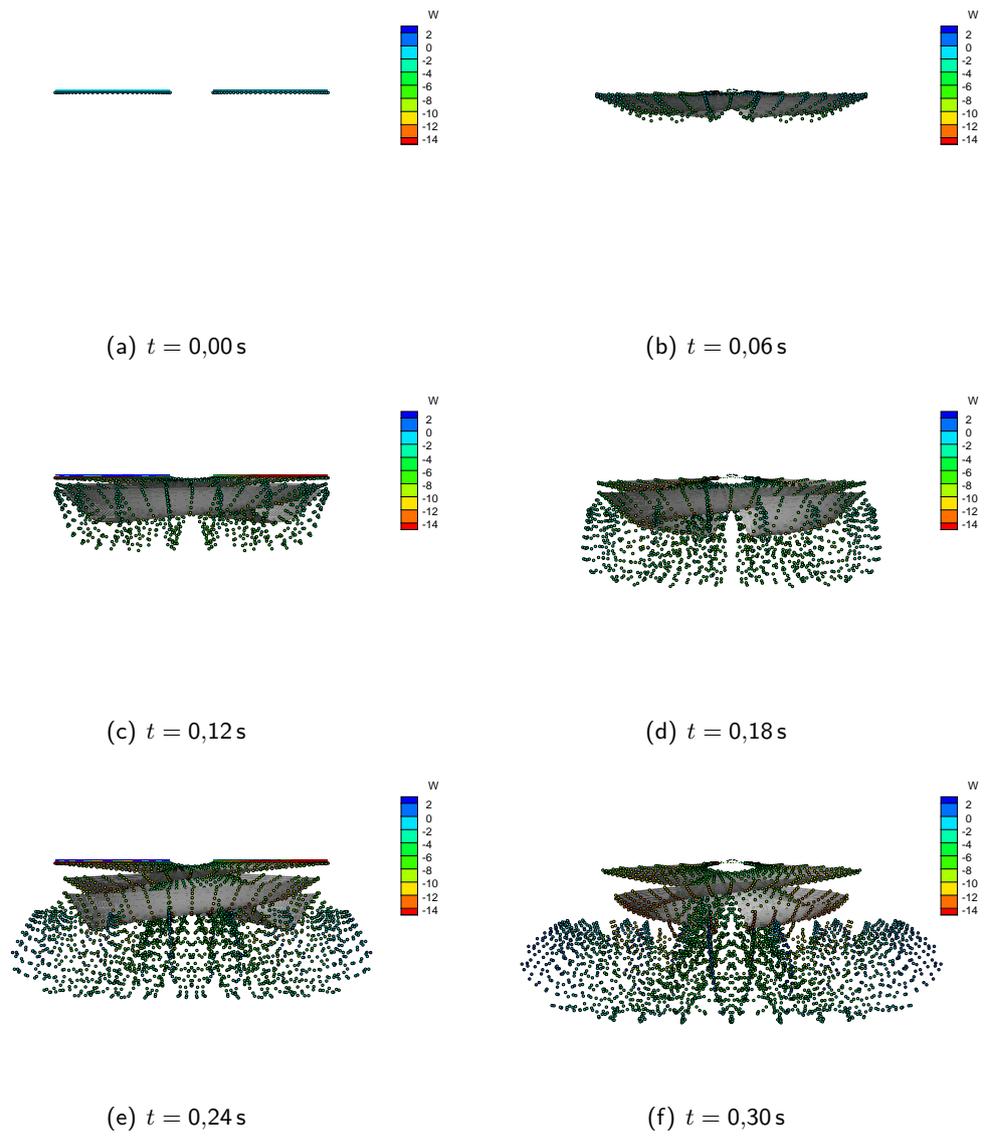


Figure 5.11: Axial velocity for a simple hovering test case and Fully Free Wake

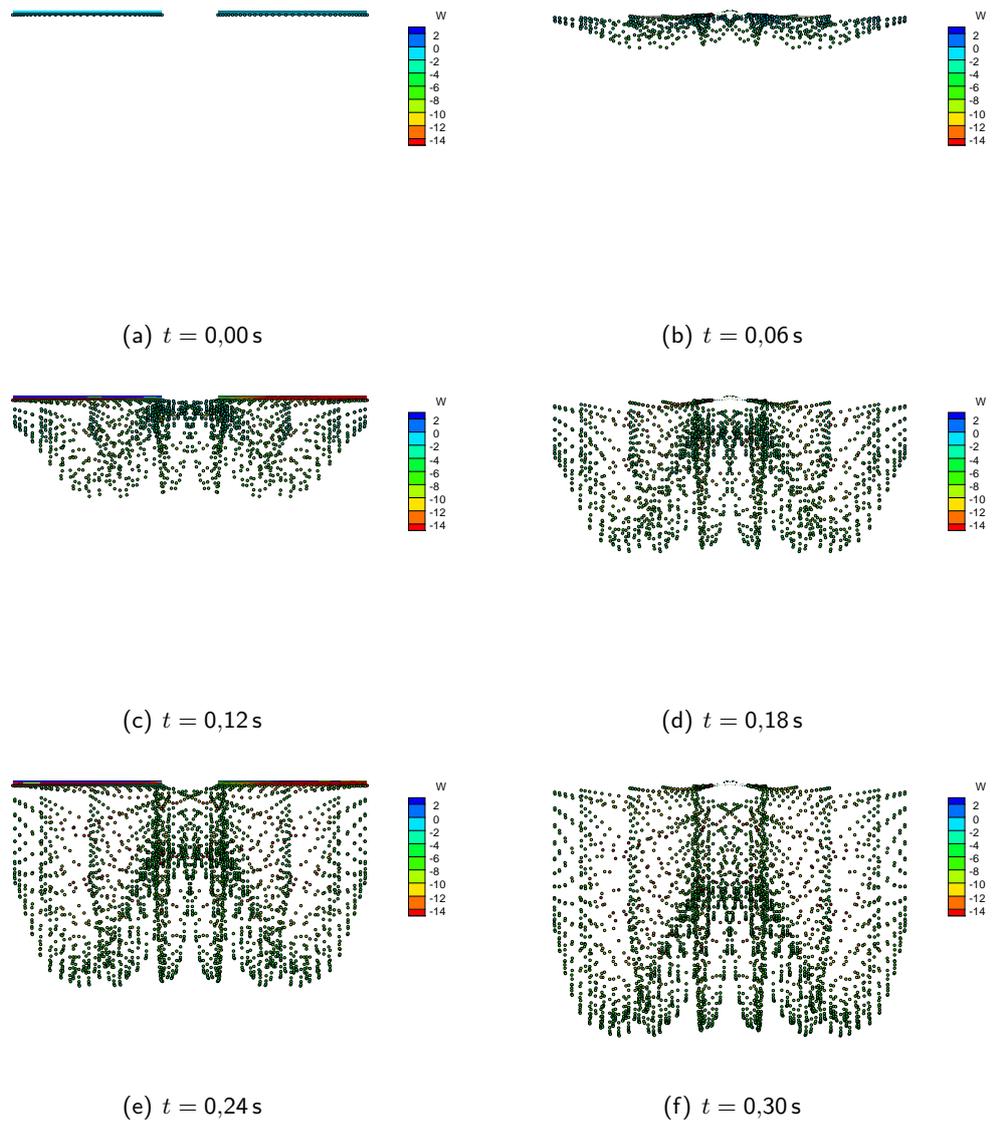


Figure 5.12: Axial velocity for a simple hovering test case and No Roll-up Wake

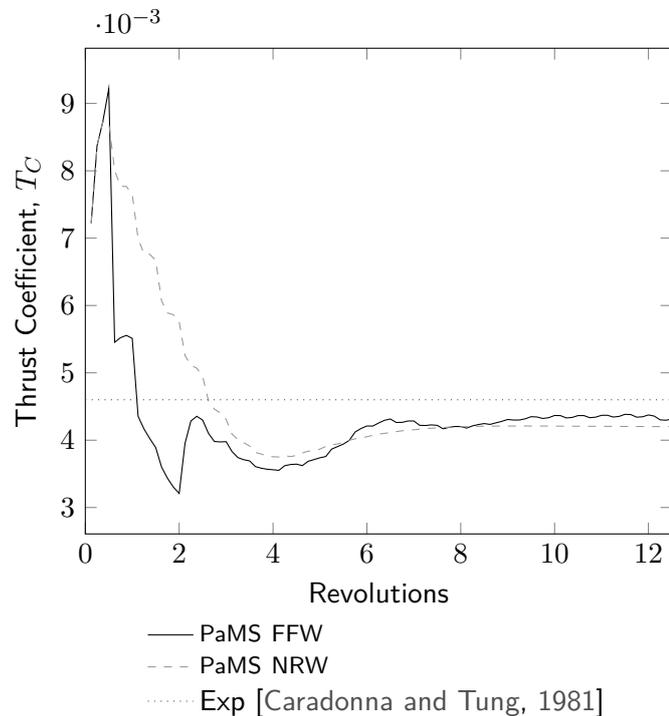


Figure 5.13: Thrust coefficient in hovering

5.5 Motion law test

In order to test and verify the possibility to provide external motion routines to PaMS, here, according to the aim of this thesis, a simple tilting rotor case has been investigated.

5.5.1 Tilting Rotor

The case described in this section provides a basic analysis of a tilting phase performed by the same 7,5 ft (2,286 m) diameter rotor considered in the hovering test case in section 5.4.1. The full motion routine is reported in listing D.1 on page 123 and it has already been described in section 5.2.3, whereas in figure 5.15 a time-event correlation is shown, in order to clarify the motion law assigned to the object. Moreover, this figure serves as a legend to understand the color scheme adopted on the backgrounds of figures 5.17 to 5.19, where the different color bands refers to the time phases illustrated in figure 5.15 itself. In particular these phases are:

- at $t = 0$ s, from a stationary position, the rotor begins its revolutions around the shaft axis with an angular speed of $\Omega = 1250$ rpm;
- as it starts to revolve, the rotor is subjected to a vertical translation at a constant rate of 10 m/s;
- the climb phase ends at $t = 0,096$ s, when a hovering phase lasting $t = 0,048$ s begins;

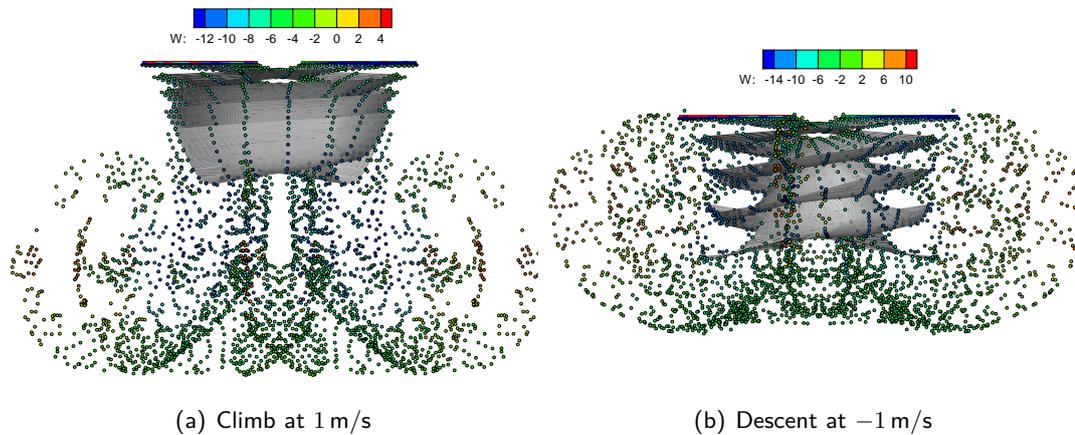


Figure 5.14: Axial velocity for climb and descent with Fully Free Wake

- at $t = 0,144$ s, until $t = 0,264$ s, the tilt does happen at $\Omega_t = \frac{\pi}{2} \cdot \frac{100}{12}$ rad/s, so that a quarter of revolution is completed into the tilting time (amounting to 0,12 s);
- when the tilt is complete, a horizontal translation is done at 10 m/s until the final simulation time of 0,6 s is reached.

As a consequence of the imposed velocities, the motion of the rotor is purely longitudinal and the trajectory described by a point ideally integral to the aircraft (whose attitude is assumed to be fixed in space) is *L-shaped*, whereas each point belonging either to the rotor or to its nacelle other than the center of rotation, is characterized by a more complex planar trajectory, as clarified in figure 5.16, where a V-22 Osprey is ideally depicted in subsequent moments of the tilting phase. Another insight into the kinematics of the motion is provided by figures 5.17 and 5.18, where the time history of the pivot coordinates and velocity are reported, respectively. Obviously the data reported in figure 5.18 are coincident with those imposed into the DATAIN file and shown in figure 5.15.

To conclude the motion-focused analysis discussed in the present section, in figure 5.19 the time evolution of one of the two rotor tips is depicted. As expected, this plot shows a general sinusoidal trend (whose amplitude is equal to the rotor diameter) for the three components of the evaluation point (which are related to the global, fixed reference frame): this is obviously related to the circular motion verified when the device is not tilting. In particular, the probed point is initially characterized by a z value amounting to 1 m: this information, if compared with the pivot data of figure 5.17 suggests that the rotor has an eccentricity of 1 m with respect to the tilt pivot. Furthermore, during the ascent the oscillating components are x and y , meaning that the blades are rotating in the xy plane. Meanwhile the z coordinate is regularly increased of 1 m, in agreement with figure 5.17. When the tilt starts at $t = 0,144$ s all the three components are oscillating: another uniform circular motion is superimposed to the pre-existing one. In this phase both the x and z average values are decreasing, compatibly with the sketch shown in figure 5.16. At $t = 0,264$ s the tilt is over and the oscillating component of x is dumped out: from this point

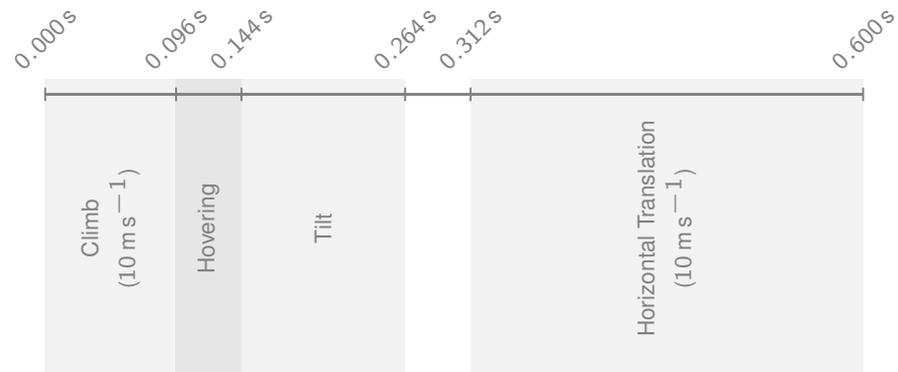


Figure 5.15: Time-event correlation for the tilting rotor case

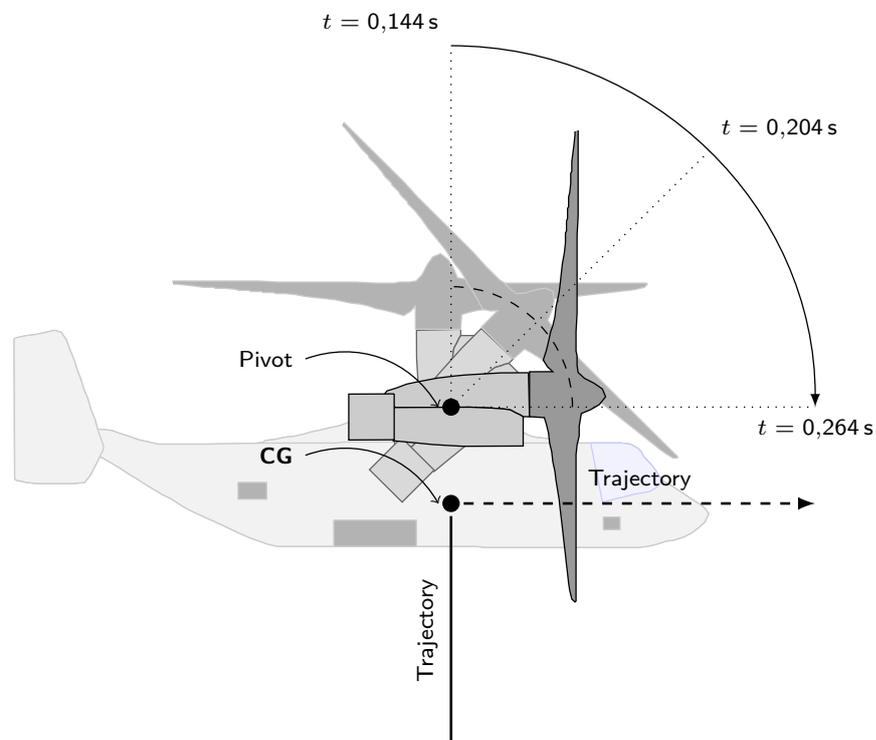


Figure 5.16: Path of the rotor center in the tilting rotor case

ahead the motion is bounded to the yz plane and, starting from $t = 0,312s$, the whole device travels for nearly 3m along the $-x$ direction. The last dutiful note concerns the combination of the three curves depicted in figure 5.19, which give a three-dimensional representation of the point trajectory (a constant pitch helix whose axis is L -shaped).

Finally, figure 5.20 shows the vorton wake in six different moments of the simulation.

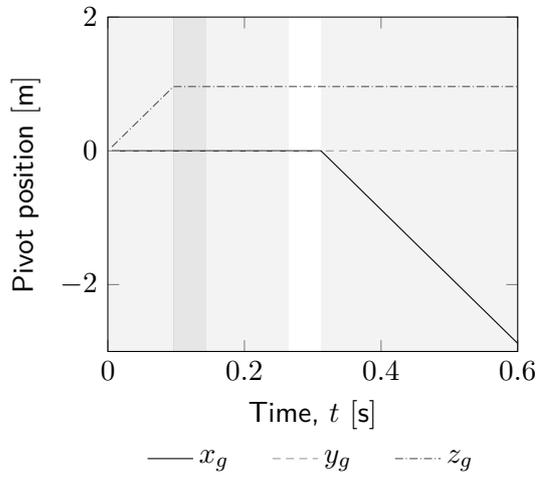


Figure 5.17: Pivot position history for the tilting rotor case (the background color refers to figure 5.15)

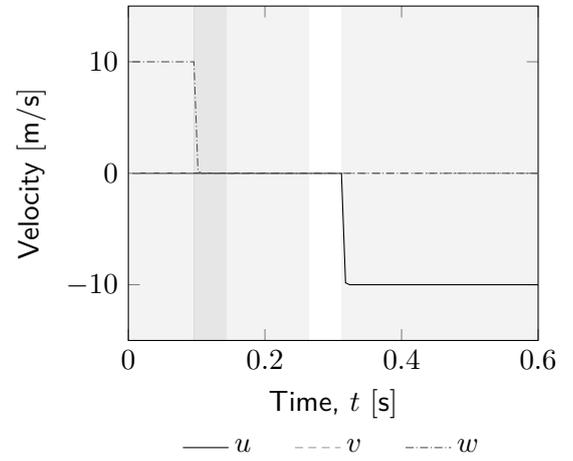


Figure 5.18: Velocity history for the tilting rotor case (the background color refers to figure 5.15)

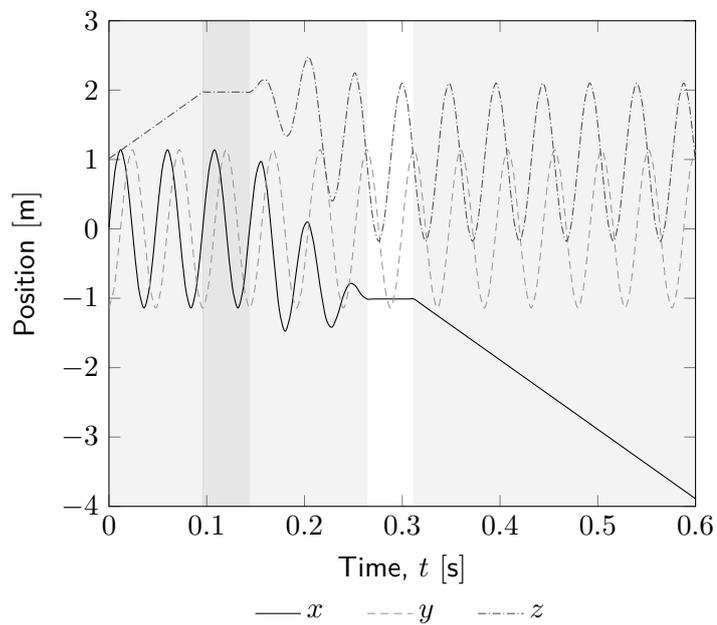


Figure 5.19: Rotor tip trajectory for the tilting rotor case (the background color refers to figure 5.15)

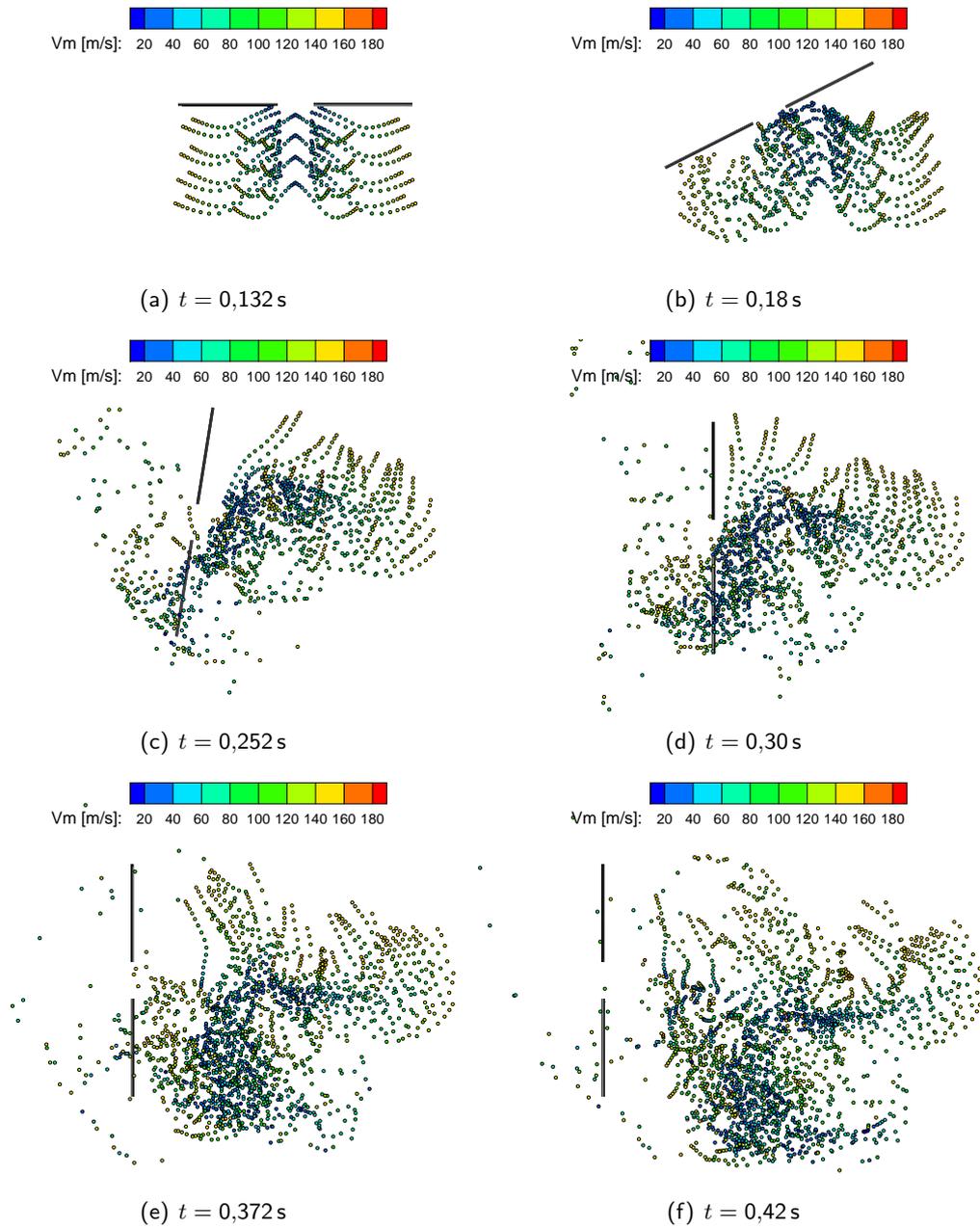


Figure 5.20: Average velocity for a preliminary tilting rotor case

6 Simulation

“*L*earning by doing, peer-to-peer teaching, and computer simulation are all part of the same equation. ”

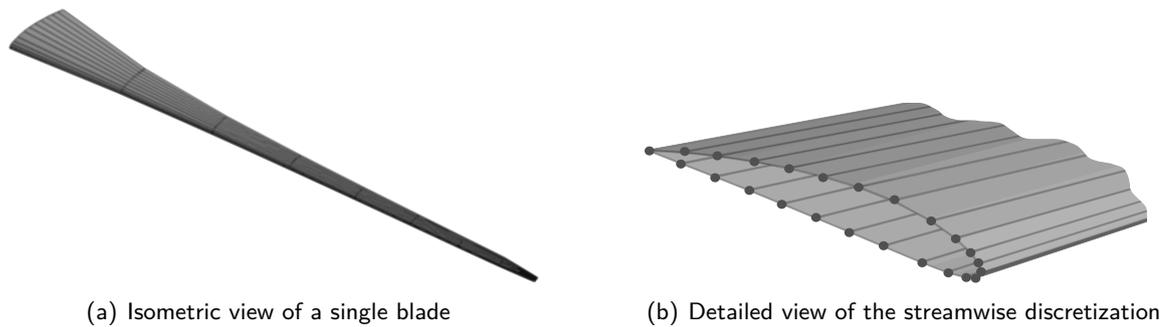
— *Nicholas Negroponte*

Having laid the foundations for a complete understanding of the aerodynamic problem dealt in this thesis, in the current chapter the results of the numerical analyses based on PaMS Vortex Particle Method implementation are reported. In particular all the steps leading to the study of the wing-wake interaction in a complex kinematic configuration such as the tilt phase of one of the two fore mounted E-Pteron propellers are described in detail.

6.1 Geometry discretization

As stated in section 2.4, the first step aimed at the aerodynamic study of the potential flow-field surrounding a body by means of the VPM concerns the discretization of its geometry into panel elements. Obviously, as opposite to the Computational Fluid Dynamics (CFD), the employed method requires the generation of a surface mesh, that in the present case involves only the propeller and the canard equipping the E-Pteron VUAV, since the hub and the nacelle have not been included into the calculation due to computational costs reasons.

The adopted meshing strategy is the same for both the objects: eight panels have been spread along the wingspan and twelve in the streamwise direction, both on the upper and the lower sides of the wing/blade. This choice is justified by two cooperating factors: the number of time steps to be employed in the simulation and the overall number of panels located in the domain. In fact, as it easy to think about, the computational cost and time depend on both these points, so that a compromise has to be found. However, the number of time steps is imposed by both the angular speed of the propeller (which rules the time resolution, and hence the time step amplitude) and



(a) Isometric view of a single blade

(b) Detailed view of the streamwise discretization

Figure 6.1: E-Pteron fore propeller discretization

the duration of the tilting phase (that is a design parameter). In the actual case, given that the design tilt time for the E-Pteron VUAV is of 5 s (*i.e.* the simulation has to span for at least 5 s), a reduced revolution speed – with respect to the ones discussed in section 4.2.2 – of 1250 rpm has been set for the rotor, so that a time step of 0,004 s could have been employed. Considering one more second of simulation for an ascent/forward flight phase, this unavoidable choice implies an overall number of time steps equal to 1500 and hence a not negligible computational time is expected. Therefore the total number of panels should be reasonably limited, and here comes the discretization strategy exposed before, which – since a total of three blades is present – leads to 960 panels.

6.1.1 E-Pteron fore propellers

The fore propellers proposed in section 4.2.1 did represent a greater challenge in the discretization process with respect to the canard, due to a more complex geometry characterized by a small taper ratio, an elevated twist variation along the wingspan and a little dimension in the streamwise direction. As a consequence, at the blade tip, the resulting discretization could result in narrow panels with high aspect ratio and located nearby each other. In particular, this latter factor may cause numerical issues in the computation of the inductions and should thus be avoided, as it is possible. The adopted solution to this problem is depicted in figure 6.1(b), where the employed streamwise panel distribution is highlighted by means of some big marks. In this image one should note that the panel density is greater in the fore part of the blade (*i.e.* near its LE) as here the greatest curvature of the airfoil shape is present. Moving towards the TE the panel density decreases until the 30% of the chord is reached: from this point forward such a density becomes constant and no tightening is provided at the TE. Finally, as it can be seen in the final result shown in figure 6.1(a), the discretization has been intensified toward the blade tip, where the highest velocities are obviously expected to be more significant.

6.1.2 E-Pteron canard

The canard, already described in figure 4.5, has a simpler geometry than the rotor blade, and hence it has been possible to employ a more conventional panel distribution. In effect, in addition

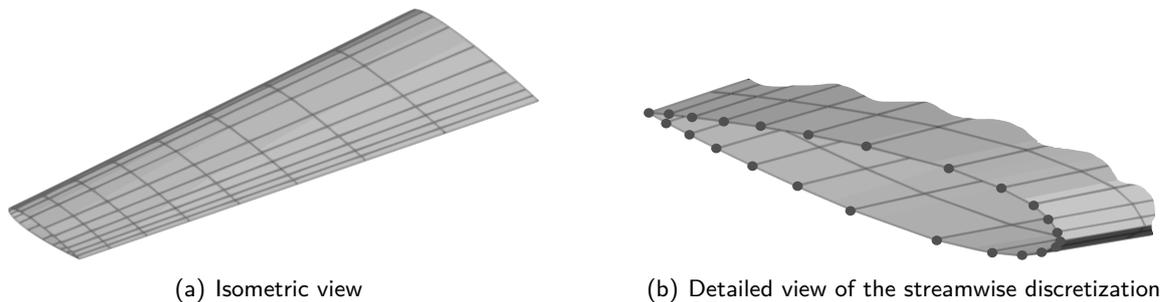


Figure 6.2: E-Pteron canard discretization

to the LE, as shown in the detailed view of figure 6.2(b), the panel frequency should be intensified at the wing Trailing Edge, too. Moreover, as depicted in figure 6.2(a), also in this case the spanwise panel density has been increased towards the wing tip, where the propeller is mounted and hence where the propeller-wing interaction is expected.

6.2 Propeller performance estimation

Before introducing the tiltrotor implementation, the geometries and the discretizations described in sections 4.2.1 and 6.1.1 have been employed to carry out a preliminary study concerning the isolated propeller. In particular, the analysis here described is aimed at the computation of the characteristic curves of the designed propeller in order to obtain a basis of comparison with the results reported in section 4.2.2.

The numerical investigation on the lone rotor has been performed at five different angular speeds n spacing from the lower limit of the E-Pteron flight envelope up to upper one, characterized by the achievement of the angular speed barrier discussed in section 4.2.2. Moreover, since the author was intended to perform a comparison among different methods as broad-spectrum as possible, four different techniques have been employed in the dimensional analyses aimed at the achievement of the thrust versus airspeed function: *Xrotor* (already exploited to carry out the results of figure 4.8 on page 55), a homemade BEMT code, a Trefftz technique and a pressure integration one, both by means of PaMS.

About *Xrotor* and the BEMT code, it could be argued that the first is based on the same theory which rules the latter, so that the expected results should be coincident. This is partially true, however the software coded by Mark Drela adopts some higher order refinements that are not present in the homemade software. Nevertheless, on the other hand, some corrections are available in this second code, too. In fact it can take into account both the tip losses (estimated by means of the *Prandtl correction* function [Tognaccini, 2011]) and the hub drag. In particular this latter correction has been excluded from the calculations here performed since the hub has been not included in the geometry to be analyzed with PaMS. Moreover, whereas *Xrotor* takes as input some relevant parameters employed to calculate the airfoil polar curves, the BEMT code extracts the sectional aerodynamic coefficients from *xfoil* [Drela, 2000, web]. Therefore, on the one hand *Xrotor* may result difficult to set up but is potentially not affected by drag estimation-

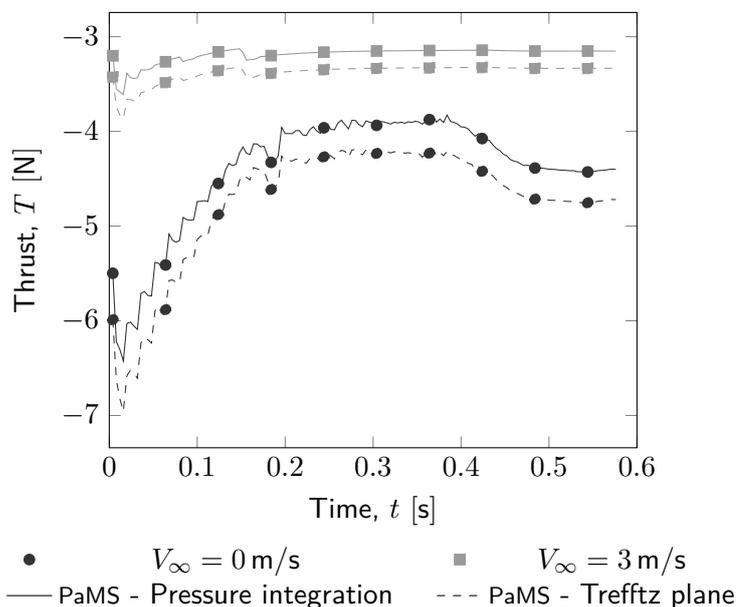


Figure 6.3: Thrust evolution through time at 1250 rpm for different free-stream velocities

related errors and, on the other hand, the BEMT code may produce bad results at low angles of attack due to *xfoil* limitations in drag computation.

Concerning the analyses performed by means of PaMS, they have been carried out using a set up (*i.e.* a DATAIN file) similar to that employed for the rotor hovering test case described in section 5.5.1. Obviously, the curves of interest have to be evaluated at the steady state, so that the unsteady calculation performed by the software should end at a sufficiently high final time t_{end} so that the transient could be considered as decayed. At this purpose, an example of time evolution of the thrust developed by the rotating propeller is reported in figure 6.3, where one could observe that the greatest is the free-stream velocity, the smallest is the transient time: this is due to the fact that the wake convection away from the propeller disk is faster at higher airspeeds.

The results of this preliminary analysis are shown in figure 6.4. As it can be noticed from figure 6.5, the agreement between *Xrotor* and PaMS predictions is always above the 94%. Moreover, figure 6.5(b) reveals that both the percentage error and the one averaged over the various values of the airspeed are increasing functions of the number of revolutions in the time unit. This behaviour can be justified by the fact that all the PaMS analyses have been carried out using the same temporal resolution (*i.e.* the same time step), so that the spatial resolution – and hence the accuracy of the solution – decreases as the propeller angular speed increases.

Finally, in figure 6.4(f) the dimensionless thrust profile as a function of the advance ratio J is depicted. The curves shown in this plot which concern the Vortex Particle Method are obtained from those shown in figures 6.4(a) to 6.4(e) and thus are presented as a collection of scatter data rather than as a continuous function. The bulk of these scatter data is concentrated at low values of J since the higher ones are achieved in a lower number of cases (when $n = 3000$ rpm

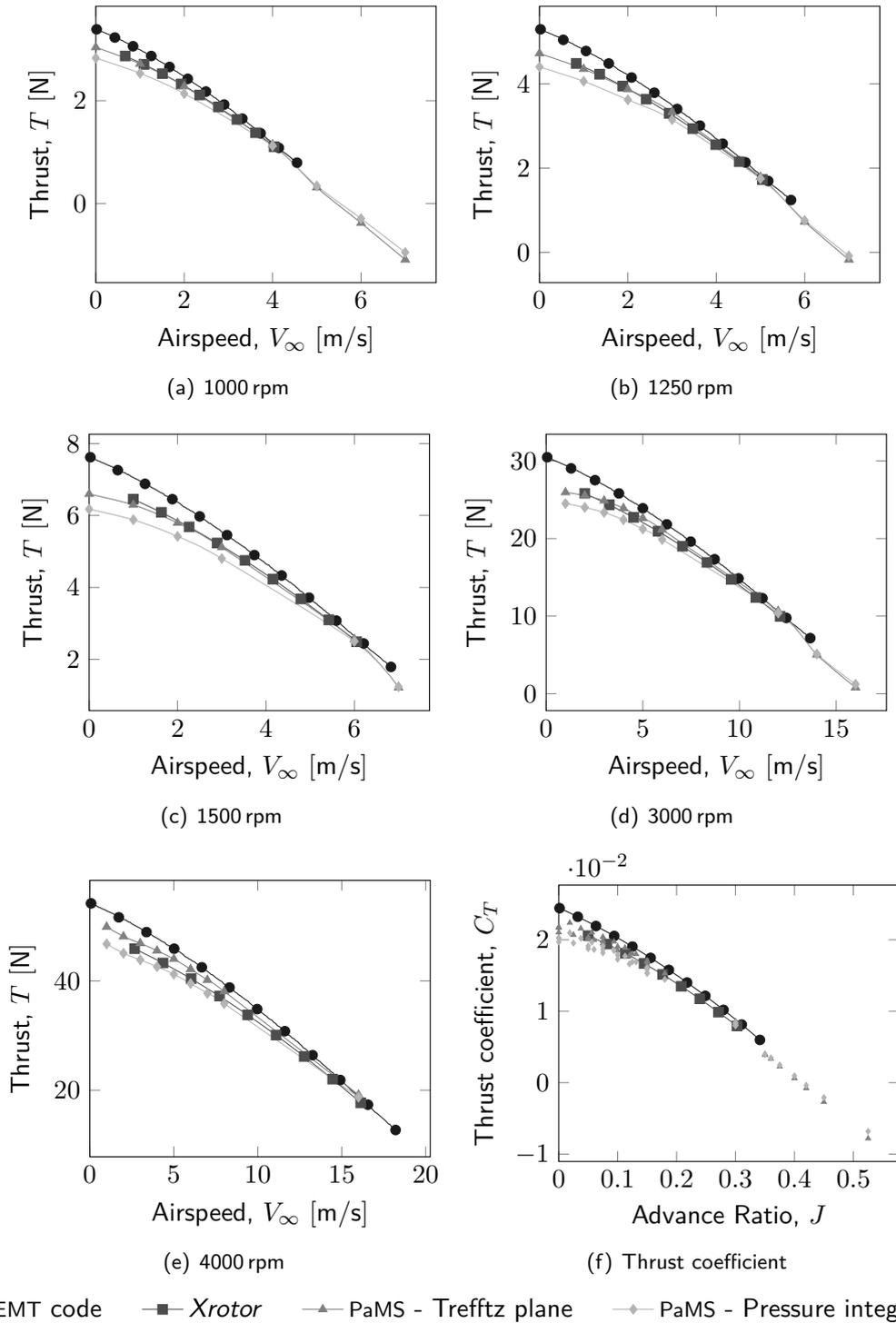


Figure 6.4: Propeller performances estimated by means of different techniques

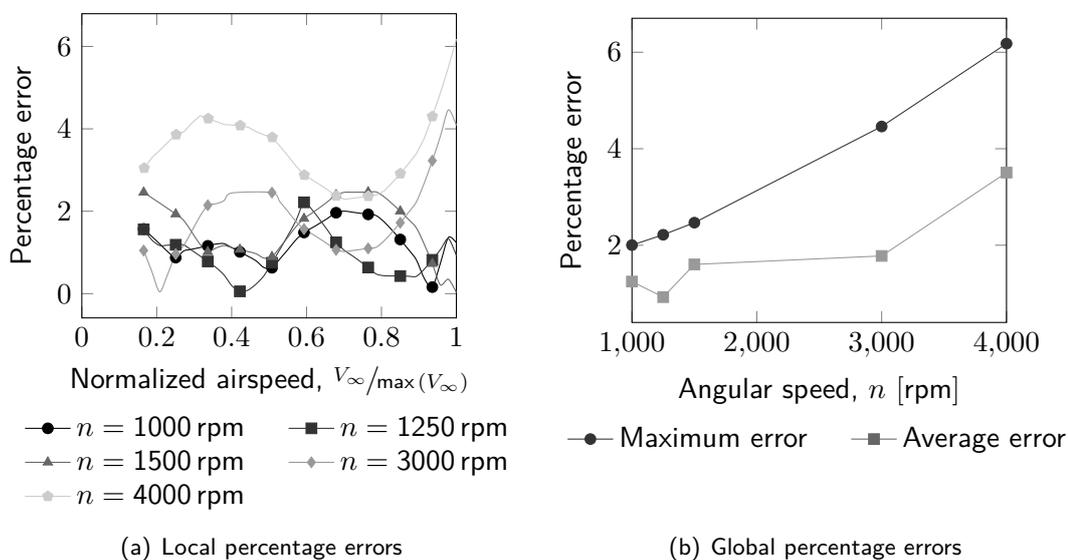


Figure 6.5: Percentage error on thrust for PaMS. The reference value is the *Xrotor* solution

and 4000 rpm).

6.3 Tilting rotor

After the propeller performance estimation carried out in section 6.2, another preliminary analysis has been performed: in order to implement and test the final motion law, the tilt phase of the isolated rotor has been investigated. In particular a longitudinal, pure tilt motion has been imposed by means of a law similar to that reported in listing D.1 on page 123 and already described in section 5.2.3 on page 77. In particular, the rotor axis is always contained in the $x - z$ plane and hence it rotates around the y axis describing a perfect quarter of circle: in fact, the simulated case is the numerical equivalent of a bench test, where the pivot is motionless in all the phases of the test (and hence even before and after the conversion). This choice is related to the fact that the motion routines built in PaMS are purely kinematic and hence the trajectory of the pivot due to the thrust generated by the rotor is unknown. As a consequence the coupling of the dynamics equations with the VPM is not in the scope of this thesis and hence it is deferred to future works.

In particular, a time-event correlation of the employed motion law is graphically represented in figure 6.6. This image is also useful to better distinguish between the different phases of the motion, since it can be used as a legend to interpret the background colors of figures 6.7 and 6.11. Finally, the three phases of the considered motion are:

- at $t = 0,00$ s, from a stationary position the propeller starts to rotate around its axis at a constant angular speed of 1250 rpm, and the rotor lies in a plane parallel to the $x - y$ one;
- at $t = 0,72$ s, once, based on the results of section 6.2, the steady state has been reached, the tilt begins at a rate of 3,125 rpm, so that an angle of 90 deg is travelled in a tilt time



Figure 6.6: Time-event correlation for the tiltrotor simulation

of 4,8 s;

- starting from $t = 5,52$ s and until the simulation end at $t = 6,00$ s, the rotor continues to revolve around its axis (which is now horizontal) at 1250 rpm, without moving forward.

The results of the analysis described above are plotted in figure 6.7, where it should be noted that, according with what has been said in section 6.2, the transient is already decayed when the tilt begins. In fact, since the tilt time is quite long and the modulus of the aerodynamic force (*i.e.* the thrust) produced by the propeller is almost constant from $t = 0,72$ s ahead, one can conclude that the current case of study is characterized by a quasi-steady flow. Furthermore, as expected, during the first steady phase depicted in figure 6.6 (*i.e.* for $t < 0,72$ s) the thrust vector is aligned with the z axis, which is vertical and points upwards, whereas after the tilt has been completed it is directed along the x axis. As it can be seen, the conversion is such that in airplane mode the thrust points in the opposite direction with respect to the x axis. This happens because the author intended to use the same law of motion for the basic tiltrotor configuration, too: usually a vehicle is drawn in a three-dimensional space fixing the reference frame on its foremost point and aligning the x axis with the fuselage, pointing towards the airplane tail. Therefore, in such a coordinate system (known as *body reference frame*), a pushing propeller should produce thrust along the $-x$ direction.

Another consequence of the transient decay is that the x and z components of the aerodynamic force during the tilt phase are nearly given by the analytical expressions

$$F_x = -T_{st} \sin(\Omega_t (t - t_{t_0})) \quad \text{and} \quad F_z = T_{st} \cos(\Omega_t (t - t_{t_0}))$$

where $T_{st} = 4,32$ N denotes the value assumed by the thrust at the steady state, $t_{t_0} = 0,72$ s is the starting tilt time and Ω_t is the tilt angular speed.

6.4 Propeller-wing interaction

At this stage it is possible to include the canard into the analysis performed in section 6.3 on the isolated propeller. In particular, also in this case the motion law sketched in figure 6.6 has been used, whereas the canard described in section 4.1 on page 48 has been considered as a non-aerodynamic body, so that it influences the solution being an obstacle for the propeller wake only. This choice simplifies the problem set up. In fact, since at the beginning of the simulation

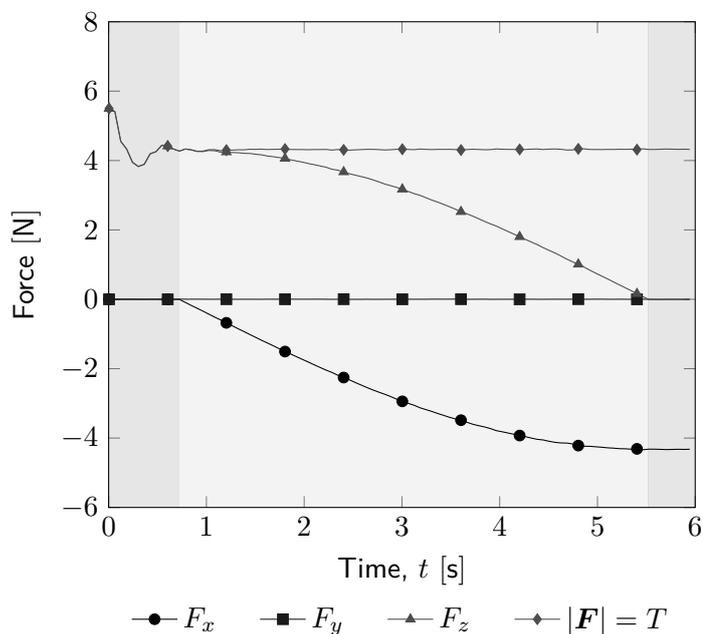


Figure 6.7: Force history for the isolated tilting propeller (the background color refers to figure 6.6)

the canard works at very high angle of attacks, the Kutta condition does not hold and the closure of the potential problem cannot be achieved (the reader should remember that flow separation is beyond the potential methods). Moreover, as a consequence, the forces computed by PaMS and acting on this body are not physical and hence should not be considered, therefore the application of an engineering method for the estimation of the drag at these high incidences has been developed in section 6.4.1.

The complete DATAIN file employed in this case is reported in listing E.1 on page 125. Of particular relevance on this code is the definition of the multibody problem, which, as described in section 5.2.2, requires the presence of `nbody` different *Geometry and Closure* and *Body Motion* sections. The first of these two logical sections should be clear to the reader, whereas the latter one deserves an insight, in order to gain a better understanding of the set up of multibody problems. At this purpose the core part of the *Body Motion* sections related to both the propeller and the canard are respectively reported in listings 6.1 and 6.2, so that the reader can focus his attention on them.

Listing 6.1: Propeller body motion, extracted from the DATAIN file

```
&GEOTRA GT(1,1)=-0.182 GT(1,2)=0.0 GT(1,3)=0.0 &END
&GEOTRA GT(2,1)=0.0 GT(2,2)=0.0 GT(2,3)=0.0 &END
&GEOTRA GT(3,1)=0.0 GT(3,2)=0.0 GT(3,3)=0.0 &END
&GEOROT GR(1,1)=0.0 GR(1,2)=+90.0 GR(1,3)=0.0 &END
&GEOROT GR(2,1)=0.0 GR(2,2)=0.0 GR(2,3)=0.0 &END
&GEOROT GR(3,1)=0.0 GR(3,2)=0.0 GR(3,3)=0.0 &END
```

```

&SYSTRA ST(1,1)=0.0 ST(1,2)=-1.1 ST(1,3)=0.0 &END
&SYSTRA ST(2,1)=0.0 ST(2,2)=0.0 ST(2,3)=2.0 &END
&SYSTRA ST(3,1)=0.0 ST(3,2)=0.0 ST(3,3)=0.0 &END
&SYSROT SR(1,1)=0.0 SR(1,2)=0.0 SR(1,3)=0.0 &END
&SYSROT SR(2,1)=0.0 SR(2,2)=0.0 SR(2,3)=0.0 &END
&SYSROT SR(3,1)=0.0 SR(3,2)=0.0 SR(3,3)=0.0 &END

```

Listing 6.2: Canard body motion, extracted from the DATAIN file

```

&GEOTRA GT(1,1)=-0.163 GT(1,2)=0.0 GT(1,3)=0.0 &END
&GEOTRA GT(2,1)=0.0 GT(2,2)=0.0 GT(2,3)=0.0 &END
&GEOTRA GT(3,1)=0.0 GT(3,2)=0.0 GT(3,3)=0.0 &END
&GEOROT GR(1,1)=0.0 GR(1,2)=0.0 GR(1,3)=0.0 &END
&GEOROT GR(2,1)=0.0 GR(2,2)=0.0 GR(2,3)=0.0 &END
&GEOROT GR(3,1)=0.0 GR(3,2)=0.0 GR(3,3)=0.0 &END
&SYSTRA ST(1,1)=0.0 ST(1,2)=0.0 ST(1,3)=0.0 &END
&SYSTRA ST(2,1)=0.0 ST(2,2)=0.0 ST(2,3)=0.0 &END
&SYSTRA ST(3,1)=0.0 ST(3,2)=0.0 ST(3,3)=0.0 &END
&SYSROT SR(1,1)=0.0 SR(1,2)=0.0 SR(1,3)=0.0 &END
&SYSROT SR(2,1)=0.0 SR(2,2)=0.0 SR(2,3)=0.0 &END
&SYSROT SR(3,1)=0.0 SR(3,2)=0.0 SR(3,3)=0.0 &END

```

Obviously, the aim of the two combined *Body Motion* sections is to correctly place the two geometries one with respect to the other. Moreover, since in the current analysis a relative motion between the parts has to occur, the arrangement that has to be attained should be referred to the starting time of the simulation. The target configuration for the rough tiltrotor case taken into account is shown in figure 6.8, where some useful lengths are also reported. In this image O_c represents the canard reference frame, which is coincident with the global one and is located on the nose of the whole vehicle (here not considered). On the other hand, O_p is the propeller coordinate system and here it is represented after the $ST(1,2) = -1.1$ operation reported in listing 6.1. In effect, according to figure 4.1 the propeller is located at the canard tip, which is 1,100m away from the global symmetry plane $X - Z$, along the $-Y$ direction. Furthermore, the hinge is located 0,083m behind the canard LE, which has a global x coordinate of 0,080m, so that the whole canard has been translated of $-0,163$ m in this direction and in his own reference frame (*i.e.* a GEOTRA has been employed). In fact, the strategy adopted in order to implement the tilt motion consists into aligning the hinge axis with the Y one. At this stage the propeller center lies on the hinge axis and the rotor is oriented so that the thrust is produced along the $-x$ direction. Therefore other corrections have to be applied to the rotor positioning: an eccentricity of 0,182m between the hinge axis and the rotor center has to be provided and a rotation around the Y axis is required, too. These tasks are accomplished by means of two GEO operations, so that O_p still lies on the hinge axis and a GEOROT can be used in the motion routine, as in listing D.1. The final result is shown in figure 6.9 for both the start and the end of the simulation.

The meticulous reader could have noted that in listing 6.1 a constant z velocity of 2.00 m/s is imposed to the propeller, even though this parameter is not compatible with the motion that the propeller is supposed to execute. In fact this value is overridden in the motion routine – where the command $SYST(1,2,3)=0.0$ is present – and, according to the dissertation carried out in

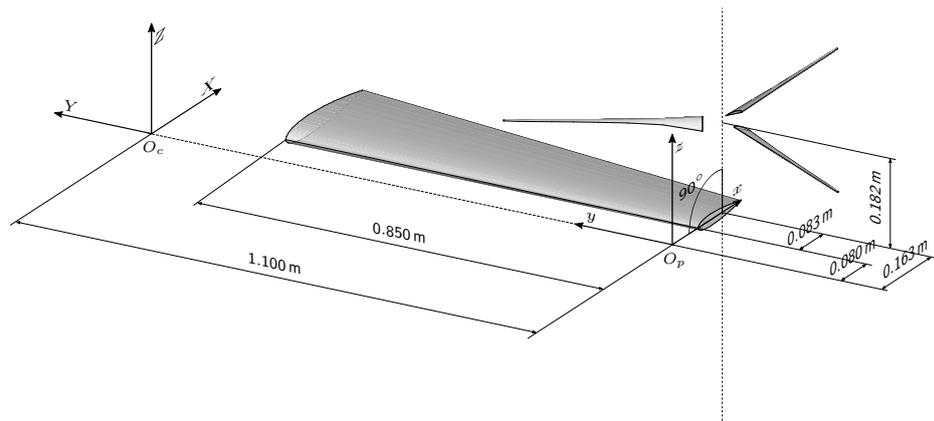


Figure 6.8: Reference frames operations for the tiltrotor

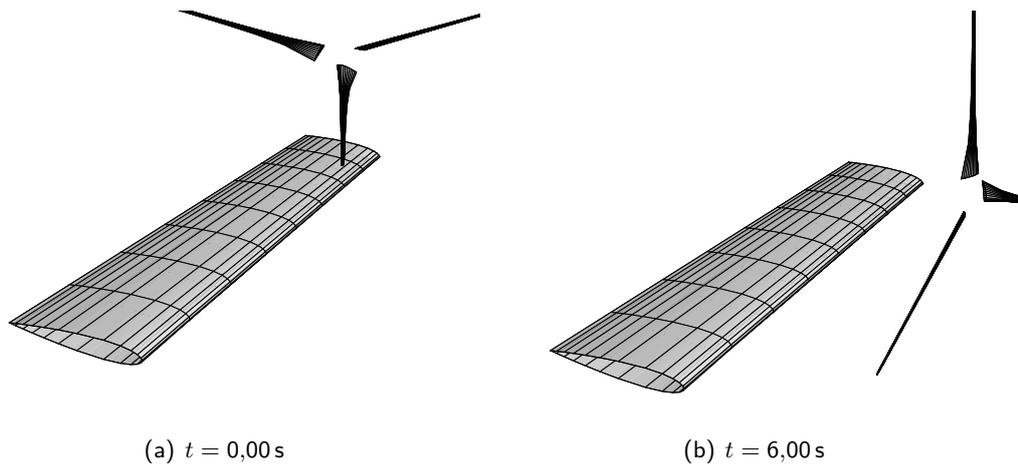


Figure 6.9: Assembly mesh for the tiltrotor

section 5.2.2, it has been written in order to help PaMS to identify the edges which require the enforcement of the Kutta condition. This is due to the fact that the software performs this task considering also the initial movement prescribed through the DATAIN.

In figure 6.10 the streamlines related to both the final hovering time and to the final simulation one (*i.e.* $t = 0,72$ s and $t = 6,00$ s, respectively) are depicted. As expected, since the axial velocity increases due to the speed up imposed at the rotor disk and according to the mass conservation law $\dot{m}/\rho = VA = V\pi D^2/4 = \text{const}$ (remember that the flow is assumed to be incompressible), the particle paths are helices whose diameter decreases as the travelled distance increases. Moreover, the small pitch of the helices shown in figure 6.10(a) is justified by the zero free-stream velocity, so that it is expected to become greater and greater as V_∞ increases. Furthermore, being the helix angle a function of both the axial and tangential velocities ($\theta_h = \arctan(\sqrt{u^2 + v^2}/w) = \arctan(\Omega r/w)$, where V has been decomposed in the propeller reference frame), this pitch varies with the radial coordinate r and it is greater for small values of r since here the tangential speed Ωr is smaller and hence comparable with the axial one. Moreover, as highlighted by the colors of figures 6.10(a) and 6.10(b), the velocity magnitude $V = \sqrt{u^2 + v^2 + w^2} = \sqrt{(\Omega r)^2 + w^2}$ linearly increases moving from the propeller hub towards its tip.

Finally, the force history for the basic tiltrotor configuration considered here is reported in figure 6.11, whereas a time-stepping graphical representation of the solution is shown in figure 6.12. The differences between these results and those concerning the isolated tilting rotor (depicted in figure 6.7) are just minimal and related to the transient time. This could be caused by the relative position of the two bodies, whose overlap is restricted to less than half of the rotor wake. Moreover, the intersection area between the canard and the wake is an inverse function of the tilt time and, since the wing chord is much greater of its thickness, a cosine relation nearly holds. For this reason the presence of the canard is relevant at the early stages of the simulation only, even though a half-span located propeller could be more influenced by the wing presence.

6.4.1 Canard drag estimation

As said in section 6.4, the closure of the potential problem by means of the enforcement of the Kutta condition to the canard is not possible in the early stages of the simulation, since in these phases the canard is hit by the high angle of attack flow generated by the propeller (whose axis is normal to the canard mean plane). However, in order to properly dimension the canard beams and to formulate a suitable control law for the two tilting propellers (*i.e.* to find the relation $n = n(t)$ which allows to perform a certain manoeuvre and to counteract the forces induced by the propellers wakes on the canard), it is useful to have at least an estimation of the drag acting on the canard due to the rotors.

At this purpose an engineering method has been employed:

- as shown in figure 6.13, the actual pressure distribution over the upper surface of the canard has been extracted by the PaMS generated FILEOUT.tecplot file, and in particular the solution at the end of the hovering time (*i.e.* at $t = 0,72$ s) has been considered, so that the transient has decayed;
- since on the lower side of the canard huge expansion is expected due to flow separation and hence no valid information are provided by PaMS, here the pressure P_l has arbitrarily

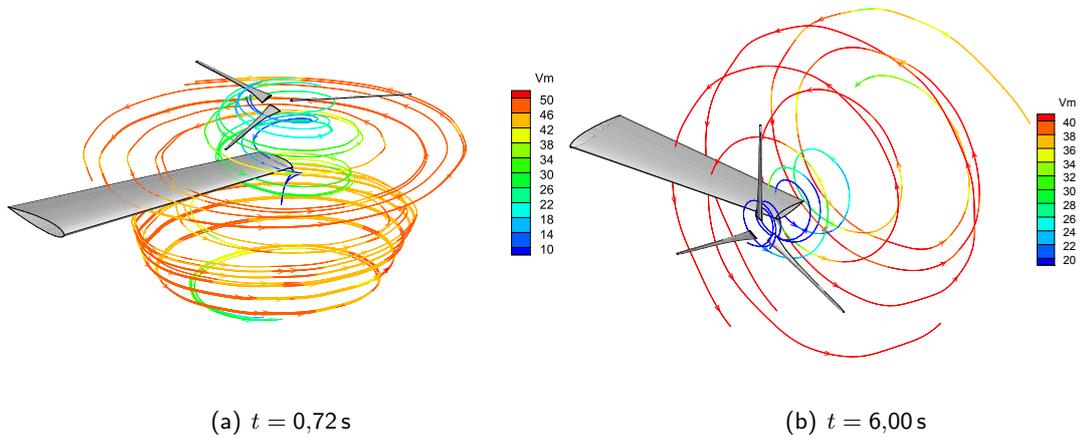


Figure 6.10: Streamlines for the tiltrotor case

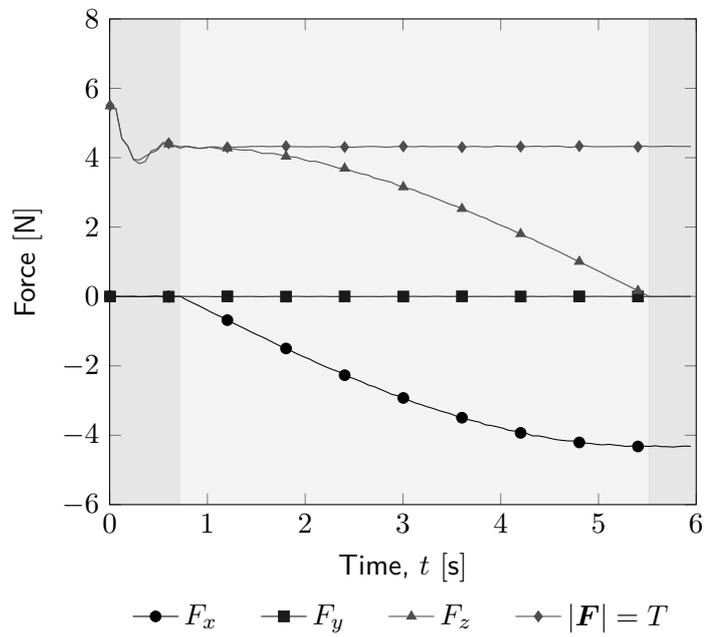
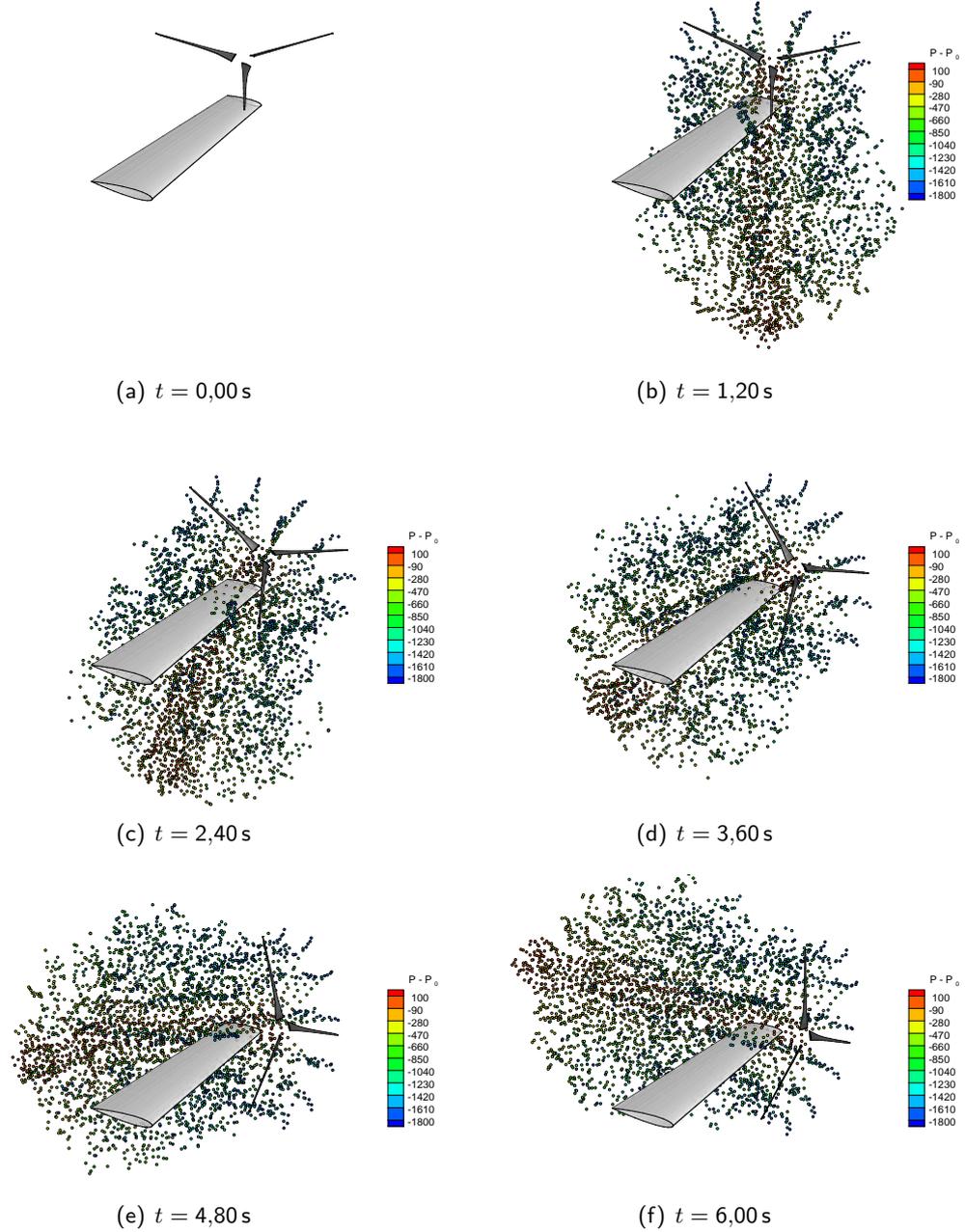


Figure 6.11: Force history for the tiltrotor configuration (the background color refers to figure 6.6)

**Figure 6.12:** Pressure distribution for the tiltrotor case

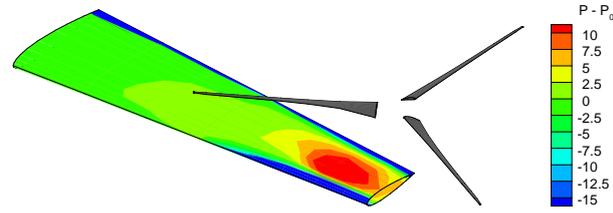


Figure 6.13: Pressure distribution over the canard at $t = 0,72$ s

been set to a value such that $C_{pl} = -0.1$. In fact, this value is typical for separated flows behind both flat plates [Fail et al., 1957] and backward-facing steps [Rajasekaran, 2011]. Moreover, it should be noticed that, since the free-stream velocity is zero, the classical definition of the pressure coefficient does not hold and hence a reference speed equal to $\max(|\mathbf{V}|) = 52.64$ m/s has been employed. In particular, two different pressure distributions based on this value have been considered: the first one assumes a uniform distribution over the lower surface of the canard and should thus be conservative, whereas the other takes into account that the difference $P - P_0$ is maximum near the wing tip and almost zero nearby its root. Specifically, this latter solution is intended to be more realistic and is based on an exponentially modulated sine, according to

$$\bar{P}_l = - \frac{\sin\left(\frac{2y}{b_c}\pi\right) e^{-5y}}{\max\left[\sin\left(\frac{2y}{b_c}\pi\right) e^{-5y}\right]} P_l,$$

where the $[\bar{\cdot}]$ symbol refers to the spanwise pressure distribution and the subscript c denotes the canard. Finally, the chordwise distribution is assumed to be a sine whose amplitude is equal to $c_c(2y/b_c) = c_c(\bar{y})$, so that the ambient pressure is recovered on the edges of the canard.

- The pressure difference evaluated on the basis of the previous steps is then numerically integrated over the canard surface, resulting in the drag force and in its application point (*i.e.* the center of pressure, C.P.). In particular the area of each panel has been projected on the canard mean plane, being this process equivalent to taking the pressure integral in the direction parallel to the incoming airflow.

Even though this method is obviously approximated and highly sensible to the value assigned to the pressure acting on the expansion side of the canard, it may be extremely useful, for example, to assess the minimum rate of rotation n_h of the fore propellers which allows the hovering. In fact, the data shown in figure 4.9 on page 56 do not take into account the apparent weight due to the drag force induced by the rotating propellers. In this context a shift of the hovering point towards higher values of n (and hence a reduction of the vertical flight envelope) is expected and could be estimated by means of an iterative procedure:

Table 6.1: Estimation of the drag acting on the canard due to the propeller. The data are referred to half of the total wingspan

Estimated drag, [N]	
Uniform	Modulated sine
29.21	3.93

1. the first attempt value for n_h is taken to be the one evaluated in section 4.2.2 (*i.e.* 2688 rpm);
2. the analysis is performed and the drag force is estimated;
3. this value of the drag is added to the vehicle weight;
4. the new value of n_h is found by means of the propeller characteristic curves;
5. the convergence is checked and, wheather needed, the process is iterated jumping back to the step 2.

However the implementation of this procedure is beyond the objectives of this thesis and it is therefore recommended to be carried out in future works. Nonetheless, the results related to the first two steps of the first iteration have been calculated by means of the two drag estimation techniques described before in this paragraph and are thus reported in table 6.1. In particular, both the uniform and the modulated sine pressure distributions on the lower side of the canard imply a center of pressure located nearby the wing tip and whose x coordinate does not differ much (some millimeters) from those of the nacelle pivot. As a consequence no severe torque and a relatively slight bending are expected to act on the canard.

6.4.2 Canard lift distribution

An approach similar to that applied in section 6.4.1 has been employed in this paragraph to the lift distribution. In particular, the simulation time $t = 6,00\text{s}$ has been considered and the corresponding pressure distribution over the top surface of the canard has been depicted in figure 6.14(a) as a contour and in figure 6.14(b) as pressure profiles along the wingspan. Since at the considered time the canard is hit by a flow almost parallel to its chord (*i.e.* almost lying in its mid-plane), in this case the potential problem is well conditioned and the data provided by PaMS can be employed without additional assumptions. Moreover, as in section 6.4.1, it should be noticed that, since the free-stream velocity (and hence the dynamic pressure) is zero, in the current case the pressure coefficient C_P cannot be defined in the canonical way so that the data reported in figure 6.14(b) have been nondimensionalized with the maximum value assumed by the pressure difference $P - P_0$ and should be considered as a source of qualitative information only. From figure 6.14 one can notice that, as expected, the propeller influence on the pressure distribution is bounded to the wing tip, where a negative angle of attack is induced by the propeller rotation (as it can be observed in figure 6.10(b)). For this reason, even though the canard is symmetric, untwisted and with zero angle of incidence, the overall lift acting on it at $t = 6,00\text{s}$

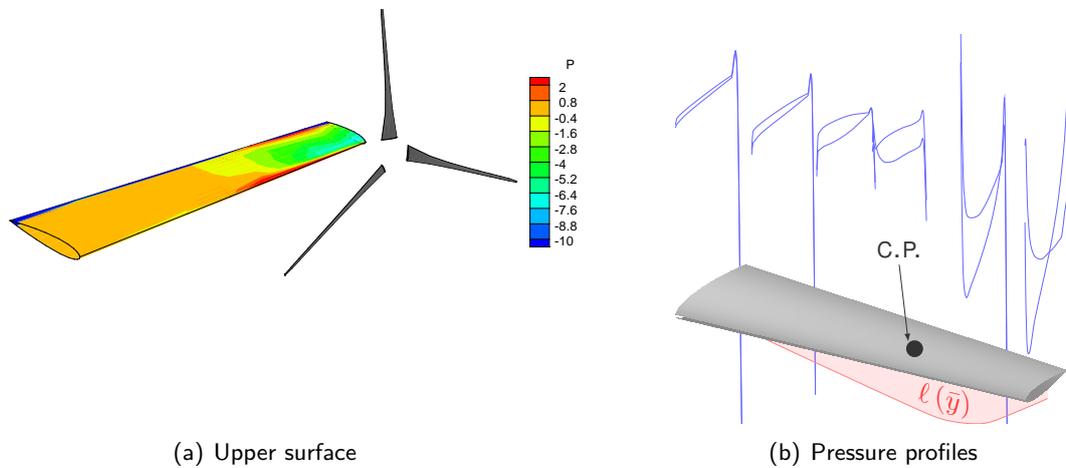


Figure 6.14: Pressure distribution for the tiltrotor case at $t = 6,00$ s

is not zero, but slightly negative ($L = -0,12$ N) and thus a bending moment of nearly 0.1 N·m acts on the canard due to the propeller induced flow. Finally, if the two fore propellers are not counter-rotating, a roll torque of about 0.2 N·m is expected due to this force, whereas a peach moment (whose magnitude depends on the location of the center of gravity and whose direction is related to the sense of rotation of the two propellers) rises for co-rotating rotors. On the other hand, during the tilt, if the two rotors have the same sense of rotation and due to the *Draper's law*, the nacelle tilt causes both a yaw and a slight roll moment.



Conclusions and future developments

6.5 Conclusions

Even though the CFD calculations are far more flexible than the potential flow solvers, meaning that their scope of applicability is wider, this technique has been proved to be inadequate to the study of complex configurations due to the limitations of the actual hardware capabilities. On the other hand, if the incompressibility and lack of viscous effects assumptions are met, the panel methods can be successfully employed as a bridge over some of the CFD gaps. For example, this is the case of a small tiltrotor performing the conversion from helicopter to airplane mode.

In this context, the aim of the present work was to prove the possibility to exploit the Vortex Particle Methods to predict the time evolution of the thrust generated by a tilting rotor, with and without the presence of a wing acting as an obstacle to the development of the propeller wake. In particular, at first a suitable propeller has been designed by means of an inverse method based on the Blade Element Momentum Theory and on the available requirements, such as the thrust to be produced and the blade radius. The off-design performances of this propeller have been investigated and, on their bases, the flight envelope has been estimated for both vertical and forward balanced flight. At this stage, the effectiveness and the accuracy of the VPM have been proved comparing the potential results with those produced by two different BEMT codes. This comparison produced good agreement between the two techniques and, moreover, the potential method gave some hints about the flow field both on the body surfaces and in the vortons wake.

Since the target simulation involves the rotation of the propeller also around the wing-wing axis, a suitable motion routine has been written and tested on the isolated rotor. In particular a virtual bench test has been considered, that is to say that the tilt pivot is motionless in the three-dimensional space and hence the free-stream velocity is zero.

Finally, the lone propeller configuration has been refined by adding a fixed wing which interacts with the rotor wake shielding the vortons floating around it. In particular, since a potential solution is not suitable for separated flows, the wing has been considered as a non aerodynamic body, so that, for instance, it is not possible to evaluate the drag acting on it due to the propeller action. Nevertheless an engineering method has been applied in order to at least estimate this drag

force. Furthermore, the thrust evolution over time has been computed for this wing & propeller configuration, aside with the lift distribution on the first one when the tiltrotor is in airplane mode. Moreover, these data will be employed to optimize the flight mechanics and dynamics and the controllability of the vehicle E-Pteron.

The extreme conclusion of this work is thus that the VPM could be successfully employed in the early stages of the design process of an aircraft, thanks to low computational costs and hence to the possibility to test different configurations. For instance, considering a conventional airplane equipped with two co-rotating propellers, due to the fact that an asymmetric flow-field is expected even in forward flight, a nonzero lateral force Y rises. In this context the method could be exploited to find out the optimum shape of the vertical fin which allows to balance the propellers induced lateral force Y in cruise conditions.

6.6 Future developments

The work presented in this volume is far than complete and it is thus intended to be extended by several future investigations. At first, with the goal to employ the data in the development of a real vehicle, since it has been proved that the required rates of rotation for the propellers are quite higher than the ones considered in this thesis, it could be useful to repeat the analyses performed in the previous chapters for a wider range of angular speeds. Moreover, other simulations than the bench test ones should be carried out and at this purpose it could be extremely useful to equip the potential flow solver with the capability to take into account the dynamics of the object under investigation. Even though this seems the best way to perform a complete and resolute analysis, a solution simpler to code could consist on a hand made coupling implemented by means of an iterative process. Whatever the technique employed, the results found out could be precious in a control perspective, that is to say to understand the angular speed required by each propeller to stabilize the vehicle in a particular situation.

Furthermore, in order to evaluate the flight envelope with higher level of accuracy, some empirical methods usually adopted for the helicopters or the wind turbines [see for example Montgomerie, 1996] could be employed to estimate the drag of both the fuselage and the wing during the vertical flight phases. Obviously these considerations are expected to reduce the flight envelope.

Appendices

A

Points cloud treatments

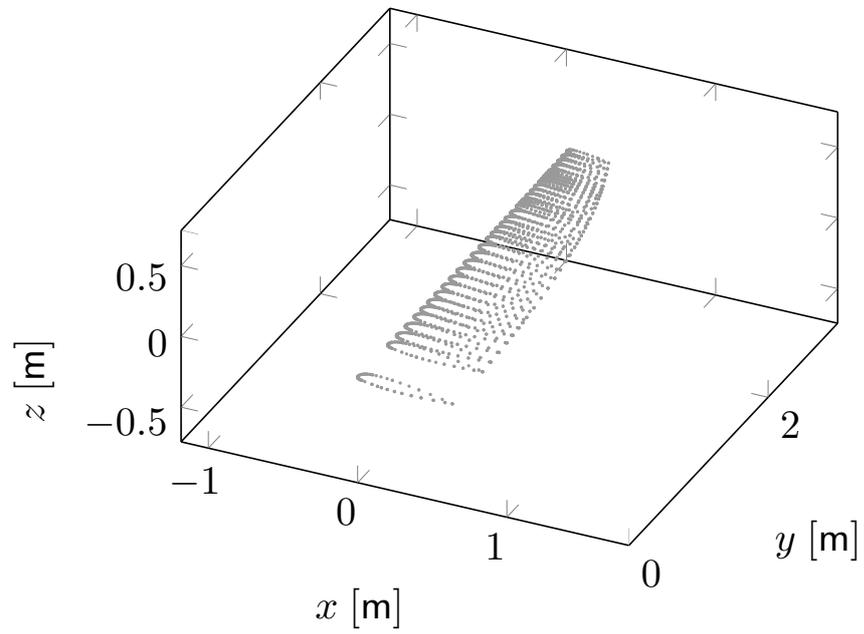
The E-Pteron geometry file provided in order to perform the analysis discussed in this document was a IGES CAD file, so that it could be immediately employed to discretize the canard geometry into panels. However, on the other hand, the IGES alone is not sufficient to extrapolate some aerodynamic characteristics of the wing such as, for instance, the chord distribution over the wingspan or the function $\Lambda(\bar{y})$. As a consequence an algorithm able to extract these information from the CAD file itself has been elaborated and it is illustrated in this appendix.

The basic idea is to convert the IGES file in the STL format, where the data related to the body, discretized into triangular elements, are provided. In fact, since the coordinates of these triangles vertices are explicitly listed in the STL file, they can be easily employed to construct a points cloud, which nevertheless requires some treatments in order to appear rightly structured and hence employable to its intended purpose. In fact, as it can be observed in figure A.1, although it could appear rather ordered, the original data structure is characterized by the presence of some messy elements, as emphasized in figure A.1(b). This figure in effect shows that even though the original data set is made of point groups located at different stations along the spanwise direction, some points appear to be isolated from the bulk, hence breaking this structure. As a consequence, the chosen approach has been to locate the main span stations containing the most of the points belonging to the same wing section and to condensate the other points in the nearest station. In order to achieve this task, some *influence areas* have been defined for each identified section, as shown in figure A.2. Here one can notice that each area boundary lies halfway between two subsequence sections, so that in general the influence of a station in the surrounding region is not symmetrical on its two sides. At this stage, each point influenced by a main span location and not belonging to it (*i.e.* each fault highlighted in figure A.1(b)) is transferred on it by means of a coordinate shift. This procedure quickly results in a data structure where each single point belongs to a well defined wing section, so that these airfoils can be isolated and studied independently in order to evaluate quantities such as the chord or the twist distribution along the wingspan. Moreover, the data related to different airfoils could be employed to perform an aerodynamic or stability analysis based on the *strip theory*. However, it should be noticed that the airfoil points are almost useless if the location of both the Leading Edge and the Trailing Edge is not known,

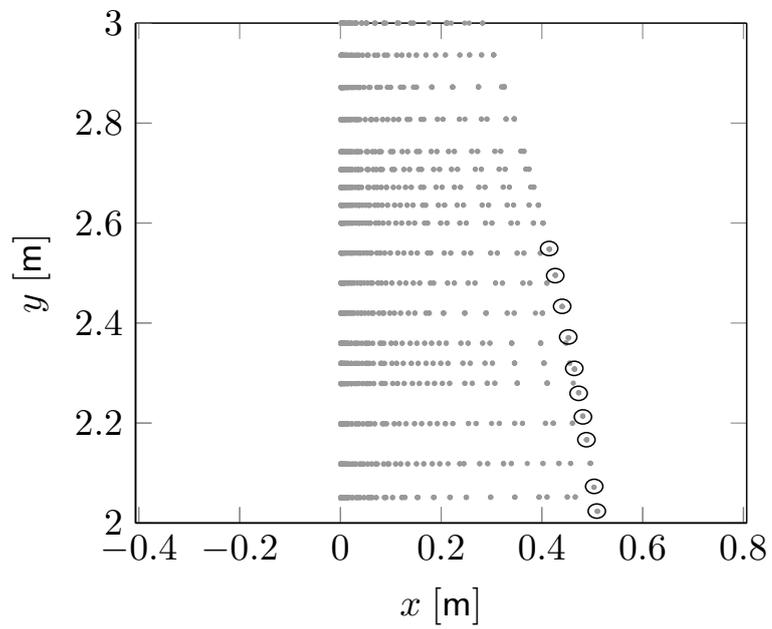
since these two points are necessary to identify the chord and hence the angle of incidence of a wing section. In the method shown here, these key points have been evaluated assuming that they are the two furthest points in a single airfoil, leading to the result depicted in figure A.3. Finally, knowing the position of both the LE and the TE, all the relevant geometrical quantities can be hence evaluated: the chord and the twist distribution, the focal line, the dihedral angle, the sweep distributions and the planform related data, such as the wing surface and the aspect ratio. All these variables are depicted in figure A.4, where the marks represent the calculated data and the solid lines are referred to a polynomial best fitting. In particular, figure A.4(d) shows that the greatest errors are related to the Trailing Edge sweep, Λ_{TE} : this could have been expected since, as shown in figure A.1(b), the most of the faults in the original data set are located in this zone, moreover a missing TE point could lead to a non regular TE line. At this purpose it should be pointed out that the employed code, reported in listing A.1, includes a further TE correction for the points that are not correctly aligned with their neighbours. In a similar fashion, figure A.4(b) shows that the twist data set is formed by two distinct groups of points: one is lying on the top of the best fit line, whereas the other is located on its bottom. This behaviour is due to the fact that the TE of the analyzed wing is open, rather than sharp, so that in effect two different TE points do exist which may be selected by the algorithm almost with the same probability.

Listing A.1: Points cloud treatment algorithm

```
1  clc; close all; clear all;
2
3  % Set tolerances to organize the points cloud into a data structure
4  xTol = 2;
5  yTol = 0.5;
6
7  % Import stl file in ASCII format, as exported from Catia V5
8  [vert, ~] = import_stl_fast('wingTrasl.stl', 2);
9
10 % Sort vert in spanwise direction
11 vert = sortrows(vert, -2);
12 vert(:,2) = abs(vert(:,2));
13
14 % Span location of the airfoils
15 shortVert = vert(vert(:,1) < xTol, :);
16 spanStations = [shortVert(diff(shortVert(:,2)) > yTol, 2); ...
17     shortVert(end,2)];
18
19 % Evaluate the influence area related to each section and reorganize
20 % the points so that their spanwise locations are always coincident
21 % with an airfoil location
22 influenceArea = diff(spanStations)/2;
23
24 for s=2:numel(spanStations) - 1
25     for i=1:numel(vert(:,2))
26         if abs(vert(i,2) - spanStations(s)) < influenceArea(s)
27             vert(i,2) = spanStations(s);
28         end
29     end
30 end
```



(a) Isometric view



(b) Some faults in a magnified region

Figure A.1: E-Pteron wing: original points cloud

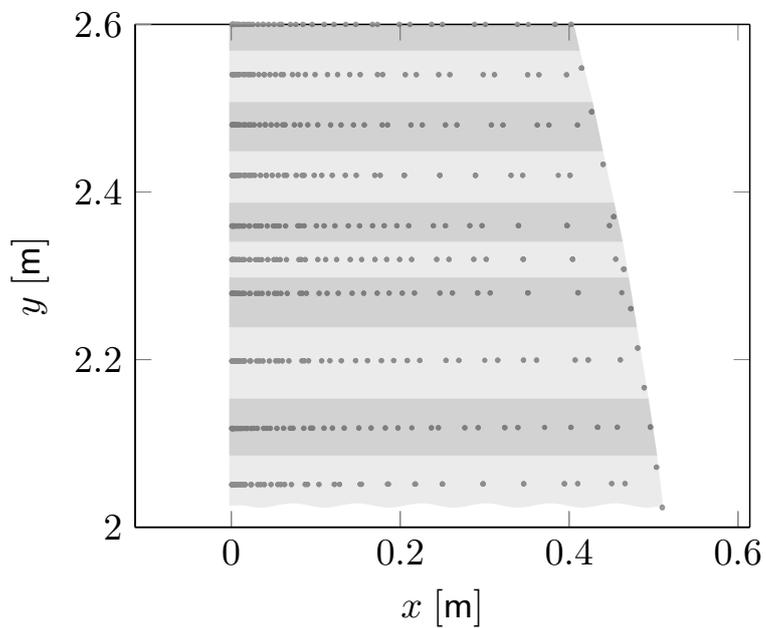


Figure A.2: Influence areas

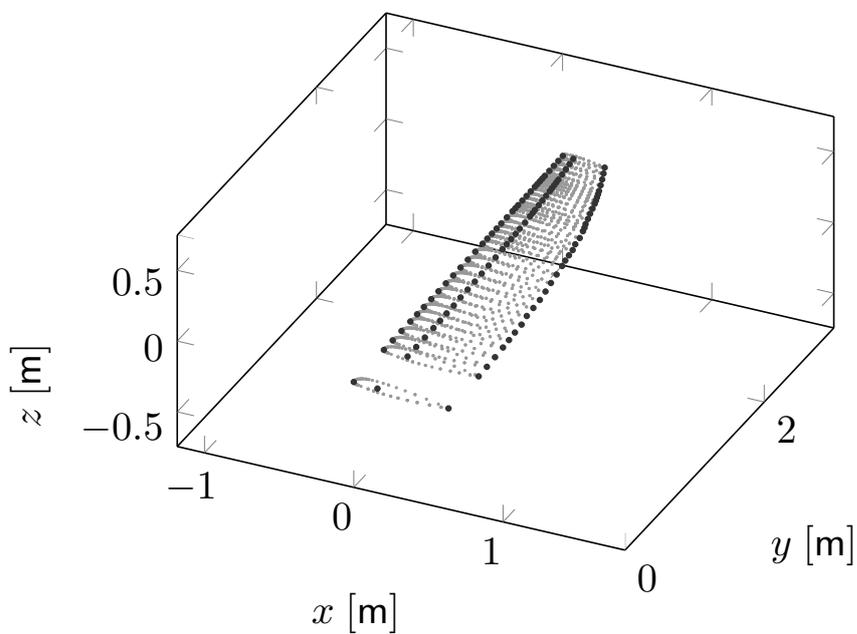


Figure A.3: E-Pteron wing: treated points cloud with detected LEs, TEs and focal line

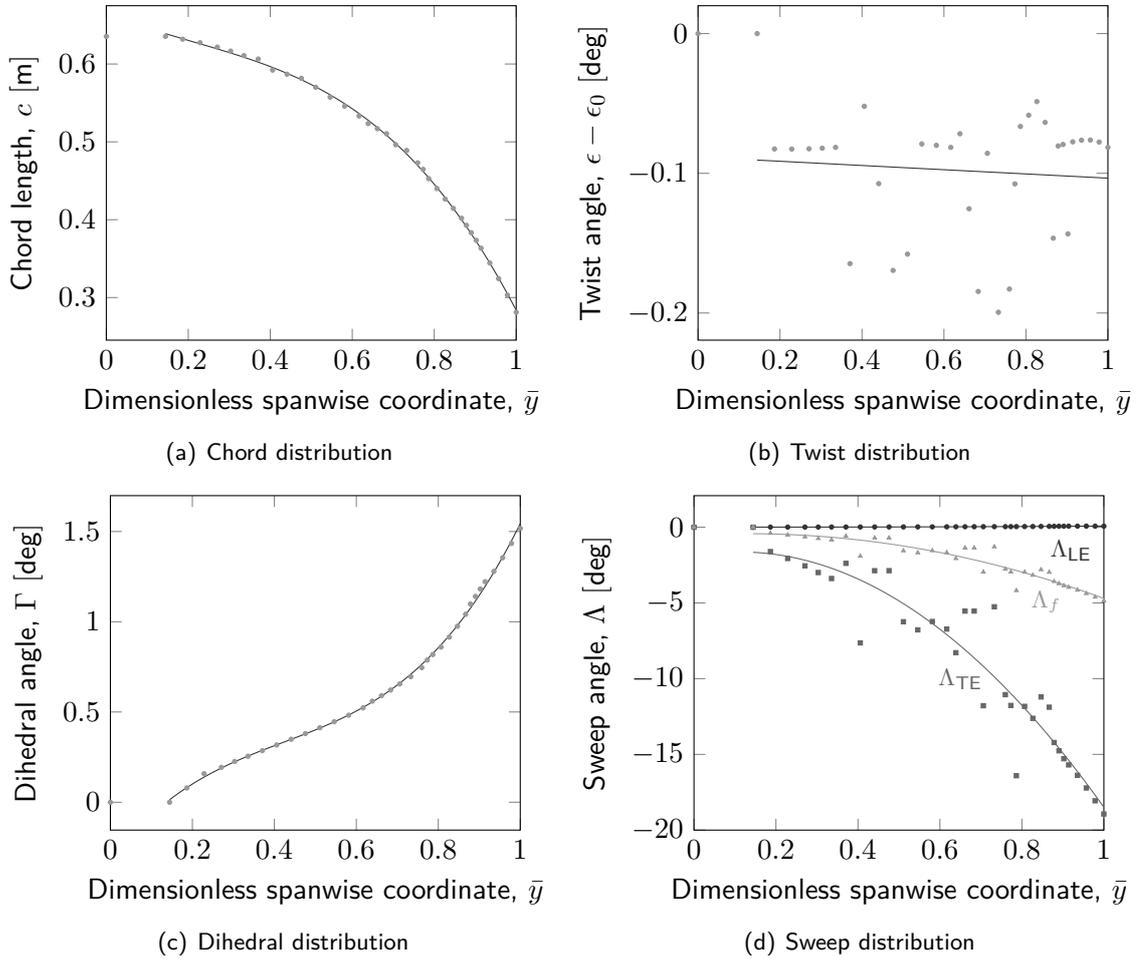


Figure A.4: E-Pteron wing characteristics: the marks refer to calculated data, whereas the solid lines are related to their polynomial best fitting

```

30     end
31 end
32 % Create a data structure where the points are recollocated in the
33 % respective spanwise locations
34 for i=1:numel(spanStations)
35     wingSection{i} = vert(find(vert(:,2) == spanStations(i)), :);
36 end
37
38 % Find the leading edge and the trailing edge for each section,
39 % assuming that those are the furthest points in an airfoil
40 for s=1:numel(spanStations)
41     x = wingSection{s}(:,1);
42     z = wingSection{s}(:,3);
    
```

```

44     [dist, idx] = pdist2([x,z], [x,z], 'euclidean', 'Largest', 1);
45     idx1 = min(unique(idx(dist==max(dist))));
46     idx2 = max(unique(idx(dist==max(dist))));

48     if x(idx1) > x(idx2)
49         LE.idx(s) = idx2;
50         TE.idx(s) = idx1;
51     else
52         LE.idx(s) = idx1;
53         TE.idx(s) = idx2;
54     end

56     LE.x(s) = x(LE.idx(s));
57     LE.z(s) = z(LE.idx(s));
58     TE.x(s) = x(TE.idx(s));
59     TE.z(s) = z(TE.idx(s));
60 end

62 % Fix trailing edge faults
63 for s=2:numel(spanStations)-1
64     if TE.x(s) < TE.x(s-1) && TE.x(s) < TE.x(s+1)
65         TE.x(s) = (TE.x(s-1) + TE.x(s+1))/2;
66         TE.z(s) = (TE.z(s-1) + TE.z(s+1))/2;
67         wingSection{s} = [wingSection{s}; [TE.x(s), spanStations(s),
68             TE.z(s)]];
69         TE.idx(s) = numel(wingSection{s}(:,1));
70     end
71 end

72 % Find the focal line
73 for s=1:numel(spanStations)
74     focus.x(s) = (LE.x(s) + (LE.x(s) + TE.x(s))/2)/2;
75     focus.z(s) = (LE.z(s) + (LE.z(s) + TE.z(s))/2)/2;
76 end

78 % Evaluate chord, wing twist, dhiedral angle and sweep
79 chord = sqrt((LE.x - TE.x).^2 + (LE.z - TE.z).^2);
80 twist = atand((LE.z - TE.z)./(LE.x - TE.x));
81 dhiedral = atand(diff(LE.z)./diff(spanStations));
82 dhiedral = [dhiedral(1), dhiedral];
83 dhiedral(3) = (dhiedral(2) + dhiedral(4))/2;
84 sweep.LE = atand(diff(LE.x)./diff(spanStations));
85 sweep.LE = [sweep.LE(1), sweep.LE];
86 sweep.TE = atand(diff(TE.x)./diff(spanStations));
87 sweep.TE = [sweep.TE(1), sweep.TE];
88 sweep.f = atand(diff(focus.x)./diff(spanStations));
89 sweep.f = [sweep.f(1), sweep.f];
90

```

```
% Evaluate wing planform
92 for s=1:numel(spanStations)
    wingPlanform.LE(s) = min(wingSection{s}(:,1));
94    wingPlanform.TE(s) = max(wingSection{s}(:,1));
    end
96
% Global data
98 S = 2*polyarea([wingPlanform.LE, fliplr(wingPlanform.TE), ...
    wingPlanform.LE(1)], [spanStations;flipud(spanStations); 0]')/1e6;
100 b = 2*spanStations(end)/1e3;
    AR = b^2/S;
102
% Polynomial approximations
104 N = 100;
    yMesh = linspace(spanStations(2), spanStations(end), N);
106 chordPP = polyfit(spanStations(2:end)', chord(2:end), 3);
    twistPP = polyfit(spanStations(2:end)', twist(2:end), 1);
108 dhiedralPP = polyfit(spanStations(2:end)', dhiedral(2:end), 3);
    sweepPP.LE = polyfit(spanStations(2:end)', sweep.LE(2:end), 2);
110 sweepPP.TE = polyfit(spanStations(2:end)', sweep.TE(2:end), 2);
    sweepPP.f = polyfit(spanStations(2:end)', sweep.f(2:end), 2);
112
% Normalized wing airfoil at root
114 airfoil.x_temp = wingSection{1}(:,1)/chord(1);
    airfoil.y_temp = wingSection{1}(:,3)/chord(1);
116
    airfoil.x = cosd(-twist(1))*airfoil.x_temp - ...
118         sind(-twist(1))*airfoil.y_temp;
    airfoil.y = sind(-twist(1))*airfoil.x_temp + ...
120         cosd(-twist(1))*airfoil.y_temp;
122
    airfoil.up = sortrows([airfoil.x(airfoil.y >= 0), ...
        airfoil.y(airfoil.y >= 0)], -1);
124
    airfoil.dw = sortrows([airfoil.x(airfoil.y < 0), ...
        airfoil.y(airfoil.y < 0)], 1);
```

B e-Pteron in Digital Datcom+

Listing B.1: Digital Datcom+ input for E-Pteron

```

1  *
2  *   e-Pteron, full configuration
3  *
4
5  DIM M
6  DERIV DEG
7  DAMP
8  PART
9
10 $FLTCON WT=45.4, LOOP=2.0,
11          NMACH=1.0, MACH(1)=0.0,
12          NALT=1.0, ALT(1)=0.0,
13          RNNUB=0.5E6,
14          NALPHA=20.0,
15          ALSCHD(1)= -16.0, -8.0, -6.0, -4.0, -2.0, 0.0, 2.0, 4.0, 8.0,
16                   9.0, 10.0, 12.0, 14.0, 16.0, 18.0, 19.0, 20.0, 21.0,
17                   22.0, 24.0,
18          STMACH=0.6, TSMACH=1.4, TR=1.0$
19
20 $OPTINS SREF=3.2351, CBARR=0.4704, BLREF=6.0, ROUGFC=0.25E-3$
21
22 $SYNTHS XCG=1.6, ZCG=0.0,
23          XW=0.075, ZW=0.0, ALIW=0.0,
24          XH=2.00999, ZH=0.1108, ALIH=3.01,
25          XV=3.65813, ZV=0.10986,
26          SCALE=1.0, VERTUP=.TRUE.$
27
28 $BODY NX=20.0,
29          X(1)=0.0, 0.0139, 0.0270, 0.0817, 0.1558, 0.2912, 0.4604,
30          0.6647, 0.9872, 2.0275, 2.2477, 2.6506, 2.9381, 3.2008,

```

APPENDIX B. E-PTERON IN DIGITAL DATCOM+

```

31      3.4519, 3.6754, 3.9123, 4.1683, 4.2248, 4.2281,
      R(1)=0.0, 0.0374, 0.0589, 0.1055, 0.1518, 0.2177, 0.2664,
33      0.3042, 0.3441, 0.3435, 0.3233, 0.2710, 0.2237, 0.1776,
      0.1343, 0.0998, 0.0705, 0.0404, 0.0202, 0.0,
35      ZU(1)=0.0, 0.0212, 0.0336, 0.0531, 0.0706, 0.0881, 0.1058,
      0.1233, 0.1415, 0.1586, 0.1571, 0.1538, 0.1500, 0.1482,
37      0.14645, 0.1447, 0.1322, 0.1178, 0.1132, 0.1129,
      ZL(1)=0.0, -0.0285, -0.0349, -0.0568, -0.0788, -0.1009,
39      -0.1209, -0.1323, -0.1534, -0.1702, -0.1593, -0.1291,
      -0.0988, -0.0686, -0.0383, -0.0081, 0.0226, 0.0485,
41      0.0858, 0.1129,
      BNOSE=1.0, BTAIL=1.0,
43      ITYPE=1.0, METHOD=1.0$

45      $WGPLNF CHRDR=0.325,   CHRDTp=0.1506,   CHRDBP=0.325,
      SSPN=1.10,   SSPNE=0.99,   SSPNOP=1.10,
47      SAVSI=0.0,
      CHSTAT=0.0, TWISTA=0.0,
49      DHDADI=0.0,
      TYPE=1.0$

51      NACA W 4 0012
53      NACA H 5 23012
      NACA V 4 0012
55      NACA F 4 0012

57      $HTPLNF CHRDR=0.63461,   CHRDTp=0.28071,   CHRDBP=0.63461,
      SSPN=3.0,   SSPNE=2.7352,   SSPNOP=2.567,
59      SAVSI=0.0,
      CHSTAT=0.0, TWISTA=-0.05,
61      DHDADI=0.0,
      TYPE=1.0$

63      $VTPLNF CHRDTp=0.255353,   SSPNE=0.65038,   SSPN=0.65038,   CHRDR=0.7696,
65      SAVSI=42.23,   CHSTAT=0.0,   TYPE=1.0$

67      $PROPWR AIETLP=0.0,   NENGSP=2.0,   THSTCP=0.0,
      PHALOC=0.075,   PHVLOC=0.0,   PRPRAD=0.4,
69      BWAPR3=0.0174,   BWAPR6=0.0086,   BWAPR9=0.0058,   BAPR75=8.5735,
      NOPBPE=3.0,   YP=1.10,   CROT=.FALSE.$

71      CASEID TOTAL: e-Pteron (Simple)

```

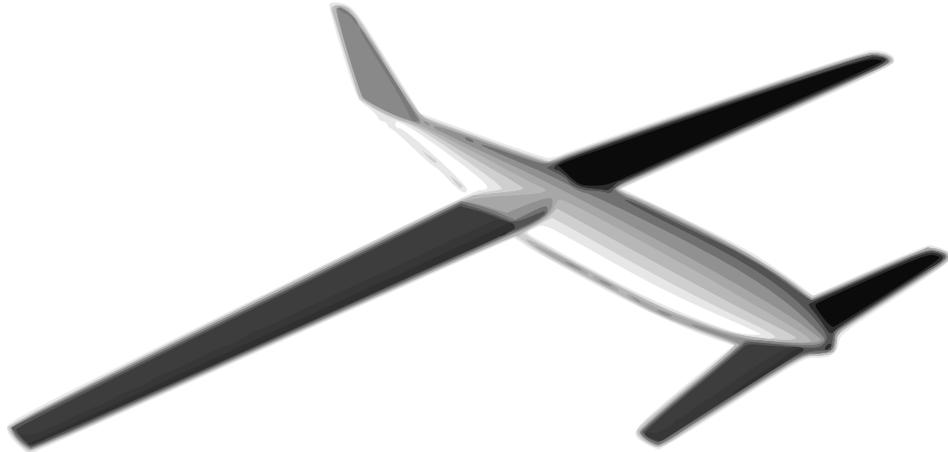


Figure B.1: E-Pteron geometry produced by listing B.1. The engines have been excluded due to graphical reasons

C An Example of PaMS DATAIN File

Listing C.1: Typical DATAIN file for PaMS

```

1  -----
2  &RUN          runname="PaMS-simulation"                ver='BS'    &END
3  -----
4  &INFTY        velinf(1,1)=0.0    velinf(2,1)=0.0    velinf(3,1)=0.0    &END
5  &INFTY        omginf(1,1)=0.0    omginf(2,1)=0.0    omginf(3,1)=0.0    &END
6  &INFTY        density( 1)=1.225    stpress( 1)=0.0    soundsp( 1)=340.3 &END
7  -----
8  &TIME         dt=0.003                tstart=0.0        tend=0.300        &END
9  &TIME         nsubi=2                    &END
10 -----
11 &INVERTER     omegainv=0.0            convinv=1.E-6      maxiter=100        &END
12 &INVERTER     maxblkiter=5            nblkmin=1          mfile='Y'          &END
13 -----
14 &CUTOFF       RFF(1)=1.E+9            RFF(2)=1.E+9      &END
15 -----
16 &BODY         nbody =1                relmot='N'         symplane=0         &END
17 &BODY         cbdgap=0.0                    &END
18 -----
19 &MOTION       nsubimot=2                onlymot='N'        &END
20 -----
21 &FILEOUT      fofreq      = 1                    &END
22 &FILEOUT      tecplotgp  = 'Y'            tecplotpn  = 'Y'            tecplottm  = 'Y'    &END
23 &FILEOUT      tecpress   = 'N'            tecpresstm = 'N'            &END
24 &FILEOUT      tectreff   = 'Y'            tectrefftm = 'Y'            &END
25 &FILEOUT      patran     = 'N'            patrantm   = 'N'            &END
26 &FILEOUT      tecplot3d  = 'N'            tecplot3dtm = 'N'          &END
27 &FILEOUT      sfo        = 'N'            sfoname    = 'pams.tgz'    &END
28 -----
29 &OUT3D        out3dcen(1)=0.    out3dcen(2)=0.    out3dcen(3)=1.    &END
30 &OUT3D        out3ddim(1)=0.25  out3ddim(2)=4.    out3ddim(3)=0.25  &END

```

APPENDIX C. AN EXAMPLE OF PAMS DATAIN FILE

```

31 &OUT3D      ngpout3d(1)=50      ngpout3d(2)=1      ngpout3d(3)=50      &END
-----
33
-----
35
&GEOIN      geoname="rotor_8deg.geo"      icb=0      &END
37 &GEOIN      orient      =+1      orignumb  ='N'      maxang=180.0      &END
&SCALE      sclfac(1)=1.0      sclfac(2)=1.0      sclfac(3)=1.0      &END
39 &KWAKE      ang=140.      tmsfac=1.0      kwnumb    ='N'      &END
&KWAKE      nkwtts =400      nkwtstrig = 0      nkwtspan  =16      &END
41 &KWAKE      xanru = 0.      yanru = 0.      zanru = 0.      &END
&BC         bctype(1)='D'      bctype(2)='R'      bctype(3)='P'      &END
43 &BC         bcnvel =0.0      bcidou   ='N'      &END
&BCLIM      corerad  =0.2      coremod  ='R'      pt2wknd  =0.      &END
-----
45
&GEOTRA     GT(1,1)=0.0      GT(1,2)=0.0      GT(1,3)=0.0      &END
47 &GEOTRA     GT(2,1)=0.0      GT(2,2)=0.0      GT(2,3)=0.0      &END
&GEOTRA     GT(3,1)=0.0      GT(3,2)=0.0      GT(3,3)=0.0      &END
49 &GEOROT     GR(1,1)=0.0      GR(1,2)=0.0      GR(1,3)=0.0      &END
&GEOROT     GR(2,1)=0.0      GR(2,2)=0.0      GR(2,3)=0.0      &END
51 &GEOROT     GR(3,1)=0.0      GR(3,2)=0.0      GR(3,3)=0.0      &END
&SYSTRA     ST(1,1)=0.0      ST(1,2)=0.0      ST(1,3)=0.0      &END
53 &SYSTRA     ST(2,1)=0.0      ST(2,2)=0.0      ST(2,3)=0.0      &END
&SYSTRA     ST(3,1)=0.0      ST(3,2)=0.0      ST(3,3)=0.0      &END
55 &SYSROT     SR(1,1)=0.0      SR(1,2)=0.0      SR(1,3)=0.0      &END
&SYSROT     SR(2,1)=0.0      SR(2,2)=0.0      SR(2,3)=1250.      &END
57 &SYSROT     SR(3,1)=0.0      SR(3,2)=0.0      SR(3,3)=0.0      &END
-----

```

D An Example of Motion Routine

Listing D.1: Example of motion routine for PaMS

```

2      SUBROUTINE MOTION_R(IB,KTS,IRK,DTrk,
3          +                XLRSc, YLRSc, ZLRSc,
4          +                XLRsvX,XLRsvY,XLRsvZ,
5          +                YLRsvX,YLRsvY,YLRsvZ,
6          +                ZLRsvX,ZLRsvY,ZLRsvZ)
7
8      C-----
9      C PURPOSE:          in this routine motion laws can be inserted
10     C CALLED BY:       newpos_rk4
11     C EXTERNAL REFERENCES: none
12     C ENVIRONMENT:    FORTRAN 90
13     C AUTHOR:         Paolo Caccavale
14     C DEVELOPMENT HISTORY: DATE INITIALS DESCRIPTION LUG-08
15     C-----
16     C INPUT : TM
17     C OUTPUT:
18
19     C-----
20     C F90 ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
21     USE COMMONH
22     IMPLICIT INTEGER (I-N)
23     IMPLICIT REAL(KIND=nBYTEdim) (A-H,O-Z)
24     C-----
25
26     IB =IB
27     KTS =KTS
28     IRK =IRK
29     DTrk=DTrk
30

```


E PaMS DATAIN file for a tiltrotor

Listing E.1: PaMS DATAIN file for a wing & canard tiltrotor configuration

```

-----
2  &RUN          runname="PaMS-simulation"                ver='BS'      &END
-----
4  &INFTY        velinf(1,1)=0.0      velinf(2,1)=0.0      velinf(3,1)=0.0  &END
   &INFTY        omginf(1,1)=0.0      omginf(2,1)=0.0      omginf(3,1)=0.0  &END
6  &INFTY        density( 1)=1.225      stpress( 1)=0.0      soundsp( 1)=340.3 &END
-----
8  &TIME         dt=0.004                tstart=0.0          tend=6.000        &END
   &TIME         nsubi=2                                &END
10 -----
   &INVERTER     omegainv=0.0          convinv=1.E-6        maxiter=100        &END
12  &INVERTER     maxblkiter=5          nblkmin=1            mfile='Y'          &END
-----
14  &CUTOFF       RFF(1)=1.E+9          RFF(2)=1.E+9        &END
-----
16  &BODY         nbody =2              relmot='Y'           symplane=2         &END
   &BODY         cbdgap=0.0                                &END
18 -----
   &MOTION       nsubimot=1          onlymot='N'          &END
20 -----
   &FILEOUT       fofreq      = 1                                &END
22  &FILEOUT       tecplotgp  = 'Y'          tecplotpn  = 'N'          tecplottm  = 'Y'  &END
   &FILEOUT       tecpress   = 'N'                                tecpresstm = 'N'  &END
24  &FILEOUT       tectreff   = 'Y'                                tectrefftm = 'Y'  &END
   &FILEOUT       patran     = 'N'                                patrantm   = 'N'  &END
26  &FILEOUT       tecplot3d  = 'N'                                tecplot3dtm = 'N'  &END
   &FILEOUT       sfo        = 'Y'          sfoname    = 'pams.tgz'  &END
28 -----
   &OUT3D         out3dcen(1)=0.          out3dcen(2)=0.          out3dcen(3)=1.    &END
30  &OUT3D         out3ddim(1)=0.25      out3ddim(2)=4.          out3ddim(3)=0.25  &END

```

APPENDIX E. PAMS DATAIN FILE FOR A TILTROTOR

```

32  &OUT3D      ngpout3d(1)=50      ngpout3d(2)=1      ngpout3d(3)=50      &END
-----
34
-----
36  &GEOIN      geoname="propeller_b.geo"      icb=1      &END
37  &GEOIN      orient      =+1      orignumb  ='N'      maxang=180.0      &END
38  &SCALE      sclfac(1)=0.001      sclfac(2)=0.001      sclfac(3)=0.001      &END
39  &KWAKE      ang=140.      tmsfac=1.0      kwnumb    ='N'      &END
40  &KWAKE      nkwtts =120      nkwttsrig = 0      nkwtspan  =2      &END
41  &KWAKE      xanru = 0.      yanru = 0.      zanru = 0.      &END
42  &BC         bctype(1)='D'      bctype(2)='R'      bctype(3)='I'      &END
43  &BC         bcnvel =0.0      bcidou    ='N'      &END
44  &BCLIM      corerad  =0.02      coremod   ='G'      pt2wknd   =0.      &END
-----
46  &GEOTRA     GT(1,1)=-0.182      GT(1,2)=0.0      GT(1,3)=0.0      &END
47  &GEOTRA     GT(2,1)=0.0      GT(2,2)=0.0      GT(2,3)=0.0      &END
48  &GEOTRA     GT(3,1)=0.0      GT(3,2)=0.0      GT(3,3)=0.0      &END
49  &GEOROT     GR(1,1)=0.0      GR(1,2)=+90.0      GR(1,3)=0.0      &END
50  &GEOROT     GR(2,1)=0.0      GR(2,2)=0.0      GR(2,3)=0.0      &END
51  &GEOROT     GR(3,1)=0.0      GR(3,2)=0.0      GR(3,3)=0.0      &END
52  &SYSTRA     ST(1,1)=0.0      ST(1,2)=-1.1      ST(1,3)=0.0      &END
53  &SYSTRA     ST(2,1)=0.0      ST(2,2)=0.0      ST(2,3)=2.0      &END
54  &SYSTRA     ST(3,1)=0.0      ST(3,2)=0.0      ST(3,3)=0.0      &END
55  &SYSROT     SR(1,1)=0.0      SR(1,2)=0.0      SR(1,3)=0.0      &END
56  &SYSROT     SR(2,1)=0.0      SR(2,2)=0.0      SR(2,3)=0.0      &END
57  &SYSROT     SR(3,1)=0.0      SR(3,2)=0.0      SR(3,3)=0.0      &END
-----
60
-----
61  &GEOIN      geoname="canard.geo"      icb=2      &END
62  &GEOIN      orient      =+1      orignumb  ='N'      maxang=180.0      &END
63  &SCALE      sclfac(1)=0.001      sclfac(2)=0.001      sclfac(3)=0.001      &END
64  &KWAKE      ang=140.      tmsfac=1.0      kwnumb    ='N'      &END
65  &KWAKE      nkwtts =108      nkwttsrig = 0      nkwtspan  =2      &END
66  &KWAKE      xanru = 0.      yanru = 0.      zanru = 0.      &END
67  &BC         bctype(1)='D'      bctype(2)='R'      bctype(3)='I'      &END
68  &BC         bcnvel =0.0      bcidou    ='N'      &END
69  &BCLIM      corerad  =0.002      coremod   ='G'      pt2wknd   =0.      &END
-----
70
-----
71  &GEOTRA     GT(1,1)=-0.163      GT(1,2)=0.0      GT(1,3)=0.0      &END
72  &GEOTRA     GT(2,1)=0.0      GT(2,2)=0.0      GT(2,3)=0.0      &END
73  &GEOTRA     GT(3,1)=0.0      GT(3,2)=0.0      GT(3,3)=0.0      &END
74  &GEOROT     GR(1,1)=0.0      GR(1,2)=0.0      GR(1,3)=0.0      &END
75  &GEOROT     GR(2,1)=0.0      GR(2,2)=0.0      GR(2,3)=0.0      &END
76  &GEOROT     GR(3,1)=0.0      GR(3,2)=0.0      GR(3,3)=0.0      &END
77  &SYSTRA     ST(1,1)=0.0      ST(1,2)=0.0      ST(1,3)=0.0      &END
78  &SYSTRA     ST(2,1)=0.0      ST(2,2)=0.0      ST(2,3)=0.0      &END
79  &SYSTRA     ST(3,1)=0.0      ST(3,2)=0.0      ST(3,3)=0.0      &END

```

APPENDIX E. PAMS DATAIN FILE FOR A TILTROTOR

```
80 &SYSROT   SR(1,1)=0.0      SR(1,2)=0.0      SR(1,3)=0.0      &END
    &SYSROT   SR(2,1)=0.0      SR(2,2)=0.0      SR(2,3)=0.0      &END
82 &SYSROT   SR(3,1)=0.0      SR(3,2)=0.0      SR(3,3)=0.0      &END
-----
```



Bibliography

Abbott, I. H. and A. E. von Doenhoff

1959. *Theory of wing sections: Including a Summary of Airfoil Data*, Dover Books on Aeronautical Engineering. Dover Publications.

Acree, C. W.

2012. Integration of rotor aerodynamic optimization with the conceptual design of a large civil tiltrotor. Technical Report 562061, NASA Ames Research Center, Moffett Field, California 94035, USA.

Beshears, T. and G. I. Peterson

2004. The HV-911 eagle eye tiltrotor VUAV: An exciting future for coast guard aviation. Technical report, American Society of Naval Engineers.

Betz, A.

1919. Screw propellers with minimum loss of energy. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, Pp. 193–217.

Betz, A.

1922. The theory of screw propeller. *NACA Report*, (83).

Bhatia, H., G. Norgard, P. Valerio, and P.-T. Bremer

2013. The helmholtz-hodge decomposition – a survey. *IEEE Transactions on Visualization and Computer Graphics*, 19(X):1386–1404.

Blumenthal, O.

1905. Über die zerlegung unendlicher vektorfelder. *Mathematische Annalen*, 61(2):235–250.

Caradonna, F. and C. Tung

1981. Experimental and analytical studies of a model helicopter rotor in hover. NASA Technical Memorandum NASA-TM-81232, NASA.

BIBLIOGRAPHY

- Carré, B. A.
1961. The determination of the optimum accelerating factor for successive over-relaxation. *The Computer Journal*, 4:73–78. Issue 1.
- Chorin, A. J. and P. S. Bernard
1973. Discretization of a vortex sheet, with an example of roll-up. *Journal of Computational Physics*, 13(3):423–429.
- Cottet, G.-H. and P. D. Koumoutsakos
2000. *Vortex Methods: Theory and Practice*. Cambridge University Press.
- De Nicola, C.
2013. Appunti per un corso di aerodinamica degli aeromobili.
- Di Francesco, G., G. Scapaticci, and M. Mattei
2014. Transition control and redundancy management for a tilt rotor UAV using genetic algorithms. 29th Congress of the International Council of the Aeronautical Sciences. St. Petersburg, Russia.
- Fail, R., J. A. Lawford, and R. C. W. Eyre
1957. Low-speed experiments on the wake characteristics of flat plates normal to an air stream. *Aeronautical Research Council Reports and Memoranda*, (3120).
- Froude, R. E.
1889. On the part played in propulsion by differences of fluid pressure. *Transactions of the Institution of Naval Architects*, XXX:390–423.
- Froude, W.
1878. On the elementary relation between pitch, slip and propulsive efficiency. *Transactions of the Institution of Naval Architects*, XIX:47.
- Hirsch, C.
2007. *Numerical Computation of Internal and External Flows: Fundamentals of Computational Fluid Dynamics*, 2nd edition.
- Jeffrey, A. and D. Zwillinger
2007. *Table of Integrals, Series, and Products*, 7th edition.
- Johnson, W.
1994. *Helicopter Theory*. Dover Publications.
- Johnson, W.
2013. *Rotorcraft Aeromechanics*, Cambridge Aerospace Series. Cambridge University Press.
- Katz, J. and A. Plotkin
1991. *Low Speed Aerodynamics: from wing theory to panel methods*.
- Kellogg, O. D.
2010. *Foundations of Potential Theory*, Dover Books on Physics. Dover Publications.

BIBLIOGRAPHY

- Kreuzig, E.
2010. *Advanced Engineering Mathematics*, 10th edition.
- Kroo, I.
2001. Drag due to lift: Concepts for prediction and reduction. *Annual Review of Fluid Mechanics*, 33:587–617.
- Kutta, M. W.
1902. Lifting forces in flowing fluids. *Illustrierte Aeronautische Mitteilungen*.
- Leishman, G.
2007. Is there a case for the tiltrotor? *RUSI defence systems*, Pp. 52–55.
- Losito, V.
1983. *Fondamenti di aeronautica generale*. Accademia Aeronautica.
- Maisel, M. D., D. J. Giulianetti, and D. C. Dugan
2000. *The History of the XV-15 Tilt Rotor Research Aircraft: From Concept to Flight*, The NASA History Series, NASA SP-2000-4517 edition. Office of Policy and Plans, NASA History Division. Monographs in Aerospace History #17.
- Meola, C. and G. de Felice
1996. *Fondamenti lineari per la fluidodinamica numerica*.
- Montgomerie, B.
1996. Drag coefficient distribution on a wing at 90 degrees to the wind. Technical report, Netherlands Energy Research Foundation. ECN-C–95-061.
- Monti, R. and R. Savino
2003. *Aerodinamica*.
- Rajasekaran, J.
2011. On the flow characteristics behind a backward-facing step and the design of a new axisymmetric model for their study. Master's thesis, University of Toronto.
- Rankine, W. J. M.
1865. On the mechanical principle of the action of propellers. *Transactions of the Institution of Naval Architects*, VI:13–39.
- Reid, J. K.
1996. A method for finding the optimum successive over-relaxation parameter. *The Computer Journal*, 9:200–204. Issue 2.
- Strickland, J. H. and R. S. Baty
1994. A three-dimensional fast solver for arbitrary vorton distributions. Technical Report SAND93-1641, Sandia National Laboratories.

BIBLIOGRAPHY

- Strickland, J. H., G. F. Homicz, V. L. Porter, and A. A. Gossler
2002. A 3-d vortex code for parachute flow predictions: Version 1.0. Technical Report SAND2002-2174, Sandia National Laboratories.
- Tognaccini, R.
2011. *Lezioni di Aerodinamica dell'Ala Rotante*, 0.1 edition. Eliche, rotori ed aeromotori.
- van der Vorst, H. A.
1992. BI-CGSTAB: a fast and smoothly converging variant of BI-CG for the solution of non-symmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 13:631–644. Issue 2.
- von Helmholtz, H. L. F.
1858. Über integrale der hydrodynamischen gleichungen, welche den wirbelbewegungen entsprechen. *Journal für die reine und angewandte Mathematik*, 1858(55):25–55.
- Weick, F. E.
1931. Full scale wind tunnel tests with a series of propellers of different diameters on a single on a single fuselage. *NACA Report*, (339). Langley Memorial Aeronautical Laboratory.
- Willis, D. J.
2006. An unsteady, accelerated, high order panel method with vortex particle wakes. Master's thesis, Massachusetts Institute of Technology.
- Willis, D. J., J. Peraire, and J. K. White
2005. A combined pFFT-multipole tree code, unsteady panel method with vortex particle wakes. *International Journal for Numerical Methods in Fluids*, 53:1399–1422.
- Winckelmans, G. S.
1989. *Topics in vortex methods for the computation of three- and two-dimensional incompressible unsteady flows*. PhD thesis, California Institute of Technology.
- Winckelmans, G. S. and A. Leonard
1993. Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows. *Journal of Computational Physics*, 109:247–273. Issue 2.



Webography

Acree, C. W.

2012. NASA large civil tiltrotor concept. <http://halfdome.arc.nasa.gov/Research/Programs/LCTR.html>.

AugustaWestland

2014. AW609 tiltrotor to perform first ever customer demonstration flights at heli-expo. <http://www.agustawestland.com/news/aw609-tiltrotor-perform-first-ever-customer-demonstration-flights-heli-expo>.

Caccavale, P.

2008. <http://www.fluere.it/>.

Drela, M.

2000. XFOIL: Subsonic airfoil development system. <http://web.mit.edu/drela/Public/web/xfoil/>.

Drela, M.

2001. Xrotor download page. <http://web.mit.edu/drela/Public/web/xrotor/>.

Holy Cows, Inc.

2002. Datcom by Holy Cows, Inc. <http://www.holycows.net/datcom/>.

OpenMP ARB

1997. The OpenMP API specification for parallel programming. <http://openmp.org/wp/>.

Tabak, E. G.

2002. Vortex stretching in incompressible and compressible fluids. <http://www.math.nyu.edu/faculty/tabak/vorticity.pdf>.



Index of Names and Citations

A

Acree, Cecil, xxvi

B

Berliner, Henry, xxii
Bernard, Peter S., 36
Beshears, Troy, xxiv
Betz, Albert, 43, 44
Bhatia, Harsh, 6, 7
Blumenthal, Otto, 7

C

Caccavale, Paolo, 35, 38, 39, 59, 60, 83
Caradonna, F. X., 83, 84, 87
Carré B. A., 64
Chorin, Alexandre Joel, 36

D

de Felice, Giuseppe, 64
de Nicola, Carlo, 31
Di Francesco, Gabriele, 48–50, 53
Drela, Mark, 53, 94

E

Eyre, R. C. W., 105

F

Fail, R, 105
Froude, Robert Edmund, 43
Froude, William, 44

H

Helmholtz, Hermann Ludwig Ferdinand von,
6, 7
Hirsch, Charles, 64

J

Jefrey, Alan, 7, 9, 16
Johnson, Wayne, xxviii, 40, 43, 44

K

Katz, Joseph, 11, 12, 17, 19, 20, 23, 27, 29,
30, 32, 33, 39
Kellogg, Oliver Dimon, 12, 24
Kreyzig, Erwin, 8
Kutta condition, 102
Kutta, Martin Wilhelm, 11

L

Lawford, J. A., 105
Lehberger, George, xxii
Leishman, Gordon, xxvi, xxvii, xxix, xxxi
Leonard, Anthony, 35, 36

M

Mattei, Massimiliano, 48–50, 53
Meola, Carlo, 64
Monti, Rodolfo, 3

P

Peraire, Jaime, 38, 39
Peterson, Gordon, xxiv
Plotkin, Alen, 11, 12, 17, 19, 20, 23, 27, 29,
30, 32, 33, 39

R

Rajasekaran, Jagannath, 105
Rankine, William John Macquorn, 43, 44
Reid, J. K., 64

S

Savino, Raffaele, 3
Scapatucci, Giuseppe, 48–50, 53
Strickland, James H., 36

T

Tabak, Esteban G., 5
Tognaccini, Renato, 40, 53, 94
Tung, C, 83, 84, 87

V

van der Vorst, Hendrik Albertus, 64

W

Weick, Fred E., 42, 43
White, Jacob K., 38, 39
Willis, David Joe, 36–39
Winckelmans, Grégoire Stéphane, 35–37

Z

Zwillinger, Daniel, 7, 9, 16



Index of Vehicles

A

AugustaWestland
AW609, xxiv, 53

B

Bell
Bell XV-15, xxiii, xxiv, xxviii
Bell XV-3, xxiii
Bell-Boeing Eagle Eye, xxiv, 48
Bell-Boeing Quad TiltRotor, xxiv
Bell-Boeing V-22 Osprey, xxii–xxiv, xxvi,
xxviii–xxxi, 53, 88
Bell/Augusta BA609, see AW609

E

E-Pteron, 48–52, 54, 56, 57, 92–94, 111,
113–115, 118

L

Lockheed
Lockheed C-130 HERCULES, xxiv
Lockheed C-130J, xxix–xxxi

N

NASA
NASA Large Civil TiltRotor, xxvi, xxxii

S

Sikorsky
Sikorsky CH-53E, xxviii, xxix
UH-60 Blackhawk, xxviii, xxix

T

Transcendental Model 1-G, xxii

Index

B

- Bernoulli equation, 9, 10, 30
- Blade Element Momentum Theory, 40, 44, 45, 53, 55, 94, 95
- blade element theory, 42, 44
- boundary condition, 10, 11
 - Dirichlet, 23, 24, 27, 30, 32
 - mixed, 23
 - Neumann, 23, 24, 27
 - no slip, 10

C

- collocation point, 24, 27, 28, 32
- control point,
 - see collocation point
- core function, 36, 74, 75

D

- derivative
 - convective, xii
 - material, see substantial derivative
 - substantial, xii
- doublet, 14–18, 20, 21, 23, 26, 28, 30, 32, 34, 37, 39
 - axis, 17

E

- Einstein summation, xii
- Euler equations, 4, 5

F

- free stream, 15, 16

G

- Green's identity, 12
 - first, 24
 - second, 13
 - third, 13, 14
- Grid Point index, 81

H

- Helmholtz
 - decomposition, 6–8
 - theorem, see Helmholtz decomposition
 - vortex theorem, 5, 12, 25

I

- influence coefficient, 28

K

- Kutta condition, 10–12, 22, 23, 25, 26, 28, 30, 32, 39, 62, 64, 70–72, 82, 99, 102
- Kutta-Joukowski theorem, 26

L

- Laplace's equation, 8–10, 14, 15, 17, 22, 23, 33
- leakage flow, 26, 72

M

momentum theory, 42–44

N

Navier-Stokes

continuity equation, 3, 8
 energy equation, 4
 equations, xii, 3, 4, 9, 10
 momentum equation, 4

P

PaMS, 59, 60, 62–67, 69–81, 83, 87, 92, 94,
 95, 97, 99, 102, 106
 DATAIN, 60, 62–64, 66–70, 73, 74, 76,
 77, 81–84, 88, 99, 100, 102, 121,
 125
 DATAOUTforce file, 79, 81
 DATAOUTgeom file, 79–81
 FILEOUT.tecplot, 81, 82, 102
 FILEOUT.tectreff, 81, 82
 geo file, 60–62, 70
 motion routine, 77, 87, 123
 REPORT file, 79
 Poisson’s equation, 8, 10, 35, 37
 potential flow, 6, 12, 17, 26, 30
 propeller-slipstream theory, *see* momentum
 theory
 proprotor, xxi

R

Range Specific Transport Efficiency, xxx, xxxi
 Reynolds number, 5

S

shear stress tensor, 4
 sink, 15–17, 23
 source, 14–17, 19, 20, 22, 23, 30, 32
 Specific Productivity, xxx, xxxi
 stream function, 8

T

tiltrotor

critics to, xxvi
 history, xxii

Time Step index, 81

Trefftz

plane, 33, 34
 technique, 34, 79, 82, 94

U

UAV, xxiv, 48–50, 92, 93

V

velocity potential, 8
 vortex, 15, 17, 19, 21, 23, 26, 39
 particle, *see* vorton
 stick, *see* vorton
 stretching, 5, 36
 vorticity equation, 5, 36
 vorton, 34–39, 72, 74, 75, 81
 core radius, 35, 36, 74
 method, 34, 37, 74, 92
 wake, 37–39, 83
 VUAV, *see* UAV

W

wake
 fully free, 71, 73, 83, 84
 model, 25, 37, 38
 no roll-up, 83
 roll-up, 32, 33, 73
 shape, 26
 strength, 25