

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

ME599 _ MASTER THESIS IN MECHANICAL ENGINEERING

Development of a Doublet-Lattice Method Program for Aeroelastic Analysis in Conceptual Aircraft Design

Author

Alessandro A. GASTALDI

alessandro.gastaldi@epfl.ch

Supervisors

Prof. Mark SAWLEY (EPFL)
Jan B. Vos (CFS Engineering SA)
Aidan JUNGO (CFS Engineering SA)



February 3, 2017

Foreword

I would like to express my profound gratitude to those who have been my colleagues for the past six months, and who have made my Master Thesis with CFS Engineering a very pleasant and instructive experience. To Jan Vos in particular, for giving me the opportunity to work on this project. To Aidan, for his advice and his help with CEASIM, and to Alain and Dominique, for the interesting discussions around coffee each day.

I wish to thank also Prof. Mark Sawley, for his willingness to supervise this project and for his helpful suggestions.

Tomas Melin, whose advice and experience with the Vortex-Lattice Method was precious.

Finally, I thank my parents, to whom I truly owe everything, as well as all those who have made these past five years a time that I will fondly remember.

The present project has been performed in the framework of the AGILE project (Aircraft 3rd Generation MDO for Innovative Collaboration of Heterogeneous Teams of Experts).

Contents

1	Introduction and Problem Statement	1
2	Computational Flutter Analysis	4
3	Theory of the Doublet-Lattice Method	6
3.1	Elementary solutions of the Linearised Aerodynamic Potential Equation	6
3.2	The Integral Equation for a Pressure Doublet Sheet and its Kernel Function	9
3.3	The Doublet-Lattice Method	12
3.4	Boundary Conditions	17
3.5	Aerodynamic Coefficients and Generalised Aerodynamic Forces	18
4	Implementation in Python	20
4.1	Objectives	20
4.2	Program Structure	22
4.3	CPACS Compatibility	26
4.4	Performance Optimisation	32
5	Results for Selected Test Cases	33
5.1	Verification of the Kernel Function	33
5.2	Planar Rectangular Wing	39
5.3	Influence of Panel Aspect Ratio	42
6	Concluding Remarks	53
A	Additional notes on the Theory of Potential Flow	58
A.1	Equations of Motion for Inviscid Flow	58
A.2	Fundamentals of Potential Flow	59
A.3	The Linearised Aerodynamic Potential Equation	61
A.4	Boundary Conditions	63
B	Refinement of the Doublet-Lattice Method	64
B.1	Quartic Approximation of the Kernel Function	64
C	Input and Output Files	66
C.1	Aircraft Geometry Configuration File	67
C.2	Aircraft Geometry Coordinate File	68
C.3	Operating Conditions File	69
C.4	Settings File	70

Nomenclature

Coordinate Systems

x, y, z	Position of receiving point in global coordinate system
$\bar{x}, \bar{y}, \bar{z}$	Position of receiving point in coordinate system of emitting panel
x_0, y_0, z_0	Relative position of receiving point with respect to emitting point
ξ, η, ζ	Position of emitting point in global coordinate system
$\bar{\xi}, \bar{\eta}, \bar{\zeta}$	Position of emitting point in coordinate system of emitting panel
t	Time
ω	Frequency

Roman Symbols

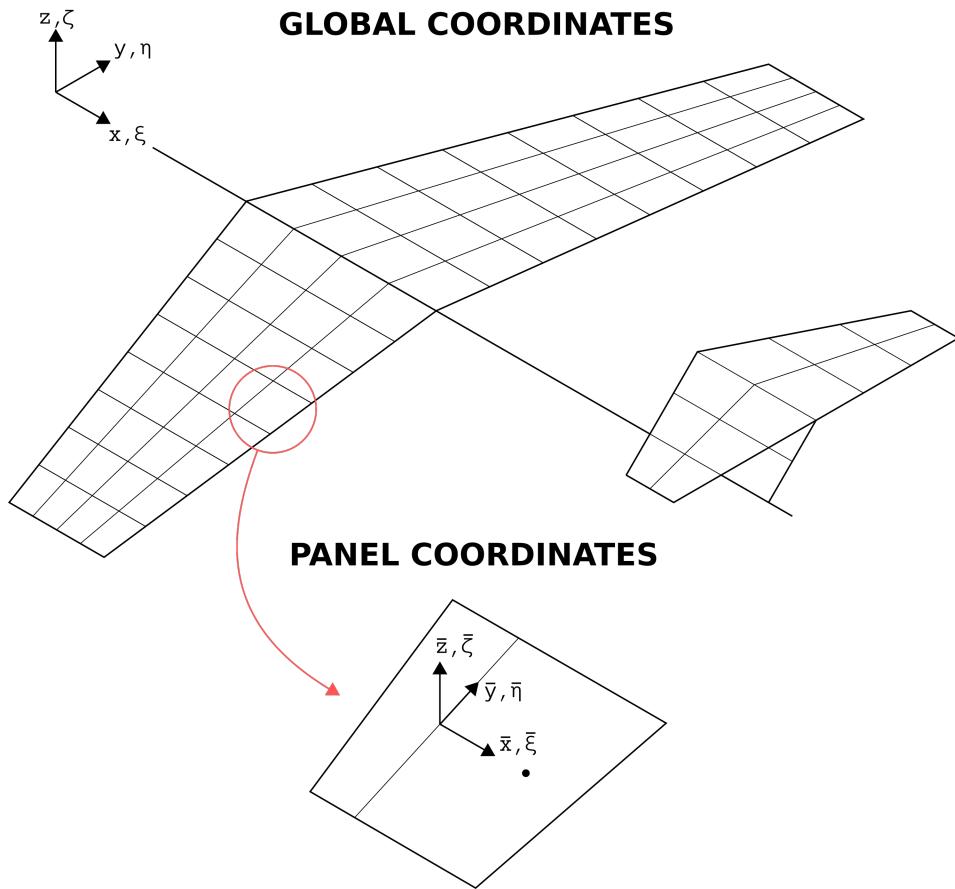


Figure 0.1: Coordinate systems on example configuration.

A	Surface area
AR	Aspect ratio
A_1, A_2	Parabolic/quartic interpolation coefficient (planar, non-planar)
A_Ψ	Source and doublet strengths (acceleration potential)
A_ϕ	Source and doublet strengths (velocity potential)
B_1, B_2	Parabolic/quartic interpolation coefficient (planar, non-planar)
$[B]$	Matrix of generalised structural damping
C_1, C_2	Parabolic/quartic interpolation coefficient (planar, non-planar)
C_L	Lift coefficient (wing), see (3.51)
D_1, D_2	Quartic interpolation coefficient (planar, non-planar)
$[D]$	Matrix of total downwash factors
E_1, E_2	Quartic interpolation coefficient (planar, non-planar)
F	Time-dependent integration constant, see (A.8)
G	Time-dependent integration constant, see (A.12)
H	Integral, see (3.35)
I_0	Integral, see (3.22)
I_1	Integral, see (3.21)
I_2	Integral, see (3.21)
J_0	Integral, see (3.22)
K	Kernel function, see (3.17)
K_1	Planar component of kernel function, see (3.19)
K_2	Non-planar component of kernel function, see (3.19)
$[K]$	Matrix of generalised structural stiffness
L_{ref}	Reference length
M	Mach number
$[M]$	Matrix of generalised structural masses
\vec{N}	Surface normal
P_1, P_2	Polynomial approximation of the kernel numerator (planar, non-planar)
$[Q]$	Matrix of generalised aerodynamic forces
R	Hyperbolic radius
S	Flight vehicle surface
T_1	Geometric parameter, see (3.19)
T_2	Geometric parameter, see (3.19)
U	Velocity magnitude
\vec{a}	Acceleration field
a	Small-perturbation propagation velocity
c	Chord or average chord
c_p	Pressure coefficient

c_L	Lift coefficient (strip), see (3.50)
e	Panel half-width
\hat{e}	Unit direction vector
g	System damping (flutter analysis)
h	Out-of-plane deformation
i	Imaginary unit $\sqrt{-1}$
k	Reduced frequency
k_1	Modified reduced frequency, see (3.19)
p	Pressure or complex eigenvalue (flutter analysis)
q	Generalised coordinate
r	Cylindrical radius
\vec{u}	Velocity field
u	x -component of velocity field
u_1	Integration variable, see (3.19)
v	y -component of velocity field
w	z -component of velocity field
w_N	Normal component of velocity field

Greek Symbols

Δ	Parameter, see (3.39)
Γ	Dihedral angle
Λ	Sweep angle
Ψ	Acceleration potential
α	Parameter, see (3.37)
β	Prandtl-Glauert factor
δ_1	Parameter, see (3.40a)
δ_2	Parameter, see (3.40a)
γ	Specific heat ratio
λ	Chord-wise integration variable or oscillation wavelength
ϕ	Velocity potential
ρ	Density
σ	Span-wise integration variable
τ	Small-perturbation travel time, see (3.4)
$\vec{\omega}$	Vorticity field

Subscripts and Superscripts

$\square_{1/4}$	Quarter-chord line of panel or wing
\square_{LE}	Leading edge

\square_{TE}	Trailing edge
\square_{ref}	Reference value
\square_r	Receiving panel
\square_s	Emitting panel
\square_∞	Far-field
$\square^{(0)}$	Steady component
$\square^{(1)}$	Unsteady planar component
$\square^{(2)}$	Unsteady non-planar component
$\square^{(i) \text{ or } (j)}$	Relative to modes i, j respectively
$\bar{\square}$	Steady, undisturbed flow or amplitude of harmonically oscillating value
$\hat{\square}$	Non-dimensional variable
$\tilde{\square}$	Unsteady perturbation

1 Introduction and Problem Statement

Since the first successful flights of the early 20th century, the challenges of manned aviation have inspired significant advances in science and engineering. In little over a century, civil air transport has become ubiquitous and indispensable, carrying the equivalent of half the world's population to their destination every year [1]. Freight and passenger volumes have increased ten-fold in the last 40 years alone [1, 2].

As environmental and economic concerns become increasingly urgent, the industry now faces a multi-faceted challenge. The Flightpath 2050 report issued by the High-Level Group on Aviation Research of the European Commission presents a highly ambitious vision for the future of the aviation industry: to increase competitiveness, sustainability and safety through research, technology and innovation [3].

Current efforts in aircraft design are directed toward cutting development times and costs, and achieving comprehensive, life-cycle-oriented optimisation of the product. Traditional design strategies are governed by what is known as *Cayley's paradigm*, after pioneering aeronautical engineer Sir George Cayley (1773 - 1857): the aircraft is conceived as an assembly of weakly coupled subsystems, individually designed and optimised to perform distinct subsets of the primary functions required for flight. In this framework however, multi-disciplinary considerations become an afterthought, often delayed to more advanced stages of the design process.

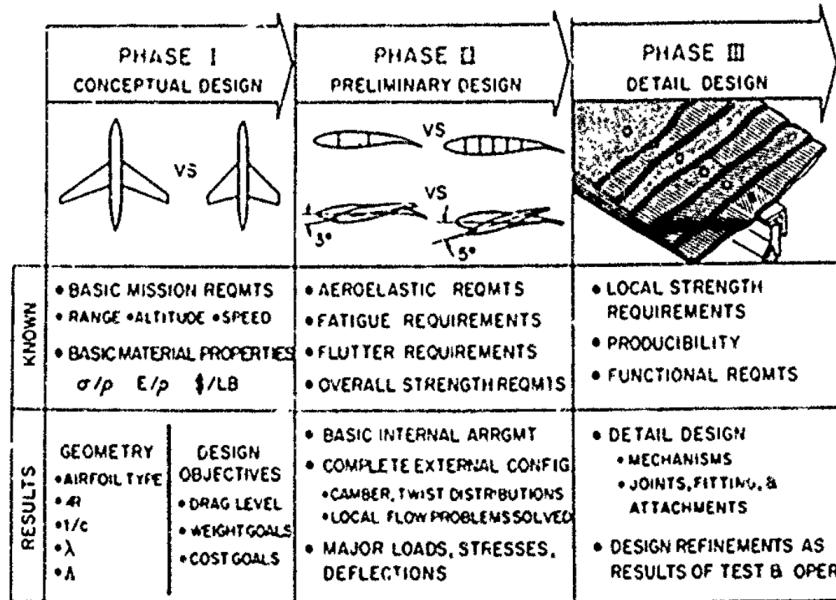


Figure 1.1: Overview of the three major phases of aircraft design [4].

From the first formulation of the customer's requirements to production, the design and development of an aircraft can be decomposed in three principal phases : conceptual, preliminary, and detail design (see Fig. 1.1).

As the development progresses, the aircraft is defined in increasing detail and greater resources are allocated to its continued development. The objective of conceptual design is to formulate a proposal, among many possible configurations, of the product that is to meet the design requirements. Once a promising and feasible configuration is accepted for further development, the preliminary design phase can start. The aircraft layout will undergo minor but repeated revisions as the design matures toward a final proposal; should this proposal be accepted for full-scale development, the detail design phase is initiated. This final step is the culmination of development and testing efforts, toward the complete definition of every component to production quality.

Despite representing a fraction of the product development time, conceptual design will determine up to 80% of the committed development costs [5]. Efficient design during this phase heavily relies on experience and semi-empirical methods to rapidly evaluate the performance (in terms of aerodynamics, weight, propulsion, ...) of multiple, continuously evolving configuration proposals, in order to reduce the inherent large uncertainties and, by extension, mitigate potentially unwanted redesigns. These *handbook* methods typically employ empirical and analytical correlations to scale experimental data to produce estimates of the projected aircraft's performance.

As novel aircraft configurations are proposed in response to growing challenges in aircraft design, the semi-empirical handbook methods risk being solicited beyond their domain of validity, unable to account for complex interactions between the aircraft's components. For analogous reasons, classic approaches based on Cayley's paradigm may no longer be sufficient to guide manufacturers towards competitive, *first-time right* solutions.

To surpass these limitations, there is a need for rapid prototyping and uncomplicated re-design of aircraft configurations and components, with reliable and automatic evaluation of performance, using robust physics-based models, all while adopting a multi-disciplinary approach from cradle to grave. The objective is to provide the decision-maker with as much information as possible, as early as possible.

The CEASIM environment (*Computerised Environment for Aircraft Synthesis and Integrated Optimisation Methods*) currently being developed at CFS Engineering aims to fulfil this purpose by serving as a unifying framework for software tools dedicated to aerodynamics, structures and flight dynamics [6]. Its centralising approach reduces the complexity of interaction between various analysis tools for aircraft design. As product development progresses, results generated with increasing fidelity can gradually be included to generate and refine the performance database.

For the estimation of aerodynamic loads, this progression from low- to high-fidelity methods could be summarised as follows:

- L0** Semi-empirical handbook methods
- L1** Linear potential flow methods
- L2** Numerical resolution of Euler equations
- L3** Numerical resolution of Navier-Stokes equations

The reconciliation of data generated with varying levels of uncertainty produces reliable information earlier during the design process, at reduced computational expense.

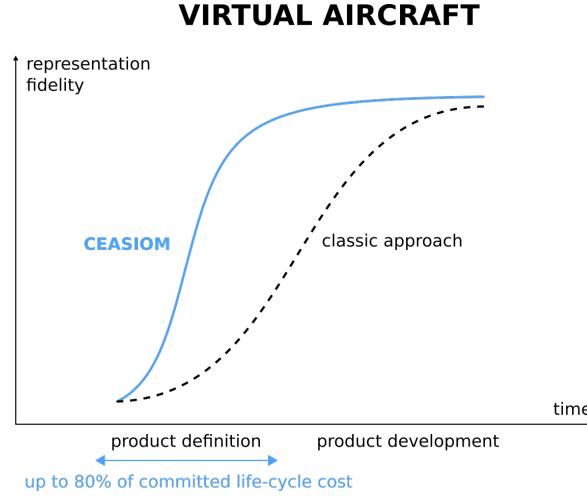


Figure 1.2: Increased fidelity at early design stages using CEASIOM (adapted from [6]).

It is challenging to extend this approach to aero-elasticity. It must be ensured that the final aircraft structure will not be susceptible to unwanted static and dynamic interactions with aerodynamic loads, which often requires experiments and complex, inherently multi-disciplinary modelling. Erroneous prediction or belated detection of undesirable aero-elastic behaviour can lead, in the best of cases, to costly and non-optimal *a posteriori* adjustments to the design. However, sufficiently accurate estimations of an aircraft's structural weight become available only at later stages in the design process, once its subsystems are defined to some degree of detail.

The current project stems from a need to address the problems outlined by the above considerations: the objective is to lay the groundwork for a tool capable of providing low-to mid-fidelity estimates, at low computational cost, of the dynamic aero-elastic behaviour of arbitrary aircraft configurations at the conceptual design stage, to bridge the existing gap between the handbook methods and high-fidelity computational tools already implemented in CEASIOM framework.

This report serves to document in detail the development of a program for the calculation of the unsteady aerodynamic characteristics of arbitrary aircraft configurations, using the Doublet-Lattice Method. The intention is to provide the foundations of a numerical tool dedicated to flutter analysis during conceptual design.

First, the theory behind the Doublet-Lattice Method is presented in Section 3. Technical aspects related to the implementation of the method and the development of the code are provided in Section 4. In Section 5 results are presented for the validation process using selected test cases.

2 Computational Flutter Analysis

Flutter is a dynamic instability arising from the interaction of elastic deformations of a structure with the unsteady aerodynamic loading it induces. Predicting the onset of this phenomenon is an inherently inter-disciplinary problem, governed by coupled inertial, elastic and aerodynamic effects.

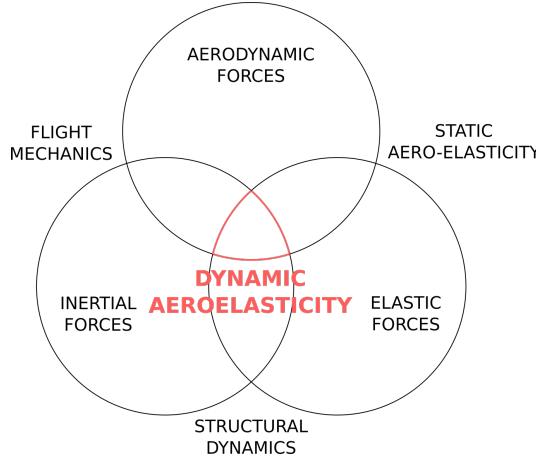


Figure 2.1: Representation of the interactions between phenomena in aero-elasticity. The study of flutter belongs in the field of dynamic aeroelasticity, at the intersection between inertial, elastic and aerodynamic effects.

The objective of flutter analysis is to determine the *flutter* or *critical* velocity U_{flutter} for a range of operating conditions; this is the airspeed at which the response of the aircraft structure due to interactions with the aerodynamic loading becomes unstable.

If one assumes harmonically oscillating motion, the flutter velocity is the lowest airstream velocity for which the oscillations are no longer damped. There exist multiple methods, of varying sophistication, to determine the point at which this occurs; in this chapter we concisely present perhaps the simplest of these, the *K-method*, in order to illustrate the role of the Doublet-Lattice Method in flutter analysis.

The K-method, also known as the $V - g$ or the british method, is based on the repeated solution of the following eigenvalue problem, shown here without derivation:

$$\left\{ \left[\frac{k^2}{L_{\text{ref}}^2} [M] + \frac{\rho_\infty}{2} [Q(M_{\infty,k})] \right] p^2 + [K] \right\} \vec{q} = 0 \quad (2.1)$$

where $p \equiv \omega/(1+ig)$ is a complex eigenvalue, \vec{q} the vector of generalised coordinates of the system, $[M]$ the matrix of generalised structural masses, $[K]$ the generalised structural stiffness matrix and $[Q]$ the matrix of generalised aerodynamic forces, a function of the free-stream Mach number M_∞ and of the reduced frequency $k \equiv \omega L_{\text{ref}}/U_\infty$, with ω the circular frequency of oscillation, U_∞ , ρ_∞ the free-stream velocity and density, and L_{ref} a reference length (characteristic of the flow, typically half the reference chord). In some formulations, a generalised structural damping matrix $[B]$ is also included.

The reduced frequency k gives a measure of the unsteadiness of the flow. It can be seen as the ratio of the characteristic length L_{ref} of the flow and U_∞/ω , the characteristic wavelength of the unsteady perturbations. For most aircraft, the reduced frequency rarely attains values in excess of the order of unity.

For a complete derivation of the equations of motion for wings in flutter, leading to (2.1), the reader is directed to the very complete work of Bisplinghoff, Ashley and Halfman [7].

For the K-method of flutter prediction, the eigenvalue problem (2.1) is solved for a set of Mach numbers M_∞ and reduced frequencies k . The velocity U , circular frequency ω and damping g of the system are tracked at each iteration. These are obtained from the eigenvalues p using the relations $g = 2\text{Re}(p)/\text{Im}(p)$, $\omega = k\text{Im}(p)/L_{ref}$ and $U = \text{Im}(p)$. Finally, the flutter point is determined graphically on a so-called $V - g$ plot, in which the system damping is traced against the system velocity for each deformation mode. The flutter point is determined by identifying where the system damping vanishes, at the lowest corresponding velocity.

It is clear from the above that computational methods for the detection of flutter require three components: a method for the evaluation of the aircraft's structural mass and stiffness properties, a method for the estimation of the generalised unsteady aerodynamic loads, and an interpolation scheme to ensure the correct coupling of both models. The fidelity of the analysis is dependent on the fidelity of these components, which must be chosen in function of the requirements of the aero-elastic study.

It can be argued that, of these, the accurate estimation of unsteady aerodynamic loads presents the greater challenge. The choice of structural model is often not limiting in terms of computational cost, and is rather dictated by the information available at early stages of aircraft design.

By contrast, the selection of the method to be used for the estimation of the unsteady aerodynamic loads is generally determining for the precision of the results and the computational cost incurred to obtain them. In the context of the conceptual design phase, it is desirable to rapidly and repeatedly estimate the susceptibility of multiple candidate aircraft configurations to unwanted, potentially destructive phenomena, such as flutter, as early as possible. The evaluation of the generalised aerodynamic forces [Q] must then be rapid and applicable to a variety of geometries, even if these are not defined in detail.

The Doublet-Lattice Method is a numerical method for linearised potential flow based on results of oscillating lifting-surface theory, introduced by Albano and Rodden [8] in 1969, and later refined by Rodden [9, 10]. The method may be regarded as an extension of Hedman's Vortex-Lattice Method [11] to unsteady aerodynamics. Among other, similar techniques, the Doublet-Lattice Method stands out as it is still extensively in use today for flutter clearance where the complexity of dynamic Fluid-Structure Interaction simulations using high-fidelity Computational Fluid Dynamics is not affordable. The method allows for the calculation of unsteady aerodynamic loads and is applicable to very general aircraft configurations, represented simply by thin lifting surfaces. As such, this method lends itself particularly well to flutter analysis at early design stages, where the uncertainty in the knowledge of the aircraft outweighs the increased accuracy that could be obtained by taking into account viscosity and non-linear effects.

3 Theory of the Doublet-Lattice Method

This chapter is intended to provide the reader with the theory leading to the formulation of the Doublet-Lattice Method for general aircraft configurations presented in Section 3.3. In Section 3.1 we present source and doublet flows, which serve as building blocks in many computational methods using the potential flow assumption. In Section 3.2, an integral equation is derived for the normal velocity induced by an oscillating lifting surface modeled as a continuous sheet of time-varying acceleration potential doublets. Working forms for numerical evaluation of the so-called kernel function are provided. Treatment of boundary conditions for thin lifting surfaces is addressed in Section 3.4.

We take as a starting point the governing Partial Differential Equation (PDE) for the behavior of small perturbations of the velocity potential ϕ in inviscid, irrotational and compressible flow, linearised around a uniform, parallel flow with velocity U_∞ along x :

$$(1 - M_\infty^2) \tilde{\phi}_{xx} + \tilde{\phi}_{yy} + \tilde{\phi}_{zz} - \frac{2U_\infty}{a_\infty^2} \tilde{\phi}_{xt} - \frac{1}{a_\infty^2} \tilde{\phi}_{tt} = 0 \quad (3.1)$$

with a_∞ the speed of sound in the far-field flow, and $M_\infty \equiv U_\infty/a_\infty$ the corresponding Mach number. The theory leading to this so-called *linearised aerodynamic potential equation* is found in Appendix A. The boundary conditions needed to solve the PDE follow from the unperturbed flow condition at infinity and tangential flow over the lifting surface S ,

$$\tilde{\phi}|_{\text{far-field}} = 0 \quad \text{and} \quad \frac{\partial S}{\partial t} + \nabla \tilde{\phi} \cdot \nabla S = 0 \quad (3.2)$$

where the latter condition holds only on S .

The tilde denotes time-dependent perturbations to the steady far-field flow $\bar{\phi} = U_\infty x$. In behalf of readability, this notation is omitted in the following.

3.1 Elementary solutions of the Linearised Aerodynamic Potential Equation

The Doublet-Lattice Method employs the linearity of the governing PDE to solve complex problems (e.g. flow around a flight vehicles) by superposition of elementary solutions.

The Moving Source A particularly relevant example is the *uniformly moving source*, which may be assimilated to the unit-impulsive solution, or Green function, of equation (3.1). The velocity potential perturbation induced at a receiving point (x, y, z) by a source in (ξ, η, ζ) moving along x with constant velocity U_∞ is

$$\phi_{\text{source}}(x, y, z, \xi, \eta, \zeta, t) = \frac{1}{R} f_\phi(t - \tau) \quad (3.3)$$

where f_ϕ is the perceived strength of the source, a function of time and the delay required for a disturbance to travel from the emitting source point to the receiving point:

$$\tau = \frac{-M_\infty(x - \xi) + R}{a_\infty \beta_\infty^2} \quad (3.4)$$

where we have introduced the hyperbolic radius $R \equiv [(x - \xi)^2 + \beta_\infty^2(y - \eta)^2 + \beta_\infty^2(z - \zeta)^2]^{1/2}$ and the Prandtl-Glauert factor $\beta_\infty^2 \equiv 1 - M_\infty^2$.

The reader may find a complete derivation of this result in [12]. Here, we present merely a qualitative description in order to clarify the origin of the parameter τ , also referred to as *retarded time*.

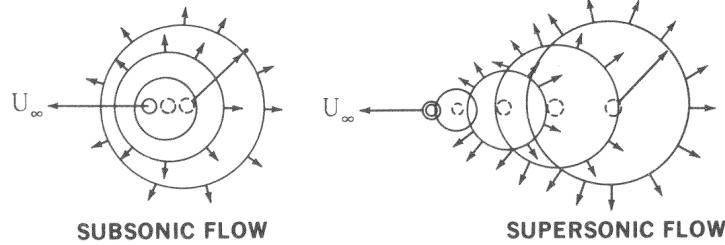


Figure 3.1: Series of acoustic pulses with origin moving at constant velocity U_∞ [12].

Consider a succession of stationary acoustic pulses, continuously distributed along the direction of uniform motion x (see Figure 3.1). At any moment t , one and only one point on this line emits a perturbation that radiates outward with velocity a_∞ . The origin (ξ, η, ζ) of each pulse travels along x with uniform velocity U_∞ .

The radius of the perturbation wave emitted at some anterior time t^* is $\varrho = a_\infty(t - t^*)$. This pulse will arrive in a receiving point (x, y, z) when

$$\varrho^2 = [x - \xi + U_\infty(t - t^*)]^2 + (y - \eta)^2 + (z - \zeta)^2 \quad (3.5)$$

In this case, the delay $t - t^*$ is equivalent to the retarded time τ . Substituting the expression for ϱ into (3.5) yields a quadratic equation in τ satisfied by (3.4).

The moving source solution as such is of limited interest for the representation of lifting surfaces, as it cannot directly generate pressure differences. Furthermore, we are primarily interested in the unsteady pressure distribution over a body, provided appropriate boundary conditions; in order to avoid intermediate calculations, it is then preferable to express the solution directly in terms of the pressure.

The Acceleration Potential Source For small perturbations around the uniform mean flow U_∞ along x , Prandtl's acceleration potential Ψ differs from the pressure by a constant only,

$$\Psi = \frac{p - p_\infty}{\rho_\infty}$$

It is related to the velocity potential by the material derivative

$$\Psi(x, y, z, t) = \left[\frac{\partial}{\partial t} + U_\infty \frac{\partial}{\partial x} \right] \phi(x, y, z, t) \quad (3.6)$$

which gives the inverse relation,

$$\phi(x, y, z, t) = \frac{1}{U_\infty} \int_{-\infty}^x \Psi(\lambda, y, z, t - \frac{x - \lambda}{U_\infty}) d\lambda \quad (3.7)$$

where λ is the integration variable.

It can be verified that the acceleration potential also satisfies the linearised aerodynamic potential equation (3.1), and therefore admits the same elementary solutions. The definition of an acceleration potential point source follows by straightforward analogy with (3.3),

$$\Psi_{\text{source}}(x, y, z, \xi, \eta, \zeta, t) = \frac{1}{R} f_\Psi(t - \tau) \quad (3.8)$$

with R , β_∞ , and τ defined as previously.

In the following we restrict our analysis to sinusoidally oscillating solutions of the form $\Psi_{\text{source}} = A_\Psi \exp(i\omega t)$, where A_Ψ is the amplitude, ω the oscillation frequency and $i \equiv \sqrt{-1}$ the imaginary unit. The acceleration potential source may be written explicitly:

$$\Psi_{\text{source}} = \frac{A_\Psi}{R} \exp[i\omega(t - \tau)] = \frac{A_\Psi}{R} \exp(-i\omega\tau) \exp(i\omega t) = \bar{\Psi}_{\text{source}} \exp(i\omega t)$$

where $\bar{\Psi}_{\text{source}}(x, y, z, \xi, \eta, \zeta)$ is the modulus. Using (3.4) we obtain

$$\bar{\Psi}_{\text{source}} = \frac{A_\Psi}{R} \exp\left[\frac{i\omega}{a_\infty \beta_\infty^2} [M_\infty(x - \xi) - R]\right] \quad (3.9)$$

The Acceleration Potential Doublet Equation (3.9) is the oscillatory moving source solution in terms of the pressure. It remains to formulate an elementary solution to (3.1) which could be used to generate a pressure differential. For this, we introduce the *doublet*, constructed by superimposing two infinitesimally adjacent sources of equal and opposite strengths.

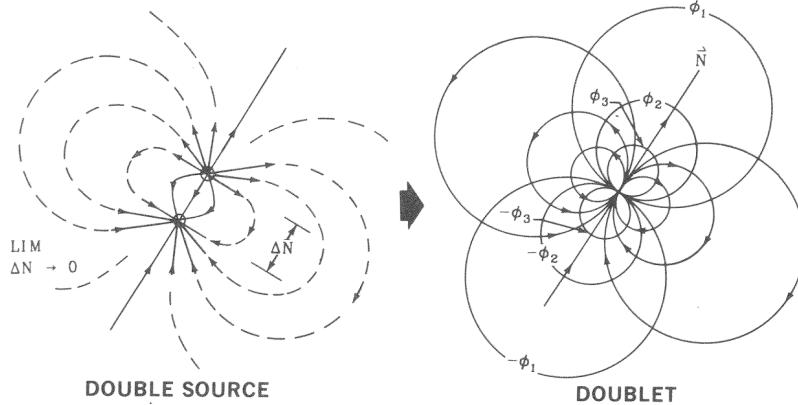


Figure 3.2: Derivation of a doublet from infinitesimally close equal and opposite point sources [12].

Mathematically, this is equivalent to taking the directional derivative of the expression for the source solution Ψ_{source} along the vector \vec{N}_s separating the source and the sink; in practice, \vec{N}_s is the normal vector of the lifting surface at (ξ, η, ζ) .

$$\Psi_{\text{doublet}} \equiv \bar{\Psi}_{\text{doublet}} \exp(i\omega t) = \frac{\partial \Psi_{\text{source}}}{\partial \vec{N}_s} \quad (3.10)$$

with modulus

$$\bar{\Psi}_{\text{doublet}} = \frac{\partial}{\partial \vec{N}_s} \left\{ \frac{A_\Psi}{R} \exp\left[\frac{i\omega}{a_\infty \beta_\infty^2} [M_\infty(x - \xi) - R]\right] \right\} \quad (3.11)$$

3.2 The Integral Equation for a Pressure Doublet Sheet and its Kernel Function

Equation (3.11) is the desired elementary solution. We will now outline the derivation of the integral formula for the normalwash induced by a continuous sheet of acceleration potential doublets used to model the pressure difference generated by a thin wing. The reference material for this section may be found in Section IV of Andrews and Vivian [13].

The Integral Equation Substitution of relation (3.7) into (3.11) permits to express the magnitude of the velocity potential induced by an acceleration potential doublet as

$$\bar{\phi}(x, y, z, \xi, \eta, \zeta) = -\frac{A_\Psi}{U_\infty} \frac{\partial}{\partial N_s} \exp\left[\frac{i\omega}{U_\infty}(x-\xi)\right] \int_{-\infty}^{x-\xi} \frac{1}{R'} \exp\left\{i\omega\left[\frac{\lambda}{U_\infty} + \frac{M_\infty \lambda}{a_\infty \beta_\infty^2} - \frac{R'}{a_\infty \beta_\infty^2}\right]\right\} d\lambda \quad (3.12)$$

where the radial measure R' in the integrand is a function of the integration variable λ ,

$$R' \equiv [\lambda^2 + \beta_\infty^2(y-\eta)^2 + \beta_\infty^2(z-\zeta)^2]^{1/2} \quad (3.13)$$

The normal velocity component \bar{w}_N induced at some receiving point (x, y, z) is obtained from the velocity potential by differentiation along the local surface normal \vec{N}_r .

$$\bar{w}_N(x, y, z, \xi, \eta, \zeta) = -\frac{A_\Psi}{U_\infty} \frac{\partial}{\partial N_r} \frac{\partial}{\partial N_s} \exp\left[\frac{i\omega}{U_\infty}(x-\xi)\right] \int_{-\infty}^{x-\xi} \frac{1}{R'} \exp\left\{i\omega\left[\frac{\lambda}{U_\infty} + \frac{M_\infty \lambda}{a_\infty \beta_\infty^2} - \frac{R'}{a_\infty \beta_\infty^2}\right]\right\} d\lambda \quad (3.14)$$

The directional derivatives can be decomposed as

$$\frac{\partial}{\partial N_s} = \cos \Gamma_s \frac{\partial}{\partial z} - \sin \Gamma_s \frac{\partial}{\partial y} \quad (3.15a)$$

$$\frac{\partial}{\partial N_r} = \cos \Gamma_r \frac{\partial}{\partial z} - \sin \Gamma_r \frac{\partial}{\partial y} \quad (3.15b)$$

where we use Γ_s and Γ_r to denote the angle between the surface and the xy -plane at the sending point s and receiving point r , respectively. Notice that $\exp[i\omega(x-\xi)/U_\infty]$ has been taken out of the integral, as it is independent of the normal directions \vec{N}_s and \vec{N}_r .

It can be shown that the pressure differential induced by an infinitesimal element of the acceleration potential doublets sheet is related to its strength A_Ψ as

$$\Delta \bar{p} = 4\pi \rho_\infty A_\Psi d\xi d\sigma \quad (3.16)$$

with ξ and σ the tangential coordinates. Substituting (3.16) into the expression for the normal velocity magnitude (3.14), integrating over a sheet S of pressure doublets and dividing by U_∞ , we obtain the desired integral equation in non-dimensional form

$$\frac{\bar{w}_N}{U_\infty} = \frac{-1}{4\pi \rho_\infty U_\infty^2} \iint_S \Delta \bar{p}(\xi, \eta, \zeta) K(x-\xi, y-\eta, z-\zeta) d\xi d\sigma \quad (3.17)$$

Equation (3.17) has been simplified by introducing the so-called *kernel function* K ,

$$K(x_0, y_0, z_0) \equiv \exp\left[\frac{i\omega x_0}{U_\infty}\right] \frac{\partial}{\partial N_r} \frac{\partial}{\partial N_s} \int_{-\infty}^{x_0} \frac{1}{R'} \exp\left[i\omega \frac{\lambda - M_\infty R'}{U_\infty \beta_\infty^2}\right] d\lambda \quad (3.18)$$

where it was convenient to define the relative coordinate system (x_0, y_0, z_0) defined as

$$\begin{aligned} x_0 &\equiv x - \xi \\ y_0 &\equiv y - \eta \\ z_0 &\equiv z - \zeta \end{aligned}$$

The integral formula (3.17) relates the unknown pressure differential $\Delta\bar{p}$ generated by a thin lifting surface, modeled as a sheet of acceleration potential doublets, to the velocity normal to the surface \bar{w}_N (henceforth referred to as *normalwash*) in a receiving point. The latter follows from the impermeability condition at the lifting surface (see Section 3.4).

The Kernel Function The remaining difficulty lies in evaluating the kernel function K , which shows singular behavior as the receiving point approaches the doublet sheet.

Rodemich [13] shows that the kernel function may be cast in the form

$$K(x_0, y_0, z_0) = \exp\left[\frac{-i\omega x_0}{U_\infty}\right] \frac{K_1 T_1 + K_2 T_2}{r^2} \quad (3.19)$$

where $r \equiv (y_0^2 + z_0^2)^{1/2}$ and

$$\begin{aligned} T_1 &= \cos(\Gamma_r - \Gamma_s) \\ T_2 &= \left[\frac{z_0}{r} \cos \Gamma_r - \frac{y_0}{r} \sin \Gamma_r \right] \left[\frac{z_0}{r} \cos \Gamma_s - \frac{y_0}{r} \sin \Gamma_s \right] \end{aligned}$$

For the demonstration of this result, the reader is directed to Vivian and Andrews [13]. Landahl [14] proposes a simplified expression for the terms K_1 and K_2 , of the form

$$\begin{aligned} K_1 &= I_1 + \frac{M_\infty r}{R} \frac{\exp(-ik_1 u_1)}{(1+u_1^2)^{1/2}} \\ K_2 &= -3I_2 - \frac{ik_1 M_\infty^2 r^2}{R^2} \frac{\exp(-ik_1 u_1)}{(1+u_1^2)^{1/2}} - \frac{M_\infty r}{R} \left[(1+u_1^2) \frac{\beta_\infty^2 r^2}{R^2} + \frac{M_\infty r u_1}{R} + 2 \right] \frac{\exp(-ik_1 u_1)}{(1+u_1^2)^{3/2}} \end{aligned}$$

with

$$\begin{aligned} k_1 &\equiv \frac{\omega r}{U_\infty} \\ u_1 &\equiv \frac{M_\infty R - x_0}{\beta_\infty^2 r} \end{aligned}$$

and the integrals

$$\begin{aligned} I_1(u_1, k_1) &\equiv \int_{u_1}^{\infty} \frac{\exp(-ik_1 u_1)}{(1+u^2)^{3/2}} du \\ I_2(u_1, k_1) &\equiv \int_{u_1}^{\infty} \frac{\exp(-ik_1 u_1)}{(1+u^2)^{5/2}} du \end{aligned}$$

We recall the hyperbolic radius in terms of the new coordinates, $R = (x_0^2 + \beta_\infty^2 r^2)^{1/2}$.

The integrals I_1 and I_2 are to be evaluated numerically. Integration by parts yields

$$I_1(u_1, k_1) = \exp(-ik_1 u_1) \left\{ 1 - \frac{u_1}{(1+u_1^2)^{1/2}} - ik_1 I_0(u_1, k_1) \right\} \quad (3.21a)$$

$$\begin{aligned} 3I_2(u_1, k_1) &= \exp(-ik_1 u_1) \left\{ (2+ik_1 u_1) \left[1 - \frac{u_1}{(1+u_1^2)^{1/2}} \right] \right. \\ &\quad \left. - \frac{u_1}{(1+u_1^2)^{3/2}} - ik_1 I_0(u_1, k_1) + k_1^2 J_0(u_1, k_1) \right\} \end{aligned} \quad (3.21b)$$

with I_0 and J_0 the integrals

$$I_0(u_1, k_1) \equiv \exp(ik_1 u_1) \int_{u_1}^{\infty} \left[1 - \frac{u}{(1+u^2)^{1/2}} \right] \exp(-ik_1 u) du \quad (3.22a)$$

$$J_0(u_1, k_1) \equiv \exp(ik_1 u_1) \int_{u_1}^{\infty} \left[1 - \frac{u}{(1+u^2)^{1/2}} \right] u \exp(-ik_1 u) du \quad (3.22b)$$

Laschka [15] proposes an accurate approximation of I_0 and J_0 for $u_1 \geq 0$.

$$I_0(u_1, k_1) \approx \sum_{n=1}^{11} \frac{a_n \exp(-ncu_1)}{n^2 c^2 + k_1^2} (nc - ik_1) \quad (3.23a)$$

$$\begin{aligned} J_0(u_1, k_1) &\approx \sum_{n=1}^{11} \frac{a_n \exp(-ncu_1)}{(n^2 c^2 + k_1^2)^2} \{ n^2 c^2 - k_1^2 + nc u_1 (n^2 c^2 + k_1^2) \\ &\quad - ik_1 [2nc + u_1 (n^2 c^2 + k_1^2)] \} \end{aligned} \quad (3.23b)$$

with the constant $c = 0.372$ and the coefficients a_n as given in Table 3.1. The error induced by this approximation has been shown not to exceed 0.135%.

n	coefficients a_n (L11)
1	0.241 861 98
2	-2.791 802 7
3	24.991 079
4	-111.591 96
5	271.435 49
6	-305.752 88
7	-41.183 630
8	545.985 37
9	-644.781 55
10	328.727 55
11	-64.279 511

Table 3.1: Coefficients for Laschka's L11 approximation [15].

In order to evaluate I_1 and I_2 for $u_1 < 0$ we can make use of their symmetry,

$$I_1(u_1, k_1) = 2 \operatorname{Re} I_1(0, k_1) - \bar{I}_1(-u_1, k_1) \quad (3.24a)$$

$$I_2(u_1, k_1) = 2 \operatorname{Re} I_2(0, k_1) - \bar{I}_2(-u_1, k_1) \quad (3.24b)$$

where the overbar denotes the complex conjugate.

3.3 The Doublet-Lattice Method

Equations (3.21), (3.23) and (3.24) are used to evaluate the kernel function (3.19) given an emitting point (ξ, η, ζ) and a receiving point (x, y, z) . In this section we introduce additional approximations used to solve the integral equation (3.17), leading to the formulation of the Doublet-Lattice Method for general aircraft configurations to be implemented our program, following the works of Albano [8], Giesing [16, 17] and Rodden [9, 18, 10, 19].

Geometry Idealisation Albano and Rodden [8] originally proposed to divide the geometry into n_{tot} of trapezoidal elements (Figure 3.3), referred to as *boxes* or *panels*, aligned chord-wise with the far-field flow and over which the pressure is assumed constant.

GEOMETRY DISCRETISATION

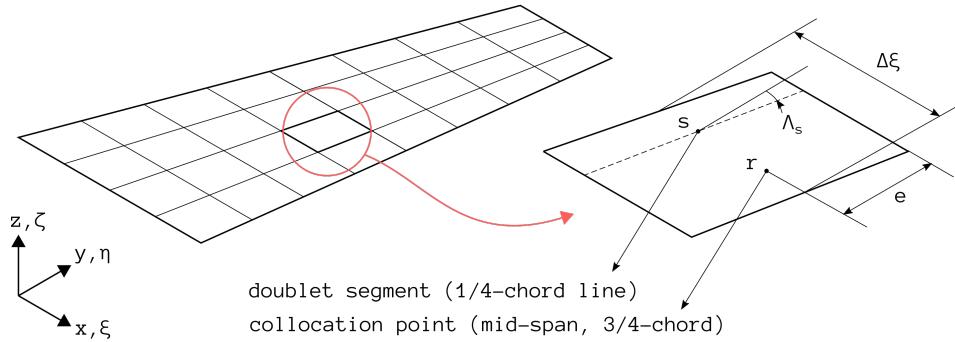


Figure 3.3: Discretisation of the lifting surface geometry and overview of panel properties.

In view of this discretisation, the integral equation (3.17) for the induced normalwash at a receiving panel r becomes a sum of the contributions from each emitting panel s , or

$$\begin{aligned} \frac{\bar{w}_{N,r}}{U_\infty} &= \sum_s \frac{-\Delta \bar{p}_s \Delta x_s}{4\pi \rho_\infty U_\infty^2} \iint_s K(x-\xi, y-\eta, z-\zeta) d\xi d\sigma \\ &= \sum_s -\Delta \bar{c}_{p,s} \frac{\Delta x_s}{8\pi} \iint_s K(x-\xi, y-\eta, z-\zeta) d\xi d\sigma \end{aligned}$$

with Δx_s the chord of the emitting panel and $\Delta \bar{c}_{p,s}$ the non-dimensional pressure coefficient

$$\Delta \bar{c}_{p,s} = \frac{\Delta \bar{p}_s}{\frac{1}{2} \rho_\infty U_\infty^2} \quad (3.25)$$

In addition, the influence of the doublet sheet over each element s is lumped onto a spanwise segment along the 1/4-chord line, further simplifying the integral to

$$\frac{\bar{w}_{N,r}}{U_\infty} = \sum_s -\Delta \bar{c}_{p,s} \frac{\Delta x_s}{8\pi} \int_s K(x-\xi_t, y-\eta, z-\zeta) d\sigma$$

The flow properties are evaluated at *collocation* or *control points* placed at midspan of the 3/4-chord line. It has been shown (empirically and a posteriori) that this placement of control points yields optimal results and satisfies the Kutta condition at the trailing edge.

The previous expression may be cast into the linear system

$$\{\bar{w}_N\} = [D]\{\Delta \bar{c}_p\} \quad \text{or} \quad \bar{w}_{N,r} = \sum_{s=1}^{n_{\text{tot}}} D_{rs} \Delta \bar{c}_{p,s} \quad (3.26)$$

where the $n_{\text{tot}} \times n_{\text{tot}}$ matrix $[D]$ relating the lifting pressure at a panel s to the normalwash at some other panel r , is the so-called matrix of *total downwash factors*,

$$D_{rs} \equiv \frac{\Delta x_s}{8\pi} \int_s K(x - \xi_{1/4}, y - \eta, z - \zeta) d\sigma \quad (3.27)$$

Low-Frequency Behavior It can be shown that the doublet segment reduces to a horseshoe vortex at the $1/4$ -chord line as the (reduced) frequency of motion approaches zero; the Doublet-Lattice Method then becomes equivalent to Hedman's Vortex-Lattice Method [11].

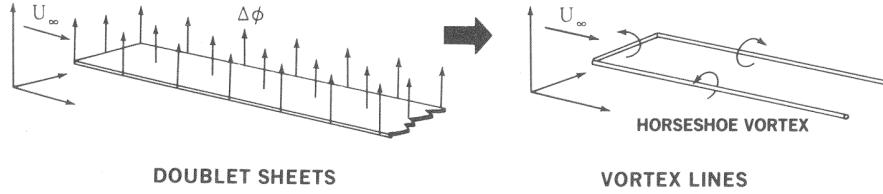


Figure 3.4: Equivalence between a steady, semi-infinite doublet sheet and a horseshoe vortex [12].

The steady component of the downwash factor D_{rs} may be computed exactly, whereas the integral in (3.27) is only approximated. In light of this observation, the results may be improved by restricting the approximation to the *incremental* downwash factor D_{rs}' , obtained by subtracting the limiting value $K^{(0)} \equiv \lim_{\omega \rightarrow 0} K$ from the integrand. The steady contribution $D_{rs}^{(0)}$ of a horseshoe vortex is determined exactly and added a posteriori,

$$D_{rs} = D_{rs}^{(0)} + D_{rs}' \quad (3.28)$$

The low-frequency limit $K^{(0)}$ of the kernel function K is given by [13, 8],

$$K_1^{(0)} \equiv \lim_{\omega \rightarrow 0} K_1 = 1 + \frac{x_0}{R} \quad (3.29)$$

$$K_2^{(0)} \equiv \lim_{\omega \rightarrow 0} K_2 = -2 - \frac{x_0}{R} \left(2 + \frac{\beta_\infty^2 r^2}{R^2} \right) \quad (3.30)$$

Parabolic Approximation of the Integrand In the original method it is proposed to fit the numerator of the integrand in the kernel function (3.19) with a parabolic function of the spanwise element coordinate $\bar{\eta}$,

$$D_{rs}' = \frac{\Delta x_s}{8\pi} \int_{-e}^{+e} \frac{P(\bar{\eta})}{r^2} d\bar{\eta}$$

$$\text{with } P(\bar{\eta}) = A\bar{\eta}^2 + B\bar{\eta} + C \approx (K_1 T_1 + K_2 T_2) \exp\left[\frac{-i\omega}{U_\infty} (\bar{x} - \bar{\eta} \tan \Lambda_s)\right] - (K_1^{(0)} T_1 + K_2^{(0)} T_2)$$

where the overbar denotes the coordinates are expressed in the plane of the sending element, relative to the center (ξ_c, η_c, ζ_c) of the doublet segment (see Figure 0.1):

$$\begin{aligned} \bar{x} &\equiv x - \xi_c & \bar{\xi} &\equiv \xi - \xi_c \\ \bar{y} &\equiv (y - \eta_c) \cos \Gamma_s + (z - \zeta_c) \sin \Gamma_s & \bar{\eta} &\equiv (\eta - \eta_c) \cos \Gamma_s + (\zeta - \zeta_c) \sin \Gamma_s \\ \bar{z} &\equiv (z - \zeta_c) \cos \Gamma_s - (y - \eta_c) \sin \Gamma_s & \bar{\zeta} &\equiv (\zeta - \zeta_c) \cos \Gamma_s - (\eta - \eta_c) \sin \Gamma_s \end{aligned}$$

in which Γ_s and Λ_s are the dihedral and sweep angles of the emitting panel.

This approximation was improved by Rodden et al [9], who distinguish the planar and non-planar terms in the expression for the incremental downwash factor D_{rs}' .

$$D_{rs} = D_{rs}^{(0)} + D_{rs}^{(1)} + D_{rs}^{(2)} \quad (3.31)$$

where the superscripts (0) , (1) and (2) denote respectively the steady, planar unsteady and non-planar unsteady contributions.

In fact, separate treatment of the planar and the non-planar components improves results for lifting surface elements aligned streamwise and slightly offset w.r.t. to the surface normal. Such configurations lead to large variations of the kernel function over the panel, rendering the original parabolic approximation inadequate.

For brevity we will simply present the resulting equations, without derivation.

$$D_{rs}^{(1)} = \frac{\Delta x_s}{8\pi} \int_{-e}^{+e} \frac{P_1(\bar{\eta})}{r^2} d\bar{\eta} \quad (3.32a)$$

with $P_1(\bar{\eta}) \equiv A_1 \bar{\eta}^2 + B_1 \bar{\eta} + C_1 \approx \left\{ K_1 \exp\left[\frac{-i\omega(\bar{x} - \bar{\eta} \tan \Lambda_s)}{U_\infty}\right] - K_1^{(0)} \right\} T_1$

$$D_{rs}^{(2)} = \frac{\Delta x_s}{8\pi} \int_{-e}^{+e} \frac{P_2(\bar{\eta})}{r^4} d\bar{\eta} \quad (3.32b)$$

with $P_2(\bar{\eta}) \equiv A_2 \bar{\eta}^2 + B_2 \bar{\eta} + C_2 \approx \left\{ K_2 \exp\left[\frac{-i\omega(\bar{x} - \bar{\eta} \tan \Lambda_s)}{U_\infty}\right] - K_2^{(0)} \right\} T_2^*$

where we have introduced $T_2^* \equiv r^2 T_2$.

The coefficients A_1, B_1, C_1 and A_2, B_2, C_2 of the parabolic interpolating functions are obtained from the values of the kernel numerator at three points along the doublet segment: $\bar{\eta} = -e$ (inboard), $\bar{\eta} = 0$ (midspan) and $\bar{\eta} = +e$ (outboard).

$$\begin{aligned} A_1 &= [P_1(+e) - 2P_1(0) + P_1(-e)]/2e^2 & A_2 &= [P_2(+e) - 2P_2(0) + P_2(-e)]/2e^2 \\ B_1 &= [P_1(+e) - P_1(-e)]/2e & \text{and} & B_2 = [P_2(+e) - P_2(-e)]/2e \\ C_1 &= P_1(0) & C_2 &= P_2(0) \end{aligned}$$

yielding, after substitution into (3.32) and integration by parts,

$$D_{rs}^{(1)} = \frac{\Delta x_s}{8\pi} \left\{ [(\bar{y}^2 - \bar{z}^2)A_1 + \bar{y}B_1 + C_1]H + \left(\frac{1}{2}B_1 + \bar{y}A_1\right) \log \left[\frac{(\bar{y}-e)^2 + \bar{z}^2}{(\bar{y}+e)^2 + \bar{z}^2} \right] + 2eA_1 \right\} \quad (3.34a)$$

$$\begin{aligned} D_{rs}^{(2)} &= \frac{\Delta x_s}{16\pi\bar{z}^2} \left\{ [(\bar{y}^2 - \bar{z}^2)A_2 + \bar{y}B_2 + C_2]H \right. \\ &\quad + \frac{1}{(\bar{y}+e)^2 + \bar{z}^2} [[(\bar{y}^2 + \bar{z}^2)\bar{y} + (\bar{y}^2 - \bar{z}^2)e]A_2 + (\bar{y}^2 + \bar{z}^2 + \bar{y}e)B_2 + (\bar{y} + e)C_2] \\ &\quad \left. - \frac{1}{(\bar{y}-e)^2 + \bar{z}^2} [[(\bar{y}^2 + \bar{z}^2)\bar{y} - (\bar{y}^2 - \bar{z}^2)e]A_2 + (\bar{y}^2 + \bar{z}^2 - \bar{y}e)B_2 + (\bar{y} - e)C_2] \right\} \end{aligned} \quad (3.34b)$$

in which it remains to evaluate the integral

$$H \equiv \int_{-e}^{+e} \frac{d\bar{\eta}}{(\bar{y} - \bar{\eta})^2 + \bar{z}^2} = \frac{1}{|\bar{z}|} \tan^{-1} \left[\frac{2e|\bar{z}|}{\bar{y}^2 + \bar{z}^2 - e^2} \right] \quad (3.35)$$

where the arctangent is taken in the interval $[0, \pi]$.

Rodden et al [10] present a refined scheme using quartic (polynomial order four) function to approximate the numerator of the integrands P_1 and P_2 , requiring the kernel function to be evaluated at five points along the doublet segment, see Appendix B. This higher-order fit boasts improved accuracy over a larger range of panel aspect ratios.

Remaining Singularities It is necessary to address the behavior of Equation (3.35) as $\bar{z} \rightarrow 0$. Rodden et al [9] propose to rewrite the integral as

$$H = \delta_1 \frac{2e}{\bar{y}^2 + \bar{z}^2 - e^2} (1 - \alpha \frac{\bar{z}^2}{e^2}) + \delta_2 \frac{\pi}{|\bar{z}|} \quad (3.36)$$

where the parameters δ_1 and δ_2 are used to restrict H to the correct quadrants (see below). The parameter α introduced above is defined as

$$\alpha \equiv \frac{e^2}{\bar{z}^2} \left\{ 1 - \frac{\bar{y}^2 + \bar{z}^2 - e^2}{2e|\bar{z}|} \tan^{-1} \left[\frac{2e|\bar{z}|}{\bar{y}^2 + \bar{z}^2 - e^2} \right] \right\} \quad (3.37)$$

Substituting (3.36) into the last expression of (3.34) then yields the more accurate form

$$\begin{aligned} D_{rs}^{(2)} &= \frac{\Delta x_s e}{8\pi(\bar{y}^2 + \bar{z}^2 - e^2)} \left\{ \frac{2(\bar{y}^2 + \bar{z}^2 + e^2)(e^2 A_2 + C_2) + 4\bar{y}e^2 B_2}{[(\bar{y}+e)^2 + \bar{z}^2][(\bar{y}-e)^2 + \bar{z}^2]} \right. \\ &\quad \left. - \frac{\delta_1 \alpha + \Delta}{e^2} [(\bar{y}^2 + \bar{z}^2)A_2 + \bar{y}B_2 + C_2] \right\} \end{aligned} \quad (3.38)$$

to be used when $|(\bar{y}^2 + \bar{z}^2 - e^2)/2e\bar{z}| > 0.1$. The parameter Δ is defined as

$$\Delta = 1 - \delta_1 + \delta_2 \frac{\pi}{|\bar{z}|} \left[\frac{\bar{y}^2 + \bar{z}^2 - e^2}{2e} \right] \quad (3.39)$$

We distinguish the following cases in function of $\bar{y}^2 + \bar{z}^2 - e^2$, which indicates the position of the receiving point with respect to the wake of the emitting doublet segment :

$$\delta_1 = 1 \quad \delta_2 = 0 \quad \text{for} \quad \bar{y}^2 + \bar{z}^2 - e^2 > 0 \quad (3.40a)$$

$$\delta_1 = 0 \quad \delta_2 = 1/2 \quad \text{for} \quad \bar{y}^2 + \bar{z}^2 - e^2 = 0 \quad (3.40b)$$

$$\delta_1 = 1 \quad \delta_2 = 1 \quad \text{for} \quad \bar{y}^2 + \bar{z}^2 - e^2 < 0 \quad (3.40c)$$

For $\bar{y}^2 + \bar{z}^2 - e^2 \leq 0$ expression (3.36) becomes singular with $\bar{z} \rightarrow 0$, behaving as $H \sim \pi/|\bar{z}|$ or as $H \sim \pi/2|\bar{z}|$ when $\bar{y}^2 + \bar{z}^2 - e^2 = 0$. This singularity is neutralised in D_{rs} as contributions from the terms involving H in $D_{rs}^{(1)}$ and $D_{rs}^{(2)}$ are equal and of opposite sign (see Rodden et al [9]). Still, numerical difficulties may arise for small values of \bar{z} : Giesing et al [16] report the limiting value $\bar{z}/e = 0.0025$, below which the loss of accuracy becomes significant. For such small values of \bar{z} , the problem may simply be assimilated to the planar case ($\bar{z} = 0$).

Conversely, (3.36) is well-behaved when $\bar{y}^2 + \bar{z}^2 - e^2 > 0$. Under these circumstances, the convenient series expansion

$$\alpha \approx \frac{4e^4}{(\bar{y}^2 + \bar{z}^2 - e^2)} \sum_{n=2}^7 \frac{(-1)^n}{2n-1} \left[\frac{2e\bar{z}}{\bar{y}^2 + \bar{z}^2 - e^2} \right]^{2n-4} \quad (3.41)$$

may be used when $|2e\bar{z}/(\bar{y}^2 + \bar{z}^2 - e^2)| \leq 0.3$.

We consider finally the case where $\bar{z} = 0$. Firstly, the contribution of the non-planar term $D_{rs}^{(2)}$ vanishes. The remaining singularity of the integral H is circumvented using the concept of principal values (see Mangler [20]),

$$H = \oint_{-e}^{+e} \frac{d\bar{\eta}}{(\bar{y} - \bar{\eta})^2} = \frac{1}{\bar{y} - e} - \frac{1}{\bar{y} + e} = \frac{2e}{\bar{y}^2 - e^2} \quad (3.42)$$

where the crossed integral sign denotes integration in the sense of principal values.

It is a requirement for the Doublet-Lattice Method that panels on nearly wings panels be aligned in span-wise strips. This way, the singularity of H in $\bar{y} = e$ is avoided.

Summary We now have the mathematical tools required to construct the matrix of total downwash factors [D]. Planar and non-planar contributions in equation (3.31) are computed using (3.34), (3.36) and (3.37), with relations (3.38) and (3.41) when appropriate, and the kernel function is evaluated using the equations presented in Section 3.2. The steady component is computed exactly using the Vortex-Lattice Method. Finally, the unsteady pressure distribution $\Delta\bar{c}_p$ is obtained by solving the resulting linear system (3.26).

In practice, it is convenient to introduce the non-dimensional variables

$$\hat{x} \equiv x/L_{\text{ref}}, \quad \hat{y} \equiv y/L_{\text{ref}}, \quad \hat{z} \equiv z/L_{\text{ref}}, \quad \hat{t} \equiv t(U_\infty/L_{\text{ref}}), \quad k \equiv \omega(L_{\text{ref}}/U_\infty) \quad (3.43)$$

where k is the *reduced frequency*. When expressed in terms of the new variables, it is easily verified that the equations presented in this section depend only on the far-field Mach number M_∞ , the reduced frequency k , and the aircraft geometry.

3.4 Boundary Conditions

We recall the boundary conditions the linearised aerodynamic potential equation (3.1),

$$\tilde{\phi} \Big|_{\text{far-field}} = 0 \quad \text{and} \quad \frac{\partial S}{\partial t} + \nabla \tilde{\phi} \cdot \nabla S = 0 \quad (3.44)$$

The far-field boundary condition is implicitly satisfied as the perturbation induced by sources and doublets dies out with increasing relative distance. For a thin lifting surface,

$$S(x, y, z, t) = h_0(x, y, z) - h(x, y, z, t) \pm h_t(x, y, z) = 0 \quad (3.45)$$

where $h(x, y, z, t)$ denotes the time-dependent mid-plane deformation, and $h_0(x, y, z)$ and $h_t(x, y, z)$ the surface mid-plane and its thickness envelope, both supposed time-invariant. Substitution into (3.44) gives

$$w_N = \frac{\partial h}{\partial t} + U_\infty \frac{\partial h}{\partial x} \quad (3.46)$$

In flutter analysis of arbitrary aircraft configurations, a number of mid-plane deformation modes h_i is determined from the aircraft structure's known (or estimated) properties, using computational methods in structural dynamics.

For simple wings however, it is convenient to represent deformation modes as polynomial functions of order n_X, n_Y, n_Z in X, Y, Z , the components of some coordinate system. In the present implementation of the Doublet-Lattice Method, this coordinate system is the non-dimensionalised coordinate system (x, y, z) of the aircraft, although in future development it may be worth considering allowing the definition of polynomial modes defined in the plane of the wing in question. The modes are assumed harmonic with frequency ω .

The out-of-plane deformation is thus constrained to the following form:

$$h(x, y, z, t) = \left[\sum_{j=0}^{n_X} \sum_{k=0}^{n_Y} \sum_{l=0}^{n_Z} a_{jkl} X^j Y^k Z^l \right] \exp(i\omega t) \quad (3.47)$$

which gives, upon substitution into the boundary condition (3.46) for the wing,

$$\begin{aligned} w_N &= \frac{\partial h}{\partial t} + U_\infty \frac{\partial h}{\partial x} \\ &= \left\{ i\omega \left[\sum_{j=0}^{n_X} \sum_{k=0}^{n_Y} \sum_{l=0}^{n_Z} j a_{jkl} X^{j-1} Y^k Z^l \right] + U_\infty \left[\sum_{j=0}^{n_X} \sum_{k=0}^{n_Y} \sum_{l=0}^{n_Z} a_{jkl} X^j Y^k Z^l \right] \right\} \exp(i\omega t) \end{aligned} \quad (3.48)$$

with complex modulus \bar{w}_N such that $w_N = \bar{w}_N \exp(i\omega t)$,

$$\bar{w}_N = i\omega \sum_{k=0}^{n_Y} \sum_{l=0}^{n_Z} a_{0kl} Y^k Z^l + \sum_{j=1}^{n_X} \sum_{k=0}^{n_Y} \sum_{l=0}^{n_Z} a_{jkl} (i\omega X^j Y^k Z^l + U_\infty j X^{j-1} Y^k Z^l) \quad (3.49)$$

Upon introduction of the non-dimensional variables (3.43), the boundary condition for the normalwash \bar{w}_N is dependent only on the reduced frequency k , the aircraft geometry and the normalised deformation modes $\hat{h}_i \equiv h_i / L_{\text{ref}}$.

3.5 Aerodynamic Coefficients and Generalised Aerodynamic Forces

In the preceding sections of this chapter, we have shown how the distribution of unsteady aerodynamic loads over an arbitrary configuration of lifting surfaces can be calculated using the Doublet-Lattice Method.

It may sometimes be required to calculate conventional aerodynamic coefficients from the resulting pressure coefficients. In addition to the information this data provides on the oscillatory aerodynamic characteristics of the considered wing or aircraft, it also serves as an excellent measure to be used for the validation of the program.

Lift Coefficient The non-dimensional lift coefficient may be calculated for an individual strip of panels using the formula

$$c_L(y) = \frac{\int_{x_{LE}(y)}^{x_{TE}(y)} \Delta \bar{c}_p dx}{\pi c(y)} \approx \frac{\sum_j \Delta \bar{c}_{p,j} \Delta x_j}{\pi c(y)} \quad (3.50)$$

where $\Delta \bar{c}_{p,j}$ the pressure coefficient at panel j , Δx_j is the mid-span chord of panel j , and $c(y) = x_{TE}(y) - x_{LE}(y)$ is the mid-span chord of the strip. Here, $x_{LE}(y)$ and $x_{TE}(y)$ are the x -coordinates of the leading and trailing edges at mid-span of the strip, respectively.

By extending the integration to the entire wing, we obtain the expression for the lift coefficient of the wing,

$$C_L = \frac{\iint_A \Delta \bar{c}_p dA}{\pi A_{ref}} \approx \frac{\sum_j \Delta \bar{c}_{p,j} A_j}{\pi A_{ref}} \quad (3.51)$$

where A_j is the panel surface area, and A_{ref} is the reference area, typically the surface area of the wing.

In the definitions above, taken from Bennekers and Labrujère [21], the summations \sum_j are intended as summations over the panels that constitute the considered wing or strip. It should be noted that the dependence of certain values on the y -coordinate is highlighted to indicate that these values are likely to vary along the span of the considered lifting surface, in the most general case. For vertical fins, this should be replaced with a dependence on the z -coordinate.

Generalised Aerodynamic Forces The Generalised Aerodynamic Forces (GAFs) Q_{ij} required to solve the flutter equation (2.1) can be calculated using the expression

$$Q_{ij} = -\frac{1}{b^2} \iint_A \Delta \bar{c}_p^{(j)} h^{(i)} dx dy \approx -\frac{1}{b^2} \sum_{k=1}^{n_{tot}} \Delta \bar{c}_{p,k}^{(j)} h_k^{(i)} A_k \quad (3.52)$$

where the superscripts (i) and (j) the i -th rigid or elastic deformation mode and the j -th pressure mode (pressure distribution resulting from the j -th rigid or elastic deformation mode), respectively. Here, b is the wing half-span, A_k is again the surface area of panel k and h_k are the normalised deflection at panel k . The summation is performed over all n_{tot} panels k constituting the studied wing or aircraft.

The matrix $[Q]$ of GAFs can be seen as the projection of the aerodynamic loads onto the *generalised* or *modal* coordinates of the system introduced in Chapter 2; each coefficient Q_{ij} represents the contribution of the pressure distribution resulting from the j -th structural deformation mode to the virtual work performed along the i -th structural deformation mode. It is implicitly assumed that the unsteady aerodynamic loads may be expressed as the linear superposition of the pressure distributions induced by a number of independent deformation modes.

For the solution of the flutter equation, it would only remain to interpolate the GAFs onto the structural grid.

4 Implementation in Python

In the preceding chapters we have reviewed the theoretical tools required for the calculation of unsteady aerodynamic characteristics by the Doublet-Lattice Method.

This chapter serves to fully document the final program and the development process, beginning with a review of the requirements to be fulfilled in Section 4.1. Relevant implementation aspects, such as code structure (Section 4.2) and efforts toward optimisation of performance in terms of execution time (Section 4.4) are then discussed. Compatibility of the program with CEASIOMpy and its centralised database format CPACS is described and demonstrated in Section 4.3.

As a preliminary note, the choice of programming language should be addressed. While it lacks the raw performance of FORTRAN and C/C++, Python boasts a flexible, readable syntax, which allows for rapid prototyping of new code, and simplifies maintenance and extension of existing code. It is freely available, well-documented and supported by an active community of users. A number of well-rounded libraries dedicated to scientific computing are available, making Python a viable alternative to high-level scientific programming environments such as MATLAB. Finally, Python applications can be complemented with components written in other programming languages where performance is critical.

4.1 Objectives

The primary objective of the project is straightforward: to develop a program that is able to produce the unsteady aerodynamic characteristics used in flutter analysis calculations for arbitrary aircraft configurations, using the Doublet-Lattice Method. One could refer to the set of scripts written purely to this end as the *core* of the program.

Then, a number of additional requirements arise from the following considerations: first, the program must be structured as to naturally accommodate its future extension to structural dynamics, a requirement for flutter analysis of arbitrary configurations. In addition, several auxiliary scripts must be written in order for the program to interact with the user, and with CEASIOMPy, the Python implementation of CEASIOM [5, 6], an integrated design environment for conceptual aircraft design currently in development at CFS Engineering.

In the following we summarise how the above-mentioned requirements outline the final product and influence the implementation of the method.

Unsteady Aerodynamics Solver As previously stated, the requirements for the Doublet-Lattice Method solver are self-evident: the solver must be able to produce sufficiently accurate results within sufficiently low execution times, for arbitrary aircraft configurations. Of course, the inherent limitations of the method are to be taken into account when evaluating the program's precision and performance.

More specifically, the solver must be able to output the unsteady pressure distribution, aerodynamic coefficients and Generalised Aerodynamic Forces (GAFs) for given operating conditions. These are required as input for the solution of the flutter equations. The Doublet-Lattice Method itself requires the geometry (lifting surface configuration) and the structural deflection of the flight vehicle as inputs (see Section 3).

Efforts during the development process are directed toward verification of the correct implementation of the method, and optimisation of the resulting program in terms of execution time. The validation process is detailed in Chapter 5, and measures taken to improve the program's performance are presented in Section 4.4.

Integration in CEASIOMPy Despite the seemingly secondary nature of the task, adapting the program to communicate efficiently with other modules and with the user tends to represent a significant fraction of the programming efforts incurred throughout the development process. Correctly identifying the requirements at an early stage is essential to minimise unnecessary work, and helps streamline the interaction between the different components of the program.

The program is to be used and further extended at CFS Engineering. In this sense, the structure of the code should provide a suitable framework for the remaining development. Naturally, particular attention must be paid to provide readable, well-documented code. Section 4.2 contributes to this purpose.

During the conceptual design phase, calculations must be performed in rapid succession for a number of quickly evolving prototype configurations. For this reason, it is important that the modification of the aircraft geometry be made straightforward, by allowing direct manipulation of crucial design parameters. This is also beneficial if the calculations must be run repeatedly by an external routine, such as CEASIOMPy, for creation of a performance database or for optimisation purposes. In these cases, it is also important the the pre- and post-processing steps be robustly automated as to minimise the need for user intervention during large calculation campaigns.

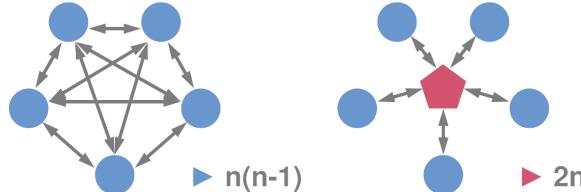


Figure 4.1: Centralised approach provided by the CPACS format [22, 23]. Using a common format reduces the complexity of the interactions between various components of an application.

The CEASIOMPy environment uses CPACS (*Common Parametric Aircraft Configuration Schema*) an aircraft database format developed at DLR [22, 23], to centralise the interactions between its components. In addition to the previous requirements, the program must be able to infer the idealised wing configuration used in the Doublet-Lattice Method from the complete definition of the aircraft found in the CPACS files. This way, its integration in CEASIOMPy can be made seamless.

Finally: the Doublet-Lattice Method relies on the Vortex-Lattice Method to evaluate the steady downwash factors (see Section 3). Rather than to implement a rudimentary Vortex-Lattice code in the current program, it is chosen to await the expected release of the existing, mature program TORNADO [24] in Python, to be integrated in CEASIOMPy. The future compatibility between the programs is to be ensured.

4.2 Program Structure

As is mentioned in the introduction to this section, one of the strengths of the Python programming language is its flexibility: routines constituting the program are separated in independent modules, which may be individually modified and reused without complication. Related subroutines are grouped in relevant modules to give a coherent structure to the program files.

The code consists of three modules: `utils`, `cas` and `dlm`. The first contains general purpose, boiler-plate routines required for reading and writing of data, as well as processing user inputs. The `cas` module contains the scripts used for the case setup: the definition of the aircraft geometry and the operating conditions. The last module contains routines for the discretisation of the aircraft geometry, and the Doublet-Lattice Method implementation. A fourth, empty directory named `fem` is included to act as a placeholder for the structural dynamics module during development.

The root folder contains the main script (named accordingly), which can be run from the command-line to execute the program.

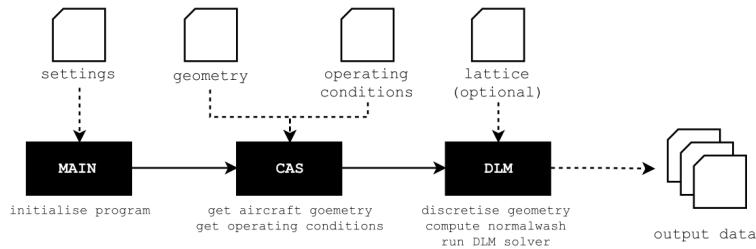


Figure 4.2: Schematic representation of Doublet-Lattice program.

Configuration Files Input parameters are read from configuration files located in the working directory; by default, this is the `files` folder located in the program files. Three files are required: one for the execution settings, one for the operating conditions, and one for the aircraft geometry.

The settings file must be passed as an argument when calling the program, as it contains the location of the remaining files. Herein, the user also sets the order of approximation of the kernel function (see Appendix B) and controls for the automatic geometry discretisation.

The operating conditions file is used to set the free-stream Mach number M_∞ and the reduced frequency k of excitation, as well as the reference length L_{ref} used to normalise the geometry and boundary conditions. The user may define polynomial deformation modes applied to the entire aircraft or to selected wings. These must be provided if no tool for structural dynamics is available (as is currently the case).

In order to allow the user to perform analyses of simple geometries without resorting to the CPACS format, it is also possible to define the aircraft geometry in a third input file, containing either the geometric properties of its parts or the coordinates of their corner points. The definition of the aircraft geometry is detailed further in this section.

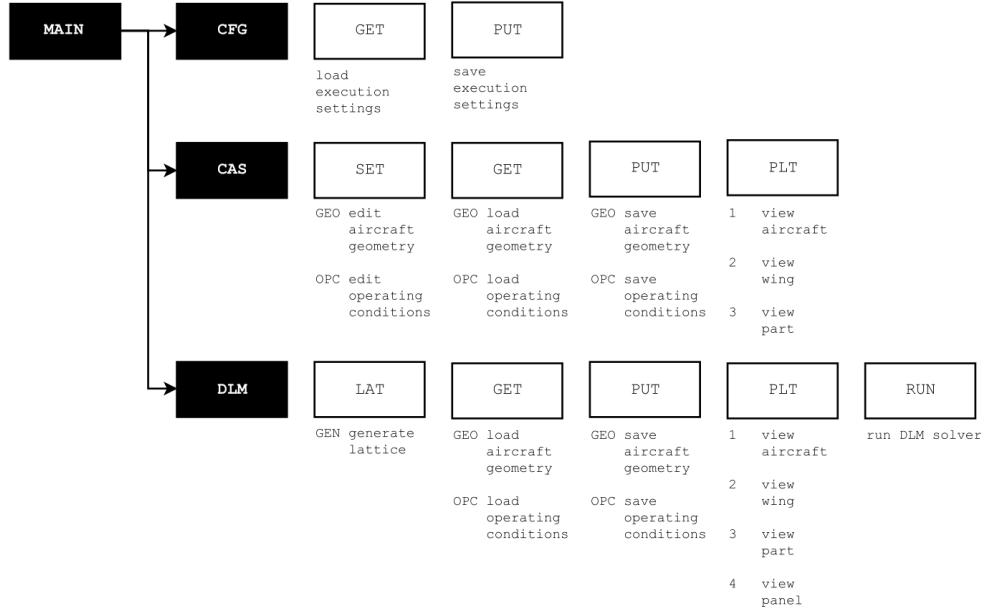


Figure 4.3: Schematic representation of Doublet-Lattice program's optional user interface.

For the input data to be correctly interpreted by the program, the configuration files must follow a certain format (see Appendix C). In order to facilitate the writing of configuration files, an optional text-based user interface has been added to the program. This allows users to rapidly and interactively edit a given case before programmatically generating the configuration files required for normal operation of the program.

Utility Routines Module The `utils` module contains auxiliary functions that are often reused throughout the code. These mainly perform tasks related to the input and output of configuration and data files, the processing of user input and the optional user interface. More abstract programming objects are defined in this module, which are mainly used to initialise the data structure and for error handling.

Case Setup Module The `cas` module serves as a pre-processor for the Doublet-Lattice solver. It contains the routines required to generate, modify and review the aircraft geometry and operating conditions, as well as the definitions of the corresponding data structures.

The aircraft geometry is composed of a number trapezoidal *parts*, which can be grouped in function of the *wing* they belong to. Parts can be added, removed, grouped and modified individually to represent complete aircraft configurations. For example, the main wing of the D150 model shown in Figure 4.6 is composed of three parts, while its horizontal and vertical stabilisers are each modelled by a single part. Using this definition, complex configurations remain straightforward to model and their discretisation is simplified; see the joined wing of the configuration displayed in Figure 4.9, composed of 50 trapezoidal components.

The components constituting the aircraft geometry must be defined by the user, either in the `.parts` or `.coord` configuration files, which contain the geometric parameters and the coordinates of the corner points of the component, respectively (see Appendix C). Alternatively, the geometry definition may be performed directly within the text-based user interface. The parameters required to generate each component are summarised and illustrated in Figure 4.4. If the aircraft geometry is instead defined in a CPACS `.xml` file, the program is able to reduce the three-dimensional geometry to its corresponding idealised lifting surface configuration, composed of thin trapezoidal parts; these parts correspond to *segments* defined in the CPACS aircraft definition (see Section 4.3). The properties defined in Figure 4.4 are automatically calculated for each part.

GEOMETRY DEFINITION

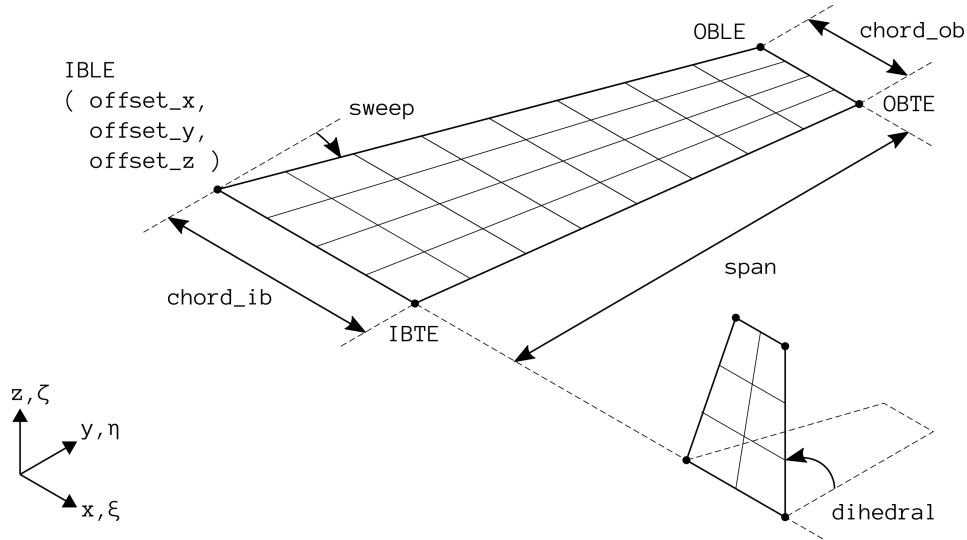


Figure 4.4: Definition of geometric properties of aircraft part. By convention, the corner points are ordered from in-board leading edge (IBLE) to out-board leading edge (OBLE) to out-board trailing edge (OBTE) to in-board trailing edge (IBTE). This ordering is consistent with the TORNADO Vortex-Lattice code.

Operating conditions are not yet included in the CPACS format, and must therefore be provided in a separate input `.oper` file. The Doublet-Lattice Method requires a Mach number M_∞ , a reduced frequency k and a reference length L_{ref} to be defined. The latter is typically taken to be the semi-chord of the main wing. It is used to non-dimensionalise the aircraft geometry and boundary conditions; aerodynamic coefficients are calculated using the wing and strip dimensions in terms of the non-dimensional coordinates.

Additionally, the user may provide a number of polynomial deformation modes $h^{(i)}$, defined in function of the global, non-dimensional coordinates (x, y, z) . Each mode may be restricted to a particular wing or applied to the entire aircraft.

Doublet-Lattice Method Module Once the case is properly defined, the data structures of the aircraft geometry and the operating conditions are sent to the core of the program, the `dlm` module. This module contains the routines for the discretisation of the geometry, the calculation of the downwash boundary conditions from the assumed modes and the computation of the matrix [D] of downwash factors.

The data structure for the discretised geometry was discussed with the developer of the TORNADO Vortex-Lattice Program [24], in order to ensure future compatibility of the present program with a Python version of TORNADO currently in the works. Stored are the coordinates of the control points, the doublet points (points along the doublet segments at which the kernel function is evaluated) and the panel corner points. For performance, these are stored in contiguous memory blocks - this is required if the data is to be passed to C++. Additional information is provided for each panel, namely the identifier of the aircraft part it belongs to, in order to selectively apply deformation modes, and a setting for symmetry.

In fact, it is possible to model only half the aircraft when the geometry and boundary conditions satisfy symmetry about the x , z -plane. This can be indicated in the geometry definition. The program is then able to take advantage of symmetry to calculate the pressure distribution in a reduced number of operations.

Currently, the program writes the matrix of downwash factors, the pressure distribution and the downwash to separate output files. These are easily read back into Python for further post-processing. The lift coefficient of the configuration is also computed and written to file. Each set of data is generated once for every mode included in the analysis.

Data Structures In summary, the `main.py` script run from the command-line initialises and passes the input data to the `cas` pre-processor module, where the aircraft geometry and operating conditions are generated and in turn passed to the `dlm` module, which performs the calculation of the unsteady pressure distributions.

The data passed between the program's routines is stored in a number of simple data structures, containing:

```

settings execution settings
aircraft aircraft geometry (trapezoidal lifting surface parts)
oper     operating conditions and polynomial modes
grid_dlm grid points for Doublet-Lattice Method
data      numerical data for calculations
log       warnings and errors

```

The `settings`, `aircraft` and `oper` data structures are constructed entirely from the input files, and are described in further detail in Appendix C. They are designed so that individual parts, modes, and other parameters may easily be accessed and manipulated. As was previously mentioned, the `grid_dlm` structure contains the coordinates of the points constituting the grid used by the Doublet-Lattice Method, stored in contiguous memory blocks for performance, ready to be passed to the C++ solver. Resulting downwash factors and pressure distributions are stored in similar format in the `data` object. Finally, a `log` data structure has been added to contain the errors, warnings and information messages raised during execution of the programs, which are written to a log file.

4.3 CPACS Compatibility

Among the objectives of the project, it is mentioned that efforts should be made to facilitate the future integration of the program or of a selection of its components in CEASIOM. Because the latter makes use of the CPACS database format to achieve a centralised structure, it is sufficient to ensure compatibility with this file format to allow the passing of data between the current program and any other tool included in CEASIOM (as illustrated in Figure 4.1).

It must also be taken into account that the program is likely to be executed repeatedly in sensitivity and rapid optimisation procedures typical of early design stages, meaning that the pre-processing routines should be automated as much as is possible in order to eliminate the need for human intervention at each model evaluation.

Consequently, the present program was extended with routines for the extraction of idealised lifting surface configurations from the CPACS aircraft definition and for automatic discretisation of the resulting geometry.

The CPACS format is structured in a hierarchical manner, intended to encompass all aspects of aircraft design from more general considerations to the most detailed definition. The components of the aircraft model are sorted by type (fuselages, wings, engines, etc.). Among these, we are interested only in the wings (this includes all lifting surfaces, such as fins, canards and winglets). An aircraft may have multiple *wings*, each partitioned into a number of *segments* with distinct properties.

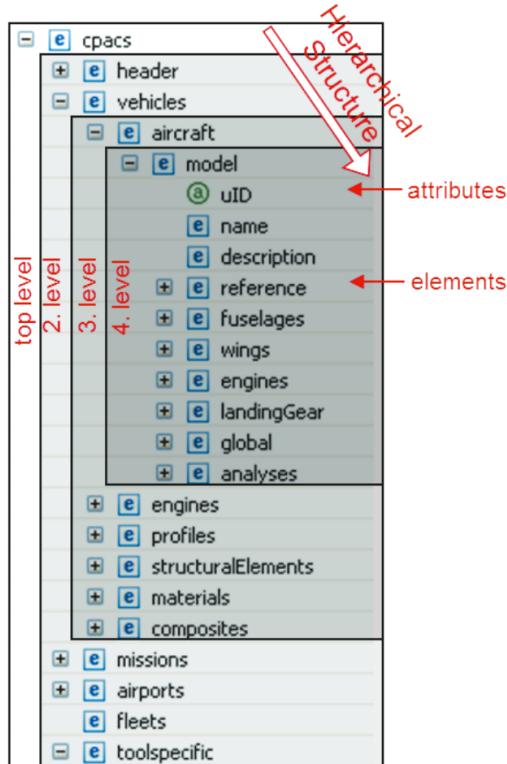


Figure 4.5: Hierarchical structure of the CPACS common aircraft definition developed at DLR [22].

CFS Engineering have provided a set of functions that extract the four corner points IBLE, OBLE, OBTE and IBTE of each segment from the CPACS file structure. The routines are contained within the `utils.TIXI` module (named after the library used to navigate the CPACS database, also developed by DLR), and are used in the `cas` pre-processor module to generate trapezoidal parts of thin lifting surface for the program as defined in Figure 4.4. In this sense, the parts defined in the Doublet-Lattice model correspond one-to-one with the segments defined in the CPACS file.

The resulting models are represented in Figures 4.6b, 4.7b, 4.8b, 4.9b, taken from CPACS definitions of selected aircraft from the AGILE Hangar. It is seen that even non-conventional geometries such as the strut-braced and box-winged aircraft are represented correctly.

Once the representation of the lifting surfaces by trapezoidal parts is obtained, it remains to sub-divide each part into panels for the Doublet-Lattice Method. Because the number of parts constituting a single wing can grow to be quite large depending on its geometry and how it is defined in the CPACS file (see the NASA CRM model of Figure 4.7 and the box-winged aircraft of Figure 4.9 for example), it can not be expected of the user to set a number of span- and chord-wise divisions manually for each part.

Instead, an additional routine for the automatic discretisation of the geometry is included in the `dlm` module. Only a single parameter needs to be provided, namely the number of chord-wise divisions on the main wing of the aircraft.

The procedure functions as follows: the main wing is identified as the wing containing the component with maximum root chord. A reference panel chord is calculated by dividing the average chord of the main wing by the parameter passed to the automatic paneling function, the number of chord-wise divisions. The reference panel chord is then used to determine the number of chord-wise divisions on every remaining wing. The number of span-wise strips for each parts is calculated ensuring minimum average panel aspect ratio.

In view of the guidelines for convergence given in Section 5.3, a minimum of four chord-wise divisions is enforced. Beyond this minimum, it is left to the user to ensure that the number of chord-wise divisions is sufficient with respect to the wavelength of the oscillations. For good measure, panels with aspect ratios beyond 3 (10 if a quartic approximation is used) are highlighted to warn the user of the potentially poor discretisation.

Further improvement of the automatic paneling feature is needed to ensure the accuracy of the Doublet-Lattice Method; for nearly coplanar surfaces, it is a requirement that panels be aligned in stream-wise strips, as to circumvent the singularity present in the integral H of (3.35). This is likely to require modification of the geometry as defined in the CPACS files.

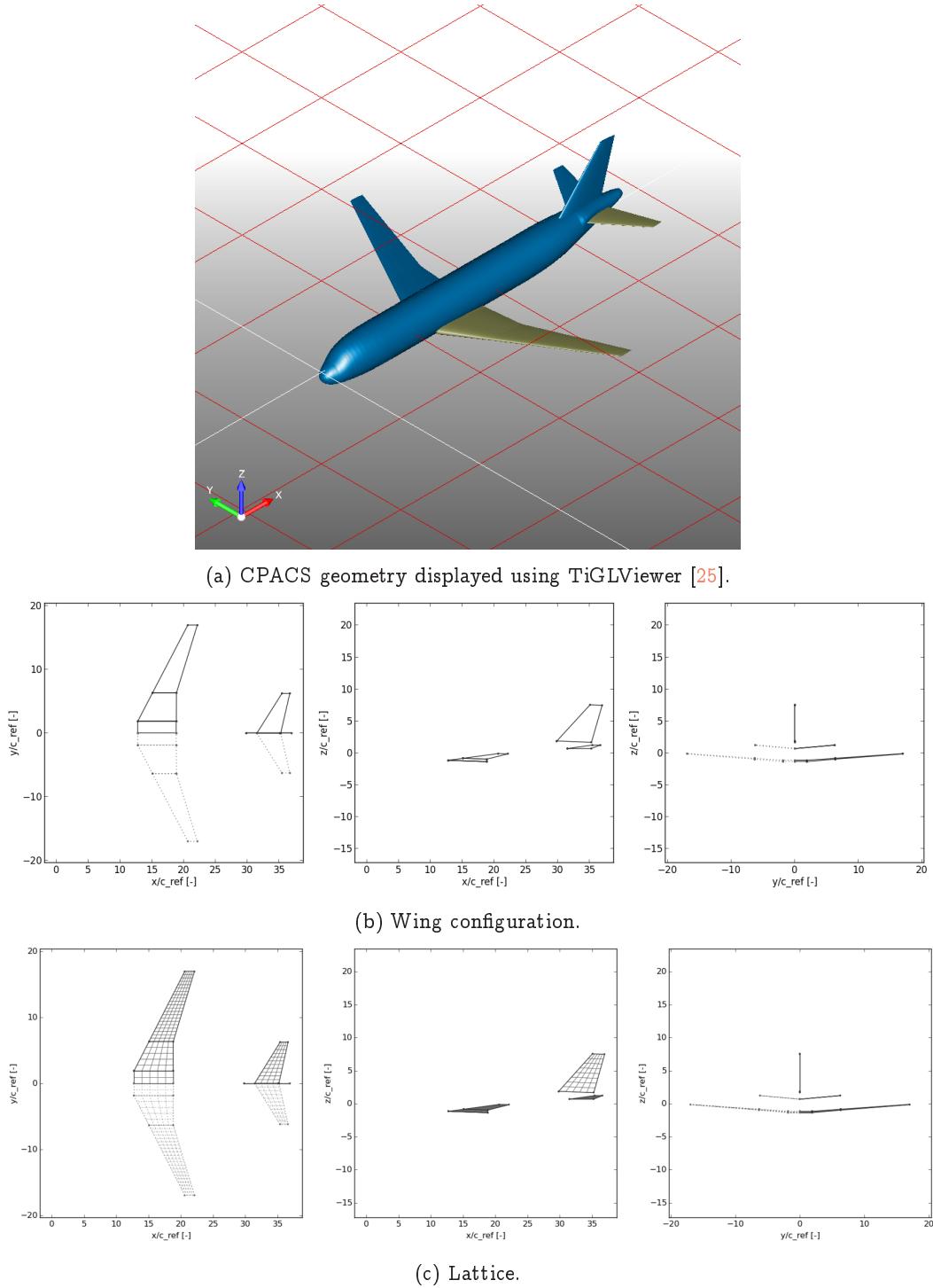
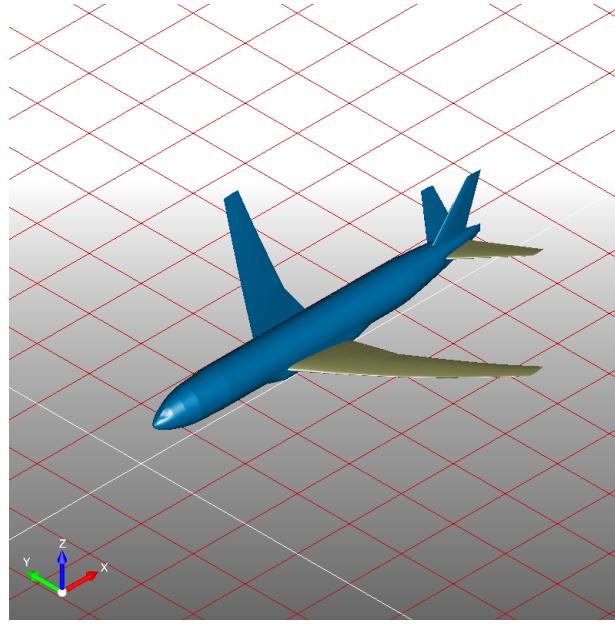
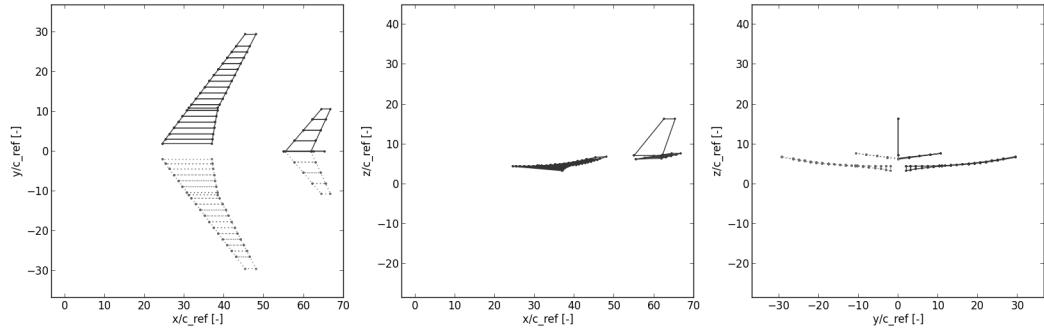


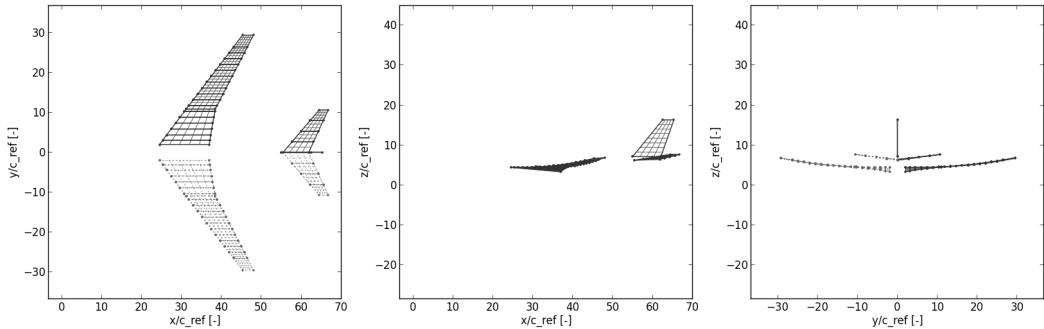
Figure 4.6: D150 model from the AGILE Hangar. This model is the most complete and serves as a reference for the ongoing development of CEASIOM.



(a) CPACS geometry displayed using TiGLViewer [25].



(b) Wing configuration.



(c) Lattice.

Figure 4.7: NASA CRM model from the AGILE Hangar. The geometry of the aircraft is obtained from a CAD model, leading to many segments of inconsistent width along the wing. The automatic geometry discretisation routine attempts to maintain panel aspect ratio as close as possible to 1.0 in each segment.

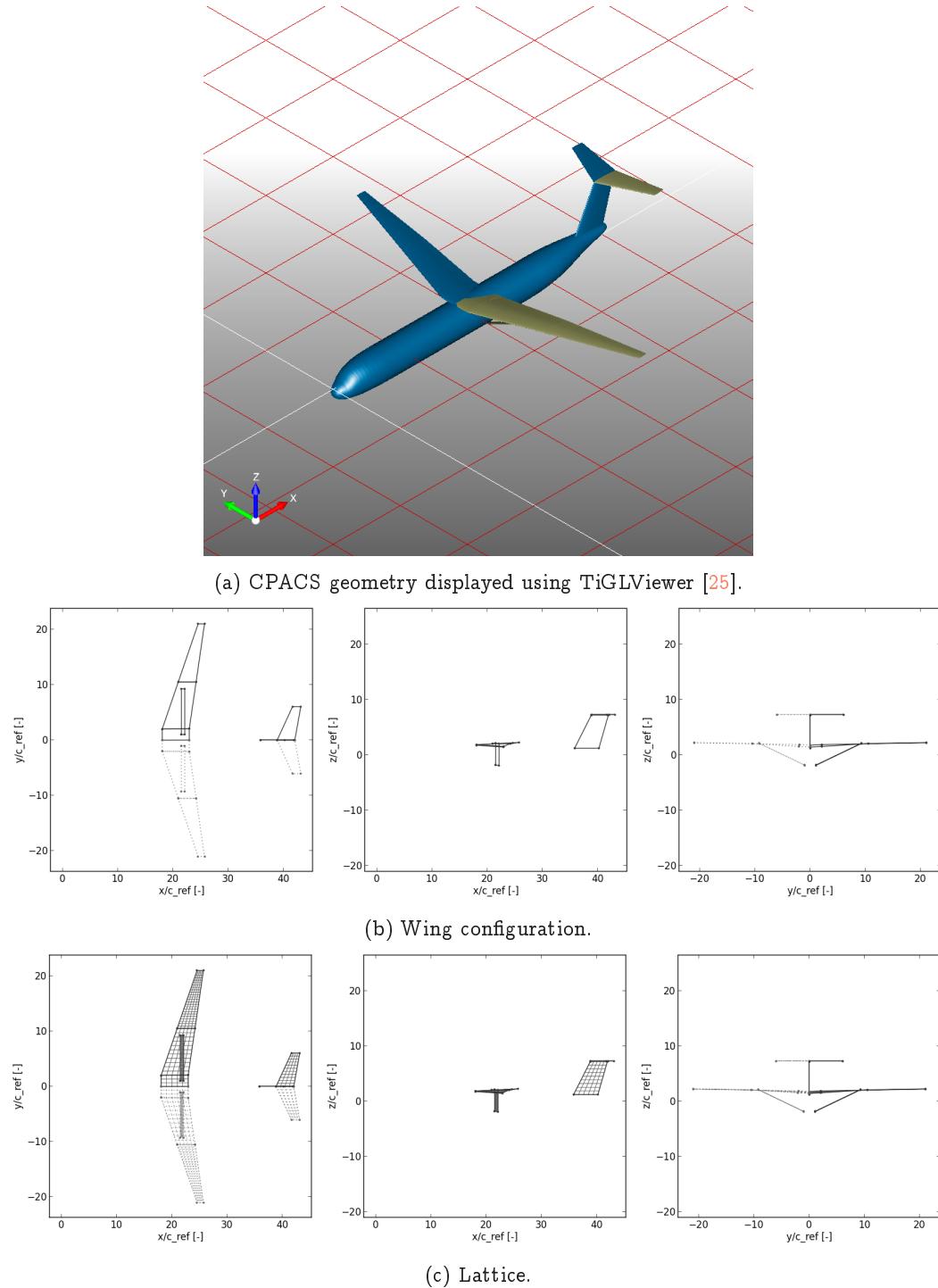


Figure 4.8: DLR Strut-Braced Wing model from the AGILE Hangar. While it bears some similarities with conventional aircraft configurations, this innovative design is studied for its potential to improve aircraft efficiency by relieving the structural loading on high-aspect ratio wings.

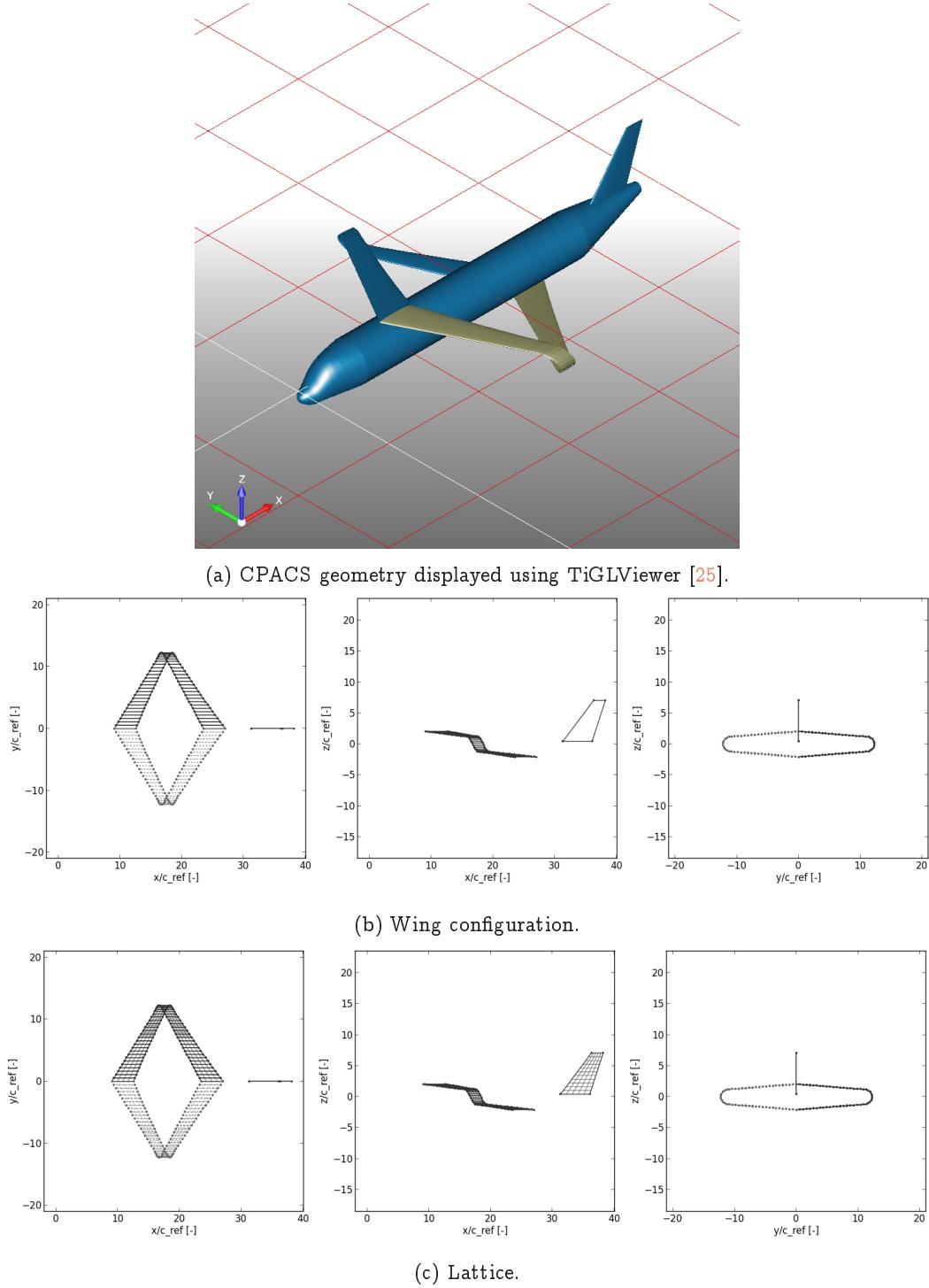


Figure 4.9: Box-winged aircraft model from the AGILE Hangar. The configuration is inspired by the work of Prandtl which shows that, under certain conditions, the joined-wing design minimises induced drag for given lift and span [26].

4.4 Performance Optimisation

Rather unsurprisingly, profiling shows that the program bottlenecks, in terms of execution time, at the calculation of the downwash factors: in a first, naive Python implementation, this process represented 95 – 99% of the total execution time. Because the cost of this procedure grows with $\mathcal{O}(n_{\text{tot}}^2)$, where n_{tot} is the total number of panels that constitute the discretised geometry, execution times quickly become unacceptable. Performance is an important aspect of the program if calculations are run repeatedly for a set of design points - at early stages of the aircraft design process, this is typically the case.

The above observation suggests that performance optimisation efforts should be directed towards accelerating the double (nested) iteration over each panel in the calculation of the matrix of downwash factors. This was done by re-implementing these routines in C++ using the Python/NumPy API. The execution times measured for the simple case of a plunging rectangular wing undergoing a plunging mode are compared in Figure 4.10.

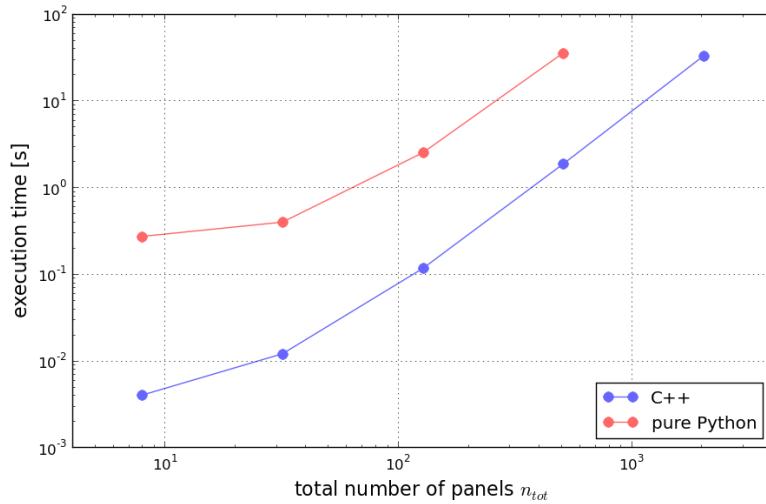


Figure 4.10: Execution times measured for the simple case of a plunging rectangular, using implementations of the Doublet-Lattice Method in pure Python and C++.

The intention of this benchmark is to obtain an estimate of the improvement obtained by porting the performance-critical part of the code to C++. The execution times for the Python version are not necessarily representative of the achievable performance: there exist a number of packages and libraries that greatly improve the performance of Python for scientific calculations. It was chosen to implement selected routines in C++ in order to minimise the reliance of the program on third-party libraries, while keeping the code reasonably readable. From Figure 4.10 it is seen that a reduction in the execution time of over an order of magnitude has been obtained; a significant improvement.

5 Results for Selected Test Cases

Although the effectiveness of the Doublet-Lattice Method has been demonstrated through research and its continued application in industry, the correct implementation of the method in the present program remains to be ascertained by comparing results obtained for selected test cases found in the literature. The application of the current program to simple problems is also a welcome opportunity to illustrate the significance of certain parameters, and to explore the limitations of the Doublet-Lattice Method.

For the validation process, we use a set of problems of increasing complexity, in order to gradually include a larger number of parameters.

5.1 Verification of the Kernel Function

Before performing any complete computation of the unsteady pressure distribution over test configurations, it is useful to verify the correct behaviour of the elementary unit employed in the calculations: the kernel function K . The mathematical formulation of the kernel function and its numerical evaluation are presented in detail in 3.2.

We consider here the case wherein the receiving panel r and the emitting panel s are one and the same. Rodden et al [18] have calculated the variation of the planar component of the kernel numerator along the quarter-chord line of the emitting panel, for multiple values of the panel aspect ratio AR , in order to determine its influence on the accuracy of the method. Figures 5.1a, 5.1d and 5.1g, taken from the original paper, show the real part of the planar kernel numerator, evaluated exactly and approximated using the parabolic interpolation $P_1(\bar{\eta})$ introduced in (3.34) and the enhanced quartic interpolation $P_1(\bar{\eta})$ of Appendix B.1. The steady component $K_1^{(0)}$ of the kernel, as seen in (3.30), is subtracted in all the calculations.

property	value
Mach number M_∞	0.8
reduced frequency k	1.0
reference length L_{ref}	0.5 units

Table 5.1: Operating conditions for the first test case, from Rodden et al [18].

The panel chord is fixed at $c = 1.0$ units, and the panel span is increased to vary the panel aspect ratio from $AR = 1.0$ to $AR = 5.0$. The problem is made non-dimensional using the panel half-chord as reference length, $L_{\text{ref}} = 0.5$ units. Operating conditions used for this first test case are summarised in Table 5.1.

The results obtained are shown in Figure 5.1. An exact correspondence is observed between the incremental planar kernel numerator of the present implementation and the values reported in the original paper. It should be noted that in the original results, the evolution of the kernel function is plotted against the y -coordinate, normalised by the panel span, with $y = 0$ at the inboard edge of the panel. For the current results, it was preferred to use a similar measure based on the coordinate $\bar{\eta}$ instead, in the interest of consistency. The definition of $\bar{\eta}$ may be found in Section 3.3 (see also Figure 0.1).

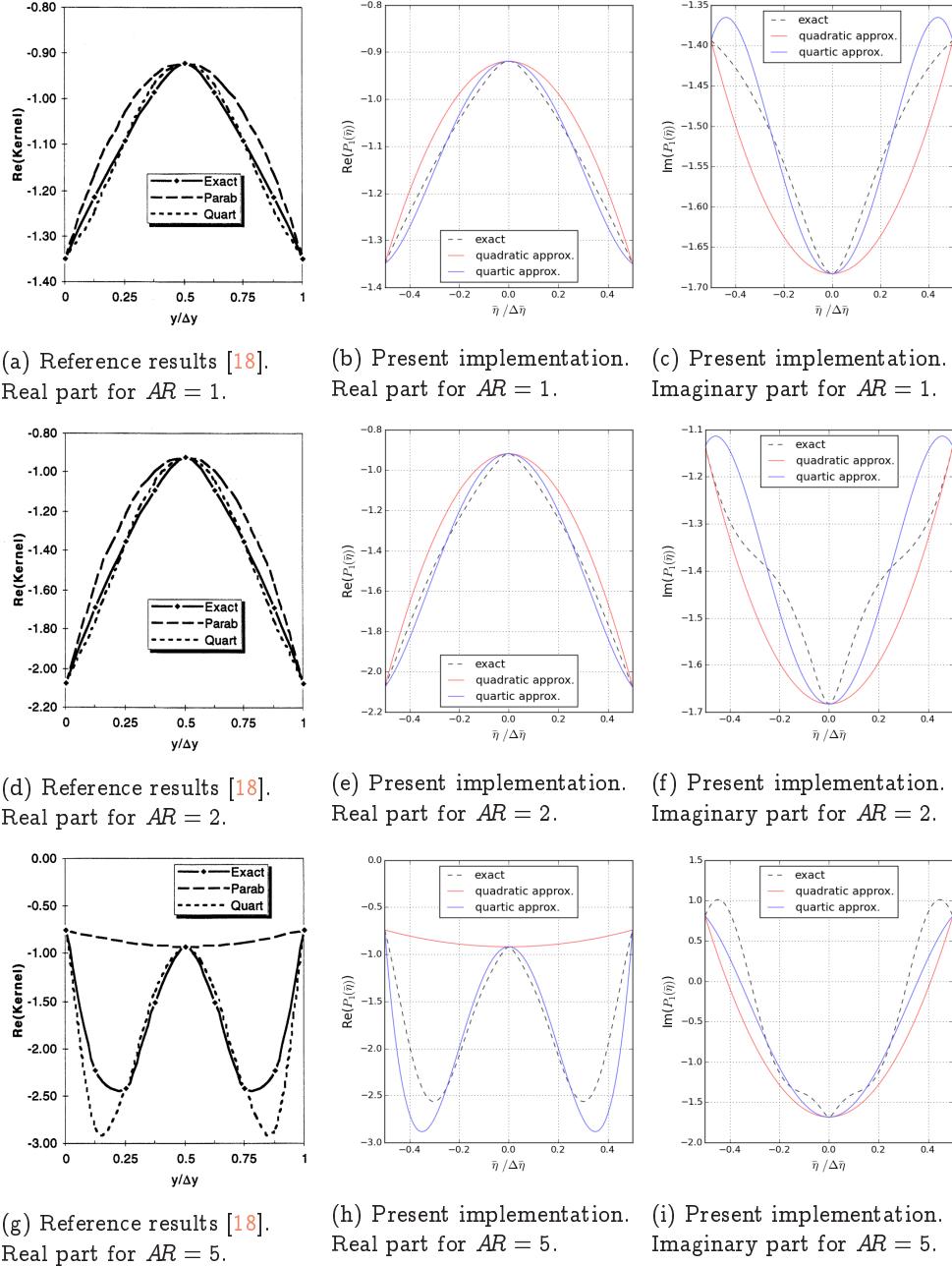


Figure 5.1: Variation of the incremental planar kernel numerator along the doublet line of the panel. The kernel function is computed exactly and compared to the parabolic and quartic approximations. The panel chord is fixed at $c = 1.0$, and the aspect ratio AR is varied by increasing panel span. Results were obtained using $M_\infty = 0.8$ and $k = 1.0$, with reference length $L_{ref} = 0.5$.

The imaginary component is not studied in the report used as reference for the first test case. In order to complete the verification of the current implementation, a second case is considered, taken from a convergence study of the enhanced Doublet-Lattice Method by the same authors, Rodden et al [19]. The operating conditions are shown in Table 5.2.

property	value
Mach number M_∞	0.8
reduced frequency k	2.0
reference length L_{ref}	3.5 units

Table 5.2: Operating conditions for the second test case, from Rodden et al [19].

Results obtained for a panel of chord $c = 0.2$ units and aspect ratio $AR = 10$, with reference length $L_{\text{ref}} = 3.5$, are presented in Figure 5.3. Once more, a good match is found between the original report and the current implementation, if not for a slight discrepancy in the imaginary parts, toward the outer borders of the panel. This is likely due to the low number of points at which the kernel and its approximations are evaluated in the reference paper; indeed, it is easily verified in Figure 5.2 that 9 points are not sufficient to capture the small oscillation of the kernel function observed at the outer borders of the panel.

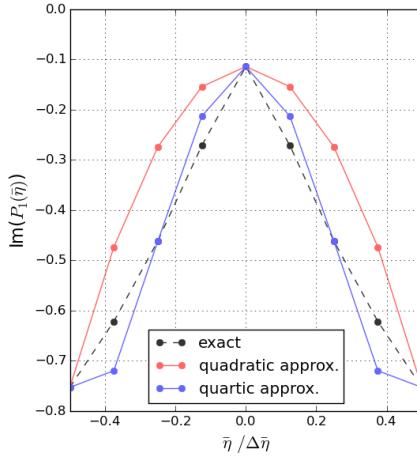


Figure 5.2: Span-wise variation of the imaginary component of the planar incremental kernel numerator, $AR = 10$. The function and its approximations are evaluated at 9 points only.

Beyond providing partial verification of the present program, these results illustrate the improvement a quartic approximation of the kernel numerator constitutes over its quadratic counterpart. Since the kernel is integrated, the error should be measured with respect to the area under the curves. In this sense, errors in the quartic approximation tend to self-cancel. This improvement is seen consistently for the imaginary part of the kernel numerator, and is most flagrant for the real part in Figures 5.1g and 5.1h. The convergence of the Doublet-Lattice Method is further commented in Section 5.3.

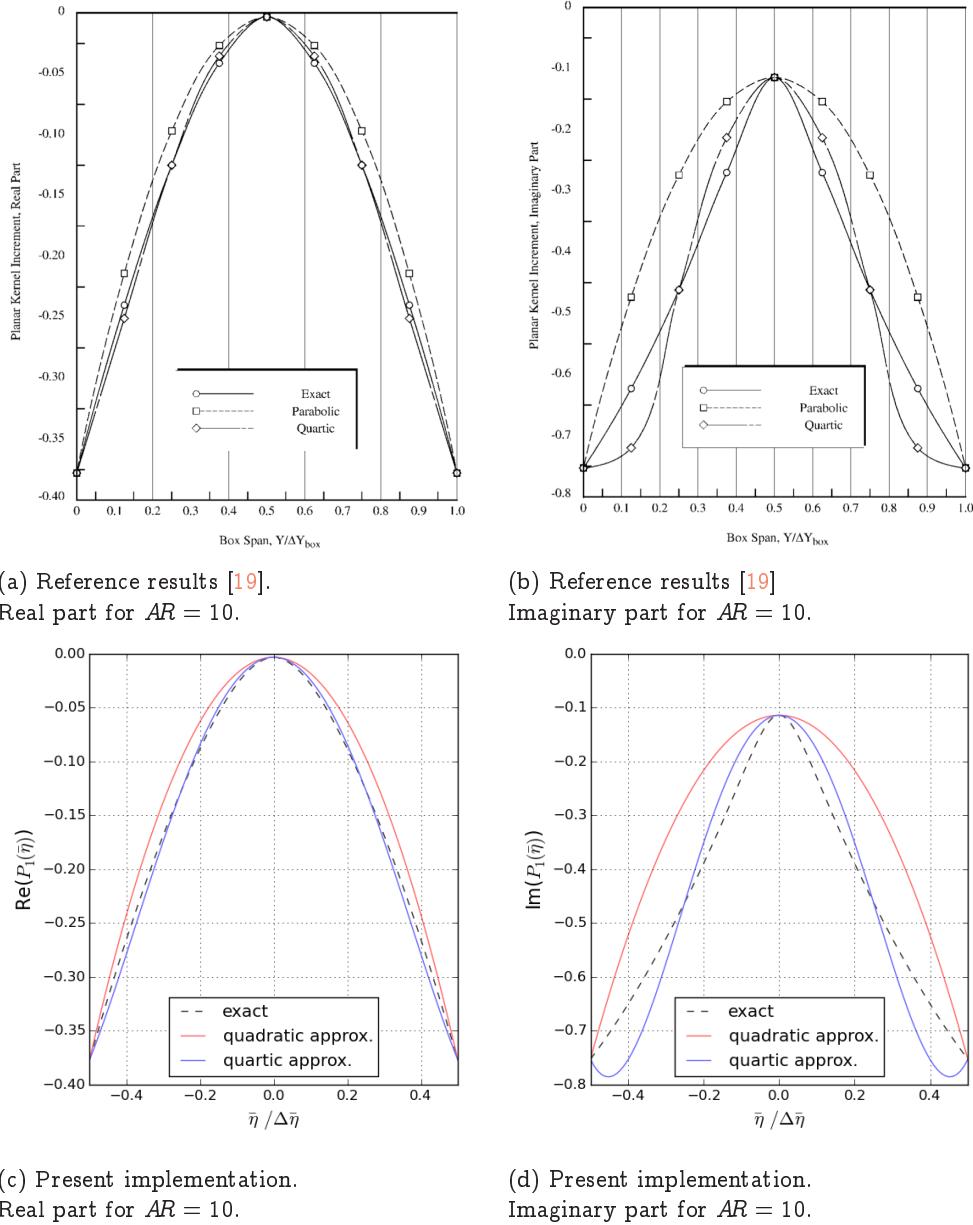


Figure 5.3: Variation of the incremental planar kernel numerator along the doublet line of the panel. The kernel function is computed exactly and compared to the parabolic and quartic approximations. The panel chord is fixed at $c = 0.2$, and the aspect ratio $AR = 10$. Results were obtained using $M_\infty = 0.8$ and $k = 2.0$, with reference length $L_{ref} = 3.5$.

In the preceding considerations, only the planar part of the approximated oscillatory kernel function has been considered. Recall from (3.31) that for a receiving panel r and a sending panel s , the unsteady downwash factor D'_{rs} was expressed as the summation of a planar component $D^{(1)}_{rs}$ and a non-planar component $D^{(2)}_{rs}$. This formulation was originally introduced by Rodden et al [9] in order to improve the behaviour of the Doublet-Lattice Method for configurations containing nearly-coplanar components.

In the same article, the authors verify the continuous and regular behaviour of the kernel function as the normal gap between an emitting panel and a receiving panel in its wake approaches zero.

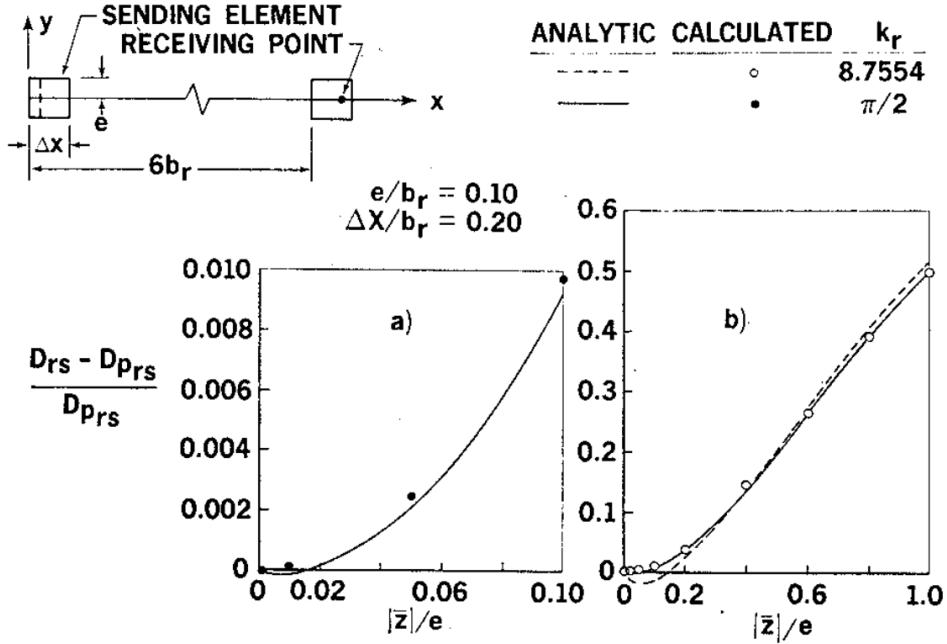
We consider two distinct panels with chords $\Delta x = 0.2L_{\text{ref}}$ and half-widths $e = 0.1L_{\text{ref}}$. The receiving panel lies directly in the wake of the emitting panel (at a distance of six times the reference length). The receiving panel is given a slight vertical offset in the direction of the normal vector of the emitting panel. This gap is made to vary in the range $0.0 \leq |\bar{z}/e| \leq 1.0$, so that $\bar{y}^2 + \bar{z}^2 - e^2 \leq 0$.

The original results, seen in Figure 5.4a along with the configuration of the panels used for the test case, confirm that despite the singular nature of the integral H of (3.36), the contribution of the non-planar part smoothly goes to zero as the gap between the panels is decreased. It should be noted that the term $D^{(p)}_{rs}$ is the planar component calculated for the case where $|\bar{z}/e| = 0$.

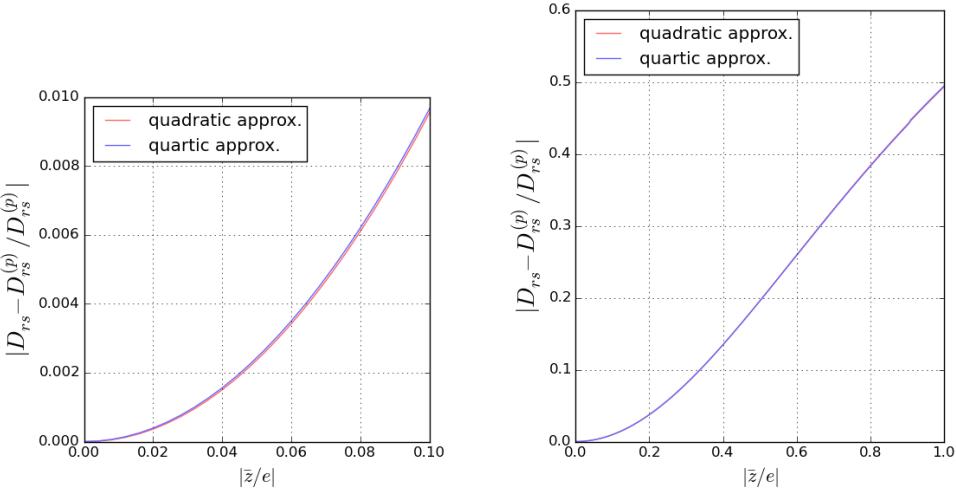
property	value
Mach number M_∞	0.0
reduced frequency k	$\pi/2$
reference length L_{ref}	1.0 units

Table 5.3: Operating conditions for the second test case, from Rodden et al [19].

This experiment is reproduced using the current program, in an effort to ascertain the correct behaviour of the non-planar component $D^{(2)}_{rs}$. From the results obtained, shown in Figures 5.4b and 5.4c, it is seen that the current implementation reproduces the reference result quite accurately. No notable difference is observed between the quadratic and the quartic approximations.



(a) Reference results [9]. Non-planar part of the approximated downwash factor, compared to its exact value (evaluated analytically, see reference).



(b) Present implementation ($k = \pi/2$).
Non-planar part of downwash factor.

(c) Present implementation ($k = \pi/2$).
Non-planar part of downwash factor (detail).

Figure 5.4: Behaviour of the non-planar component of the downwash factor $D_{rs}^{(2)}$ for a receiving point lying directly in the wake of the emitting panel. The panels are separated vertically by a small distance \bar{z} . In 5.4a, configuration of the panels is shown along with the reference results [9], compared to values obtained analytically. In 5.4c and 5.4b, both the parabolic and quartic approximations are shown. The panel chords are $c = 0.2$, and the half-widths $e = 0.1$. Results were reproduced using $M_\infty = 0.0$ and $k = \pi/2$, with reference length $L_{ref} = 1.0$.

5.2 Planar Rectangular Wing

We begin by considering the simplest possible geometry: a planar, unswept and untapered (rectangular) wing of aspect ratio $AR = 2.0$ and non-dimensional area $A_{\text{ref}} = 4.0$, symmetric about the x, z -plane. The aircraft undergoes a rigid-body plunging mode of unit amplitude, with reduced frequency $k = 1.0$, at Mach $M_{\infty} = 0.5$.

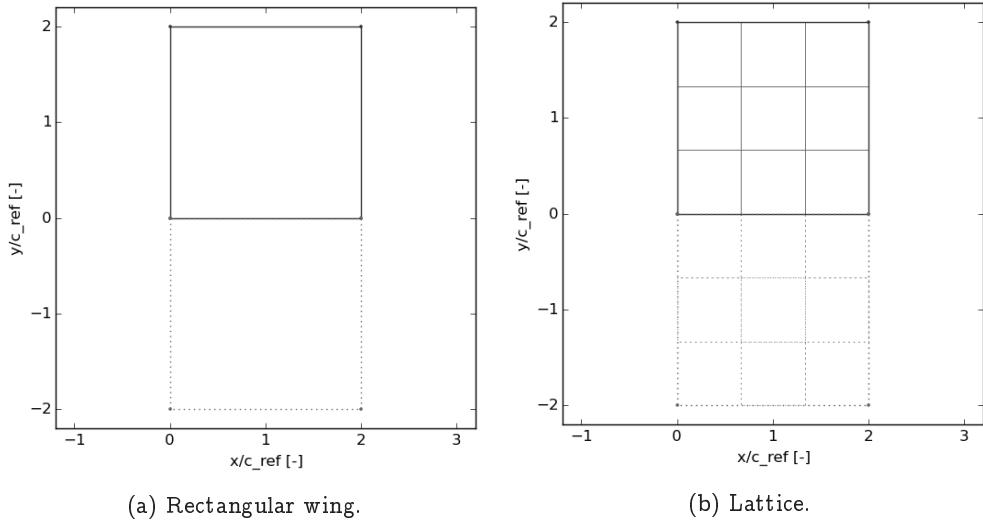


Figure 5.5: Blairstream 2100 Attack Fighter [27] wing configuration, projected in the x, y -plane. Dotted lines indicate parts of the geometry that are not modeled. Coordinates are normalised by the reference length $L_{\text{ref}} = 6.0$.

This elementary configuration is used by Blair [27] to demonstrate a simple example code to illustrate their overview of the mathematics leading to the Doublet-Lattice Method. All geometric and operational parameters are kept identical to the ones used in the reference paper; these may be found in Tables 5.4 and 5.5.

property	value
offset (x)	0.0 m
offset (y)	0.0 m
offset (z)	0.0 m
chord (inboard)	12.0 m
chord (outboard)	12.0 m
span	12.0 m
sweep	0.0 deg
dihedral	0.0 deg
symmetry	x, z -plane
number of panels (chord)	3
number of panels (span)	3

Table 5.4: Geometric properties of the Blairstream 2100 Attack Fighter [27] wing.

The problem is made non-dimensional with reference length $L_{\text{ref}} = 6.0$, i.e. the half-chord of the wing. Only the right half of the wing is modeled, taking advantage of symmetry. The geometry is discretised by three panels chord-wise and three panels span-wise. Figures 5.5 and 5.6 show the wing geometry in terms of non-dimensional coordinates, before and after discretisation.

property	value
Mach number M_∞	0.5
reduced frequency k	1.0
reference length L_{ref}	6.0 m

Table 5.5: Operating conditions for the Blairstaff 2100 Attack Fighter [27] wing.

Table 5.6 compares the computed distribution of the complex pressure coefficient $\Delta \bar{c}_p$, as defined in (3.25), to those obtained in the original report at each control point of the lattice. The calculation is performed using both types of panels (see Section 3.3 and Appendix B). Panels are numbered in chord- then span-wise order, as shown in Figure 5.6. The complex lift coefficient of the wing is also calculated, using (3.51).

The results obtained with the present implementation of the Doublet-Lattice Method are in very good agreement with those obtained in the reference paper. Pressure coefficients obtained using the higher-order approximation to the kernel function numerator differ noticeably from these results. This discrepancy is likely due to the poor discretisation of the wing geometry. However, the values obtained are in perfect agreement with those reported by Da Ronch and Cavagna [28], also computed using a quartic approximation.

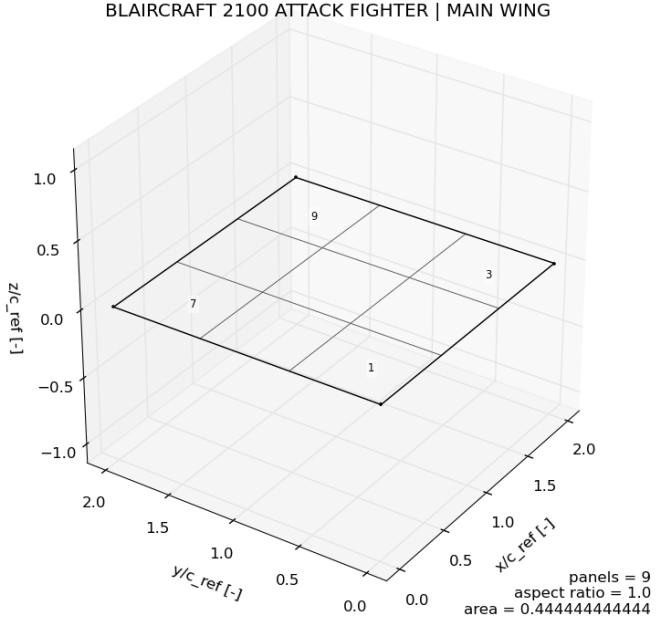


Figure 5.6: Blairstraft 2100 Attack Fighter wing (right half), discretised. Note the numbering of the panels. Average values for panel aspect ratio and area are shown in the bottom-right corner. Coordinates are normalised by the reference length $L_{\text{ref}} = 6.0$.

panel	$\Delta \bar{c}_p$ Blair 1992 [27]	$\Delta \bar{c}_p$ present (quadratic)	$\Delta \bar{c}_p$ present (quartic)
1	$-5.4900e-01 + i6.2682e+00$	$-5.4902e-01 + i6.2682e+00$	$-5.6098e-01 + i5.7936e+00$
2	$-3.8862e+00 + i2.4495e+00$	$-3.8862e+00 + i2.4495e+00$	$-3.5519e+00 + i2.3119e+00$
3	$-3.8736e+00 + i1.1745e+00$	$-3.8736e+00 + i1.1745e+00$	$-3.5194e+00 + i1.0961e+00$
4	$-5.9144e-01 + i5.8092e+00$	$-5.9146e-01 + i5.8092e+00$	$-5.9907e-01 + i5.3863e+00$
5	$-3.6405e+00 + i2.1530e+00$	$-3.6405e+00 + i2.1530e+00$	$-3.3429e+00 + i2.0434e+00$
6	$-3.6234e+00 + i1.0281e+00$	$-3.6234e+00 + i1.0281e+00$	$-3.3065e+00 + i9.6176e-01$
7	$-5.8286e-01 + i4.5474e+00$	$-5.8287e-01 + i4.5474e+00$	$-5.8574e-01 + i4.2488e+00$
8	$-2.8983e+00 + i1.4663e+00$	$-2.8983e+00 + i1.4663e+00$	$-2.6908e+00 + i1.4079e+00$
9	$-2.8893e+00 + i7.1186e-01$	$-2.8893e+00 + i7.1186e-01$	$-2.6648e+00 + i6.6740e-01$
<hr/>			
	\bar{C}_L Blair 1992 [27]	\bar{C}_L present (quadratic)	\bar{C}_L present (quartic)
	$-2.5038e+00 + i2.8453e+00$	$-2.5038e+00 + i2.8453e+00$	$+2.3136e+00 + i2.6575e+00$
	magnitude = $3.7901e+00$	magnitude = $3.7901e+00$	magnitude = $3.5234e+00$
	phase = 131.3471 deg	phase = 131.3471 deg	phase = 131.0423 deg

Table 5.6: Complex pressure coefficients computed at control points, as provided in [27] and as computed by the program using quadratic and quartic panels. i denotes the imaginary unit. The complex lift coefficient is also calculated, using non-dimensional reference area $A_{\text{ref}} = 4.0$.

5.3 Influence of Panel Aspect Ratio

Rodden et al [19] continue their study of the influence of the panel aspect ratio on the accuracy of the Doublet-Lattice Method by comparing the lift coefficients of rectangular wings pitching about their mid-chord.

The panel aspect ratios are influenced in two ways: first, by increasing the number of span-wise strips for a fixed number of chord-wise divisions. In a second series of calculations, the number of panels is kept constant as the wing aspect ratio is varied.

property	value
Mach number M_∞	0.8
reduced frequency k	0.1, 0.5, 1.0, 2.0
reference length L_{ref}	0.5 m

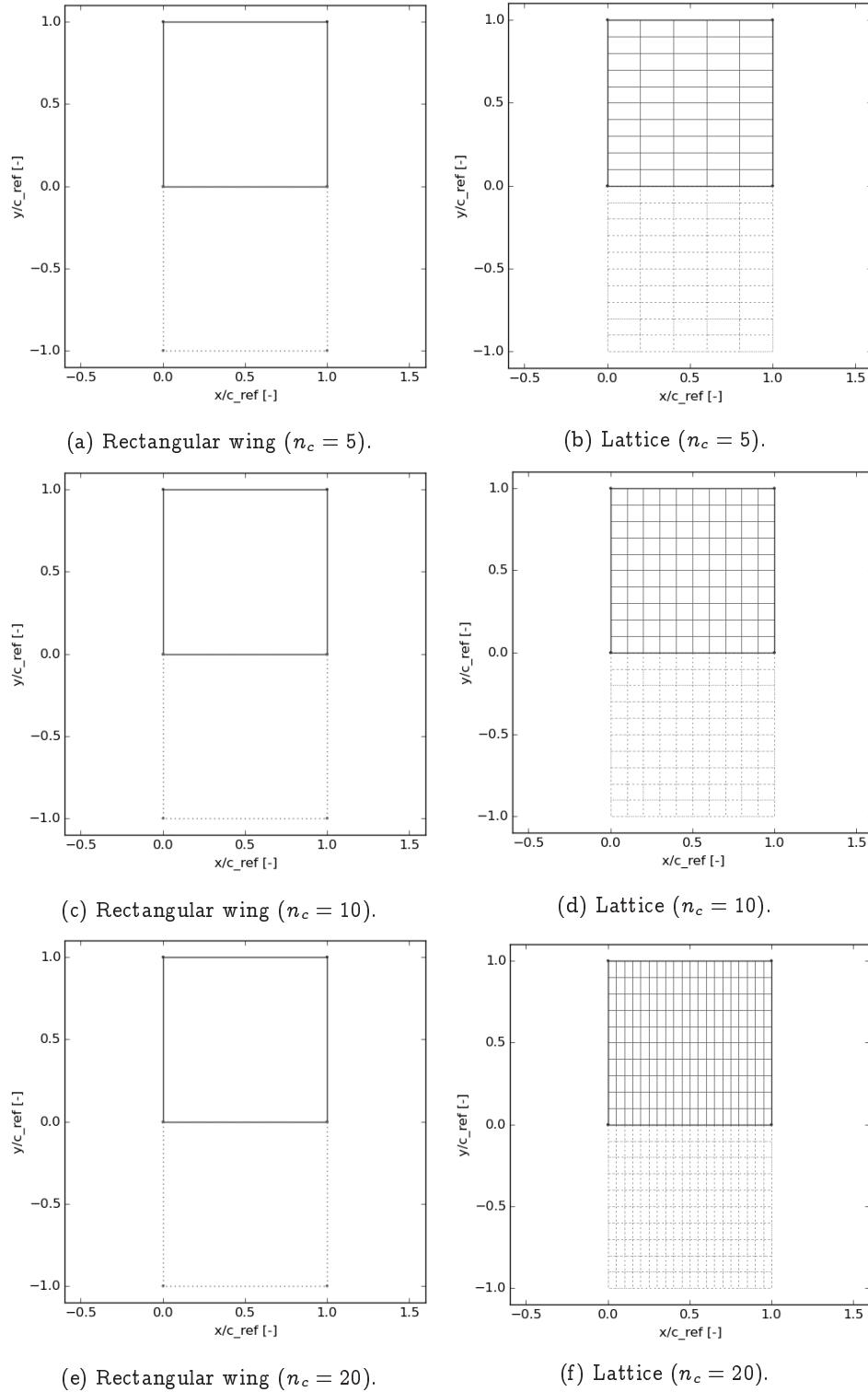
Table 5.7: Operating conditions for the convergence studies by Rodden et al [19].

Number of Span-wise Panels We consider a planar, unswept and untapered rectangular wing of unit chord and aspect ratio $AR = 2.0$. The wing is subjected to a pitching mode $h(x, y, z, t) = x_{1/2} - x$ about its mid-chord $x_{1/2}$.

Since the problem is symmetric across the x, z -plane, we consider only the right (starboard) half of the wing. The number of span-wise strips is kept constant at $n_s = 10$, and the number of chord-wise divisions is increased from $n_c = 5$ to $n_c = 100$ in order to vary the panel aspect ratio from 0.5 to 10.0. The unsteady pressure distribution over the wing is calculated for multiple values of the reduced frequency, ranging from $k = 0.1$ to $k = 2.0$, with Mach number $M_\infty = 0.8$. Operating conditions are summarised in Table 5.7, and geometric parameters for the wing are given in Table 5.8. The rectangular wing and its discretisations are shown in Figure 5.7.

property	value
offset (x)	0.0 m
offset (y)	0.0 m
offset (z)	0.0 m
chord (inboard)	1.0 m
chord (outboard)	1.0 m
span	1.0 m
sweep	0.0 deg
dihedral	0.0 deg
symmetry	x, z -plane
number of panels (chord)	5, 10, 20, 50, 100
number of panels (span)	10

Table 5.8: Geometric properties of the rectangular wing used in the convergence studies by Rodden et al [19]. In a first series of calculations, the panel aspect ratio is varied by increasing the number of chord-wise divisions.



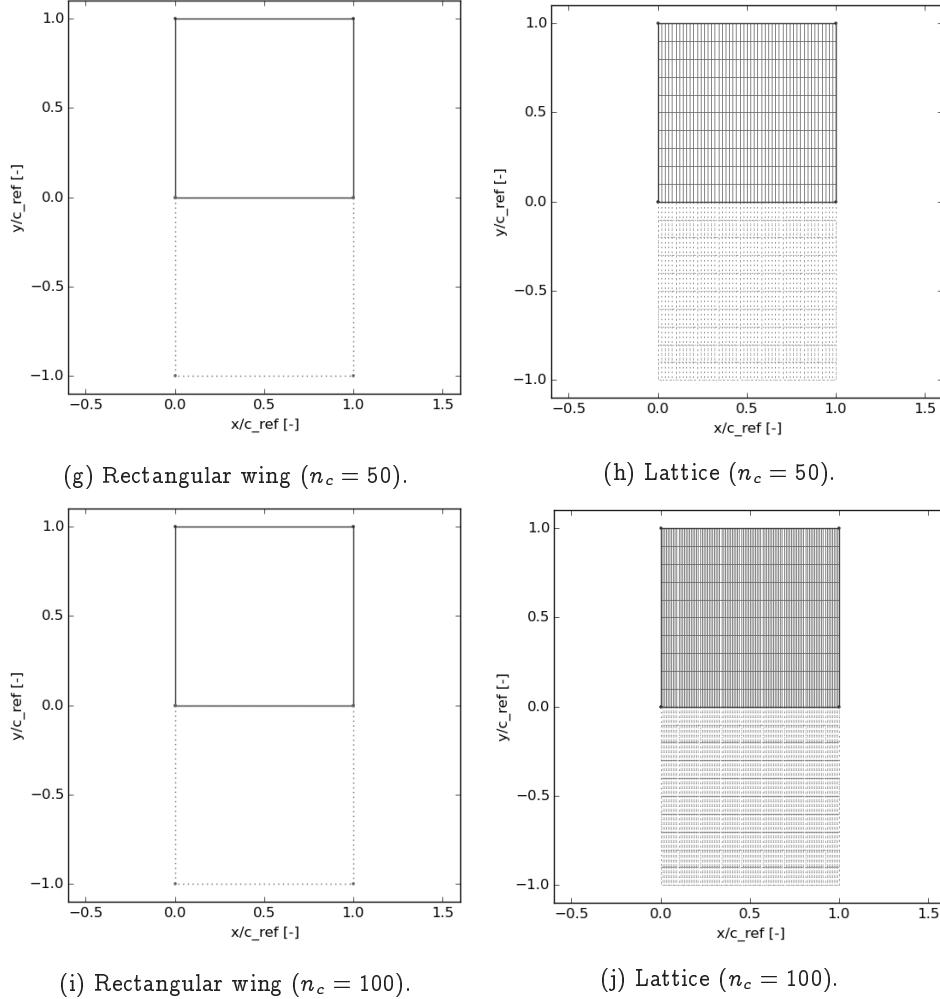


Figure 5.7: Rectangular wing configuration used in the convergence studies of Rodden et al [19], projected in the x , y -plane. Dotted lines indicate parts of the geometry that are not modeled. The number of span-wise strips is kept constant at $n_s = 10$, and the panel aspect ratios are varied by changing the number of chord-wise divisions $n_c = 5, 10, 20, 50, 100$. Coordinates are normalised by the reference length $L_{\text{ref}} = 0.5$ m.

<i>k</i>	<i>n_c</i>	panel <i>AR</i>	quadratic (NK5A) [19]	quadratic (NeoCASS) [28]	quadratic (present)
0.1	5	0.5	2.968 + <i>i</i> 0.3626	2.9691 + <i>i</i> 0.3634	2.9691 + <i>i</i> 0.3634
0.1	10	1.0	2.975 + <i>i</i> 0.3653	2.9804 + <i>i</i> 0.3699	2.9804 + <i>i</i> 0.3699
0.1	20	2.0	2.977 + <i>i</i> 0.3657	2.9912 + <i>i</i> 0.3792	2.9912 + <i>i</i> 0.3792
0.1	50	5.0	2.978 + <i>i</i> 0.3658	3.0017 + <i>i</i> 0.3912	3.0017 + <i>i</i> 0.3912
0.1	100	10.0	2.978 + <i>i</i> 0.3658	-	3.0052 + <i>i</i> 0.3958
0.5	5	0.5	3.638 + <i>i</i> 1.739	3.6422 + <i>i</i> 1.7418	3.6422 + <i>i</i> 1.7418
0.5	10	1.0	3.810 + <i>i</i> 1.731	3.8359 + <i>i</i> 1.7494	3.8359 + <i>i</i> 1.7494
0.5	20	2.0	3.870 + <i>i</i> 1.724	3.9453 + <i>i</i> 1.7793	3.9453 + <i>i</i> 1.7793
0.5	50	5.0	3.885 + <i>i</i> 1.723	4.0178 + <i>i</i> 1.8268	4.0178 + <i>i</i> 1.8268
0.5	100	10.0	3.887 + <i>i</i> 1.722	-	4.0393 + <i>i</i> 1.8464
1.0	5	0.5	4.492 + <i>i</i> 1.823	4.4966 + <i>i</i> 1.8207	4.4966 + <i>i</i> 1.8207
1.0	10	1.0	4.820 + <i>i</i> 1.479	4.8452 + <i>i</i> 1.4669	4.8452 + <i>i</i> 1.4669
1.0	20	2.0	4.920 + <i>i</i> 1.338	4.9784 + <i>i</i> 1.3133	4.9784 + <i>i</i> 1.3133
1.0	50	5.0	4.930 + <i>i</i> 1.303	5.0568 + <i>i</i> 1.2843	5.0568 + <i>i</i> 1.2843
1.0	100	10.0	4.932 + <i>i</i> 1.300	-	5.0838 + <i>i</i> 1.2899
2.0	5	0.5	4.652 + <i>i</i> 2.380	4.6568 + <i>i</i> 2.3775	4.6568 + <i>i</i> 2.3775
2.0	10	1.0	5.461 + <i>i</i> 1.729	5.4860 + <i>i</i> 1.7126	5.4860 + <i>i</i> 1.7126
2.0	20	2.0	5.681 + <i>i</i> 1.449	5.7653 + <i>i</i> 1.4133	5.7653 + <i>i</i> 1.4133
2.0	50	5.0	5.730 + <i>i</i> 1.378	5.9087 + <i>i</i> 1.3324	5.9087 + <i>i</i> 1.3324
2.0	100	10.0	5.735 + <i>i</i> 1.371	-	5.9539 + <i>i</i> 1.3232
<i>k</i>	<i>n_c</i>	panel <i>AR</i>	quartic (NK5Q) [19]	quartic (NeoCASS) [28]	quartic (present)
0.1	5	0.5	2.968 + <i>i</i> 0.3626	2.9674 + <i>i</i> 0.3656	2.9674 + <i>i</i> 0.3656
0.1	10	1.0	2.968 + <i>i</i> 0.3565	2.9751 + <i>i</i> 0.3681	2.9751 + <i>i</i> 0.3681
0.1	20	2.0	2.971 + <i>i</i> 0.3563	2.9789 + <i>i</i> 0.3688	2.9789 + <i>i</i> 0.3688
0.1	50	5.0	2.972 + <i>i</i> 0.3560	2.9853 + <i>i</i> 0.3739	2.9853 + <i>i</i> 0.3739
0.1	100	10.0	2.972 + <i>i</i> 0.3560	-	2.9893 + <i>i</i> 0.3785
0.5	5	0.5	3.638 + <i>i</i> 1.739	3.6204 + <i>i</i> 1.7605	3.6204 + <i>i</i> 1.7605
0.5	10	1.0	3.770 + <i>i</i> 1.724	3.8054 + <i>i</i> 1.7539	3.8054 + <i>i</i> 1.7539
0.5	20	2.0	3.859 + <i>i</i> 1.712	3.9004 + <i>i</i> 1.7467	3.9004 + <i>i</i> 1.7467
0.5	50	5.0	3.898 + <i>i</i> 1.706	3.9695 + <i>i</i> 1.7635	3.9695 + <i>i</i> 1.7635
0.5	100	10.0	3.902 + <i>i</i> 1.705	-	3.9965 + <i>i</i> 1.7824
1.0	5	0.5	4.492 + <i>i</i> 1.823	4.4637 + <i>i</i> 1.8733	4.4637 + <i>i</i> 1.8733
1.0	10	1.0	4.768 + <i>i</i> 1.528	4.8284 + <i>i</i> 1.5027	4.8284 + <i>i</i> 1.5027
1.0	20	2.0	4.901 + <i>i</i> 1.313	4.9635 + <i>i</i> 1.2760	4.9635 + <i>i</i> 1.2760
1.0	50	5.0	4.948 + <i>i</i> 1.212	5.0347 + <i>i</i> 1.1655	5.0347 + <i>i</i> 1.1655
1.0	100	10.0	4.953 + <i>i</i> 1.200	-	5.0623 + <i>i</i> 1.1556
2.0	5	0.5	4.652 + <i>i</i> 2.380	4.5315 + <i>i</i> 2.4436	4.5314 + <i>i</i> 2.4436
2.0	10	1.0	5.396 + <i>i</i> 1.814	5.4168 + <i>i</i> 1.7774	5.4168 + <i>i</i> 1.7774
2.0	20	2.0	5.720 + <i>i</i> 1.393	5.7389 + <i>i</i> 1.3474	5.7389 + <i>i</i> 1.3474
2.0	50	5.0	5.840 + <i>i</i> 1.194	5.8964 + <i>i</i> 1.1393	5.8964 + <i>i</i> 1.1393
2.0	100	10.0	5.854 + <i>i</i> 1.170	-	5.9483 + <i>i</i> 1.1124

Table 5.9: Complex lift coefficient \bar{C}_L for rectangular wing pitching about its mid-chord, compared with results of Rodden et al [19] and Da Ronch and Cavagna [28]. The number of chord-wise divisons n_c is varied. Values shown in this table were obtained using both the quadratic and quartic approximations to the kernel numerator. The reference area $A_{\text{ref}} = 4.0$ used to calculate the lift coefficient is the surface area of the wing in terms of the non-dimensional coordinates, rescaled by the reference length $L_{\text{ref}} = 0.5$ m. *i* denotes the imaginary unit.

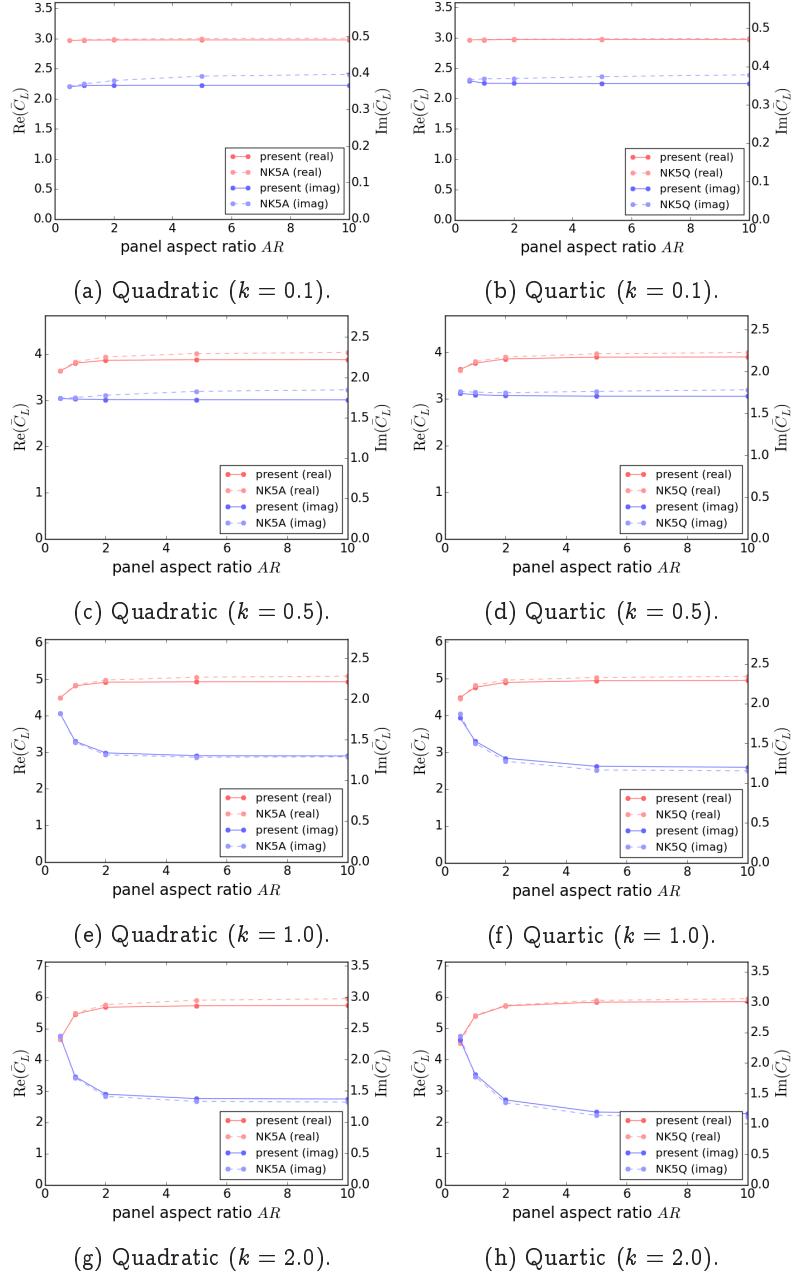


Figure 5.8: Complex lift coefficient \bar{C}_L for rectangular wings pitching about their mid-chord, in comparison with results of Rodden et al [19]. Values are plotted against panel aspect ratio, varied by increasing the number of chord-wise divisions n_c . Results were obtained using both the quadratic (left column) and quartic (right column) approximations to the kernel numerator.

The resulting lift coefficients \bar{C}_L are presented in Table 5.9, and compared with original results by Rodden et al [19] and by Da Ronch and Cavagna [28], for the quadratic and quartic approximations. Overall, the values are in reasonably good agreement; the error measured with respect to the former reference does not exceed 10%, and an exact correspondence is observed with those obtained by the latter.

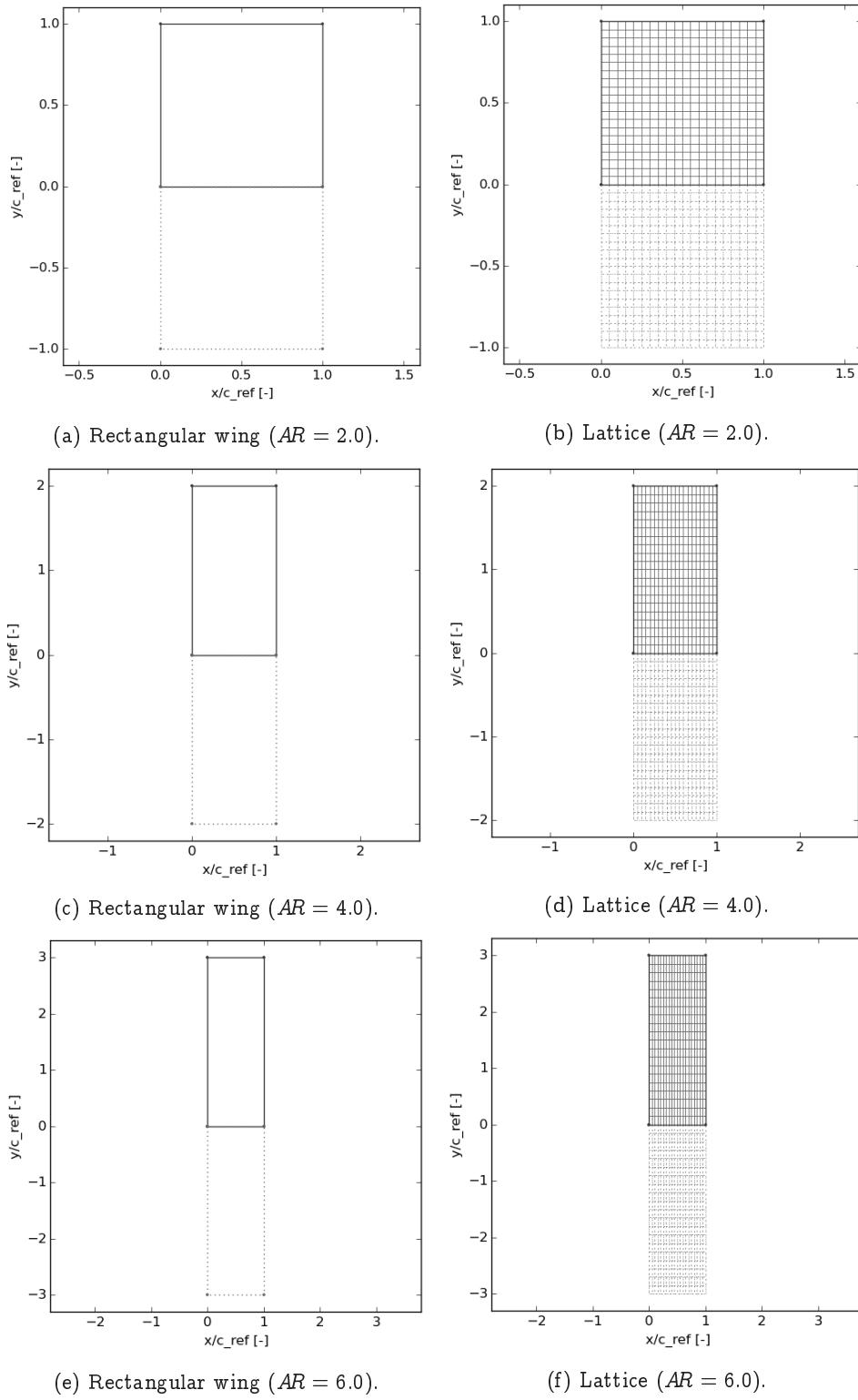
Wing Aspect Ratio Now, the panel aspect ratio is controlled by varying the wing geometry. Using the same wing used for the previous study as a reference configuration, the span is varied to take on the values $s = 1.0, 2.0, 3.0, 5.0$ and 10.0 m . The resulting geometries before and after discretisation are shown in Figure 5.9. The wings are subjected to the same pitching mode that was applied to the previous case. Aerodynamic operating conditions are given in Table 5.7 and geometric parameters are found in Table 5.10. The reference length used to non-dimensionalise the results is again equal to the wing half-chord, $L_{\text{ref}} = 0.5\text{ m}$.

property	value
offset (x)	0.0 m
offset (y)	0.0 m
offset (z)	0.0 m
chord (inboard)	1.0 m
chord (outboard)	1.0 m
span	1.0, 2.0, 3.0, 5.0, 10.0 m
sweep	0.0 deg
dihedral	0.0 deg
symmetry	x, z -plane
number of panels (chord)	20
number of panels (span)	20

Table 5.10: Geometric properties of the rectangular wing used in the convergence studies by Rodden et al [19]. In a second study, the panel aspect ratio is influence by varying the wing span.

The lift coefficient \bar{C}_L calculated for each combination of wing geometry and reduced frequency is presented in Table 5.9, along with the reference results obtained by Rodden et al [19], for both the quadratic and the quartic approximation. Once more, the values match relatively well. Furthermore, we observe again an exact correspondence with the results obtained by Da Ronch and Cavagna [28].

The discrepancies, of up to 10%, observed between the results generated using the present implementation of the Doublet-Lattice Method and those calculated using the programs NK5A (which uses a quadratic kernel numerator approximation) and NK5Q (which uses a quartic kernel numerator approximation) of the reference report are perhaps explained by the likely more accurate estimation of the steady downwash factors by the latter. In fact, as was mentioned in Chapter 4, in its current version the present program does not perform the calculation of the steady component $D_{rs}^{(0)}$ of the downwash factor D_{rs} exactly using Hedman's Vortex-Lattice Method (see (3.28)). However, this would not explain the exact agreement with the results obtained by Da Ronch and Cavagna [28] using NeoCASS [29].



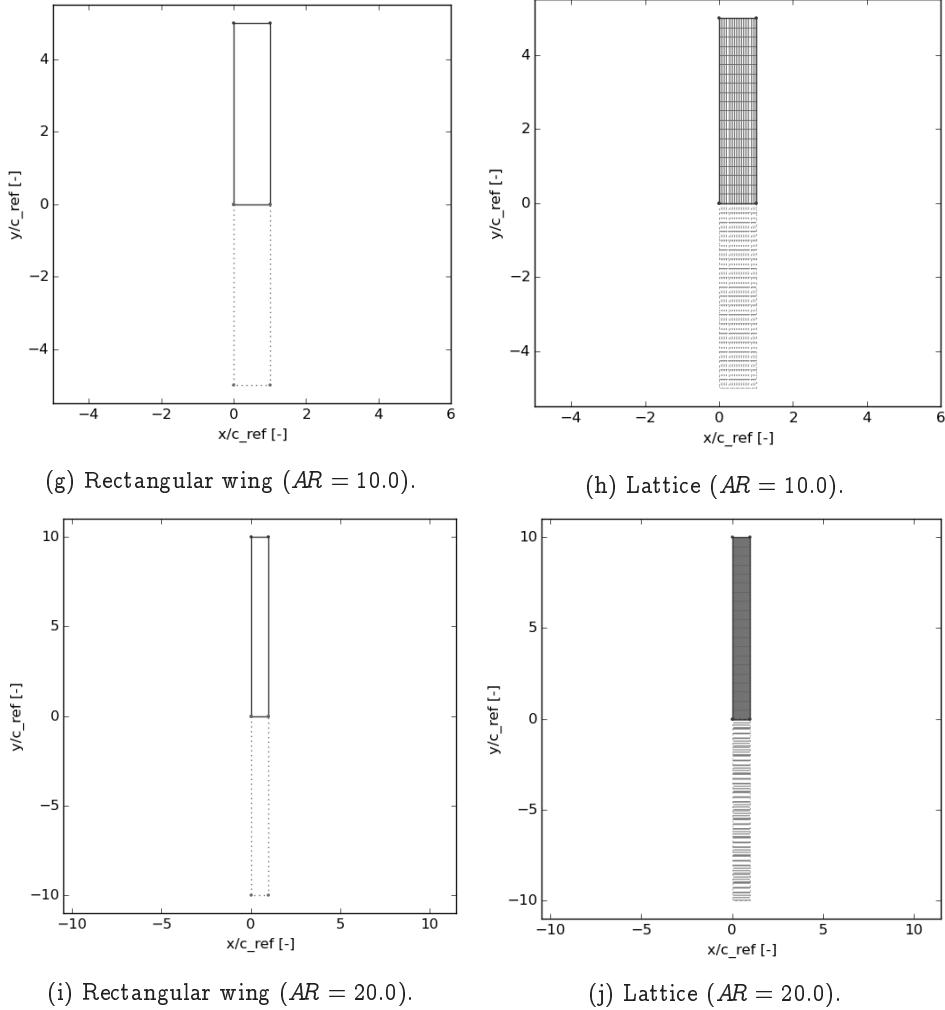


Figure 5.9: Rectangular wing configuration used in the convergence studies of Rodden et al [19], projected in the x , y -plane. Dotted lines indicate parts of the geometry that are not modeled. The wing chord is kept constant at $c = 1.0$ m, and the panel aspect ratios are equal to the wing aspect ratio (and the wing span) $AR = 1.0, 2.0, 3.0, 5.0$, and 10.0 . Coordinates are normalized by the reference length $L_{\text{ref}} = 0.5$ m.

<i>k</i>	<i>s</i>	panel <i>AR</i>	quadratic (NK5A) [19]	quadratic (NeoCASS) [28]	quadratic (present)
0.1	1.0	1.0	$2.915 + i0.3653$	$2.9174 + i0.3680$	$2.9174 + i0.3680$
0.1	2.0	2.0	$4.608 - i0.0186$	$4.6511 - i0.0051$	$4.6511 - i0.0051$
0.1	3.0	3.0	$5.445 - i0.4964$	$5.5666 - i0.4881$	$5.5666 - i0.4881$
0.1	5.0	5.0	$6.078 - i1.144$	$6.3735 - i1.1947$	$6.3735 - i1.1947$
0.1	10.0	10.0	$6.392 - i1.649$	-	$7.0384 - i1.8783$
0.5	1.0	1.0	$3.814 + i1.735$	$3.8290 + i1.7461$	$3.8290 + i1.7461$
0.5	2.0	2.0	$4.726 + i0.5393$	$4.8290 + i0.5361$	$4.8290 + i0.5361$
0.5	3.0	3.0	$4.740 + i0.2891$	$4.9213 + i0.2663$	$4.9213 + i0.2663$
0.5	5.0	5.0	$4.825 + i0.1560$	$5.1769 + i0.0953$	$5.1769 + i0.0953$
0.5	10.0	10.0	$4.843 + i0.2070$	-	$5.5466 + i0.0925$
1.0	1.0	1.0	$4.897 + i1.291$	$4.9112 + i1.2855$	$4.9112 + i1.2855$
1.0	2.0	2.0	$4.882 + i0.9511$	$4.9508 + i0.9411$	$4.9508 + i0.9411$
1.0	3.0	3.0	$4.906 + i0.8530$	$5.0330 + i0.8342$	$5.0330 + i0.8342$
1.0	5.0	5.0	$4.847 + i0.9123$	$5.2104 + i0.8731$	$5.2104 + i0.8731$
1.0	10.0	10.0	$4.888 + i1.272$	-	$5.4852 + i1.2273$
2.0	1.0	1.0	$5.647 + i1.355$	$5.6633 + i1.3473$	$5.6633 + i1.3473$
2.0	2.0	2.0	$5.770 + i1.034$	$5.8484 + i0.9853$	$5.8484 + i0.9853$
2.0	3.0	3.0	$5.775 + i1.091$	$5.9309 + i0.9957$	$5.9309 + i0.9957$
2.0	5.0	5.0	$5.681 + i1.368$	$5.9950 + i1.1909$	$5.9950 + i1.1909$
2.0	10.0	10.0	$5.150 + i2.257$	-	$5.8102 + i2.0369$
<i>k</i>	<i>s</i>	panel <i>AR</i>	quartic (NK5Q) [19]	quartic (NeoCASS) [28]	quartic (present)
0.1	1.0	1.0	$2.908 + i0.3546$	$2.9142 + i0.3680$	$2.9142 + i0.3662$
0.1	2.0	2.0	$4.587 - i0.0370$	$4.6132 - i0.0136$	$4.6132 - i0.0136$
0.1	3.0	3.0	$5.414 - i0.5164$	$5.4674 - i0.4933$	$5.4674 - i0.4933$
0.1	5.0	5.0	$6.038 - i1.166$	$6.1590 - i1.1696$	$6.1590 - i1.1696$
0.1	10.0	10.0	$6.342 - i1.691$	-	$6.6220 - i1.7840$
0.5	1.0	1.0	$3.775 + i1.709$	$3.8034 + i1.7482$	$3.8034 + i1.7482$
0.5	2.0	2.0	$4.701 + i0.5339$	$4.7714 + i0.5353$	$4.7714 + i0.5353$
0.5	3.0	3.0	$4.747 + i0.2547$	$4.8258 + i0.2348$	$4.8258 + i0.2348$
0.5	5.0	5.0	$4.873 + i0.0507$	$4.9995 + i0.0036$	$4.9995 + i0.0036$
0.5	10.0	10.0	$4.933 - i0.0355$	-	$5.2079 - i0.1263$
1.0	1.0	1.0	$4.832 + i1.335$	$4.9045 + i1.3215$	$4.9045 + i1.3215$
1.0	2.0	2.0	$4.874 + i0.9398$	$4.9227 + i0.9038$	$4.9227 + i0.9038$
1.0	3.0	3.0	$4.921 + i0.7556$	$4.9615 + i0.7077$	$4.9615 + i0.7077$
1.0	5.0	5.0	$4.989 + i0.6529$	$5.0476 + i0.5995$	$5.0476 + i0.5995$
1.0	10.0	10.0	$5.026 + i0.7201$	-	$5.2042 + i0.6783$
2.0	1.0	1.0	$5.610 + i1.432$	$5.6440 + i1.4054$	$5.6440 + i1.4054$
2.0	2.0	2.0	$5.800 + i0.9705$	$5.8112 + i0.9211$	$5.8112 + i0.9211$
2.0	3.0	3.0	$5.856 + i0.8407$	$5.8578 + i0.7896$	$5.8578 + i0.7896$
2.0	5.0	5.0	$5.878 + i0.8073$	$5.9240 + i0.7541$	$5.9240 + i0.7541$
2.0	10.0	10.0	$5.806 + i1.0623$	-	$6.0488 + i0.9236$

Table 5.11: Complex lift coefficient \bar{C}_L for rectangular wing of varying semi-span *s* pitching about its mid-chord, compared with results of Rodden et al [19] and Da Ronch and Cavagna [28]. Values shown in this table were obtained using both the quadratic and quartic approximations to the kernel numerator. The reference area $A_{\text{ref}} = 4.0$ used to calculate the lift coefficient is the surface area of the wing in terms of the non-dimensional coordinates, rescaled by the reference length $L_{\text{ref}} = 0.5$ m. *i* denotes the imaginary unit.

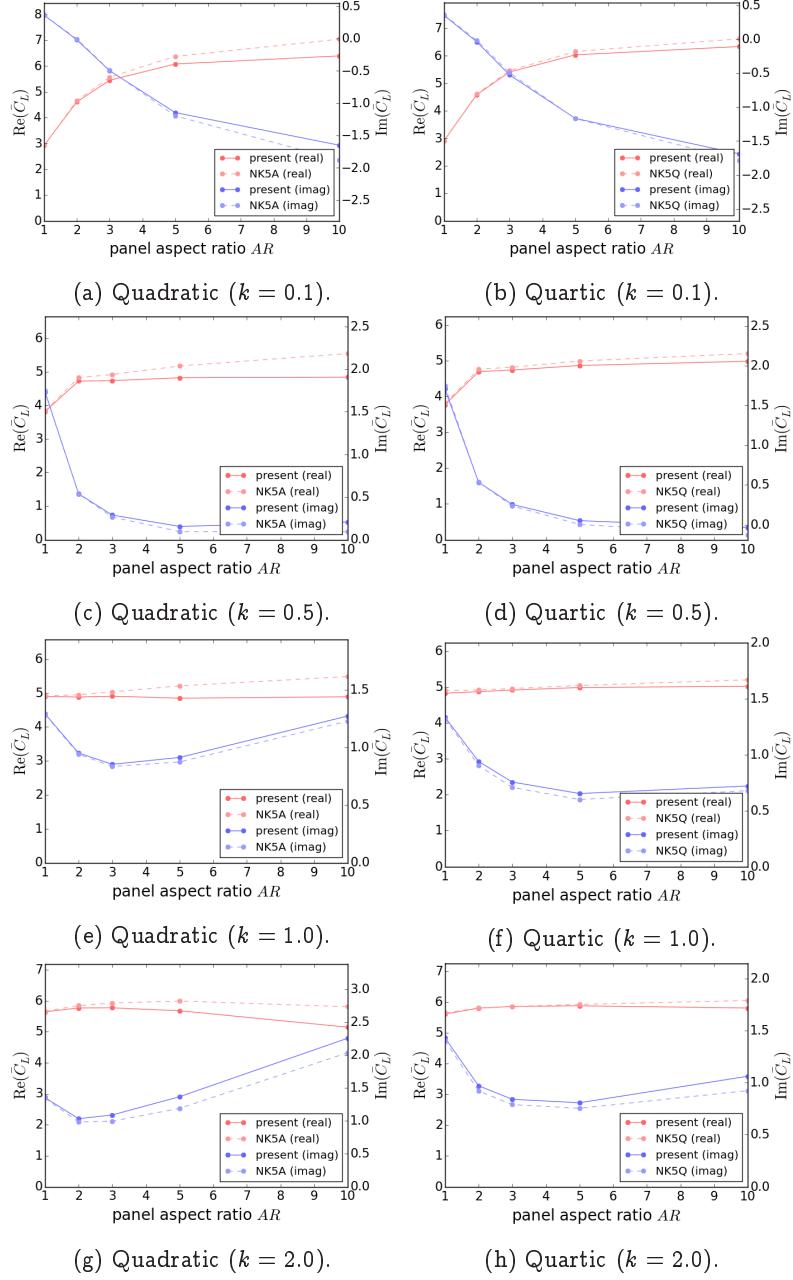


Figure 5.10: Complex lift coefficient \bar{C}_L for rectangular wings pitching about their mid-chord, in comparison with results of Rodden et al [19]. Values are plotted against panel aspect ratio, varied by increasing the wing semi-span s . Results were obtained using both the quadratic (left column) and quartic (right column) approximations to the kernel numerator.

It may be worth quantifying the error induced in the approximation of the steady downwash factors using the Doublet-Lattice Method, rather than the Vortex-Lattice Method, in a future study.

From the results of the first convergence study, shown in Figure 5.7, it is seen that the values converge as the number of chord-wise panels n_c is increased. No evident deterioration of the approximation can be attributed to the increasing panel aspect ratio within the range investigated range. However, it is observed that the convergence rate seems observed to decrease at larger reduced frequencies; this effect is especially marked for the imaginary part of the calculated lift coefficient.

A similar convergence of the lift coefficient can not be expected for the second study, as the panel aspect ratio is varied by varying the geometry of the wing.

Overall, the real parts calculated using both approximations to the kernel numerator seem to be in reasonably good agreement. At large reduced frequencies $k = 1.0$ and $k = 2.0$, the discrepancy between the imaginary parts of the lift coefficient becomes notable. The panel aspect ratio seems to influence primarily the results obtained for the imaginary parts, as is highlighted by the results of the second study (see Figure 5.9). This is also noted by the authors of the reference paper.

We recall from the examination of the planar component of the kernel numerator in Section 5.1 that the imaginary part is consistently better approximated by the quartic fit.

Based on these observations and results of previous studies [9, 18, 10], Rodden et al suggest, as a general guideline, that panel aspect ratio should approximate unity, and should be limited to 3 for the quadratic approximation. This requirement can be relaxed to 10 for the quartic approximation.

An additional rule is formulated for the number of chord-wise panels to be used, stating that the chord of each panel should not exceed a certain fraction of the wavelength $\lambda \equiv 2\pi L_{\text{ref}}/k$ of the oscillatory motion. The number of divisions per wavelength N_λ can be determined by the following restriction of the panel chord Δx :

$$\Delta x \leq \frac{1}{N_\lambda} \frac{\max(U_\infty)}{\min(k)} \quad (5.1)$$

where $\max(U_\infty)$ denotes the largest free-stream velocity of interest and $\min(k)$ the smallest reduced frequency. The recommended number of divisions per wavelength is $N_\lambda = 50$.

In a convergence study of the Doublet-Lattice Method, where the error induced by the interpolation of the kernel is isolated by eliminating integration error, van Zyl [30] conclude that the restrictions on panel aspect ratio may not be sufficient to ensure convergence of the results. Instead, it is suggested that the span- and chord-wise extent of the doublet lines should be limited. This criterion has the advantage of naturally leading to finer discretisations of highly swept or tapered planforms. The limit of one-quarter of the wavelength λ on the chord- and span-wise extent of the doublet line is proposed to ensure convergence. A minimum of four chord-wise divisions and five span-wise divisions is recommended.

6 Concluding Remarks

In the present work, the development of a Python/C++ implementation of the Doublet-Lattice Method for oscillating lifting surfaces in subsonic flows has been presented.

A particular effort was made to aggregate and compile a complete documentation of the method from the existing literature, ranging from the fundamental equations of fluid motion to details of its implementation. The role of unsteady aerodynamics in flutter analysis has also been commented.

Throughout the development process, care was taken to meet additional constraints, arising from the projected integration of the present program - or components thereof - in the CEASIOM environment for integrated aircraft design. It was sought to increase the automation and performance of the routines for the evaluation of the unsteady aerodynamic loading on arbitrary aircraft configurations.

The correct implementation of the method has been assessed in a verification process, using a variety of elementary test cases. The results obtained are certainly encouraging, and it can reasonably be concluded that the algorithm is functioning as it should. Both the planar and non-planar components of the kernel function, elementary unit of the method, have been shown to be correctly approximated.

However, it is the author's opinion that the present implementation requires further validation before the program can be applied to arbitrary configurations. Despite having demonstrated consistently accurate results, over a range of operating conditions, for the simple test cases of Chapter 5, some difficulties were encountered when attempting to reproduce results for more representative test cases. The author suspects that this is do with how (and which) reference lengths are employed in the definition of the boundary conditions and aerodynamic coefficients in the implementations of the Doublet-Lattice Method. These aspects are often not quite consistently addressed in the available documentation.

The part of error induced by the approximation of the steady component of the downwash factors remains an unknown, as the present implementation does not employ the Vortex-Lattice Method to calculates these exactly. However, this uncertainty will easily be eliminated once the Python version of TORNADO becomes available. It seems from the results of Chapter 5 also that this error is not likely to be significant, due to the good agreement found with the reference results.

It is the author's hope that the present work will serve as a basis for further development, eventually leading toward a complete program for flutter prediction at the conceptual design stage. Beyond what has already been mentioned in the above, this leaves several opportunities for future work.

The Doublet-Lattice Method allows to compute the unsteady pressure distribution over a variety of aircraft geometries, from which in turn the matrix $[Q]$ of generalised aerodynamic forces may be calculated. This term is required for the solution of the eigenvalue problem associated to the equations of motion of a wing in flutter (see Chapter 2). It would remain to determine the matrices $[K]$, $[M]$ and eventually $[B]$ of generalised structural stiffness, mass and damping, before this problem may be tackled. In addition, for complex wings, it is necessary to determine the structural deformation modes, required as inputs for

the numerically. It becomes clear that the coupling of the present Doublet-Lattice Method implementation with a structural model is a requirement for aero-elastic analysis. Care should be taken to select an adequate scheme for the interpolation of the results between the aerodynamic and structural grid.

During the conceptual design phase of aircraft design however, the information available on the inertial and elastic properties of the wings may be insufficient for the elaboration of a structural model. It may then be worthwhile looking into methods for reliable structural sizing and prediction of weight and balance properties for the early stages of aircraft design.

Finally, some future work on the Doublet-Lattice Method itself may be suggested. Giesing et al [16] describe a method that includes the interference of fuselage elements, for better prediction of wing-body interactions. In its current form, the Doublet-Lattice Method is limited to subsonic flows; several methods have been proposed to extend its domain of validity to transonic regimes [31, 32]. Once more, the value of the gain in accuracy obtainable is to be weighted against the increase in complexity of the method, especially in the conceptual design phase. In an integrated design environment such as CEASIOM, it may be simpler to rely on more specialised and higher-fidelity methods.

References

- [1] World Bank. Air transport, passengers carried. <http://data.worldbank.org/indicator/IS.AIR.PSGR>. Data from International Civil Aviation Organization, Civil Aviation Statistics of the World and ICAO staff estimates. Accessed November 15th, 2016.
- [2] World Bank. Air transport, freight (million ton-km). <http://data.worldbank.org/indicator/IS.AIR.GOOD.MT.K1>. Data from International Civil Aviation Organization, Civil Aviation Statistics of the World and ICAO staff estimates. Accessed November 15th, 2016.
- [3] European Commission. Flightpath 2050: Europe's vision for aviation., 2011.
- [4] Advisory Group for Aerospace Research and Development (AGARD). Special course on engineering methods in aerodynamic analysis and design of aircraft. Technical report, North Atlantic Treaty Organisation (NATO), January 1992.
- [5] A. Rizzi. Modeling and simulating aircraft Stability & Control - the SimSAC project. In *AIAA Atmospheric Flight Mechanics Conference*. American Institute of Aeronautics and Astronautics, August 2010.
- [6] CFS Engineering SA. CEASIOM - computerised environment for aircraft synthesis and integrated optimisation methods. www.ceasiom.com/. accessed October 20th, 2016.
- [7] Raymond L. Bisplinghoff, Holt Ashley, and Robert L. Halfman. *Aeroelasticity*. Dover Publications, Mineola, New York, first edition, 1996.
- [8] E. Albano and W. P. Rodden. A Doublet-Lattice Method for calculating lift distributions on oscillating surfaces in subsonic flows. *AIAA Journal*, 7(2), 1969.
- [9] W. P. Rodden, , J. P. Giesing, and T. P. Kalman. Refinement of the non-planar aspects of the subsonic Doublet-Lattice Method. *Journal of Aircraft*, 9(1):69–73, 1972.
- [10] W. P. Rodden, P. F. Taylor, and S. C. McIntosh Jr. Further refinement of the subsonic Doublet-Lattice Method. *Journal of Aircraft*, 35(5):720–727, 1998.
- [11] S. G. Hedman. Vortex lattice method for calculation of quasi steady state loading on thin elastic wings in subsonic flow. Technical report, Flygtekniska Försöksanstalten, October 1965.
- [12] J. P. Giesing. Basic principles and double lattice applications in potential aerodynamics. In L. Morino, editor, *Computational Methods in Potential Aerodynamics*. Springer-Verlag, Computational Mechanics Centre, Southampton, 1987.
- [13] H. T. Vivian and L. V. Andrews. Unsteady aerodynamics for advanced configurations. Part I - Application of the subsonic kernel function to nonplanar lifting surfaces. Technical report, Air Force Flight Dynamics Laboratory, May 1965.

- [14] M. T. Landahl. Kernel function for nonplanar oscillating surfaces in a subsonic flow. *AIAA Journal*, 5(5):1045–1046, 1967.
- [15] B. Laschka. Das Potential und das Geschwindigkeitsfeld der harmonisch schwingenden tragenden Fläche bei Unterschallströmung. *ZAMM*, 43(7-8):325–333, 1963.
- [16] J. P. Giesing, T. P. Kalman, and W. P. Rodden. Subsonic unsteady aerodynamics for general configurations. Part I, Vol. I - Direct application of the nonplanar Doublet-Lattice Method. Technical report, Air Force Flight Dynamics Laboratory, November 1971.
- [17] J. P. Giesing, T. P. Kalman, and W. P. Rodden. Subsonic unsteady aerodynamics for general configurations. Part I, Vol. II - Computer program H7WC. Technical report, Air Force Flight Dynamics Laboratory, November 1971.
- [18] W. P. Rodden, P. F. Taylor, and S. C. McIntosh Jr. Further refinement of the non-planar aspects of the subsonic doublet-lattice lifting surface method. In *ICAS Conference Proceedings*, pages Paper 96–2.8.2.
- [19] W. P. Rodden, P. F. Taylor, S. C. McIntosh Jr., and M. L. Baker. Further convergence studies of the enhanced Doublet-Lattice Method. *Journal of Aircraft*, 36(4):683–688, 1999.
- [20] K. W. Mangler. Improper integrals in theoretical aerodynamics. Technical report, Aeronautical Research Council, December 1952.
- [21] B. Bennekers and Th. E. Labrujere. Calculation of unsteady subsonic aerodynamic characteristics by means of the Doublet Lattice Method. Technical report, Nationaal Lucht- en Ruimtevaartlaboratorium (NLR), November 1972.
- [22] B. Nagel, D. Böhnke, P. Gollnick, P. Schmollgruber, A. Rizzi, G La Rocca, and J. J. Alonso. Communication in aircraft design: Can we establish a common language? In *28th International Congress of the Aeronautical Sciences, Brisbane, Australia*, 2012.
- [23] Deutsches Zentrum für Luft-und Raumfahrt (DLR). CPACS - a common language for aircraft design. <https://software.dlr.de/p/cpacs/home/>. accessed October 20th, 2016.
- [24] Thomas Melin. *Using Internet Interactions in Developing Vortex Lattice Software for Conceptual Design*. PhD thesis, Kungliga Tekniska Hogskolan (KTH), June 2003.
- [25] Deutsches Zentrum für Luft-und Raumfahrt (DLR). TiGL geometry library. <https://software.dlr.de/p/tigl/home/>. accessed January 30th, 2017.
- [26] L. Prandtl. Induced drag of multiplanes. Technical report, N.A.C.A. TN., March 1924.
- [27] M. Blair. A compilation of the mathematics leading to the Doublet Lattice Method. Technical report, USAF Wright Laboratory, March 1992.

- [28] Andrea Da Ronch and Luca Cavagna. A doublet-lattice method for calculating lift distributions on planar and non-planar configurations in subsonic flows. May 2007.
- [29] L. Cavagna, S. Ricci, and L. Riccobene. Structural sizing, aeroelastic analysis, and optimization in aircraft conceptual design. *Journal of Aircraft*, 48(6):1840–1855, 2011.
- [30] L. H. Van Zyl. Convergence of the subsonic Doublet-Lattice Method. *Journal of Aircraft*, 35(6):977–979, 1998.
- [31] S. Lu and R. Voss. TDLM - a transonic doublet lattice method for 3d potential unsteady transonic flow calculation. Technical report, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 1992.
- [32] A. B. Guimaraes Neto, R. G. A. Da Silva, and Padiglione P. Control-point-placement method for the aerodynamic correction of the Vortex- and the Doublet-Lattice Methods. *Aerospace Science and Technology*, 37:117–129, 2014.
- [33] Krishnamurty Karamcheti. *Principles of Ideal-Fluid Aerodynamics*. Krieger Publishing Company, Malabar, Florida, second edition, 1980.
- [34] Holt Ashley and Marten Landahl. *Aerodynamics of Wings and Bodies*. Addison-Wesley Publishing Company, inc., Reading, Massachusetts, first edition, 1965.

A Additional notes on the Theory of Potential Flow

In Section 3 of this report we concisely outline the derivation of the Doublet-Lattice Method: the oscillatory pressure distribution is approximated using the integral formula (3.17), which is obtained by superposition of elementary solutions of the so-called linearised aerodynamic potential equation (3.1).

In the interest of completeness, this supplementary section is dedicated to an overview of theoretical notions relevant to the Doublet-Lattice Method. From the equations of motion for inviscid flow, we show a derivation of the governing partial differential equation, while introducing fundamental concepts of potential aerodynamics.

A.1 Equations of Motion for Inviscid Flow

As a first step, we establish the equations of motion describing inviscid flow. Let x, y, z be the three components of the Cartesian frame of reference, and let t denote the time variable. The five state variables of interest are the pressure $p(x, y, z, t)$, the density $\rho(x, y, z, t)$, and $u(x, y, z, t), v(x, y, z, t), w(x, y, z, t)$ the components along x, y, z of the flow velocity field $\vec{u}(x, y, z, t)$. For simplicity, we omit contributions from body forces in the following.

Conservation of Mass While the volume and density of an infinitesimal fluid element may vary throughout the flow, the mass of such an element remains constant. This may be expressed in terms of the flow velocity, by observing that the rate of dilatation of a fluid volume is given by the divergence of the local velocity field, yielding

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \quad (\text{A.1})$$

known as the *equation of conservation of mass*, or *continuity equation*.

Conservation of Momentum Next, we introduce the components along x, y, z of the *equation of conservation of momentum* for inviscid flows, also known as *Euler's equation*:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = - \frac{1}{\rho} \frac{\partial p}{\partial x} \quad (\text{A.2a})$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = - \frac{1}{\rho} \frac{\partial p}{\partial y} \quad (\text{A.2b})$$

$$\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} = - \frac{1}{\rho} \frac{\partial p}{\partial z} \quad (\text{A.2c})$$

Essentially, these equations are a formulation of *Newton's second law of motion*, which states that the rate of change of momentum within a system is given by the balance of forces acting upon the system.

It is important to remind the reader that contributions from viscous effects are neglected in the derivation of Euler's equation, restricting its validity to inviscid flows.

Equation of State Under isentropic conditions, perturbations in pressure are related to perturbations in the fluid density as

$$\frac{p}{\rho^\gamma} = \frac{p_\infty}{\rho_\infty^\gamma} \quad (\text{A.3})$$

with γ the ratio of specific heat capacities. Equation (A.3) is derived from the well-known ideal gas law. If it is single-valued throughout the domain, the flow is said to be *barotropic*; for most practical applications one may safely assume this condition to hold.

A.2 Fundamentals of Potential Flow

We wish to restrict our analysis to irrotational flows, allowing us to simplify the problem by invoking the notion of *velocity potential*. Here, we present some fundamental concepts related to such flows. For further reading on the theory of potential flow, the reader is directed to the reference material used for this section: Chapter 9 of Karamcheti [33], Chapter 1 of Ashley and Landahl [34], and Chapter 5 of Bisplinghoff et al [7].

Irrational Flow The *vorticity* $\vec{\omega}(x, y, z, t)$ at some point in the flow field is related to the local rate of rotation of the fluid. Formally, it is defined as the curl of the velocity field $\vec{u}(x, y, z, t) \equiv u(x, y, z, t)\hat{e}_x + v(x, y, z, t)\hat{e}_y + w(x, y, z, t)\hat{e}_z$,

$$\vec{\omega} = \nabla \times \vec{u} \quad (\text{A.4})$$

A vector field is said to be *irrotational* if it is curl-free; for the velocity field of fluid flow it is by definition equivalent to require the vorticity vector to be everywhere null.

The vorticity field $\vec{\omega}$ has several interesting properties, which we will not list here in the interest of brevity (see *Helmholtz's vortex theorems* and *Kelvin's circulation theorem* in the reference material). Perhaps the most important result may be paraphrased as follows: in an inviscid, barotropic flow subject to conservative body forces, and in absence of heat transfer or strong curved shocks, initially irrotational flows will remain irrotational.

The Velocity Potential It can be shown that irrotationality is a necessary and sufficient requirement for the description of a vector field as the gradient of a scalar *potential field*. Hence for an inviscid, barotropic flow with an initially vorticity-free velocity field, we can guarantee the existence of a *velocity potential* $\phi(x, y, z, t)$ for all times t , such that

$$\vec{u} = \nabla\phi \quad \text{or} \quad u = \frac{\partial\phi}{\partial x}, \quad v = \frac{\partial\phi}{\partial y}, \quad w = \frac{\partial\phi}{\partial z} \quad (\text{A.5})$$

The assumption of irrotational flow allows for a welcome mathematical simplification, as a single scalar field may be used to describe the three unknown velocity components.

Equation of Motion Substitution of the velocity potential ϕ (A.5) into Euler's equation for conservation of momentum (A.2) yields, in vector form,

$$\begin{aligned} \frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} &= -\frac{1}{\rho} \nabla p \\ \Rightarrow \frac{\partial \vec{u}}{\partial t} + \nabla \left[\frac{U^2}{2} \right] - \vec{u} \times (\nabla \times \vec{u}) &= -\frac{1}{\rho} \nabla p \\ \Rightarrow \frac{\partial}{\partial t} (\nabla \phi) + \nabla \left[\frac{U^2}{2} \right] &= -\frac{1}{\rho} \nabla p \\ \Rightarrow \nabla \left[\frac{\partial \phi}{\partial t} + \frac{U^2}{2} + \int \frac{dp}{\rho} \right] &= \vec{0} \end{aligned} \quad (\text{A.6})$$

where we have introduced $U(x, y, z, t)$, the magnitude of the local velocity vector

$$U^2 \equiv \vec{u} \cdot \vec{u} = u^2 + v^2 + w^2 = \left[\frac{\partial \phi}{\partial x} \right]^2 + \left[\frac{\partial \phi}{\partial y} \right]^2 + \left[\frac{\partial \phi}{\partial z} \right]^2 = \nabla \phi \cdot \nabla \phi \quad (\text{A.7})$$

Observe that in the last equality of (A.6), the three components of the gradient vanish, implying that the quantity within the brackets may be a function only of time. Integration then yields the following result, known as *Kelvin's equation*:

$$\frac{\partial \phi}{\partial t} + \frac{U^2}{2} + \int \frac{dp}{\rho} = F(t) \quad (\text{A.8})$$

which could be seen as the unsteady equivalent of *Bernoulli's equation*. The time-dependent integration constant $F(t)$ may be eliminated artificially by introducing a modified definition of the velocity potential $\phi' \equiv \phi - \int F(t) dt$. Observe from (A.5) that this does not affect the interpretation of the velocity field. If we restrict the undisturbed flow to be parallel with uniform velocity U_∞ along the x -direction, we find $F(t) = \frac{1}{2} U_\infty^2$.

By a similar procedure the continuity equation (A.1) becomes, after dividing by ρ ,

$$\frac{1}{\rho} \frac{\partial \rho}{\partial t} + \frac{\nabla \phi \cdot \nabla \rho}{\rho} + \nabla^2 \phi = 0 \quad (\text{A.9})$$

where we introduced the familiar Laplacian operator $\nabla^2 \equiv \nabla \cdot \nabla$.

The Acceleration Potential In the context of the Doublet-Lattice Method, it is of particular relevance to include a description of Prandtl's *acceleration potential* in our survey of the theory of irrotational flow.

Euler's equation (A.2) may be rearranged as

$$\vec{a} \equiv \frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} = \nabla \left[\int \frac{dp}{\rho} \right] \quad (\text{A.10})$$

for a barotropic fluid. A consequence is that $\nabla \times \vec{a} = \vec{0}$ and, by analogy with the velocity potential, we may describe the acceleration field $\vec{a}(x, y, z, t)$ as the gradient of some scalar potential field $\Psi(x, y, z, t)$:

$$\vec{a} = \nabla \Psi \quad (\text{A.11})$$

Substituting into (A.10) and integrating shows that

$$\Psi = \int \frac{dp}{\rho} + G(t) \quad (\text{A.12})$$

where the integration constant $G(t)$ is a function of time only, analogous to $F(t)$. It is convenient to eliminate this term by introducing $\Psi' = \Psi - G(t) <$.

Under the assumption of small disturbances the acceleration potential reduces to

$$\Psi = \int_{p_\infty}^p \frac{dp}{\rho} \approx \frac{p_\infty - p}{\rho_\infty} \quad (\text{A.13})$$

in which p_∞ denotes the far-field pressure. Thus Ψ differs from the local pressure only by a constant; for this reason it is sometimes referred to as the *pressure potential*.

In oscillatory lifting surface theory, pulsating doublets of acceleration potential are used to represent the pressure differences induced across wings and around bodies.

A.3 The Linearised Aerodynamic Potential Equation

We now have the required elements to derive a single linear partial differential equation for the velocity potential. For this, we will use Kelvin's equation (A.8) and the equation of state (A.3) to eliminate the remaining state variables p , ρ , from the continuity equation (A.9). The derivation presented here closely follows Section II of Blair [27].

In order to express terms of (A.9) in function of ϕ only, we introduce the relation for the local propagation velocity of small perturbations within the fluid, the speed of sound, under isentropic conditions:

$$a^2 \equiv \left(\frac{\partial p}{\partial \rho} \right)_s = \frac{dp}{d\rho} = \frac{\gamma p}{\rho} \quad (\text{A.14})$$

where we used the equation of state (A.3).

Substituting into the first term of (A.9) yields

$$\frac{1}{\rho} \frac{\partial \rho}{\partial t} = \frac{1}{\rho a^2} \frac{dp}{d\rho} \frac{\partial \rho}{\partial t} = \frac{1}{\rho a^2} \frac{\partial p}{\partial t} = \frac{1}{a^2} \frac{\partial}{\partial t} \int \frac{dp}{\rho}$$

where the last equality follows from Leibniz's rule for differentiation of a definite integral with variable bounds. Using (A.8) we eliminate the integral to obtain

$$\frac{1}{\rho} \frac{\partial \rho}{\partial t} = -\frac{1}{a^2} \frac{\partial}{\partial t} \left[\frac{\partial \phi}{\partial t} + \frac{U^2}{2} \right] \quad (\text{A.15})$$

A similar procedure yields, for the second term of (A.9),

$$\frac{\nabla \phi \cdot \nabla \rho}{\rho} = \nabla \phi \cdot \frac{\nabla \rho}{\rho} = \frac{1}{a^2} \frac{dp}{d\rho} \nabla \phi \cdot \frac{\nabla \rho}{\rho} = \frac{1}{a^2} \nabla \phi \cdot \frac{\nabla p}{\rho} = \frac{1}{a^2} \nabla \phi \cdot \nabla \left[\int \frac{dp}{\rho} \right]$$

where once more Leibniz's rule was used to obtain the last equality. Again we recognise the last term of (A.8) in the above result to obtain

$$\frac{\nabla \phi \cdot \nabla \rho}{\rho} = -\frac{1}{a^2} \nabla \phi \cdot \nabla \left[\frac{\partial \phi}{\partial t} + \frac{U^2}{2} \right] = -\frac{1}{a^2} \left\{ \frac{\partial}{\partial t} \left[\frac{U^2}{2} \right] + \nabla \phi \cdot \nabla \left[\frac{U^2}{2} \right] \right\} \quad (\text{A.16})$$

Finally, substituting (A.15) and (A.16) back into the continuity equation (A.9) yields the *full potential equation*

$$\nabla^2\phi - \frac{1}{a^2} \left\{ \frac{\partial^2\phi}{\partial^2 t} + \frac{\partial}{\partial t}(U^2) + \nabla\phi \cdot \nabla \left[\frac{U^2}{2} \right] \right\} = 0 \quad (\text{A.17})$$

It is possible to find an expression for $a^2(x, y, z, t)$ in terms of the velocity potential and the speed of sound a_∞ of the far-field flow; however, inserting this relation into (A.17) would yield a complicated and impractical relation. Instead, we will directly replace a with a_∞ in the linearised form of the equation without inducing further error, due to the approximations made during the linearisation process.

We expand the full potential equation (A.17), using subscripts to denote differentiation,

$$\begin{aligned} \nabla^2\phi - \frac{1}{a^2}\phi_{tt} - \frac{1}{a^2}(\phi_x\phi_{xt} + \phi_y\phi_{yt} + \phi_z\phi_{zt}) \\ - \frac{1}{a^2}(\phi_x^2\phi_{xx} + \phi_y^2\phi_{yy} + \phi_z^2\phi_{zz}) \\ - \frac{1}{a^2}(2\phi_x\phi_y\phi_{xy} + 2\phi_x\phi_z\phi_{xz} + 2\phi_y\phi_z\phi_{yz}) = 0 \end{aligned} \quad (\text{A.18})$$

Suppose that the steady-state form of (A.18), obtained by setting all time derivatives to zero, admits a solution. Then, introduce small unsteady perturbations:

$$\phi(x, y, z, t) = \bar{\phi}(x, y, z) + \epsilon\tilde{\phi}(x, y, z, t) \quad (\text{A.19a})$$

$$p(x, y, z, t) = \bar{p}(x, y, z) + \epsilon\tilde{p}(x, y, z, t) \quad (\text{A.19b})$$

$$\rho(x, y, z, t) = \bar{\rho}(x, y, z) + \epsilon\tilde{\rho}(x, y, z, t) \quad (\text{A.19c})$$

$$a(x, y, z, t) = \bar{a}(x, y, z) \quad (\text{A.19d})$$

where $\epsilon \ll 1$ is a small-disturbance parameter.

Substituting the new variables into (A.18), eliminating all terms non-linear in ϵ and subtracting the steady solution gives

$$\begin{aligned} (\tilde{\phi}_{xx} + \tilde{\phi}_{yy} + \tilde{\phi}_{zz}) - \frac{1}{\bar{a}^2}(\tilde{\phi}_{tt} + 2\bar{\phi}_x\tilde{\phi}_{xt} + 2\bar{\phi}_y\tilde{\phi}_{yt} + 2\bar{\phi}_z\tilde{\phi}_{zt} \\ + \bar{\phi}_x^2\tilde{\phi}_{xx} + \bar{\phi}_y^2\tilde{\phi}_{yy} + \bar{\phi}_z^2\tilde{\phi}_{zz} \\ + 2(\bar{\phi}_x\bar{\phi}_y\tilde{\phi}_{xy} + \bar{\phi}_x\tilde{\phi}_y\bar{\phi}_{xy} + \tilde{\phi}_x\bar{\phi}_y\bar{\phi}_{xy}) \\ + 2(\bar{\phi}_x\bar{\phi}_z\tilde{\phi}_{xz} + \bar{\phi}_x\tilde{\phi}_z\bar{\phi}_{xz} + \tilde{\phi}_x\bar{\phi}_z\bar{\phi}_{xz}) \\ + 2(\bar{\phi}_y\bar{\phi}_z\tilde{\phi}_{yz} + \bar{\phi}_y\tilde{\phi}_z\bar{\phi}_{yz} + \tilde{\phi}_y\bar{\phi}_z\bar{\phi}_{yz})) = 0 \end{aligned} \quad (\text{A.20})$$

A final simplification is made by explicitly defining the steady mean flow as uniform and parallel with velocity U_∞ along x , or $\bar{\phi}(x, y, z) = U_\infty x$. We set $\bar{a} = a_\infty$ and introduce the far-field Mach number $M_\infty \equiv U_\infty/a_\infty$, to obtain

$$(1 - M_\infty^2)\tilde{\phi}_{xx} + \tilde{\phi}_{yy} + \tilde{\phi}_{zz} - \frac{2U_\infty}{a_\infty^2}\tilde{\phi}_{xt} - \frac{1}{a_\infty^2}\tilde{\phi}_{tt} = 0 \quad (\text{A.21})$$

Which is a linear partial differential equation in the desired form, the so-called *linearised aerodynamic potential equation*. It can be shown that the acceleration potential satisfies the same equation. As a final note, it must be reminded once more that this equation does not account for any effects of viscosity.

A.4 Boundary Conditions

Solutions to the linearised aerodynamic potential equation (3.1) for a particular problem, in our case external flow over an aircraft, are determined by imposing additional constraints on the velocity potential perturbation.

Consider the flow along the surface $S(x, y, z, t) = 0$ of an immersed body. Viscous effects were omitted in deriving Euler's equation (A.2), so that a no-slip condition at the surface cannot be justified. Only an impermeability condition may be enforced, meaning that the flow must be tangential to the surface; this is expressed mathematically by imposing the material derivative of the velocity vector \vec{u} to vanish along $S = 0$.

$$\frac{\partial S}{\partial t} + \vec{u} \cdot \nabla S = 0 \quad (\text{A.22})$$

The second boundary condition follows from the observation that the flow field \vec{u} away from the body must tend toward the unperturbed free-stream flow \vec{u}_∞ . In the above, we have assumed the flow at infinity to be uniform and parallel with velocity U_∞ along x .

$$\vec{u} \Big|_{\text{far-field}} = U_\infty \hat{e}_x \quad (\text{A.23})$$

Substitution of (A.5) into (A.22) and (A.23) yields the boundary conditions for Euler's equation (A.2) in terms of the velocity potential ϕ ,

$$\frac{\partial S}{\partial t} + \nabla \phi \cdot \nabla S = 0 \quad (\text{A.24})$$

and

$$\nabla \phi \Big|_{\text{far-field}} = U_\infty \hat{e}_x \quad (\text{A.25})$$

Introducing the perturbed variables of (A.19), the boundary conditions become

$$\frac{\partial S}{\partial t} + \nabla \bar{\phi} \cdot \nabla S + \nabla \tilde{\phi} \cdot \nabla S = 0 \quad (\text{A.26})$$

and

$$(\nabla \bar{\phi} + \nabla \tilde{\phi}) \Big|_{\text{far-field}} = U_\infty \hat{e}_x \quad (\text{A.27})$$

The boundary conditions (A.24) and (A.25) are satisfied for the steady component $p\bar{\phi}$ of the velocity potential. Eliminating these from the above equations yields the final expression for the boundary conditions,

$$\frac{\partial S}{\partial t} + \nabla \tilde{\phi} \cdot \nabla S = 0 \quad \text{and} \quad \nabla \tilde{\phi} \Big|_{\text{far-field}} = 0 \quad (\text{A.28})$$

B Refinement of the Doublet-Lattice Method

In Section 3.3 we present a parabolic approximation (3.32) to the numerator of the kernel function (3.19). To improve treatment of near-coplanar configurations, Rodden et al [9] propose to apply the approximation separately to the planar and non-planar terms K_1 and K_2 of the kernel. The limiting values $K_1^{(0)}$ and $K_2^{(0)}$ of the approximated terms are subtracted from the solution, and the exact steady incremental downwash factor $D_{rs}^{(0)}$ computed exactly using the Vortex-Lattice Method. Finally, the elements D_{rs} of the matrix of total downwash factors [D] are composed of

$$D_{rs} = D_{rs}^{(0)} + D_{rs}^{(1)} + D_{rs}^{(2)}$$

where the superscripts (0) , (1) and (2) denote respectively the steady, planar unsteady and non-planar unsteady contributions.

The convergence of this method becomes limited for increasing panel aspect ratios and reduced frequencies.

B.1 Quartic Approximation of the Kernel Function

For more complex configurations (high-aspect ratio wings, delta wings), satisfying these conditions may require considerable refinement of the geometry discretisation, increasing computational cost. The number of operations required to compute the matrix [D] grows as $\mathcal{O}(n_{\text{tot}}^2)$ with n_{tot} the total number of panels. In an effort to overcome this limitation, Rodden et al [10] propose an improved quartic approximation of the kernel numerator.

Equation (3.32) becomes

$$D_{rs}^{(1)} = \frac{\Delta\xi_s}{8\pi} \int_{-e}^{+e} \frac{P_1(\bar{\eta})}{r^2} d\bar{\eta}$$

with $P_1(\bar{\eta}) \equiv A_1\bar{\eta}^2 + B_1\bar{\eta} + C_1 + D_1\bar{\eta}^3 + E_1\bar{\eta}^4 \approx \left\{ K_1 \exp \left[\frac{-i\omega(\bar{x} - \bar{\eta} \tan \Lambda_s)}{U_\infty} \right] - K_1^{(0)} \right\} T_1$

$$D_{rs}^{(2)} = \frac{\Delta x_s}{8\pi} \int_{-e}^{+e} \frac{P_2(\bar{\eta})}{r^4} d\bar{\eta}$$

with $P_2(\bar{\eta}) \equiv A_2\bar{\eta}^2 + B_2\bar{\eta} + C_2 + D_2\bar{\eta}^3 + E_2\bar{\eta}^4 \approx \left\{ K_2 \exp \left[\frac{-i\omega(\bar{x} - \bar{\eta} \tan \Lambda_s)}{U_\infty} \right] - K_2^{(0)} \right\} T_2^*$

with

$$\begin{aligned} A_1 &= -[P_1(-e) - 16P_1(-e/2) + 30P_1(0) \\ &\quad - 16P_1(+e/2) + P_1(+e)] / 6e^2 & A_2 &= -[P_2(-e) - 16P_2(-e/2) + 30P_2(0) \\ &\quad - 16P_2(+e/2) + P_2(+e)] / 6e^2 \\ B_1 &= [P_1(-e) - 8P_1(-e/2) & B_2 &= [P_2(-e) - 8P_2(-e/2) \\ &\quad + 8P_1(+e/2) - P_1(+e)] / 6e && + 8P_2(+e/2) - P_2(+e)] / 6e \\ C_1 &= P_1(0) & \text{and } C_2 &= P_2(0) \\ D_1 &= -[P_1(-e) - 2P_1(-e/2) & D_2 &= -[P_2(-e) - 2P_2(-e/2) \\ &\quad + 2P_1(+e/2) - P_1(+e)] / (3e^3/2) && + 2P_2(+e/2) - P_2(+e)] / (3e^3/2) \\ E_1 &= [P_1(-e) - 4P_1(-e/2) + 6P_1(0) & E_2 &= [P_2(-e) - 4P_2(-e/2) + 6P_2(0) \\ &\quad - 4P_1(+e/2) + P_1(+e)] / (3e^4/2) && - 4P_2(+e/2) + P_2(+e)] / (3e^4/2) \end{aligned}$$

Substituting the refined approximation into for $D_{rs}^{(1)}$ and $D_{rs}^{(2)}$ and integrating by parts,

$$\begin{aligned} D_{rs}^{(1)} &= \frac{\Delta \xi_s}{8\pi} \left\{ \left[(\bar{y}^2 - \bar{z}^2) A_1 + \bar{y} B_1 + C_1 + \bar{y}(\bar{y}^2 - 3\bar{z}^2) D_1 + (\bar{y}^4 - 6\bar{y}^2\bar{z}^2 + \bar{z}^4) E_1 \right] H \right. \\ &\quad + \left[\bar{y} A_1 + \frac{1}{2} B_1 + \frac{1}{2} (3\bar{y}^2 - \bar{z}^2) D_1 + 2\bar{y}(\bar{y}^2 - \bar{z}^2) E_1 \right] \log \left[\frac{(\bar{y} - e)^2 + \bar{z}^2}{(\bar{y} + e)^2 + \bar{z}^2} \right] \\ &\quad \left. + 2e [A_1 + 2\bar{y}D_1 + (3\bar{y}^2 - \bar{z}^2 + \frac{1}{3}e^2)E_1] \right\} \\ D_{rs}^{(2)} &= \frac{\Delta \xi_s}{16\pi\bar{z}^2} \left\{ \left[(\bar{y}^2 + \bar{z}^2) A_2 + \bar{y} B_2 + C_2 + \bar{y}(\bar{y}^2 - 3\bar{z}^2) D_2 + (\bar{y}^4 + 6\bar{y}^2\bar{z}^2 - 3\bar{z}^4) E_2 \right] H \right. \\ &\quad + \frac{1}{(\bar{y} + e)^2 + \bar{z}^2} \left[[(\bar{y}^2 + \bar{z}^2)\bar{y} + (\bar{y}^2 - \bar{z}^2)e] A_2 + (\bar{y}^2 + \bar{z}^2 + \bar{y}e) B_2 + (\bar{y} + e) C_2 \right. \\ &\quad \left. + [\bar{y}^4 - \bar{z}^4 + (\bar{y}^2 - 3\bar{z}^2)\bar{y}e] D_2 + [(\bar{y}^4 - 2\bar{y}^2\bar{z}^2 - 3\bar{z}^4)\bar{y} + (\bar{y}^4 - 6\bar{y}^2\bar{z}^2 + \bar{z}^4)e] E_2 \right] \\ &\quad - \frac{1}{(\bar{y} - e)^2 + \bar{z}^2} \left[[(\bar{y}^2 + \bar{z}^2)\bar{y} - (\bar{y}^2 - \bar{z}^2)e] A_2 + (\bar{y}^2 + \bar{z}^2 - \bar{y}e) B_2 + (\bar{y} - e) C_2 \right. \\ &\quad \left. + [\bar{y}^4 - \bar{z}^4 - (\bar{y}^2 - 3\bar{z}^2)\bar{y}e] D_2 + [(\bar{y}^4 - 2\bar{y}^2\bar{z}^2 - 3\bar{z}^4)\bar{y} - (\bar{y}^4 - 6\bar{y}^2\bar{z}^2 + \bar{z}^4)e] E_2 \right] \\ &\quad \left. + \bar{z}^2 \log \left[\frac{(\bar{y} - e)^2 + \bar{z}^2}{(\bar{y} + e)^2 + \bar{z}^2} \right] D_2 + 4\bar{z}^2 \left[e + \bar{y} \log \left[\frac{(\bar{y} - e)^2 + \bar{z}^2}{(\bar{y} + e)^2 + \bar{z}^2} \right] \right] E_2 \right\} \end{aligned}$$

A more accurate expression for $D_{rs}^{(2)}$ is obtained using 3.36, yielding

$$\begin{aligned} D_{rs}^{(2)} &= \frac{e \Delta \xi_s}{8\pi(\bar{y}^2 + \bar{z}^2 - e^2)} \left\{ \frac{1}{[(\bar{y} + e)^2 + \bar{z}^2][(\bar{y} - e)^2 + \bar{z}^2]} \right. \\ &\quad \times [2(\bar{y}^2 + \bar{z}^2 + e^2)(e^2 A_2 + C_2) + 4\bar{y}e^2 B_2 + 2\bar{y}(\bar{y}^4 - 2e^2\bar{y}^2 + 2\bar{y}^2\bar{z}^2 + 3e^4 + 2e^2\bar{z}^2 + \bar{z}^4) D_2 \\ &\quad + 2(3\bar{y}^6 - 7e^2\bar{y}^4 + 5\bar{y}^4\bar{z}^2 + 6e^4\bar{y}^2 + 6e^2\bar{y}^2\bar{z}^2 - 3e^2\bar{z}^4 - \bar{z}^6 + \bar{y}^2\bar{z}^4 - 2e^4\bar{z}^2) E_2] \\ &\quad - \frac{\delta_1 \alpha + \Delta}{\bar{z}^2} [(\bar{y}^2 + \bar{z}^2) A_2 + \bar{y} B_2 + C_2 + \bar{y}(\bar{y}^2 + 3\bar{z}^2) D_2 + (\bar{y}^4 + 6\bar{y}^2\bar{z}^2 - 3\bar{z}^4) E_2] \Big\} \\ &\quad + \frac{\Delta \xi_s}{8\pi} \left\{ \frac{D_2}{2} \log \left[\frac{(\bar{y} - e)^2 + \bar{z}^2}{(\bar{y} + e)^2 + \bar{z}^2} \right] + 2 \left[e + \bar{y} \log \left[\frac{(\bar{y} - e)^2 + \bar{z}^2}{(\bar{y} + e)^2 + \bar{z}^2} \right] \right] E_2 \right\} \end{aligned}$$

with the parameters α , δ_1 , δ_2 and Δ as defined in Section 3.3. This equation is to be used when $|(\bar{y}^2 + \bar{z}^2 - e^2)/2e\bar{z}| > 0.1$.

The procedure is analogous to that described in Section 3.3 for the parabolic fit: the coefficients A_1 , B_1 , C_1 , D_1 , E_1 and A_2 , B_2 , C_2 , D_2 , E_2 are computed by evaluating the kernel numerator at five points, evenly spaced along the doublet segment: $\bar{\eta} = -e$ (inboard), $\bar{\eta} = -e/2$ (mid-inboard), $\bar{\eta} = 0$ (midspan), $\bar{\eta} = +e/2$ (mid-outboard) and $\bar{\eta} = +e$ (outboard).

C Input and Output Files

The program relies on configuration files for the automatic setup and execution of the Doublet-Lattice Method calculations. These must be located in the `files` directory of the program files, and are expected to follow a certain format. The beginning and ending of the files is delimited by a series of '='. The definitions of parts and modes are contained in sections, which are delimited by by a series of '-'. Comments can be added using the '#' symbol and are ignored when reading the files, as are empty lines.

Upon execution from the command-line, the name (without file extension) of the `.set` settings file must be provided as an input parameter using the command-line argument `-input` or `-in`. This file is expected to contain the following entries (see C.4 below):

```
run_dlm      Set to FALSE or 0 to disable the calculation of the pressure
             distribution by the Doublet-Lattice Method.
autopanels_c Parameter for automatic paneling routine: number of chord-
               wise panels on main wing. Set to FALSE or 0 to disable automatic paneling.
higher_order Set to TRUE or 1 to use the quadratic approximation of the
             kernel numerator described in Appendix B.
```

The remaining keys `settings`, `oper`, `parts`, `coord`, `CPACS` and `grid_dlm` are used to define the paths to the other input files, relative to the `files` directory.

If not using the CPACS definition presented in Section 4.3, it is also possible for the user to define the individual parts constituting the aircraft geometry using a `.parts` or `.coord` configuration file, of which examples are shown in C.1 and C.2. In either file, the aircraft entry in the header serves to set the name of the model.

In the `.parts` file, the geometric parameters of Figure 4.4 must be entered for each part. If using a `.coord` file, the user must provide the absolute x , y and z -coordinates of the in-board leading-edge point IBLE, the out-board leading-edge point OBLE, the out-board trailing-edge point OBTE, and the in-board trailing-edge point IBTE of Figure 4.4 instead.

In addition, the user may provide a name for each part using the `part` key - if it is not provided, a unique name is generated. The `wing` key is used to set the name of the wing each part belongs to, so that parts may be grouped.

Finally, the number of chord-wise divisions `panels_c` and the number of span-wise divisions `panels_c` can be defined manually for each part (this is done programmatically if valid integer is entered for the `autopanels_c` entry of the settings file).

Operating conditions are specified in the `.oper` configuration file. A set of operating conditions may be named using the `case` key. Expected entries are `mach` for the Mach number, `freq_red` for the reduced frequency, and `ref_length` for the reference length.

Polynomial modes are defined, much like aircraft parts, in a separate section for each mode. The coefficients and exponents in x , y and z for each term of the polynomial are given as elements of a list, separated by commas and delimited by square brackets. The i -th element of each list corresponds to the exponent or coefficient for the i -th term in the polynomial. Of course then, the lists must be of identical length (if this is not the case, the missing elements are padded with 0s).

C.1 Aircraft Geometry Configuration File

```

aircraft      = VAMP_D150
=====
part          = D150_VAMP_W1_shape_PART00
wing          = D150_VAMP_W1_shape

offset_x      = +1.2745600e+01
offset_y      = +0.0000000e+00
offset_z      = -1.1362800e+00
chord_ib      = +6.0720000e+00
chord_ob      = +6.0720000e+00
span          = +1.8562300e+00
sweep         = +0.0000000e+00
dihedral      = +0.0000000e+00
symmetry      = 2
#   geometric properties

panels_c      = 5
panels_s      = 2
#   geometry discretisation settings (DLM)

-----
...
-----

part          = D150_VAMP_SL1_shape_PART00
wing          = D150_VAMP_SL1_shape

offset_x      = +2.9838000e+01
offset_y      = -2.0029200e-03
offset_z      = +1.8910100e+00
chord_ib      = +5.4307000e+00
chord_ob      = +1.8953000e+00
span          = +5.7371350e+00
sweep         = +4.2598224e+01
dihedral      = +9.0000000e+01
symmetry      = 0
#   geometric properties

panels_c      = 5
panels_s      = 8
#   geometry discretisation settings (DLM)

=====
```

C.2 Aircraft Geometry Coordinate File

```

aircraft      = VAMP_D150
=====
part          = D150_VAMP_W1_shape_PART00
wing          = D150_VAMP_W1_shape

ible_x        = +1.2745600e+01
ible_y        = +0.0000000e+00
ible_z        = -1.1362800e+00

oble_x        = +1.2745600e+01
oble_y        = +1.8562300e+00
oble_z        = -1.1362800e+00

obte_x        = +1.8817600e+01
obte_y        = +1.8562300e+00
obte_z        = -1.3483200e+00

ibte_x        = +1.8817600e+01
ibte_y        = +0.0000000e+00
ibte_z        = -1.3483200e+00

#   coordinates

symmetry     = 2
panels_c      = 5
panels_s      = 2
#   geometry discretisation settings (DLM)
-----
...
-----
part          = D150_VAMP_SL1_shape_PART00
wing          = D150_VAMP_SL1_shape

ible_x        = +2.9838000e+01
ible_y        = -2.0029200e-03
ible_z        = +1.8910100e+00

oble_x        = +3.5050800e+01
oble_y        = -2.0029200e-03
oble_z        = +7.5602400e+00

obte_x        = +3.6946100e+01
obte_y        = -2.0029200e-03
obte_z        = +7.4874300e+00

ibte_x        = +3.5268700e+01
ibte_y        = -2.0029200e-03
ibte_z        = +1.6823900e+00

#   coordinates

symmetry     = 0
panels_c      = 5
panels_s      = 8
#   geometry discretisation settings (DLM)
=====
```

C.3 Operating Conditions File

```
case      = EXAMPLE
=====
# aerodynamic operating conditions
mach      = +6.000000e-01
freq_red = +1.000000e-02

# reference dimensions
ref_length = +10.000000e+00
-----
mode      = MODE01
coeff     = [-1.0]
exp_x    = [0.0]
exp_y    = [0.0]
exp_z    = [0.0]

wing      = None
-----
...
-----
mode      = MODExx
coeff     = [0.5, 0.5, 0.25]
exp_x    = [0.0, 1.0, 1.0]
exp_y    = [1.0, 0.0, 1.0]
exp_z    = [0.0, 0.0, 0.0]

wing      = MAIN WING
=====
```

C.4 Settings File

```
===== =====  
run_dlm      =           1  
#   execution settings  
  
autopanels_c =           6  
higher_order =           1  
#   doublet-lattice method  
  
settings     = example_settings.txt  
oper        =      NONE  
parts       =      NONE  
coord        = example_coords.txt  
CPACS        = CPACS/D150_AIGLE_Hangar.xml  
grid_dlm    = example_lattice.dat  
#   file locations  
===== =====
```