Advancements in Vortex Lattice Method for Improved Low-order Supersonic Aerodynamics

Hemant Joshi

Aerospace Engineering
School of Physics, Engineering and Computer Science

Submitted to the University of Hertfordshire in partial fulfilment of the requirement of the degree of Doctor of Philosophy

December 2024

I would like to dedicate this thesis to my loving parents and grandparents …

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Hemant Joshi
December 2024

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Dr. Peter Thomas, for his exceptional support and guidance throughout my PhD journey. His vast knowledge, insightful feedback, and unwavering commitment to my research have been instrumental in shaping the outcome of this thesis. Dr. Thomas's expertise and mentorship have challenged me to push the boundaries of my abilities and have played a pivotal role in my personal and academic growth.

I am also immensely grateful to the University of Hertfordshire for providing the necessary resources and facilities to conduct my research. The university's support, both in terms of laboratory facilities and financial assistance, has been invaluable in carrying out my research. I am particularly grateful for the encouragement I received to publish my findings in reputable journals and participate in international conferences to showcase my research on a global scale.

I want to extend my thanks to all the faculty members and staff at the university who have contributed to my academic journey. Their dedication to fostering a stimulating and supportive research environment has greatly enhanced my learning experience and made this thesis possible.

Additionally, I would like to thank my family and friends for their unwavering belief in my abilities and their continuous encouragement and understanding throughout this challenging endeavour. Their love, support, and motivation have been my driving force during the ups and downs of this journey.

Thank you all.
Hemant

Abstract

This research presents advancements to the Vortex Lattice Method (VLM), which transforms it into a more robust and efficient tool for supersonic aerodynamic analysis and the preliminary conceptual design of next-generation aircrafts. This thesis enhances the Vortex Lattice Method (VLM) for supersonic analysis through three contributions. First, an unsteady VLM (UVLM) with adaptive time-stepping is implemented and validated, achieving approximately 82% agreement with conventional CFD while being about 24× faster. Second, a hybrid supersonic extension is developed by coupling VLM with Taylor–Maccoll shock modelling (TMHM) to account for Mach cone and shock-cone physics. This approach yields lift and drag predictions with 90–95% accuracy across configurations including delta and highly swept wings. Third, a compressibility correction using a multiphase Lattice Boltzmann Method (LBM) is integrated, showing 92% agreement with k–ω SST RANS reference solutions, while retaining the low computational cost of VLM. These improvements were implemented using MATLAB subroutines developed within the Tornado framework to enhance the prediction of aerodynamic characteristics under complex flow conditions.

End-to-end case studies on the SCALOS configuration and the Concorde wing demonstrate that the framework bridges the gap between conventional low-order tools and high-fidelity CFD: cases requiring >68 h with URANS were completed in 15 h with the hybrid UVLM framework. These findings highlight the viability of combining VLM, Mach-cone modelling, and LBM corrections as a rapid preliminary-design methodology for supersonic aerodynamics. The advances made provide a strong foundation for future developments in aerodynamic modelling, contributing to reduced development costs and improved performance in the design of next-generation supersonic vehicles.

Table of contents

List of figures

List of tables

Nomenclature

Greek Symbols

α	Angle of Attack
α_x	Angle of attack position in x
α_y	Angle of attack position in y
β	Shock angle
Γ	Bound Circulation of an aerofoil
γ	Specific heat ratio
Γ_l	Relaxation factor
μ	Prandtl Meyer function
ω	Distribution function between lattice
ω	Frequency of oscillation
ψi	Correction term for thermodynamic distribution function
ρ	Density
τ	Relaxation time factor
θ	Machcone angle
θ_s	Half cone angle
ξ	Molecular velocity of the particle

Other Symbols

C_d Coefficient of drag

C_L Coefficient of Lift

C_p Coefficient of pressure

$C_{L\alpha}$ Lift curve slope with α

P_b Pressure acting on the base cone

P_c Pressure acting on the shock cone

T_r Reference temperature

U Speed of flow

u Radial velocity on the Machcone

u_f Free stream Velocity

v Velocity component in the direction of cone-angle

a Local speed of the sound

c Chord length of an aerofoil

dt Time step

i Time index during navigation

l Length of the shock

M Free-stream Mach number

Acronyms / Abbreviations

AICM Aerodynamic Influence Coefficient Matrix

ALU Arithmetic Logic Unit

BEM Boundary Element Method

BGK Bhatnagar Gross Krook function

CFD Computational Fluid Dynamics

DDF Double distribution function in LBM

DEM Discrete Element Method

DLM Doublet Lattice Method

DVBE Discrete Velocity Boltzmann Equation

DVM Discrete vortex method

FEM Finite Element Method

FVM Finite Volume Method

GPU Graphics Processing Unit

HLBM Hybrid Lattice Boltzmann Method

HSCT High-speed commercial transport

IFM Indicial Function Method

ISA International Standards of Atmosphere

LBM Lattice Boltzmann Method

LES Large Eddy Simulation

LOM Low order modelling

MAC Mean aerodynamic chord

MDO Multi-phase design optimisation

NASA National Aeronautics and Space Administration

NLUVLM Non-Linear UVLM

PDE Partial Differential Equation

PFEM Particle Finite Element Method

PG Prandtl-Glauret

RAM Random access memory

RANS Reynolds-averaged Navier–Stokes

RLBFS Rotated lattice Boltzmann flux solver

SCALOS Supersonic configuration at low speeds

SST Supersonic Transport

SVLM Supersonic vortex lattice method

TMHM Taylor Maccoll Hypervelocity Method

TM Taylor Maccoll

URANS Unsteady Reynolds Average Navier Stokes

UVLM Unsteady Vortex Lattice Method

VLM Vortex Lattice Method

VPM Vortex Particle Method

Chapter 1

Introduction

1.1 Framework, aims and objectives

Supersonic travel has long been a subject of fascination and ambition. Yet, its realisation has been hindered by limited research on the design parameters needed to understand the complex aerodynamic characteristics of supersonic flows fully. The accurate and efficient analysis of supersonic flows, considering compressible, steady and unsteady conditions, is crucial for optimizing the design of supersonic aircraft. In this context, the vortex lattice method (VLM) has emerged as a promising low-order modelling approach, offering fast and reliable results in aerodynamic analysis and design optimization. This PhD thesis focuses on the development and enhancement of a VLM solver called TORNADO for studying supersonic aircraft. The initial adaptation of the vortex lattice method from its traditional subsonic applications to the supersonic flow regime is achieved by modifying an existing MATLAB code, 'TORNADO'. To validate the effectiveness of the improved vortex lattice method, the solver is applied to different wing models and shapes, comparing the obtained results with existing data from reliable sources. Furthermore, this research explores the application of the unsteady vortex lattice method to investigate the aerodynamic properties of incompressible unsteady flows, offering a comprehensive understanding of complex supersonic flow phenomena. The research objectives revolve around solving the problems associated with the influence matrix and the computation of vortex strength about the Machcone. Resolving these issues is essential to reduce computed errors and enhance the overall accuracy of the Vortex lattice solver. Additionally, the inclusion of Mach cone influence allows for a better understanding of the effects of shockwaves on geometry and surface pressure during the transition from transonic to supersonic speeds. Through these efforts, this research aims to contribute to the advancement of supersonic aircraft design and optimization, paving the way for efficient and high-performance supersonic travel.

This research aim is to study the potential impact of low-order modelling for supersonic aerodynamic analysis. With improving on the potential capabilities of well-explored vortex lattice method with modern modelling techniques and advantages of computational advancements. This work is achieved through the following objectives:

- Improving VLM's capability to model and analyse the dynamic behaviour of lifting surfaces with frequency and time response analysis.
- Improving on the limitation of VLM supersonic in modelling the shockwave boundary layer around surfaces undergoing the shockcone during the transition, this work is achieved by modelling the machcone study at a higher Mach number using the Taylor Maccoll hypervelocity method.
- For improvements in supersonic modelling using VLM, work is done on the limitation of compressibility correction for supersonic flow with VLM using the compressibility correction factor from using the Lattice Boltzmann method which models the boundary conditions with finer details for supersonic flow which accounts for the boundary viscosity change and the phase change of air at high speeds.

1.2 Rationale and Research Hypothesis

Modern supersonic conceptual design requires methods that balance speed and predictive fidelity. Conventional VLM and LBM are inherently subsonic and incompressible, yet through targeted extensions both can be adapted to treat supersonic flows. This thesis hypothesises that by augmenting VLM with (i) shock physics from the Taylor–Maccoll framework and (ii) density/compressibility corrections from LBM, a low-order methodology can achieve accuracy approaching Euler/RANS solvers but at a fraction of the cost.

High-fidelity Euler and Navier–Stokes solvers were not chosen as the primary tool because their computational expense prevents wide parametric sweeps at the conceptual design stage. In this context, the hybrid low-order approach is intended as a complement: RANS is still retained for validation and final verification, while the proposed framework enables rapid exploration of configurations.

Simulating complex supersonic flows poses significant challenges primarily due to high computational costs and demanding supercomputer requirements. These simulations involve intricate interactions of shockwaves, boundary layers, and compressible flows, necessitating advanced numerical methods and extensive computational resources. The high cost and computing power required stem from the need to resolve fine details of flow physics

accurately, such as shock formation, propagation, and interaction dynamics with aircraft surfaces. Such challenges are briefly discussed as follows:

- Firstly, unsteady modelling in supersonic design involves accurately predicting transient aerodynamic behaviours such as dynamic stall, flutter, and vortex shedding. These phenomena are critical for ensuring stable and safe flight conditions but require sophisticated numerical methods due to the complex interactions between airflow and wing structures.

- Secondly, shockwave modelling is crucial for supersonic aircraft as shockwaves significantly affect aerodynamic performance and structural integrity. Properly capturing shockwave dynamics, including their formation, propagation, and interaction with the aircraft surface, is essential for optimizing design and minimizing drag and other adverse effects.

- Thirdly, compressible correction factors are necessary to accurately model the behaviour of supersonic flows, particularly in regions where multiple phases or complex interactions occur post-shock. These corrections account for changes in density, pressure, and temperature across shockwaves and ensure that simulations accurately reflect real-world conditions, enhancing the reliability of aerodynamic predictions.

Addressing these challenges involves integrating complex low-order modelling techniques and hybrid approaches that combine low-order models with reduced-order complex equations. This hybrid strategy aims to leverage the efficiency of low-order models while incorporating the detailed physics captured by reduced-order complex equations. By complementing each other's strengths, this approach enhances the capability to predict intricate supersonic flow behaviours more accurately. Ultimately, this integration reduces the computational time required for simulations while maintaining reasonable accuracy, making it suitable and efficient for the preliminary stages of aircraft design.

The advancement of low-order modelling solver TORNADO stands out as a powerful tool for aerodynamic analysis, offering Low-fidelity simulations and versatile capabilities. However, like any software tool, TORNADO has its limitations, particularly in predicting the unsteady behaviour of supersonic lifting surfaces at high speeds with complex flow analysis such as shockwave modelling and supersonic flows. This chapter presents a novel approach to address the limitations of the current version of TORNADO by introducing new modelling techniques aimed at enhancing its capabilities. The following research gaps were highlighted in this thesis to achieve improvements in the overall enhancing capability of the solver for supersonic aerodynamic design analysis.

- Dynamic Stall and Unsteady Effects: TORNADO is primarily a steady-state analysis method and struggles to capture unsteady aerodynamic phenomena such as dynamic stall, wing-rock, or vortex shedding. Enhancing the solver to account for unsteady effects would enable better prediction of aircraft behaviour during dynamic manoeuvres or in turbulent flow conditions.

- The TORNADO framework does not incorporate viscous effects or boundary layer interactions due to its reliance on the conventional Vortex Lattice Method (VLM), which is primarily designed for low-speed flow regimes. While it utilizes the Prandtl-Meyer function, this approach lacks the necessary accuracy for predicting aerodynamic forces and moments in high-speed or high-lift configurations. Developing advanced methodologies to integrate boundary layer modelling into TORNADO would significantly enhance its applicability across a broader spectrum of flight conditions.

The research work focuses on the development of subroutines within the TORNADO framework to model the dynamic analysis of the solver using an adaptive time-step response method. Central to this research is the introduction of a new solver capable of predicting the drag of supersonic lifting surfaces at high speeds, a task previously beyond the capabilities of TORNADO. Unlike conventional vortex lattice method (VLM) meshes, the novel Machcone modelling technique is employed, drawing inspiration from the Taylor Maccoll hypervelocity equation framework. This approach allows for the prediction of shockwave interactions over flat surfaces by assuming the generation of a Machcone with a certain shock angle. The correction term for surface drag is calculated by incorporating changes in velocity potential horizontal and vertical to the shockcone, thereby significantly improving the drag prediction capabilities of the solver. Furthermore, the TORNADO framework is utilized as the initialization tool for geometry generation and to define the flight state model. The introduction of the LBM is employed to incorporate compressibility correction terms for improved supersonic and hypersonic speeds. The use of both two-dimensional (D2Q9) and three-dimensional (D3Q27) lattice configurations allows for comprehensive analysis. Incorporating the no-slip and bounce-back boundary conditions over defined obstacles facilitates the identification of changes in fluid flow viscosity at each time step, essential for predicting correction terms.

By integrating these advancements into the TORNADO framework, this research aims to significantly enhance its capabilities in predicting the unsteady behaviour of supersonic lifting surfaces, thereby contributing to the advancement of aerospace design and analysis methodologies.

1.3 Dissertation Outline

- **Chapter 2**, the thesis begins with a literature survey which aims to provide detailed knowledge on the potential use of VLM for supersonic analysis. Potential research gaps, their limitations and areas for improvements are reviewed
- **Chapter 3**, providing the introduction to the improvements made in the solver. Covering in-depth discussion on methods, and modelling techniques within the framework of VLM using TORNADO as the tool for integration.
- **Chapter 4** provides the unsteady modelling development on using time response and frequency response to model the oscillating motion in the lifting surface. The framework builds on the two-dimensional analysis and three-dimensional dynamic modelling capabilities of conventional VLM.
- **Chapter 5** provides the modelling development on the use of Taylor Maccoll with VLM for Machcone modelling within the supersonic domain for improved drag prediction. This integration is achieved using the panel matching technique which is explored in this chapter with evidence of its validation with conventional CFD techniques for both two-dimensional and three-dimensional shape study.
- **Chapter 6** introduces the compressibility correction for supersonic flow using the D2Q9 lattice with Lattice Boltzmann Method (LBM). The modelling approach uses the complex mathematical modelling use of LBM for boundary conditions and collision factors in generating the change in parameters for multi-phase flows for improved correction factor of velocity and viscosity for the domain.
- **Chapter 7** Introduces the synthesis of improvements made with the choice of two case studies performed based on the wing configuration such as delta wing (Concorde) and canard wing (SCALOS) demonstrating how collectively these improvements contribute to more robust supersonic analysis and how it validates against conventional CFD RANS solver. Discussion is given on the potential benefits of this approach, both computationally and financially.
- **Chapter 8** Provides conclusion to this research, providing highlights from primary contributions made in conventional VLM with added discussion on future prospects, room for improvements and delivery of this approach towards a sustainable future analysis tool.

1.4 Publications from this research

The following peer–reviewed journal articles and conference papers were produced as a direct outcome of this doctoral research. They report key methods, validations, and applications of the work presented in this thesis.

1.4.1 Journal Articles

1. **Hemant Joshi**, et al. *[Review of vortex lattice method for supersonic aircraft design] The Aeronautical Journal*, 2023. 10.1017/aer.2023.25

2. **Hemant Joshi**, et al. *[Improved vortex lattice method for drag prediction of supersonic wings using shock cone modelling] Engineering with Computers*, 2025. 10.1007/s00366-025-02192-2

3. **Hemant Joshi**, et al. *Improvements in Vortex Lattice Method for Design Analysis of Supersonic Wings: Verification and Validation CEAS Aeronautical Journal*, under review.

1.4.2 Conference Papers

1. **Hemant Joshi**, et al. *[Multiphase Compressibility Correction in Supersonic Vortex Lattice Method Using Lattice Boltzmann Approach]* In *Proceedings of the IEEE Aerospace Conference*, 2025. 10.1109/AERO63441.2025.11068430

2. **Hemant Joshi**, et al. *[Improved drag prediction using machcone influence in vortex lattice method for supersonic aircraft]* In *AIAA SciTech Forum*, 2024. 10.2514/6.2024-1879

3. **Hemant Joshi**, et al. *Unsteady Vortex Lattice Method in TORNADO: A Fast and Cost-Effective Solver for Dynamic Analysis of Supersonic Lifting Surfaces.* In *Aerospace Europe Conference 2023, Joint 10th EUCASS–9th CEAS Conference*, 2023. 10.13009/EUCASS2023-092

Chapter 2

Literature review

2.1 History

Much work was done in the development of various potential flow solver methods during the early 1900s to study the aerodynamic characteristics of a flying object. The first-ever implementation of a computer fluid dynamics model for fluid analysis was started with Richardson [28] which was then observed and carried by Prandtl [29] in formulating the lifting line theory. The first-ever integration of single load vortex was used by Prandtl which made him develop a new method in 1938 developing an explicit finite difference method for solving boundary layer equations [30]. Many other authors worked during the same time to improve the efficiency of these methods, such as Liepmann who showed improvement in the convergence rate for Richardson's method. This research played an important role in the development of the vortex lattice method which first appeared in 1937 [31]. This become popular and was used by other authors such as Mutterperl [32] and Wessinger [33] and shortly caught the attention of many mathematicians. Improvements were done and many different variations were tried to improve the accuracy and efficiency of the solver. Subsequent use and development of VLM are illustrated in Fig. 2.1. During the 1950's Falkner's version of VLM [34] was extensively used in industry. However, the calculation effects were so large that limited the use of panels and then accuracy become a concern for some configurations. This made practitioners realize that VLM had to await the availability of greater computational capability. [1]

The first validation of the vortex lattice method for supersonic flow was given by Miranda [1] whose work used the already developed version of VLM and improved its application

[1]The literature review is published in The Aeronautical Journal as a review paper: Joshi, H. and P. Thomas (2023). Review of vortex lattice method for supersonic aircraft design. Journal of the Royal Aeronautical Society, pp.1–35. https://doi.org/10.1017/aer.2023.25.

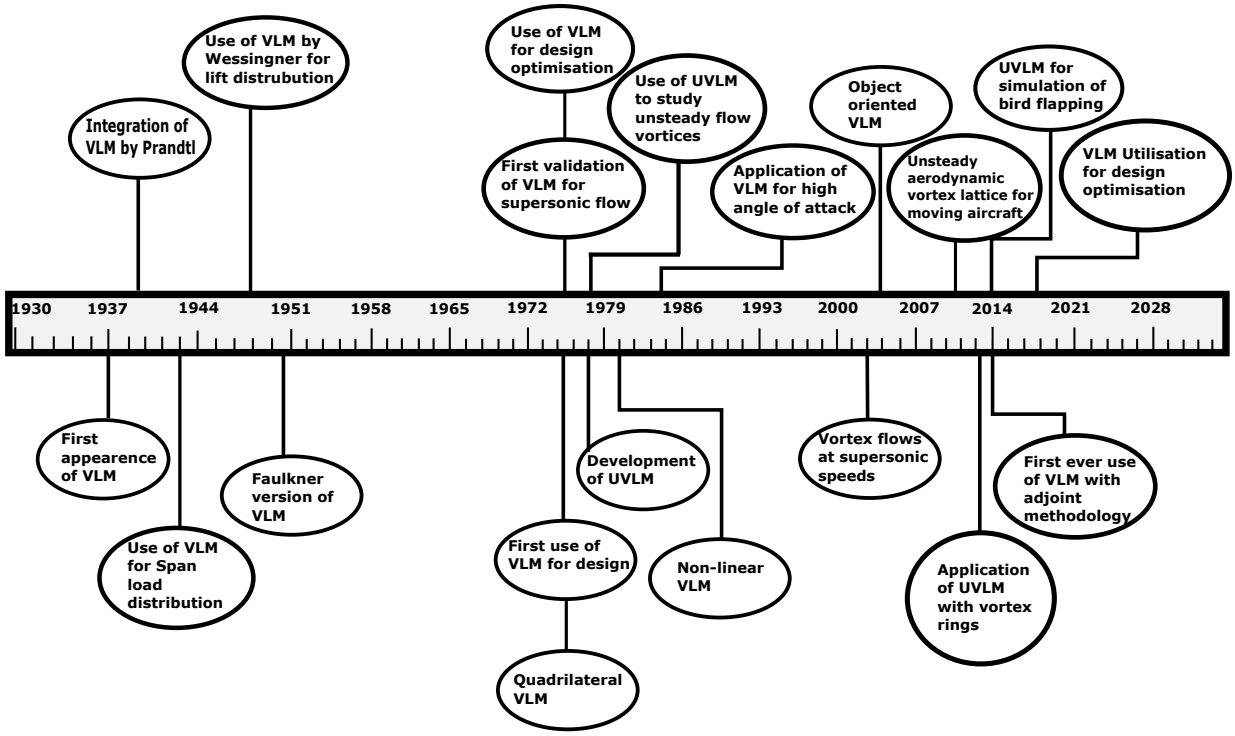


Fig. 2.1 Timeline of Vortex Lattice Method.

in studying the discrete vortices as an approximation for surface distributed vorticity whose integrals yield a residual term called 'vortex-induced velocity'. This renders the use of VLM for supersonic flow, with the developed version of VLM as shown in Fig. 2.2. NASA [35] made use of VLM for their transition from developing potential flow solvers for subsonic flow to supersonic flow which played a major role in the development of their supersonic transport (SST) mission. Further work carried out used special vortex lattice layouts, which allowed the simulation of thickness and volume with horseshoe vortices. Several versions and variations of VLM have proven to be very practical and versatile tools for aerodynamic analysis of planar and non-planar configurations. However, initially, most of the work of VLM was focused on subsonic flow.

In the early development of VLM, multiple variations were tried and tested for different purposes, out of which quadrilateral VLM was one of them. The quadrilateral panels on wing surfaces allow for easy and fast calculations, as illustrated in Fig. 2.3. The quadrilateral VLM was best designed to calculate the aerodynamic forces for high lift configurations. As an iterative method, it accounts for free wake and wake from the sides and deformation of

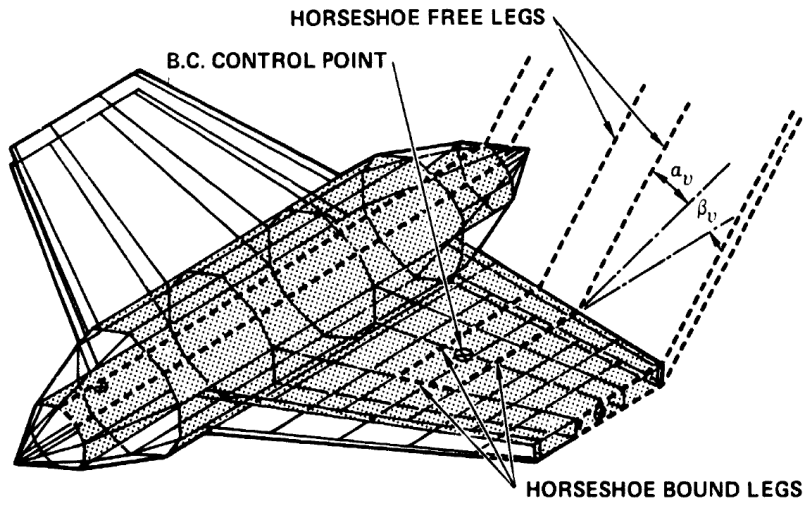


Fig. 2.2 Generalised Vortex lattice model of wing-body configuration [1].

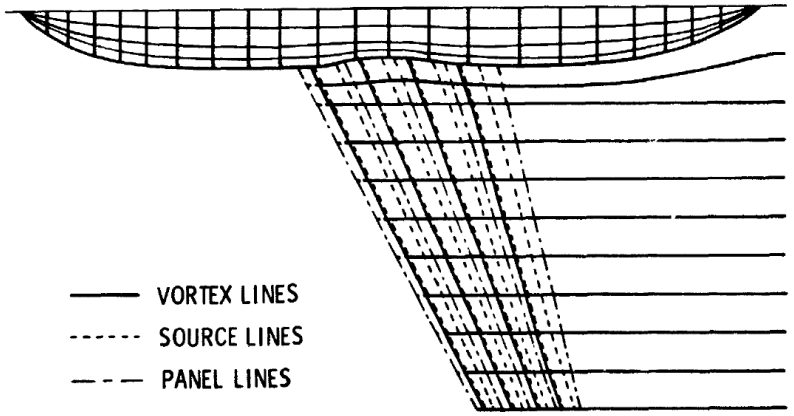


Fig. 2.3 Quadrilateral Vortex Panel, Source Panel [2]

forces. The application of VLM was also extended to be used for the design optimisation for various arbitrary wing designs and platforms. Feifel [13] used VLM for optimisation of three-dimensional arbitrary configurations with combined analysis, induced drag optimisation, and aerodynamic design purposes. NASA used VLM to study the minimum drag produced from minimum lifting surfaces [36].

A decade of use of VLM for many different conditions has opened the door to its flexibility and application for other problems besides just aerodynamic analysis. A glance at recent research papers from the last decade on the use of VLM sees its use in analysing bird flight in hover and wind turbine unsteady motions.

A new method was developed which is used to determine the mean camber surface for timed non-planar planforms with minimum vortex drag [36]. The newly proposed VLM outperforms all the previous methods and improves on overall limitations with chord loading specification. The method makes the use of Trefftz plane analysis [3] to determine the optimum span loading to calculate the minimum drag as shown in Fig. 2.4, which then predicts the mean camber surface of the wing.

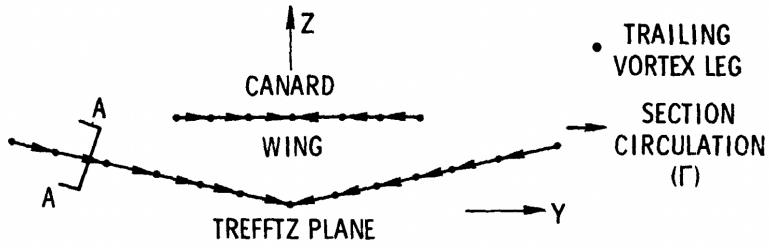


Fig. 2.4 Trefftz Plane Analysis [3].

2.2 Types of Vortex Lattice Method

The overall use of VLM in different applications in the past decade is summarised in Table 2.1, 2.2, and 2.3. It is evident that its use in the aerodynamic and design analysis of aircraft was prominent in the past but, there has been much less research available that illustrates and develops its use for aerodynamic analysis for design optimisation for supersonic aircraft.

2.2.1 Steady Vortex Lattice Method

The steady VLM has gained popularity as a computational tool, giving users and the industry a relatively simple tool to visualise and analyse the aerodynamic characteristics of a planar configuration for the desired flow condition. Visualisation and analyses tool enabled fast and informed design decisions and enhanced the speed at which modern aircraft could be designed from their initial conceptual phase. With this method, a configuration of lifting surfaces can be divided into a number of panels each of which contains an aerodynamic singularity distribution. A constant source distribution can be used to represent a body, and vortex distributions having a linear variation in stream-wise direction is used on the wing and the tail panels. The normal component of velocity induced at specified control points by each singularity distribution is calculated and makes up the coefficient of the system of linear equations relating the strengths of singularities to the magnitude of the normal

Table 2.1 Use of Vortex Lattice Methods in last decade (Linear).

Application	Year	Title	Author Name
Aerodynamic modeling	2021	High-fidelity aerodynamic modeling of an aircraft using OpenFoam- Application in CRJ700	M. Segui, F.R. Abel, R.M Botez and A. Ceruti [37]
Comparison between VLM and panel method	2020	Comparison of flow solvers: VLM and higher order panel method with analytical and wind tunnel data.	Tahura Shahid, Faiza Sajjad, Muneeb Ahsan and Sayed Farhan [38]
VLM for cross flow propeller	2019	Viscous VLM analysis of cross-flow propeller and turbines	Brenden P. Epps, Bernard T. Roesler, Richard B. Medvitz, Yeunun Choo and Jarlath McEntee [39]
Aerodynamic modeling for transonic aircraft	2016	Aerodynamic modeling of transonic aircraft using VLM coupled with transonic small disturbance for conceptual design	Daniel Chaparro, Gustaro E. C. Fujiwara, Nhan Nguyen and Eric Fing [40]
Membrane wing kits analysis	2015	Multiple wake VLM for membrane Wing Kits	Rachel Leuthold [41]
Helicopter tail rotor	2014	Review of Aerodynamic characteristics of helicopter tail rotor propeller using quasi-continuous VLM	Firdaus, Jaswar Koto, M.S. Ammoo, I.S. Ishak and Nofrizal [42]
Winglet performance	2014	Winglet performance evaluation through the vortex lattice method	Phil R. RadaeMacher [43]
Marine propeller	2013	A VLM for the prediction of the unsteady performance of marine propeller and current turbines	Lei He and Spyros A. Kinnas [44]
Sweep flight UAV	2013	Modeling with VLM and frequency sweep flight test for a fixed-wing UAV	Sanghyuk Park [45]
Floating wind turbine	2013	Unsteady aerodynamics of offshore floating wind turbine pitching motion using VLM	Mine Jeon, Seungmin Lee, Soogab Lee [46]
Aeroelastic analysis	2013	States aeroelastic analysis of very flexible wings based on non-planar VLM	Xie Changchuan, Wang Libo, Yang Chao, Liu Yi [47]
Marine propeller performance	2011	Proposal for improvement inprediction of marine propeller performance using VLM	Sung-Bu Suh [48]

Table 2.2 Use of Vortex Lattice Methods in last decade (Nonlinear).

Application	Year	Title	Author Name
Rotor noise prediction	2021	Prediction of noise from low Reynolds number rotors with different number of blades using an NLVLM	Yeongmin Jo, Thierry Jardin, Romain Gojon, Marc C. Jacob and Jean-Marc Moschetta [49]
Stall prediction	2019	NLVLM for stall prediction	Hasier Goitia, Raul Llamas [50]
Wake modeling	2018	Numerical investigation of the aerodynamic and wake structure of horizontal axis wind by using NLVLM	Hakjin Lee and Duck Joo Lee [51]
Morphing wings	2016	Analysis of the aerodynamic performance of a morphing wing tip demonstrator using a novel NLVLM	Oliviu, Sugar Gabor, Koreanschi, Andreea Botez, Ruxandra M. Mamou, Mahmoud Mebarki and Youssef [52]
Non-linear aeroelastic	2012	Non-linear aeroelastic framework based on VLM and Corotational shell finite element	Carlos Eduardo de Souza, Roberto Gil Annes da Silva and Carlos E.S Cesnik [53]

velocities. The singularity strength which satisfies the boundary condition of the tangential flow at the control points for a given Mach number and the angle of attack is determined by solving this system of equations using an iterative procedure. Once the singularity strengths are known, the pressure coefficients are calculated, and the forces and the moment acting on the configuration are determined by numerical integration. Early VLM programs were commonly written in FORTRAN IV [66].

Several versions and variations of VLM have proven to be very practical and versatile for aerodynamic analysis of planar and non-planar configurations. The success of this method is due to its simplicity of use of the numerical techniques involved and due to the high accuracy of the results obtained [67]. However, initially, most of the study with VLM was focused on subsonic flow. The use of steady VLM in studying the vortex flow under supersonic conditions was in trend during the early 20th century. The research was carried out at the NASA Langley research center (LaRC) for high-speed vortex flow [4]. Multiple test models were used such as flat-plate, cavities, bodies, missiles, and many other lifting surfaces at Mach 1.5-4.6. The presented data showcase the type of vortex structure at supersonic speeds and study their impact on the flow structures on vehicle performance and control. A clear presentation of vortex flows over a delta wing is represented in Fig. 4.5. Finally, with the

Table 2.3 Use of Vortex Lattice Methods in last decade (Unsteady).

Application	Year	Title	Author Name
wind turbine modeling	2022	An unsteady model of a horizontal axis wind turbine operating in an upstream rotor wake	D. Hankin and J.M.R Graham [54]
Rotor heat transfer	2020	Predicting rotor heat transfer using vortex blade element momentum theory and UVLM	Abdallah Samad, Gitsuzu B.S. Tagawa, Francois Morency, and Christophe Volat [55]
Reduced order modeling	2020	Parametric reduced order modeling of UVLM	Salvatore Maraniello [56]
Frequency domain solver	2018	Non-linear frequency domain solver for VLM	MAthieu Parentau and Eric laurendeau [57]
Force matrix for UVLM	2017	A model frequency domain generalised force matrix for the UVLM	G. Dimitriadis, N.F. Giannelis and G.A. Vio [58]
Unsteady lift prediction with stall	2017	Lift prediction including stall, using VLM with Kirchhoff based correction	Carlos R. dos Santos And Flairio D. Marques [59]
Unsteady drag prediction	2016	Induced drag calculations with UVLM for cambered wings	Thomas Lambert and Grigorios Dimitriadis [60]
UVLM for insect flapping wings	2016	Extended UVLM for insect flapping wings	Anh Tuan Nguyen, Joong Kwan Kim, Jeong Seob Han and Jae-hung han [61]
Rotor aircraft in hover	2015	Computational study of a transverse rotor aircraft in hover using the UVLM	Juan D. Colmenares, Omar D. Lopez, and Sergio Preidikman [62]
horizontal axis wind turbine	2014	UVLM coupled with a Linear aeroelastic model for horizontal axis wind turbine	Minu Jeon, Seungmin Lee and Soogab Lee [63]
Flapping wing in hover	2013	Modified UVLM to study the flapping wing in hover flight	Bruno A. Roccia, Sergio Preidikman and Julio C. Massa [64]
Flapping wing optimisation	2012	Analytical sensitivity analysis of an UVLM for flapping wing optimisation	Bret K. Stanford and Philip S. Beran [65]

help of the data obtained, it was safe to propose an idea for aerodynamic analysis for a design

approach for wings that uses the vortex flows for improved aerodynamic performance at supersonic speed.

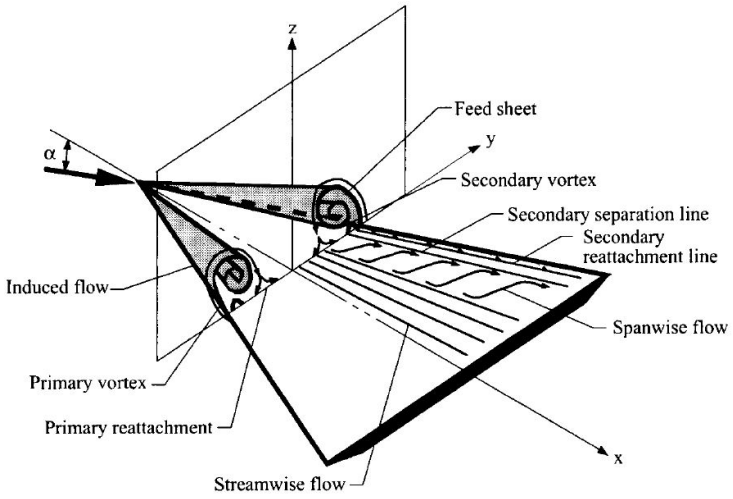


Fig. 2.5 Sketch of sharp leading-edge, wing-vortex flow on delta wing at angle of attack [4].

The quadrilateral VLM [2] with the integration of iterative wake relaxation procedure has been applied to numerous configurations with high circulation. The iterative procedure shows high convergence, and the calculations and results are in good relation with experimental data obtained. Trip-edge separation effects have been shown to be important aspects of high-lift calculations. The obtained results prove its applicability for multiple lifting surfaces with part span deflections and can include ground effect and wind tunnel interference. Robinson [68] worked with the hyperbolic characteristics of the differential equation satisfied by the velocity potential in linearised supersonic flow. This entailed the presence of fractional infinities in the fundamental solution of equations. Difficulties arising from this resulted in deriving the formulas for the field of flow with an arbitrary distribution of supersonic sources and vortices. A clear representation of smooth-wall vortex flow on a conical fore-body at the angle of attack is shown in Fig. 2.6.

Nayeh and Mook's work [69]-[70] proposed general VLM for application in unsteady flows. The method can treat unsteady flow over a finite number of lifting surfaces, fully accounting for interference. The described method is not limited by planform, twist, and maneuver if vortex bursting does not occur near any of the lifting surfaces and separation occurs only along the known lines. The results provide the position of various wakes and the distribution of circulation on every surface and in every wake. The versatility and strength of this method are demonstrated by its application for steady and unsteady flow. The steady flow example includes yawed delta wing and a cropped delta wing at a high angle of attack.

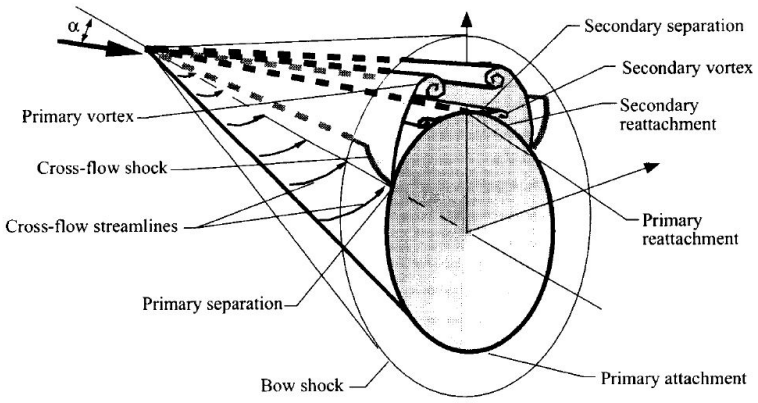


Fig. 2.6 Sketch of smooth-wall vortex flow on conical fore-body at angle of attack [4].

The unsteady flow includes a delta wing and a rectangular wing executing roll maneuvers and a rectangular wing with two flaps that oscillate periodically. Both steady and unsteady cases were discussed with promising results which satisfy the desired performance index [71].

2.2.2 Unsteady Vortex Lattice Method.

The major difference between unsteady and steady VLM is the inclusion of 3D wake modelling to account for the 3D interference during unsteady flow conditions. The transition of conventional VLM to UVLM is mainly driven by the viscous phenomenon, such as the need to study the leading-edge separation for delta wings. Unsteady characteristics of swept delta wings were being studied in the late 20th century for its improved control effectiveness [72]. The major contribution to the development of UVLM was done by Belotserkovskii [73], Rehbach[74], and researchers at Virginia Tech [75, 69] and Technion [76]. An alternative to UVLM is the doublet lattice method (DLM) which was already a successful unsteady potential flow solver written in the frequency domain. This allows the shape of a force-free wake to be obtained as part of the solution [77]. The doublet lattice method offers a fast way of computing methods with its linear methodology, but this restricts its use for small out-of-plane harmonic motion with a flat wake. The use of UVLM is increasingly popular where the free wake is important to consider because of more complex aircraft geometry. The first-ever use of this method was done by Konstadinopoulos et al. [69] which used a general method of calculating unsteady, incompressible, inviscid, 3D flows around arbitrary planforms. The method was an extension of the vortex lattice technique. The UVLM modeling of lifting surface and wake distribution using the vortex ring elements is

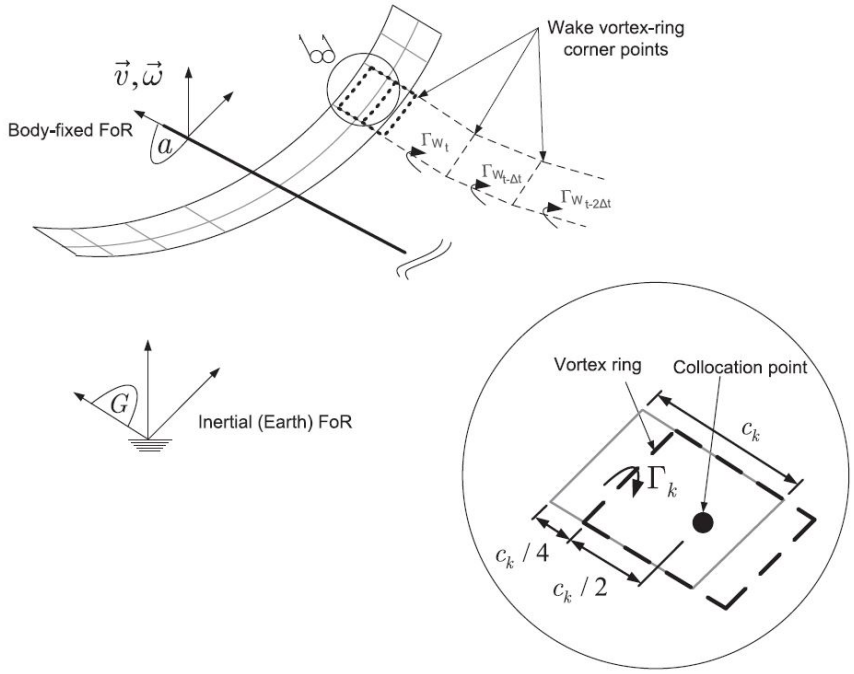


Fig. 2.7 Unsteady aerodynamics model lifting surface and wake discretization using vortex ring elements [5].

shown in Fig. 2.7. It was not limited by aspect ratio, camber, or angle of attack, if vortex breakdown does not occur above the surface of the wing and separation occurs only along the sharp edges. As the wing performs arbitrary maneuvers, the position of the wake and the distribution of circulation on the wing and in the wake are obtained as a function of time. One desirable feature of the present method is its ability to treat steady lifting flow very efficiently. In the early 1990s, a new time-domain unsteady aerodynamic technique was made by Verdon and Hall [78] to study the unsteady aerodynamic flows around 2D isolated aerofoil and, cascades of aerofoil and three-dimensional wings for small disturbances in the flow. The method was developed based on incompressible VLM. The eigenmodes of the system were computed for a given aerodynamic state and a handful of those eigenvalues were retained for the construction of reduced order models for arbitrary modes of an aerofoil motion. Soon after the development of UVLM and its popularity to study wake build-up for complex wing planforms, interest moved towards the unsteady aerodynamic optimisation of wings. Steady computational tools, such as CAESIOM [79], were already built and were in use but with the limited-to-no capability of modeling unsteady aerodynamics. A new potential flow solver called ABSOLUT, based on the theory of UVLM was developed with a time implicit function which means its solution method is concerned with the time domain, making its iterations time dependent [6].

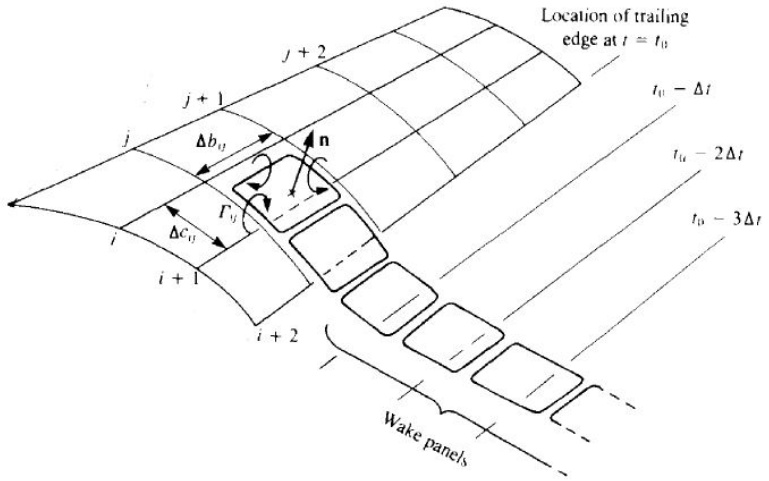


Fig. 2.8 Unsteady vortex lattice method for vortex rings [6].

Both TORNADO and ABSOLUT programs are written in the MATLAB environment with ABSOLUT making use of TORNADO's conventional steady VLM code. After the lifting surfaces are divided into panels the program starts its first iteration without any wake, but with the following iteration with new time steps including the vortex rings of the trailing edge beginning to shed at the rate of one row of rings per step as shown in Fig. 2.8. This is then followed by calculating the forces in the same way as with the classical VLM which means at first the linear system of equations is solved in order to obtain the circulations of vortex rings. Then the forces are calculated with the help of the Kutta-Joukowski [80] theorem which includes time dependency. ABSOLUT is more than capable of predicting the required aerodynamic coefficients for simple geometric configurations under unsteady conditions. A study has been made to include an extension of the elastic motion of a wing including the bending mode which provides promising results when compared to experimental results [6]. It has been checked on various flow conditions for steady computation of modern aircraft configurations, being shown to be as accurate as TORNADO. ABSOLUT provides similar results to wind tunnel experiments, especially in the case of calculation of longitudinal forces and moment solutions [6].

Another numerical method by Hernandes [81] is proposed for the unsteady solution of the aerodynamic coefficients of thin profiles in subsonic but also supersonic compressible flow. This model is made through the aerofoil discretization in uniform segments and the singularity used is a vortex incompressible flow. The method has been compared favorably with other methods from the literature. The new method is the natural extension of the

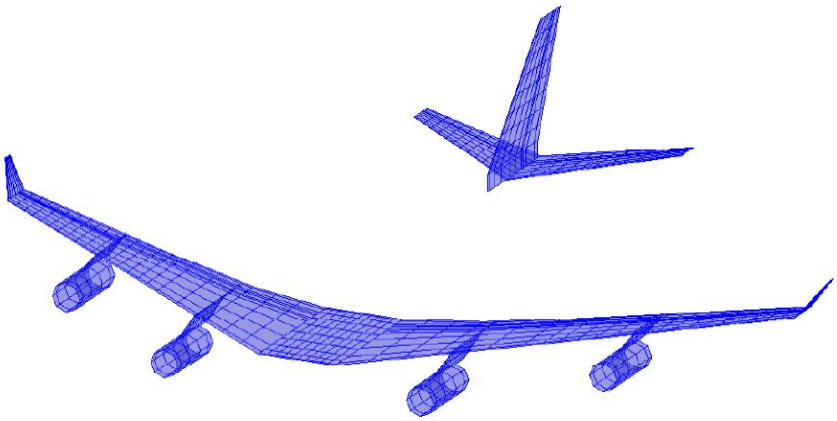


Fig. 2.9 DLM/VLM Panelling scheme [7].

compressible regime of the classical 2D VLM. The advantage of this method over others, specifically higher-order CFD methods, is that it is possible to calculate the loads and forces over the profile at a low computational cost. The proposed method is the first to use a simple numerical scheme to obtain forces and loads for an arbitrary motion of a profile in the unsteady compressible domain. To validate the method, a step change of the angle of attack (an indicial motion) was calculated. An arbitrary motion can be obtained using a superposition process or simply by changing the boundary conditions because it is a numerical scheme. Using this method, it is possible to obtain results for many important motions in applied aerodynamics, such as sharp-edged gust and aerofoil–vortex interaction, which is very important for the study of helicopter noise and cosine-gusts used in the development of civil aircraft.

Kier [7] addresses the development of aircraft models for flight loads analysis in the predesign stage. The underlying model structure consists of the non-linear equations of motion of a free-flying, flexible aircraft, which calculates the distributed aerodynamics over the entire airframe. Different possibilities in modelling the unsteady aerodynamic interaction for predesign purposes are explored and the effects on the loads are compared to access the trade-off between accuracy and speed. The methods compared and modelled were a quasi-steady VLM without any further unsteady improvements, an extended strip theory where unsteady effects are modelled by indicial functions such as Wagner's and K*ü*ssner's function, and a rational function approximation to Roger's method of the unsteady doublet lattice method.

An example of DLM and VLM paneling schemes is shown in Fig. 2.9. For the comparison of the loads, the aircraft was subjected to a longitudinal gust of the shape 1-cos, to excite the

flexible structure and unsteady aerodynamic effects. The result shows the importance of the gust tuning process, where the maximum load can occur at any gust gradient distance. The agreement of the total loads for long gradient distance was quite good, except for the fact that steady methods and the indicial function methods (IFM) lack the delay in downwash between the wing and the empennage. The VLM reacts slightly slower than the steady strip theory since the gust reaches each box control point with a respective delay. IFM takes into account the build-up of the lift and is the slowest of the three methods derived from steady aerodynamics. The purely steady methods such as VLM and strip theory shows increasing load levels as the gust gradient distances are reduced. The IFM shows the same sensitivities to the tuning process as the DLM [7] but underestimates the magnitude of the occurring loads. This study provides a statement that, with ease of use and modeling with fast simulation times, IFM is a highly suitable candidate method for predesign studies.

In Kier's analysis [7] unsteady aerodynamic coefficients of thin aerofoils in compressible subsonic and supersonic flow were studied. The lift, pitching moment, and pressure coefficient were calculated numerically for an indicial response (that is, a sudden change in the angle of attack); a thin profile penetrating the various gust profiles used in commercial aircraft design; oscillating aerofoils; the interaction of a convected vortex passing under the profile of a body (which is known as blade vortex interaction). That work utilised a numerical approach based on vortex singularities. The numerical model was created through the profile discretization in uniform segments and compressible flow vortex singularities were used. The results showed the compressibility effects of these problems, as well as how these motions are affected by the propagation velocity of the gust or vortex. Aerofoil–vortex interaction is the fundamental basis of studies on noise reduction of helicopters; the work done by Kier [7] is necessary for the development of more silent rotors. The numerical method used was, in fact, a natural extension of the compressible regime of the classical VLM in its two-dimensional version; it is the first methodology to use a simple numerical scheme to obtain forces and loads for an arbitrary motion of a profile in the unsteady compressible domain. The main physical difference is the finiteness of the disturbance propagation. The correspondence between vortex and normal dipole panels with constant density, as well as the concepts of bound and free vortices, remains valid in both subsonic and supersonic regimes and is an essential feature of the numerical scheme. This method obtained results far faster if they had been obtained via CFD and a significant correspondence between the obtained values and the references was observed. Due to ease of use and ability to study wake with time discrepancies for both subsonic and supersonic flight, the researchers started looking for its integration in MATLAB. Soon this framework was in use for studying the unsteady aerodynamics of a moving aircraft in motion using UVLM. One such integration by Bueso [6] was to develop a

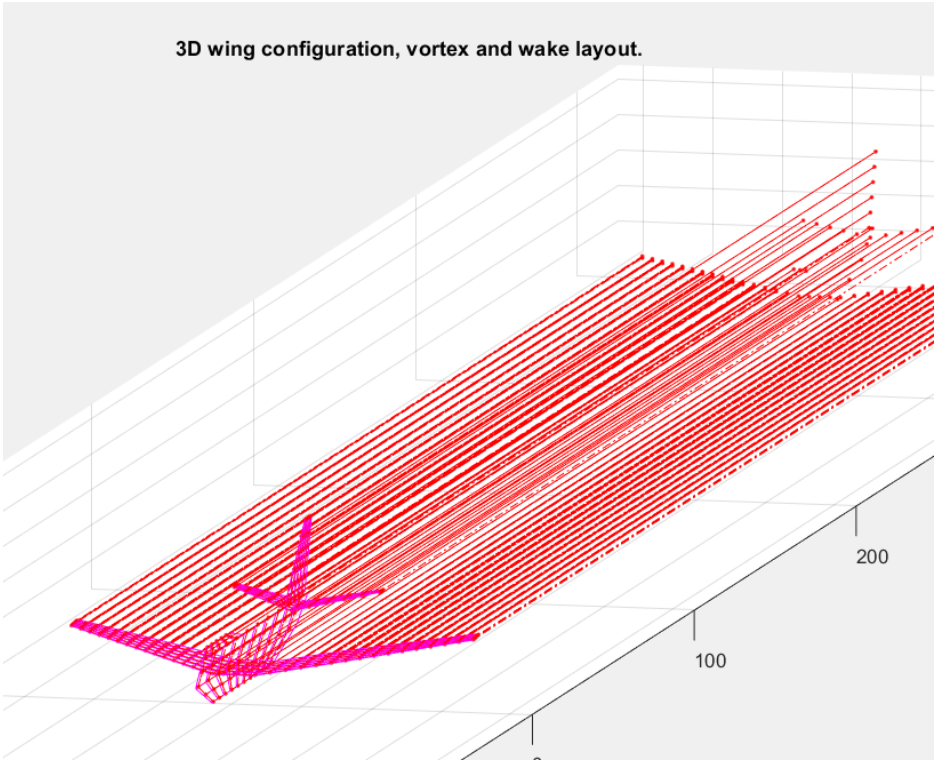


Fig. 2.10 3D Wing Configuration, wake and vortex layout in TORNADO [8].

new potential flow solver for studying unsteady aerodynamics in the MATLAB environment. To achieve this target, a VLM-based computer program was developed as shown in Fig. 2.10 . The validation of the program involves two different stages. Initially, it is compared with classical experiments and well-tested code. For a second analysis, the program is compared with wind tunnel experiments for two different aircraft configurations, classical and with the canard.

In contrast, the unsteady version has been largely overlooked in fixed-wing dynamic modelling and has been mostly exercised in other disciplines such as rotorcraft, wind turbines, or flapping wing analysis. UVLM have many applications at low speeds working on aeroelasticity and flight dynamics for various wing planforms. The UVLM is governed by Laplace's equation, which is linear and thus is constrained to the subsonic regime with negligible viscous effects. State space assembly obtained through the linearization of the equation proves extremely efficient and each point of the parametric space takes a few seconds to evaluate on a desktop computer. In addition, this formulation enables both static and dynamic stability analysis, enabling a straightforward implementation, and is ideally suited for advanced control synthesis methods. The prescribed wake assumptions provide an acceptable level of accuracy and insight before the closed-loop solution is then tested on the

non-linear solver. Finally, the state space formulation is also expected to constitute a valuable tool to address some of the pressing issues the industry is facing, such as multidisciplinary and uncertainty modelling.

2.2.3 Non-Linear Vortex Lattice Method

Non-linear aerodynamics are of great interest to the aerospace community. It not only plays a significant effect on the performance and stability of an aircraft in cruise and unsteady manoeuvring but also largely plays a vital role in aerodynamic and aeroelastic analysis for transonic speeds [82] and supersonic speeds. The transition from subsonic to supersonic is followed by complex flow interactions which can include shockwaves, separated flows and complex boundary layer interaction [83]. These non-linear interactions in play can alter the stability of a flying vehicle by the influence of complex aeroelastic behaviour [40]. In this case, the overall performance is greatly affected and can reduce the flight operation of a vehicle, making it important to understand the non-linear aerodynamic characteristics of aircraft. For the development of most modern-day aircraft, the design relies on complex aerodynamic features to produce vortices for a high-lift generation. These vortex interactions are also studied and used for maximum lift design at manoeuvring angles of attacks. The majority used canard-wing-tail configuration [84] which provides a surface that in combination with its control functions, can be used to optimise this vortex interaction. The McDonnell Douglas aircraft company (now part of Boeing) was one of the very few companies to study linear and non-linear aerodynamics for different configurations. This was done in collaboration with NASA [85].

The main difference between VLM and NLVLM is the shape of the horseshoe vortex element. In the case of VLM, the shape of a trailing vortex is a semi-infinite straight line starting from the endpoint of a bound vortex to the downstream. On the contrary, the shape of the trailing vortex in NLVLM is curved to align with the local streamline. To meet this requirement, each trailing vortex is broken into many vortex segments of finite length. The coordinates of vortex segments and the strength of the horseshoe vortices are to be determined simultaneously, and this requires an iterative solution procedure. After the development of the theory for non-linear aerodynamics and its use in the formulation of a non-linear version of VLM, NASA developed the computational system for non-linear characteristics of the wing at supersonic speed in the early 1980s. Much research has been carried out for transonic and supersonic speed to better understand the complexity of the non-linear behaviour of vortices under unsteady conditions. The already developed linearised theory could account for non-linearities in the variation of basic pressure loading with local surface slopes, could predict the degree of attainment of theoretical leading edge thrust forces, and provide an

estimate of detached leading edge vortex loading that results when theoretical thrust forces are not fully realised. The newly proposed method yielded promising results which gave significant improvement over the linearised theory for detailed pressure distribution at various angles of attack and for specific regions on the wing with 3D flow distribution. The method also offered precise, improved predictions of wing overall forces and moment coefficient. The more accurate prediction of the pitching moment, the more realistic estimate of the variation of drag with the camber surface which is vital information in the early stage of aircraft design.

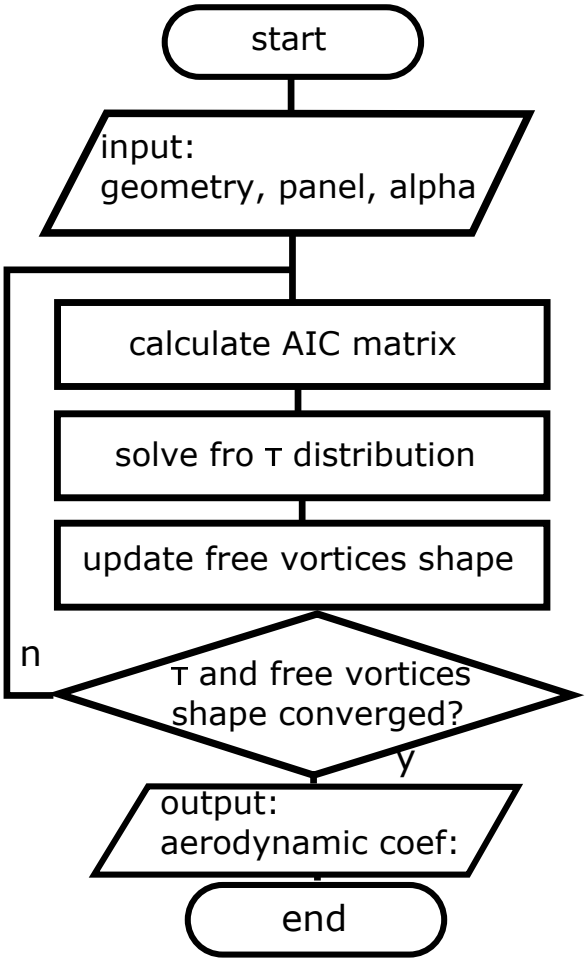


Fig. 2.11 Improved Non-Linear Vortex Lattice Method [9].

Not much work was carried forward after NASA's breakthrough research of NLVLM because this research was confined to flow conditions comprising mainly transonic and supersonic flow. Because such complexity of vortices distribution was experienced only with supersonic conditions which most civil aircraft do not encounter. However, increased interest in bringing commercial supersonic travel back provides a new impetus for research institutes

and industries to use this method or to make new variations and developments according to their requirement with modern wing analysis for supersonic flow. Despite the already developed program to study the non-linear aerodynamic characteristics there is still room for improvement. One of the limitations is associated with vortex segments affecting the convergence characteristics for NLVLM. One such work for the improvement of the iteration algorithm of NLVLM was carried out by Lee and Park [9]. Their paper mentions the rather old history of NLVLM and its limitations. The variables associated with vortex segments affect the convergence characteristics of NLVLM so studies on these variables concerning convergence have been performed in the past to minimize the instability of NLVLM. The conventional NLVLM uses feedback from the converged wake to calculate the aerodynamic influence coefficient (AIC) matrix, and feedback from the free vortices shape to update those. Whereas, the proposed method takes combined feedback from converged wake and free vortices shape as input for calculating the AIC matrix as shown in Fig. 2.11. When compared the new NLVLM approach yielded promising improvement in stability when compared to conventional NLVLM. The performance of the proposed method was demonstrated through example calculations for the rectangular and delta wing with a low-aspect-ratio.

Soon after the development of NLVLM its applications were used for the design optimisation of wings. A new formulation of the classical VLM approach for calculating the aerodynamic properties of lifting surfaces was worked out by Gabor et al. [86]. The developed method uses the effects of viscosity and yields a low computational cost. It represents a useful tool for performing rapid and accurate wing design and optimisation procedures. A two-dimensional viscous analysis of the wing was used in the formulation of the mathematical model. The viscous analysis and computed forces of the span-wise wing section, according to strip theory, were coupled with forces generated by vortex rings distributed on the wing camber surface, calculated with a fully three-dimensional vortex lifting law. The proposed method proved effective in predicting the lift and pitching moment when compared to experimental data and showed promising results in predicting wing drag. The proposed methods have been used in many applications. One such application was to modify the wing of an unmanned aerial system to increase its aerodynamic efficiency and to calculate the drag reductions obtained by an upper surface morphing technique for an adaptable regional aircraft wing. Elements of a non-linear, unsteady VLM (NLUVLM) were used for the design of an additional thrusting fin as an effective way of recovering energy in propeller wake [87].

As nonlinear aerodynamics is also a concern for confined flow conditions, the use of the NLVLM method was not particularly possible until the digital era of computing. The use of NLVLM emerged with collaboration with other methods for improved efficiency

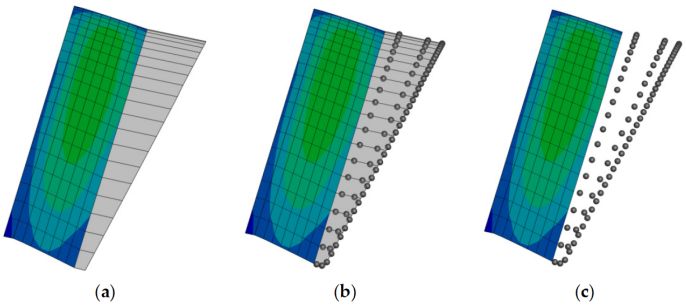


Fig. 2.12 The process to convert vortex rings to vortex particles, (a) creation of shed wake panels, (b) conversion of trailing and shed straight-line elements of wake panels to vortex particles, and (c) suppression of the trailing and shed straight-line elements of wake panels [10].

and practicality of the solver for a number of complex configurations and for different flow conditions. Multi-phase design optimisation (MDO) is often the start of aerodynamic analysis of complex configurations for accurate and fast design optimisation to fulfil given objectives and performance indices. The study of hybrid NLUVLM and vortex particle methods (VPM) by Laurendeau [10] presents a hybrid NLUVLM-VPM as shown in Fig. 2.12 to investigate the aerodynamics of rotor blades hovering in and out of ground effect. The method is of interest for the fast aerodynamic prediction of helicopters and smaller rotor blades. UVLM models the vorticity along the rotor blades and near field wakes with panels that are then converted into their equivalent vortex particle representations. The standard Vreman subgrid-scale model is incorporated in the context of a large eddy simulation for mesh-free VPM to stabilize the wake development via particle strength exchange. The computation of the pairwise interactions in the VPM is accelerated using the fast-multipole method [88]. NLUVLM is achieved with a low computational cost viscous-inviscid alpha coupling algorithm through a strip-wise 2.5D Reynolds-averaged Navier–Stokes (RANS) [11]. The schematic with the notation of the coordinate systems used in NLVLM is shown in Fig.2.13. The aerodynamics of the scaled S76 rotor blades in and out of ground effect was investigated with the proposed algorithm. The results are validated with experimental data and various high-fidelity codes [10]. Despite less application in the past for aerodynamic analysis, the new improved version of NLVLM holds high potential in aerodynamic analysis and optimisation approaches for non-linear wings at supersonic configurations.

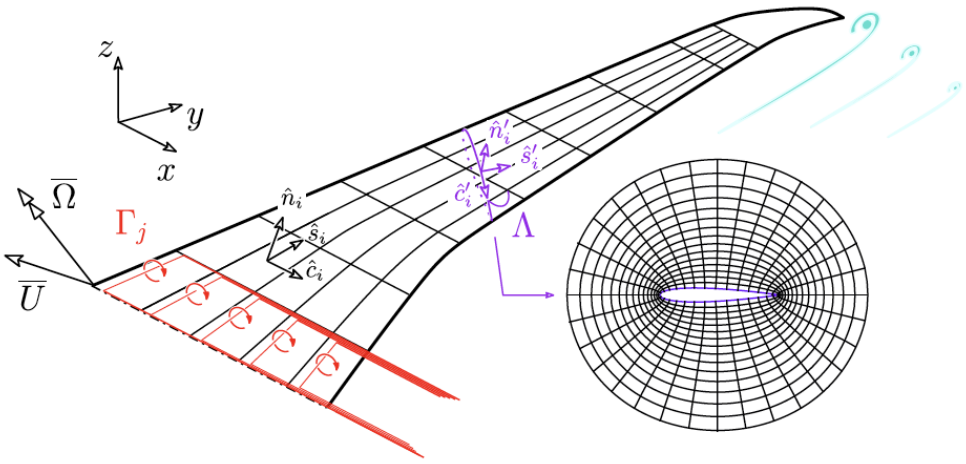


Fig. 2.13 Schematic with a notation of the coordinate system used in NLVLM model [11].

2.2.4 Validation and accuracy of VLM

The journey of VLM from its first use to reaching every milestone with improved versions for complex applications justifies its ease of use and flexible application for different flow and aircraft configurations. With NASA's journey with SST and utilization of vortex lattice method [89] for both subsonic to supersonic configuration, various VLM methods were introduced providing a relevant comparison with wind tunnel experiments and computer code in a desired to solve aerodynamic problems with quicker and cheaper alternative. Ranging from the implementation of quadrilateral vortex lattice method [2] for high flap configuration including ground effects and wind tunnel interference. The results show good agreement with wind tunnel results and rapid convergence as compared to conventional results obtained from experiments. With the early stage of the linear vortex lattice method, Lamar tested various wing shapes and configurations to validate the VLM method application for different flow and aircraft configurations [36]. This was done to gain an appreciation for the accuracy of the various implementations. Several improvements and tests were conducted for its improved accuracy when compared with the potential flow computer program originally developed and practised at Arnold Engineering Development Center [90] which was early designed for transonic propulsion wind tunnel testing. The PFP (Potential flow program) at AEDC (Arnold Engineering Development Center) has not been used as a tool to obtain absolute values, but rather as a tool to predict and verify flow fields in support of the test activities. Which was very successful in predicting the flow angularities around the surface. The major difference between using a conventional high-fidelity solver with VLM was, that it was much quicker and cheaper to run providing good arguments with accuracy when compared with

increased-order high-fidelity solvers. With period as VLM gained popularity, more testing was conducted using vortex arrangement on thin wings [89] to study the flow separation and vortex sheet formation and generally utilised from different Theordorsen's approach to linking its applications with dynamic unsteady flow analysis. The variations of VLM also gain popularity with Polhamus's leading-edge suction analogy for its use in studying the non-planar configuration stating its application as non-linear vortex lattice methods. In the early development of both applications, the vortex lattice method results showed good arguments with wind tunnel experiments results making it a preferable choice of method for early-stage analysis. The most recent bench-marking example using the VLM-based computer program VORLAX is used for different aircraft configurations for different flow conditions at both subsonic and supersonic configurations [91]. This work showcases the potential of the computer program VORLAX for its remarkably accurate aerodynamic results and that too in a few seconds of run-time on a consumer-grade computer. This work also highlights the flexibility of this solver over others in terms of panel orientation and wake analysis for both subsonic and supersonic configurations. Well, it is to be said that this solver is not an alternative for high fidelity but is a bridge between the low fidelity less accurate initial preliminary design and using it for faster convergence initial results to make relevant design changes before making any final decision analysis using high fidelity solvers. Such use of this method will be effective in reducing the analysis cost drastically.

2.3 Vortex Lattice Method-based Computer Program

Development of genetic algorithms for aerodynamic shape optimisation is a common practice in modern improvements of aerodynamic characteristics of conventional aircraft [92]. As discussed before, in the early development of VLM before computers, the only limitation which was stopping researchers from using VLM for more complex wing planform configurations was because of its complex manual calculations. The early development of computer program implementation of VLM was done by joint work of NASA research scientists Margason [93] and Lamar [94] with FORTRAN in the late 20th century. After gathering the input data for the complex planform, this program converts the lifting planform to a vortex lattice to calculate the pressure changes on the lifting surfaces. The platforms included wings with variable sweep outer panels, wings with several changes in a dihedral angle across the span, wings with twist and camber, and a wing in conjunction with either a tail or a canard. The major interest of aerodynamic characteristics was lift and pitching moment for both the flat or twisted wing, drag due to lift parameter, leading-edge thrust, leading-edge suction and their distribution on the wing, and the distribution of several span loading coefficients for many

Table 2.4 List of Computer Programs based on Vortex Lattice Method.

Solver	Year	Method	Distribution	Author	Ref.
VORLAX	1977	VLM	Commercial	NASA	[1]
WDTVOR	1985	VLM	Commercial	NASA	[12]
STRATIP	1985	VLM	Commercial	NASA	[95]
AVL	1988	VLM	Open Source	Mark Drela et al	[96]
TORNADO	2001	VLM	Open Source	Melin	[8]
XFLR5	2003	2D-Panel+ LLT/VLM/ 3D-Panel	Open Source	Deperrois	[86]
ABSOLUT	2011	UVLM	Commercial	NASA	[6]
Q3D	2012	2D panel+VLM	Commercial	Mariens et al	[97]
VSPaero	2012	2D panel+VLM	Open Source	Rob McDonald	[98]
OpenVogel	2016	UVLM	Open Source	Guillermo A. Hazebrouck	[99]

different configurations. During the early development of the FORTRAN code, there were many limitations such as the maximum number of platforms that could be studied (at the beginning this was no more than two), a maximum of 24 straight line segments could be used to define the left half of the wing, and the maximum number of horseshoe vortices was just 120. Limitations were also for variable sweep planforms and to those which have non-zero dihedral angles or two planforms which did not lie in the same plane. The computational power was another hurdle in the utility of the VLM code. One of the major drawbacks of the early VLM code as it was only designed to study the aerodynamic characteristics of a wing for a specific steady, non-compressible subsonic flow and could not model the effect of unsteady flow.

The use of VLM for design optimisation and analysis was already formulated and tested for many non-planar configurations [100] and showed promising results. Many different variations of VLM were tried and tested and compared for better efficiency and accuracy. Soon afterwards, The Boeing Company and NASA together started a joint research work

to carry out the optimisation and design of three-dimensional aerodynamic configurations of arbitrary shapes using VLM in the 1950s [13]. Their combined research influenced the formulation of a new VLM which could be applied to do combined analysis, induced drag optimisation, and aerodynamic design optimisation for three-dimensional configurations. The proposed method showed promising results when compared to analytical solutions. The first-ever testing was done for the Boeing KC-135 tanker aeroplane. The development of the computer program had the initial goal of improving the aerodynamic performance of different wing planforms. This would have helped make precise decisions on design changes before practical testing which would lead to considerable cost savings.

The extension of VLM to the supersonic application was first developed by NASA in 1977 with the use of a computer code called VORLAX [1]. In accommodating the use of VLM for supersonic flow many new programs were developed to use in relation with VORLAX [101] such as WDTVOR [12] whose purpose was to convert the wave drag input geometry for VORLAX input geometry with faster time and high accuracy.

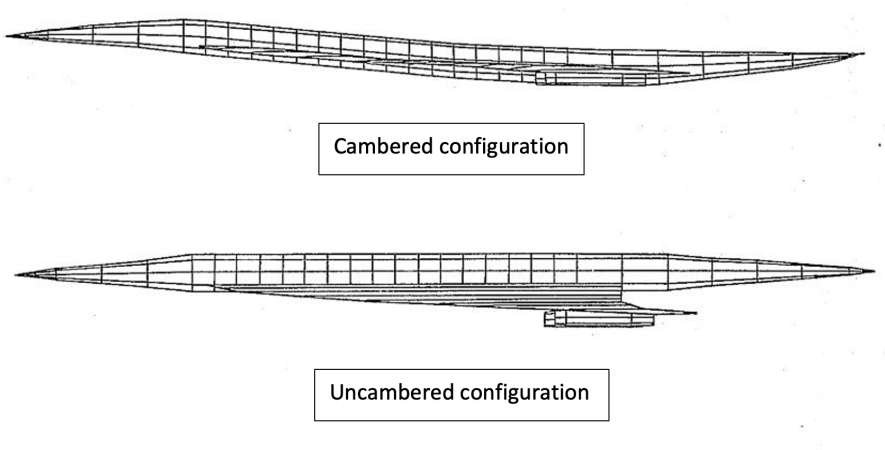


Fig. 2.14 Cambered and Un-cambered Configuration [12].

It is shown that if the discrete vortex lattice is considered as an approximation to the surface-distributed vorticity, then the concept of the generalized principal part of the integral yields a residual term to the vorticity-induced velocity field. WDTVOR was majorly used for cambered and uncambered configuration as shown in Fig. 2.14. The proper incorporation of this term to the velocity field generated by the discrete vortex lines rendered the VLM valid for supersonic flow. After the success of VORLAX, many different implementations and variations of this computer program were forged for a specific requirement such as VORSTAB [95]. A computer program for calculating lateral directional stability derivatives with vortex flow effect and STRATIP [102] to calculate the tip vortex calculation.

With the era of modern computing in the early 20th-century work done on the MDO of an aircraft is easily supported. An example is the implementation of a new set of MATLAB codes within the computational design framework of CEASIOM (which stands for the computerized environment for aircraft synthesis and integrated optimisation method). It uses analysis performed in different modules within the design process including geometry modelling, parameterization, meshing and simulation. With the framework of CEASIOM, an improved version of VLM has been used, replacing vortex panels with vortex rings. CEASIOM, as a means of aerodynamic shape optimisation, is designed to facilitate more efficient and robust designs but is limited in use to subsonic flow conditions. CEASIOM uses the MATLAB optimisation toolbox for optimisation using gradient and algorithmic-based methods which gives the user freedom to define the inclusion of constraints and predict solutions for constraint optimisation problems by mathematical nonlinear optimisation. For

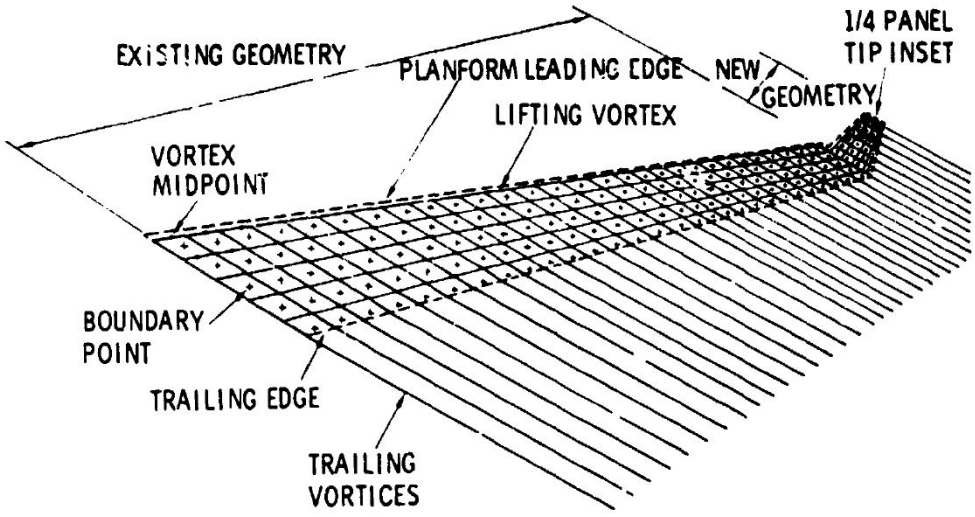


Fig. 2.15 Typical representation of wing and winglet by a multi horseshoe vortex lattice method [13].

example, a typical representation of a wing and winglet by a multi-horseshoe vortex lattice system is shown in Fig. 2.15. It starts with defining the collaborative design environment for wing shape design using the common parametric aircraft configuration schema and MDO framework as a part of conceptual design. Work was done to benchmark CEASIOM to predict flight control and flight qualities of a B-747 [103]. This was carried out with the aim to prove the effectiveness of the various modules, and testing with the different flight paths, controls and characteristics. That study also showed the co-relation between the

compressibility effect and formation of shockwaves with Euler results which contributes towards proving the utility of the CEASIOM framework.

2.4 Conceptual Design using the vortex lattice method.

2.4.1 Preliminary Design approach using Vortex Lattice Method.

Prior to the 1970's the study of aerodynamic characteristics of highly swept thin wings with sharp leading edges at subsonic speeds was hard to estimate. Multiple avenues of research were carried out and more extended versions of VLM were developed and tested [70]. From the perspective of a preliminary designer, methods which are fast, accurate to a reasonable extent, and easy to use are desirable. Once the optimised aircraft configuration is proposed, then more higher-order sophisticated methods to refine the design can be used before commencing with experimental verification of the design. In the preliminary stage of aircraft design, the designer needs to have valid estimates for lift, drag, and pitching moment to validate the design of the platform (e.g., wing, tail, and canard) and make calculated measurements for the position-to-moment center which might just be the desired aircraft center of gravity for trimmed requirement and stability margins of the design to fulfil the required performance requirements. Because of its highly efficient low-order computational analysis performance, VLM is often used in design optimisation for wing design as an early-stage preliminary design tool of wing configurations. Perhaps the first-ever use of VLM for design optimisation of subsonic aircraft was by Paulson Jr [104] who proposed the use of Rockwell-Tulinius vortex lattice theory [105] for calculating aerodynamic characteristics. A Trefftz plane optimisation procedure [14] was used to minimize the induced drag with the help of estimating span loads and a modified Trefftz plane was used to predict the induced drag for specific span loads. The representation of filler wake using the Trefftz plane on an aircraft fuselage is shown in Fig. 2.16. The first two methods show promising results in preliminary aircraft design for different planforms, cambers, and twists. The other method was majorly used to calculate the drag component for different surfaces such as flaps and control surfaces. The Rockwell-Tulinius VLM was known for predicting static and rotary stability derivatives with complex multiple lifting surface configurations of arbitrary shape. This Tulinius aircraft method [106], as programmed, was fast and easy to use which made it an excellent choice for the preliminary design study of subsonic aircraft. The same approach was carried forward by many researchers but eventually dropped when the Euler method was introduced. This became the favoured tool by the industry for aerodynamic design

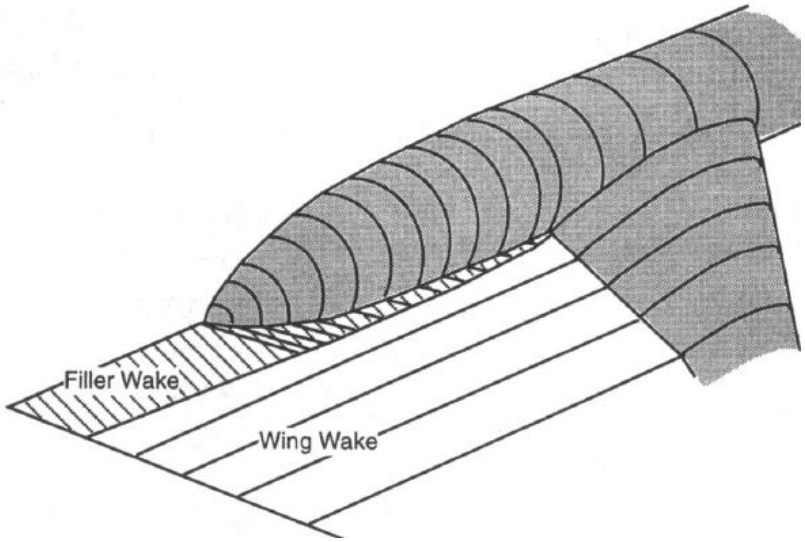


Fig. 2.16 Representation of filler wake on fuselage [14].

optimisation because of two major reasons: i) it was a readily developed and tested potential flow solver, and ii) it gave very promising and accurate results at subsonic flow conditions.

The Euler method [107], being highly developed for design analysis of subsonic or supersonic aircraft, has very complex mathematical calculations which account for slow speed but require more time and higher computational power. However, a VLM being highly flexible for different flow conditions, whilst being reasonably accurate can have promising utility for the preliminary design of aircraft [108]. VLM is also commonly used for UAV analysis, studying flight control effects, and, thus, design optimisation. Multi-fidelity tools such as CAESIOM and TORNADO have been used in studying the aerodynamic characteristics of the flapping wing of a bird [109]. Some studies look at the aerodynamic analysis of flapping wing concepts [108], with the information obtained of notable use for the future morphing wing design.

By the end of the 20th century, many approaches for the development of MDO were in trend. Researchers were focusing on implementing different flow solvers and comparing them on basic multi-objective relations such as reducing wave drag and, at the same time, reducing the lift-to-drag ratio for higher efficiency. Some were working on reducing aeroacoustic properties whilst others worked on improving aerodynamic efficiency. One such work by Pinzon [110] investigated a wing optimisation technique using VLM. Pinzon sought to optimise for a high lift-to-drag ratio of the wing at cruise velocity while maximising the theoretical cruising range and minimizing the wing weight by altering five geometric variables: wing area, wingspan, taper ratio, leading-edge sweep, and geometric twist. However, the limita-

tions of this study were the VLM being restricted to irrotational, inviscid, incompressible flow. Therefore, the given optimisation lacked the use of any compressibility correction factors and did not account for drag divergence effects. A more efficient means to solve the optimisation problem would be to incorporate a gradient-based or other optimal search-based approach as opposed to the numerical brute force technique used by Pinzon to reduce overall computational time and provide a faster solution. Improved accuracy and efficiency can be achieved with VLMs by coupling them with other methods in relation to use with parallel computers for faster analysis [111]. The integration of artificial intelligence for the design optimisation of aircraft for subsonic conditions also holds a convincing potential. In a quest to improve overall efficiency, a VLM can be used in an adjoint-based optimisation approach. The application of this approach has been explored with hypersonic and supersonic missile design [112]. A similar approach has been used to design both 2D and 3D aerodynamics of different wing configurations for transonic speeds [113].

2.4.2 Prospects of Supersonic Aircraft Design using the vortex lattice method.

Supersonic design has many trade-offs, each with its own advantages and disadvantages. The challenge, as with all optimisation problems, is to work out the best value solution for expected performance. [114–119] The design of supersonic aircraft with high swept wings is a challenging and difficult problem within modern aircraft design [120]. A high-swept wing is majorly preferred because of its low drag characteristics at supersonic conditions, as per accepted theory. Many well-known theories suggest that in order to have low drag at very high speeds the angle of the wing must be greater than the sweep-back angle of shockwaves with free-stream at high Mach numbers. As discussed above, the VLM is a very versatile and efficient potential flow solver for accurately predicting the aerodynamic characteristics of aircraft under subsonic and supersonic conditions [121]. Despite its limitations, it is easy to understand and quite flexible to changes with respect to certain flight conditions and different wing planforms [122] which makes it a critical tool for the preliminary design phase of supersonic aircraft. Conceptual design requires the fast and accurate calculation of design parameters for aircraft under given flow conditions for which VLM is a sensible choice because of the quick operation, providing reasonable accuracy and efficiency. Many computer programs have been developed and have been extensively used by many industrial and research organisations for the conceptual design phase for subsonic aircraft design prediction. From the early use of VLM for design optimisation, the focus has been on the wings [123], and lifting surfaces in general, but rarely if ever accounts for the influence from

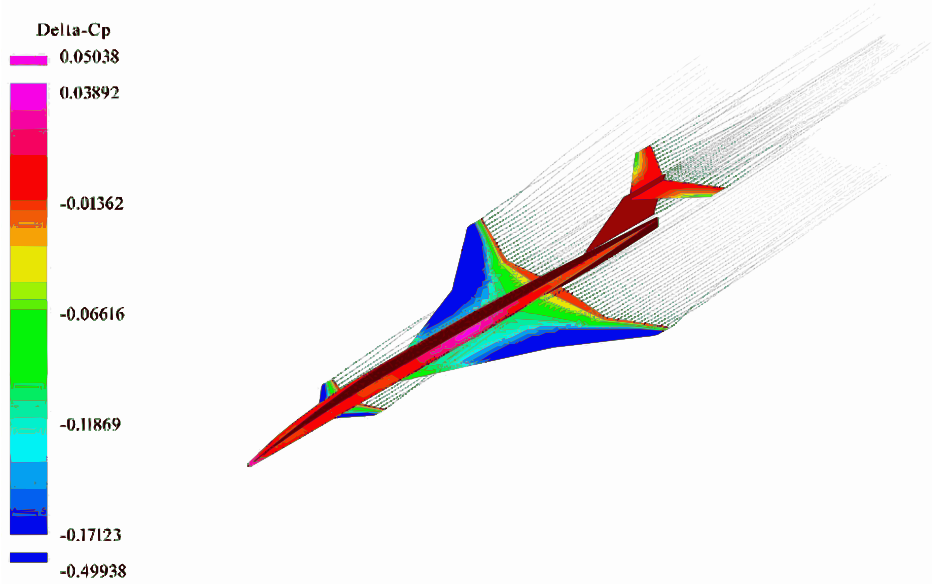


Fig. 2.17 Pressure distribution of supersonic aircraft from VSPAERO using VLM [15].

the fuselage, nacelles or other non-lifting bodies of an aircraft. With modern techniques, improved versions of VLM includes the influence from other design features and run an optimisation loop that will work for the overall design optimisation of aircraft. The use of VLM has a good argument behind it for MDO for hybrid aircraft; work processes to enrich the future of hybrid aircraft conceptual design holds a large area of research using VLM because of its fast computational offering. The recent development of hybrid aircraft configuration to reduce carbon footprints is a design priority for the aviation industry, which makes the use of aerodynamic analysis tools that can account for more parameters and support efficient and rapid MDO attractive [124]. In the past, NASA has funded many projects for a supersonic design using VLM. Progress with VLM suggested quick and cheap analysis for supersonic configuration to analyse its capabilities at low speeds. One such project 'SCALOS' supersonic configuration for low speeds is presented for aerodynamics predictions for different pitch runs and fidelity [15]. Many solvers like VSPAERO and VORSTAB are used to capture the aerodynamic characteristics of supersonic configuration for both linear and non-linear regimes. VLM used in Open-VSP was able to predict the point of change in the longitudinal stability derivative. This paper shows the capabilities of developed VLM, its comparison over other low order methods such as panel method and gives a better understanding of what needs to be changed to better predict the non-linear characteristics as VLM faces difficulty in predicting the vortex characteristics of a rounded leading edge as compared to the sharp leading edge [15]. A clear representation of use of

VLM for the prediction of pressure distribution over a supersonic configuration is shown in Fig. 2.17.

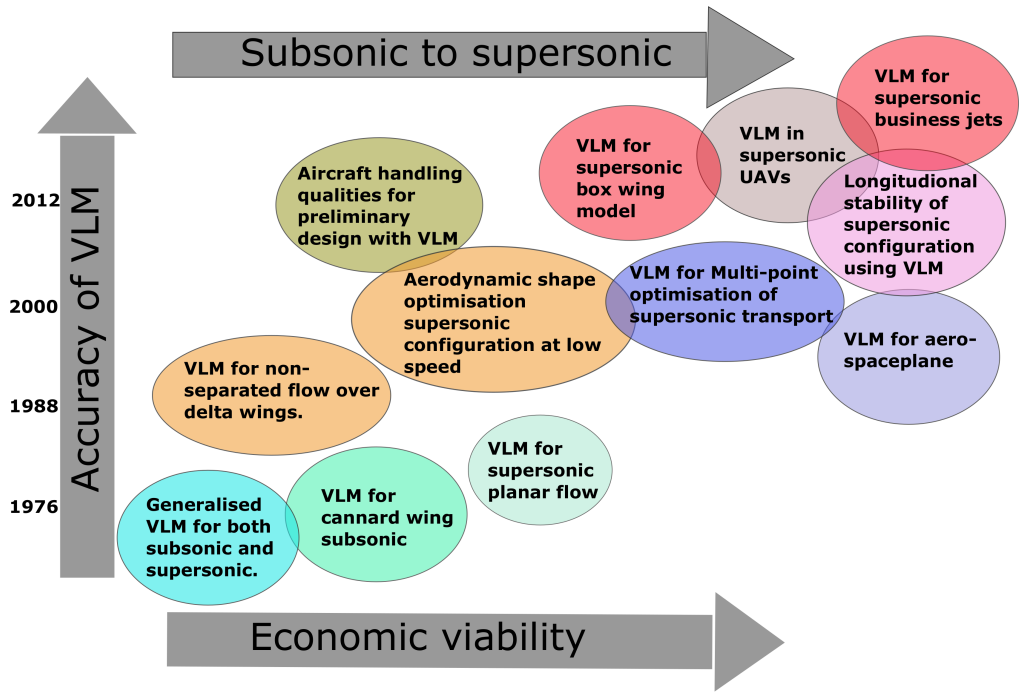


Fig. 2.18 Roadmap of VLM for supersonic configurations.

The roadmap of the vortex lattice method for its application for supersonic configuration is shown in Fig.2.18. The journey started with the development of VLM by Miranda [125] for its generalised use for subsonic and supersonic configuration. The method gained popularity when the accuracy of VLM was close to conventional CFD solver for subsonic conditions. Starting its analysis for canard wing configuration for subsonic conditions[126]. Further development of supersonic VLM was carried forward with its application by studying the planar flow at supersonic speeds improving on the limitations of the VLM [127]. Within a decade of development and use of VLM for subsonic configuration, the solver was used effectively in studying the flow separation over the delta wing [128]. The use of VLM in stability and flow separation conditions was quite an effective and efficient analysis. The early 21st century comes with solving one of the biggest design problems with supersonic configuration in research for reducing sonic boom capabilities, making them equally efficient at low speeds[15] and improving the longitudinal stability at low speeds [129] for which supersonic VLM was the most efficient method for making preliminary predictions. The recent use of VLM for supersonic configuration was used to study the spaceplane [130], in supersonic UAVs and supersonic business jet design [131], and more details are shown

in table 2.5. For the next generation of supersonic aircraft low boom is the biggest design challenge for which the study of dynamic stability is critical to examine. Dynamic stability at take-off and landing can be readily examined with a help of VLM [132]. Boeing UDP which is based on Ames's (Woodward) subsonic supersonic wing body panel method program was used to study the dynamic stability of non-planar configuration and winglets deformations [133] which utilises Carlson's leading edge thrust correction and vortex lift correction which proves to be a cheap alternative to analyse the stability derivatives than conventional solvers. When it comes to work done in designing the supersonic configuration using low-fidelity tools, The University of Michigan worked in the development of UVLM code enhanced with the Polhamus method [134] and combined with corrections from the wind tunnel experiments developed a solver with adequate fidelity while capturing the prevalent vortex phenomenon [135] involved in high angle of attack and low-speed maneuvers. Baseline supersonic configuration SCALOS was used to support the investigation to enable future parametric investigation [15].

Fig. 2.19 Northrop Grumman QSP Concept [16].

Such MDO tools work out potential solutions for multi-objective performance indices which, in the case of supersonic aircraft, can be in reducing compressibility drag, the distribution, and strength of shock waves, or the magnitude of the noise from sonic booms. In the past with the initiation of SST, there were many problems in improving these limitations [16, 136]. A lack of research and computational power to run the analysis for several different flight conditions and for different supersonic flows limited development in optimised supersonic designs. But with improved potential flow solver methods and their integration with much more efficient and powerful computer languages, there is more capability than before of supporting low-cost supersonic aircraft design. Substantial research is now underway to

reduce the impact of sonic booms [137]. With the latest design improvements one can theoretically bring down the sonic boom noise level to normal hearing conditions [138–140].

Such research is majorly favored by a fair number of new supersonic business jet companies, which are relying on modern computational capabilities to bring new designs to the market much faster than previous aircraft design schedules. Companies such as Gulfstream Aero, Boom supersonic, and many more [16] are currently working on quiet supersonic business jets. One such example by the Northrop Grumman concept is shown in Fig. 2.19. Use of VLM has been seen trending with studies for aerodynamic shape optimisation for supersonic transport especially considering the low-speed stability. VLM was used to study the low-speed constraint during optimisation for augmented vortex analysis for stability derivative computation based on the pitch-up estimation method to capture non-linear pitch-up characteristics in the model. The paper demonstrated the capabilities of VLM for highlighting the trade-offs between supersonic performance and subsonic stability. More work was carried forward with pitching moment approximation for the MDO of HSCT. Low-order VLM was used with a pitch-up estimation method to capture non-linear pitch-up characteristics in the model [151].

Table 2.5 Use of Vortex Lattice Methods in supersonic configuration.

Application	Year	Title	Author
2D-Supersonic flow	1977	Generalised VLM for subsonic and supersonic flow application	L.R. Miranda [125]
Delta wing	1981	Vortex lattice method for the calculation of the nonsteady separated flow over delta wings	Daniel Levin et. al. [141]
3D-Supersonic flow	1990	Three dimensional flow analysis by the vortex lattice method	Carlos Guillermo Rodriguez et. al. [142]
Vortex flap	1991	Study of the interaction of separated vortices in transonic and supersonic flow on improving the aerodynamic characteristics of vortex flap	T.D. Hsing [143]
Supersonic aircraft handling	1995	Computer optimisation of aircraft handling qualities during preliminary design.	D.S. Soban et. al [144]
MDO HSCT	1996	Response surface approximation for pitching moment including pitch-up in the MDO design of an HSCT	Crisafulli P et. al. [145]
Planar supersonic flow	1997	Generalised vortex lattice method for planar supersonic flow	Paulo A.O Soviero et.al [146]
Supersonic grid-fin	2005	Predicting aerodynamic characteristics of grid-fin configuration at supersonic speeds	P.theerthamalai et. al [147]
Supersonic wing design	2008	Wing design of supersonic transport by a multi-point optimisation method	K. Higuchi et. al. [148]
TORNADO	2010	TORNADO supersonic module development	Garrido Estroda [17]
Sidewash in supersonic flow	2012	Sidewash on the vertical tail in subsonic and supersonic flows	C.S. Chiu et. al. [149]
Comparative study of business jets	2014	A comparative study of the low-speed performance of two fixed planforms versus a variable geometry planform for a supersonic business jet	Aaron C. Smelsky [129]
Low-speed modeling of supersonic aircraft design	2021	Low-speed aerodynamic modeling for control related consideration in supersonic aircraft design	Thiago Guimaraes et. al. [150]
SCALOS testing	2022	Supersonic configuration at low speeds(SCALOS): Test/ Simulation correction studies	Mavriplis et. al. [15]
Dynamic stability of supersonic aircraft	2022	dynamic stability characteristics for commercial supersonic configuration at low speeds flight conditions	Magee T. et. al. [132]
Supersonic aeroelasticity	2022	An integrated low-speed aeroelastic flight dynamics framework for modeling supersonic aircraft	Thiago Guimaraes et. al. [135]
Supersonic shape optimisation	2022	Aerodynamic shape optimisation of a supersonic transport considering low-speed stability	Seraj S. et. al. [131]

2.4.3 Mach-cone influence in VLM for supersonic configurations

In implementing the supersonic extension of the Vortex Lattice Method (VLM), it is important to use precise terminology to distinguish between the geometric *Mach cone* and the physical *shock cone*, which are sometimes conflated in the literature. The Mach cone is a purely geometric locus of Mach waves associated with small-amplitude, isentropic compressions. For a free-stream Mach number $M_\infty > 1$, its half-angle is

$$\mu = \sin^{-1}\left(\frac{1}{M_\infty}\right), \tag{2.1}$$

and it defines the region of influence of a disturbance propagating at the local speed of sound.

In contrast, the shock cone arises when the flow is turned through a finite deflection, producing a real compression shock with an associated pressure rise and entropy increase. The half-angle of this shock cone, commonly denoted θ, satisfies the oblique–shock relations that underpin the *Taylor–Maccoll* conical–flow solution used in this work for supersonic VLM corrections:

$$\tan\theta = 2\cot\beta\,\frac{M_\infty^2\sin^2\beta - 1}{M_\infty^2(\gamma+\cos 2\beta)+2}, \tag{2.2}$$

where β is the flow-deflection angle and γ is the ratio of specific heats. This conical–shock treatment allows the solver to capture the redistribution of pressure and velocity across the shock layer and to update the local velocity potential on each panel.

The present supersonic VLM framework therefore couples the classical VLM influence–coefficient formulation with a Taylor–Maccoll–based conical shock model. Each lattice panel that experiences supersonic incidence is embedded in a local Mach cone geometry for wave propagation while also accounting for the finite–strength shock cone through the above relations. This dual representation provides consistent compressibility corrections to the induced–velocity field and ensures that lift and drag coefficients reflect both wave drag and shock–induced pressure changes. By maintaining this clear distinction between Mach–wave geometry and shock–wave physics, the method avoids ambiguity and improves the fidelity of supersonic VLM predictions relative to RANS and experimental data.

Subsonic and supersonic wing theories and other low-order solvers such as doublet, and vortex distribution have been reviewed and a systematic correction approach is provided which might be able to improve the correction factor associated with the accuracy of VLM for supersonic flow. It shows that care must be taken in the treatment of singularities involved in the analysis. In the development of the TORNADO's supersonic module [17] it is clearly seen the integration of Mach cone for the delta wing analysis is shown in Fig. 5.5. The relation of Mach angle with Mach speeds explaining increasing the Mach number decreases

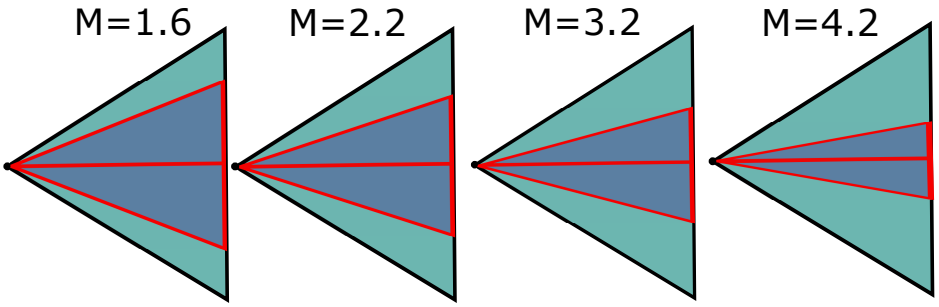


Fig. 2.20 Mach cone influence on the delta wing [17].

the Mach angle. This explains that as the flow is compressed the shock angle and the speed change with a change in temperature. The angle of the shock changes greatly with an increase in the Mach angle. To prevent this coalesce of pressure change the wing span will need to be changed which is affected by shock. Such influence of Mach cone is critical in designing supersonic configuration. Mach cone influence can also benefit from the shift in collocation point of the panels when transitioning from subsonic to supersonic speeds. We know that a change in Mach angle changes the shock's apex location, which in theory should also change the panel collocation points. When computed for the transition to supersonic speeds, these changes in location will help improve the solver's accuracy.

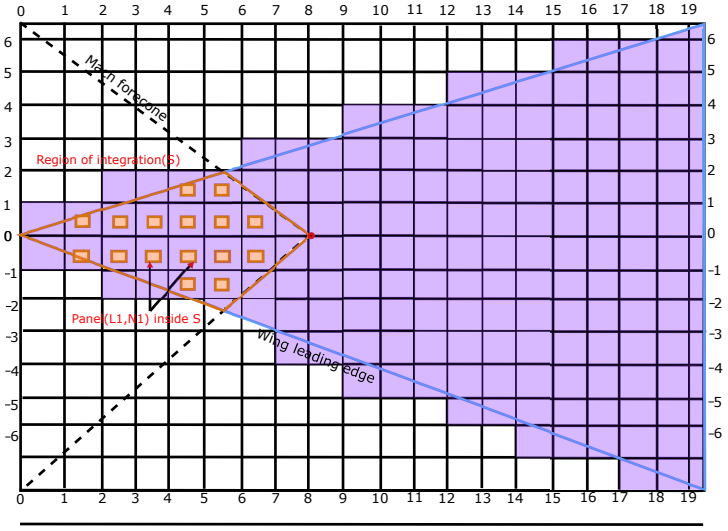


Fig. 2.21 Panel Representation of Flat Plate Delta Wings with Subsonic Leading Edge [18].

Influence of the continuous distribution of horseshoe vortices originating from wing elements divided into a number of small panels with small chord lengths and spans are shown in Fig. 2.21 The region of integration S stretches on the wing planform inside the Mach forecone originating from the field point (x,y). The panel selection and position are

influenced in relation to the Mach forecone. The change in the Mach forecone triggers the selection of panels which influence the design change of the wing for supersonic flows.

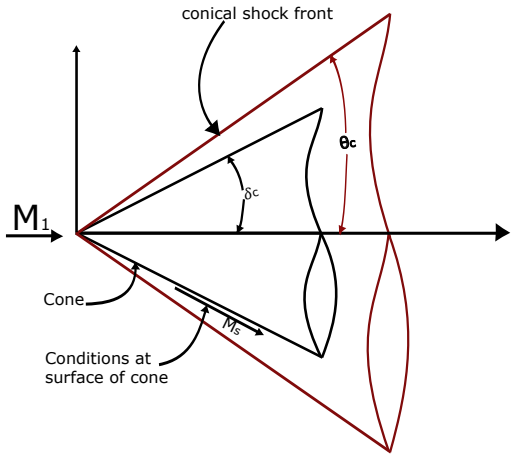


Fig. 2.22 Taylor MacColl equation representing of shock layer and oblique shock relationship for hyper-velocity flow [19].

It is the common practice to predict pressure distribution over a cone at high-speed wind channel using Taylor MacColl equation for studying the relation between oblique shock waves and Mach cone which states the normal velocity component on the cone surface becomes small. The Taylor MacColl implementation with hypervelocity calculation by Ishimatsu [19] has the potential to determine the change in velocity of the panels during its transitions from subsonic to supersonic speeds. The method can be implemented for each panel on the wing assuming each panel as a wing in itself going through supersonic flow experiences a shock cone as shown in Fig. 2.22. With the relation between the shockwave and the Mach cone of the panels, the velocity correction will be helpful in predicting the change in pressure distribution on the panels hence improving the accuracy of VLM for supersonic speeds.

$$(1-\frac{v^2}{a^2})\frac{\mathrm{d}^2\mathrm{u}}{\mathrm{d}\theta^2}+cot\theta\frac{\mathrm{du}}{\mathrm{d}\theta}+(2-\frac{v^2}{a^2})u=0 \tag{2.3}$$

Where a: is the local speed of sound, u: the velocity component in a radial direction, v: is the velocity component in θ direction and θ is the circumferential coordinate. As the velocity normal component on the cone surface is zero and even goes smaller relative to u in the shock-layer at hypervelocity. The equation 4.1 can be written as:

$$\frac{\mathrm{d}^2u}{\mathrm{d}\theta^2}+cot\theta\frac{\mathrm{d}u}{\mathrm{d}\theta}+2u=0 \tag{2.4}$$

when radial direction velocity component is defined usually as $u(\theta) = f(\theta)cos\theta$ which makes equation 4.2 as:

$$\frac{\mathrm{d}}{\mathrm{d}\theta}(\sin\theta cos^2\theta \frac{\mathrm{d}f}{\mathrm{d}\theta}) = 0 \tag{2.5}$$

equation 4.3 can be integrated analytically and $u(\theta) = f(\theta)cos\theta$ becomes:

$$u(\theta) = A[1 + cos\theta . \ln\tan\frac{\theta}{2}] + B\cos\theta \tag{2.6}$$

Remaining Taylor Maccoll equation yields:

$$v(\theta) = \frac{\mathrm{d}u}{\mathrm{d}\theta} = A[\cot\theta - \sin\theta . \ln\tan\frac{\theta}{2}] - B\sin\theta \tag{2.7}$$

in equations 4.4 and 4.5, A and B are integration constants, and these values can be determined from the oblique shock conditions:

$$u(\beta) = V_1\cos\beta \tag{2.8}$$

$$v(\beta) = -\frac{1}{\rho}V_1\sin\beta \tag{2.9}$$

V_1 is the uniform velocity and ρ is the density ratio across the shock wave given by:

$$\rho \equiv \frac{\rho_2}{\rho_1} = \frac{(\gamma+1)M_1^2\sin^2\beta}{(\gamma-1)M_1^2 sin^2\beta + 2} \tag{2.10}$$

where M: is Mach number, β: shock-wave angle, γ: specific heat ratio, ρ_1 is the density of uniform flow and ρ_2 is the density of shock behind. From equations 4.4-4.7 we have:

$$A = (1 - \frac{1}{\rho})V_1\sin^2\beta\cos\beta \tag{2.11}$$

$$B = V_1 - [(1 - \frac{1}{\rho})V_1\sin^2\beta\cos\beta] \times (\frac{1}{\cos\beta} + \ln\tan\frac{\beta}{2}) \tag{2.12}$$

This bring down the hypervelocity equation to as follows:

$$\frac{u}{V_1} = \cos\theta + (1 - \frac{1}{\rho})\sin^2\beta\cos\beta[1 - \frac{\cos\theta}{\cos\beta} + \cos\theta . \ln\frac{\tan\frac{\theta}{2}}{\tan\frac{\beta}{2}}] \tag{2.13}$$

$$\frac{v}{V_1} = -\sin\theta + (1 - \frac{1}{\rho})\sin^2\beta\cos\beta[\cot\theta + \frac{\sin\theta}{\cos\beta} - \sin\theta . \ln\frac{\tan\frac{\theta}{2}}{\tan\frac{\beta}{2}}] \tag{2.14}$$

The equations 4.11 and 2.14 used for hypervelocity prediction of high-speed hypersonic aerodynamics can be very easily implemented for supersonic configuration for supersonic speeds. Assuming each lattice or panel as a wing going through supersonic speeds encounters a shock wave can be solved to predict the change in velocity component on the panel surface which will be very well helpful in predicting the pressure changes on the panel surface, improving the supersonic VLM accuracy.

Warp-wing design using VLM.

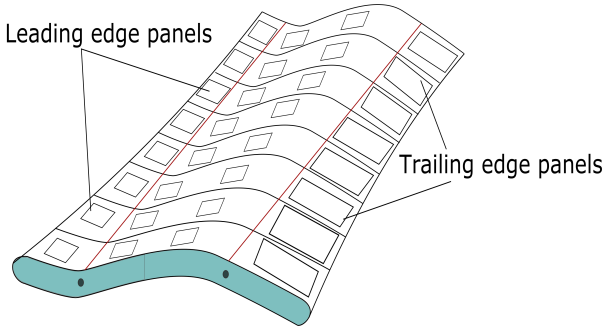


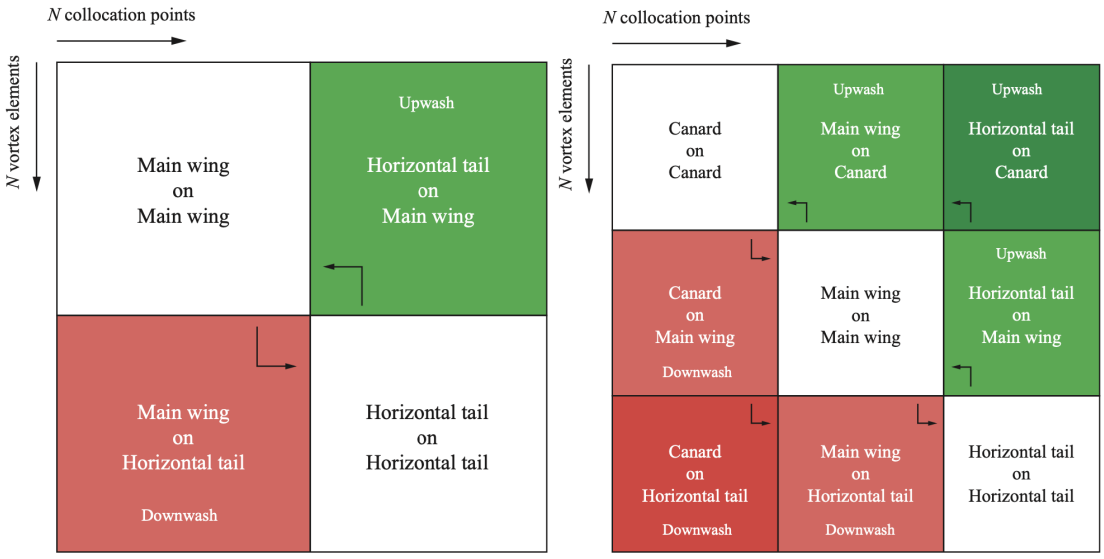
Fig. 2.23 Warp wing design using VLM.

The wing with a larger sweep angle and low aspect ratio is often used to reduce the wave drag and its influence at supersonic speeds. Keeping this configuration satisfies one application but does not perform well under low-speed take-off and landing. To overcome this problem a large area of the wing is required to generate lift and consequently reduce the take-off and landing length. Which leads to an increase in aircraft weight leading to inefficient flight. Work has been done with regards to the use of VLM in design optimisation of such configuration of wing keeping in mind the required performance index changing design parameters to achieve its goal of efficient flight using a combination of Quasi-vortex lattice method [152] and leading-edge suction analogy [148]. A new multi-point design method approach was used to design an improving lift with reducing the wing area and weight which was majorly suitable for supersonic business jet configurations. The work shows that by using the supersonic wing theory and Carlson's warp wing theory in iSIGHT-FD wing planforms are optimised at supersonic cruising speeds. Use of VLM with dividing the wings into the number of panels chordwise and spanwise focusing more on the leading edge flap as shown in Fig. 2.23 and trailing edge flaps which reduces the lift dependent drag by increasing the No. of panels on focused surfaces make desired design changes keeping

the arbitrary shape intact. Lift-dependent drag is usually reduced by combining camber and twist distribution to optimise the pressure distribution on the surface panels working out and making relevant design changes.

2.4.4 Compressibility correction and drag divergence integration with Vortex Lattice Method

For incompressible flow the change in density due to expansion and compression of fluid is small enough that it can be ignored to save a large amount of simulation cost. On the other hand, and especially when the flow is supersonic, with a significant expansion and compression of fluid being appreciable, compressible fluid analysis that considers density variation is required. To treat the compressible fluid a pressure-based calculation method is used, though in the case of high Mach numbers a density-based calculation method is preferred. To predict the effect of compressibility on down-wash interaction is proposed by Vaessen [20] which works on the compressibility correction factor k with vortex lattice method determining its influence on down-wash as shown in Fig. 2.24 and making relevant decision on design changes. the accuracy of the solver proposed was dependent on how close the tip vortex from the canard passes from the main wing.



(a) Conventional configuration. (b) Three-surface configuration.

Fig. 2.24 Structure of an N × N vortex lattice influence matrix for compressibility calculations [20].

VLM and modern implementations of UVLM model the steady and unsteady potential flow around a lifting surface for different flow conditions and parameters with the influence of vortex points or vortex rings. However, current advancement and overall VLM theory, in general, do not deal much the implementation of high-speed (compressibility) drag calculation and the amount of work done in the past in regard to this is much less than in other areas. Compressibility correction factors for standard steady VLM are typically introduced by the Prandtl-Glauert (PG) correction factor [85] [153], based on the rule which states that the pressure coefficient, C_p, at any point on a thin aerofoil surface in a compressible flow is $(1 - M_\infty^2)^{1/2}$ times the pressure coefficient at the same point on the same aerofoil in incompressible flow. The solution for the compressible flow of VLM is solved for the incompressible domain and then modified with the correction. Karman and Tsien [154] proposed a two-dimensional subsonic flow perfectly with irrotational, compressible fluid by replacing the adiabatic pressure-volume curve with a tangent line drawn at an arbitrary point on this curve. An extension of this Karman-Tsien method to supersonic jet flow was made by Coburn [155]. The basic partial differential equations satisfied by the velocity potential in a general three-dimensional domain were derived from the Eulerian momentum equations, the continuity conditions, and with respect to the relation between the speed of pressure, and propagation [156]. These relations are all analytically mentioned in literature [157] [158]. Concerning the PG rule, the prediction of induced drag is not included for three-dimensional flow. The higher-order methods require the direct use of a pressure coefficient which is not available in conventional VLM. VLMs already in use produce a delta pressure coefficient (ΔC_p) distributing the differences in pressure between the top and bottom aerofoil surfaces. Because the use of higher order correction is not applicable, and the PG rule is preferred for calculation in the three-dimensional domain, the vortices are computed as usual using VLM for incompressible strength distribution and are followed by compressible vortex distribution. After that, the incompressible vortex strength at each panel is multiplied by the PG correction factor. The induced velocity from the vortex core is calculated using compressible vortex strength and compressible vortex-induced velocity in three-dimensional space form is added to the velocity contribution from the free stream and aircraft rotation. The compressible vorticity and effective velocity at the vortex core are then used to compute the force vector acting on each panel and the force distribution is integrated to calculate the forces and moments to predict the pressure distribution on the surface. This method proved very effective and was highly accurate and efficient as a three-dimensional VLM but has certain limitations regarding supersonic conditions and large angles of attacks. A compressible VLM has been used to study the interaction of a profile wing with a convicted vortex passing under the profile, commonly known as blade vortex interaction. The technique

applied to an airfoil is known as airfoil vortex interaction [159]. The PG correction method is presently used in TORNADO's MATLAB code [17] and shows promising results when compared to other methods. Still, the implementation for the prediction of compressibility correction and drag divergence prediction for three-dimensional supersonic flight conditions is still missing. As most of the three-dimensional research for the conceptual design phase for supersonic aircraft is done under cruise conditions, the accurate modelling of drag divergence is not required. Prediction of drag divergence [160] and drag reduction [161] is still a vital area of study for supersonic aircraft design for its operation in low subsonic speeds during lift and initial manoeuvres before reaching cruising supersonic speeds [161].

2.5 Lattice Boltzmann method

The lattice Boltzmann method (LBM) is popular for its ease of modelling and efficient solver in accurately predicting the aerodynamic analysis of the lifting surface. The proficiency of this solver is rooted back from its source which is derived from the reduced order of Navier Stokes fluid equations. The earliest use of the LBM for computational modelling dates back to the early 20th century. Researchers from all over the world started exploring its potential and capability in modelling complex flows such as supersonic and hypersonic. The earliest improvements with the solver were done by exploring the range of boundary conditions suitable for different flow conditions. These viscous boundary conditions were successfully applied to fluid flows through porous media and multiphase fluid flows [162, 163]. The two common boundary conditions were bounce-back and no-slip boundary conditions as shown in Fig. 6.5, many approaches were used for these boundary conditions [164–170] The earliest use of bounce-back boundary conditions which was merely used for the stationary walls was observed with increased error in calculating the velocity change at the wall [171]. To improve on this limitation, the researcher focused on implementing a new boundary condition known as non-slip boundary conditions at the wall. In this approach, the initial calculation of the unknown distribution function is assumed to be an equilibrium distribution function with counter slip velocity for accurately predicting the fluid velocity at the wall is equal to the wall velocity. Using this approach with Poiseuille flow and Couette flow requires a nine-velocity model to demonstrate the accuracy of the boundary conditions. After exploring its numerical potential, the research inclined towards building a computer software to explore its computational capabilities earliest exploration of this approach was conducted by NASA at Langley Research Centre with the development of packages like Power-FLOW and CFL3D [172]. The work discusses the comparison between these solvers for handling the complex flow simulation for a flow over a two-dimensional NACA0012 aerofoil. The initial

results showed an increased error with PowerFLOW which was influenced by the inadequate resolutions of the boundary layer for reasons related to grid resolution while reserving the turbulence modelling. The use of square grid cells in PowerFLOW solver restricts the number of points used for modelling the boundary layer distribution across the span of the wing while keeping the computational size small making it easy to run on most computers of the time. With increasing computational power with time more research was done in improving its domain size to accommodate large grid size with increased computational requirements. Further work was done by researchers from Japan Aerospace Exploration Agency [173]. They study where flow simulations around a NACA0012 aerofoil were conducted using LBM. Initially designed for incompressible flow on orthogonal coordinates, the LBM was adapted to accurately resolve the aerofoil's boundary layer by extending it to generalized coordinates. Additionally, a suitable wall boundary condition was introduced for these generalized coordinates. The study proceeded in two main phases. First, the numerical method was validated at a Reynolds number (Re) of 500. Detailed comparisons of the boundary layer's velocity profile were made with results from the CFL3D computational fluid dynamics code (developed at NASA Langley Research Centre), demonstrating consistent

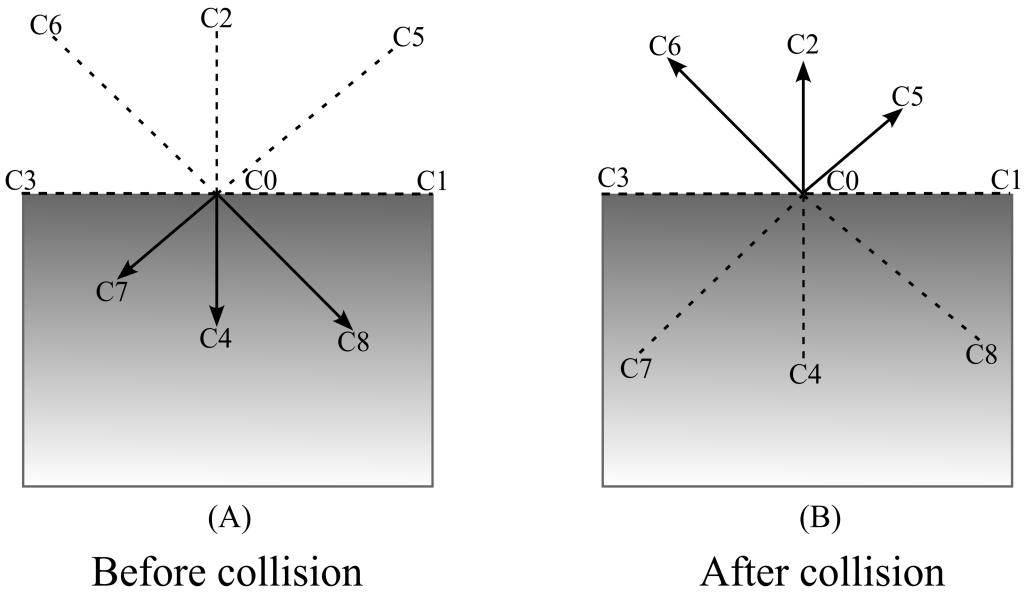


Fig. 2.25 No-slip boundary condition with collision to channel wall [21].

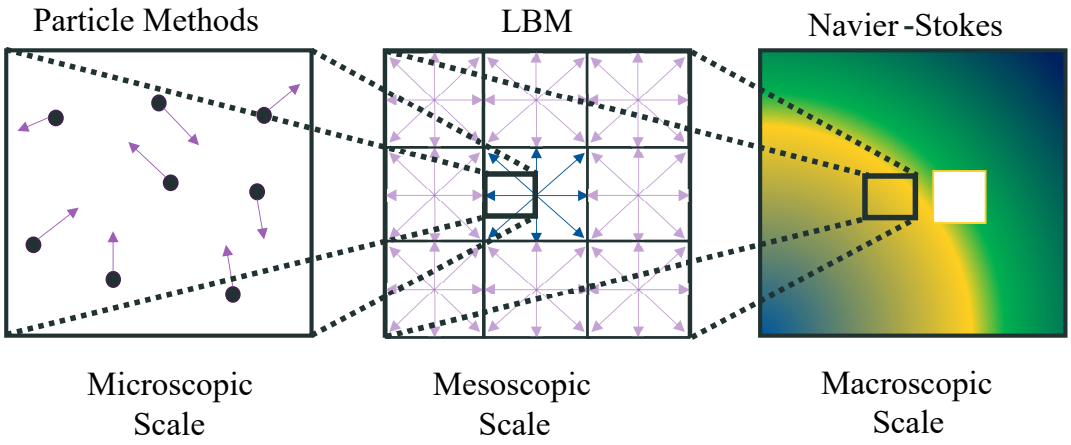


Fig. 2.26 Lattice Boltzmann within scale framework [22]

outcomes. In the second phase, the LBM was combined with the Baldwin–Lomax turbulence model [174] to analyse flows at a higher Reynolds number (Re=5×10$\hat{5}$). The resulting data were compared to experimental data and found to be in good agreement. Remarkably, with the same grid resolution, the LBM method was shown to resolve high-Reynolds-number flows as accurately as traditional Navier–Stokes solvers. This suggests the effectiveness and potential of the lattice Boltzmann method for simulating complex fluid dynamics scenarios. The scale framework benefits of using an LBM solver specifically for mesoscopic scale is shown in Fig. 2.26 LBM's potential was explored for unsteady dynamics and work was done in analysing the multi-element aerofoil for unsteady flow using the lattice Boltzmann method with a large eddy simulation approach [175]. This work describes three-dimensional time-dependent simulations conducted at varying angles of attack to analyze flow characteristics around an airfoil, previously studied using Navier–Stokes methods. The mean flow results from the simulations were compared favourably with experimental and numerical data, specifically in terms of pressure distributions and time-averaged flow patterns within the slat cove. The primary focus of the study was on unsteady flow phenomena, particularly the dynamics of the slat cove, known for generating both broadband and narrow-band noise. The simulations examined how the angle of attack influences various aspects such as the behaviour of the shear layer at the slat cusp, the shedding of vortices at the slat's trailing edge, and the convection and reattachment of vortical structures near the slat gap. The study provides insights into these complex flow behaviours under different aerodynamic conditions.

2.5.1 LBM for supersonic

With LBM improving over the period with infusing different boundary conditions, collision factor for simulating the complex flow not just in slow speed porous medium but also with

increased speed supersonic flows. Recent use of LBM for supersonic flow is explored using the double distribution function (DDF-LBM) which is capable of accurately simulating the polyatomic gases at high speeds [176]. This work extends the standard numerical equilibrium approach used in discrete velocity methods to reproduce an arbitrary number of moments of the Maxwell–Boltzmann distribution [177], essential for accurately representing thermal and compressible flows. Despite utilizing only 39 discrete velocities in three dimensions, this model effectively simulates high Reynolds-number, supersonic flows by incorporating Knudsen-number-dependent relaxation times computed analytically to ensure stability. The effectiveness of the proposed DDF-LBM is demonstrated through various simulations. A 1D Riemann problem illustrates its capability in handling discontinuities under zero-viscosity conditions, while simulations of supersonic flow past a NACA0012 aerofoil and a sphere highlight its excellent performance in low-viscosity, supersonic regimes. Notably, the DDF-LBM outperforms previous methods in terms of computational efficiency on both CPU and GPU architectures. The study suggests that this innovative approach opens up new possibilities for realistically simulating compressible flow applications using lattice Boltzmann methods alone, promising significant advancements in fluid dynamics, particularly in the realm of supersonic and low-viscosity flows. The efficient lattice Boltzmann (LB) model designed for simulating three-dimensional high-speed compressible flows was used across both subsonic and supersonic regimes [178]. The model employs a hybrid recursive regularization (HRR) collision operator on a D3Q19 lattice stencil. To simplify the implementation, an improved thermal equilibrium distribution function is derived specifically for the D3Q19 lattice, reducing the complexity of correction terms typically required in LB methods. For handling supersonic flows with shocks, the model incorporates a straightforward shock-capturing scheme and upwind-biased discretization of correction terms. The simulation process involves recovering mass and momentum equations through efficient streaming, collision, and forcing procedures on the D3Q19 lattice. Additionally, a non-conservative formulation of the entropy evolution equation is utilized and solved using a finite volume method. The effectiveness of the proposed method is evaluated through several test cases, including 2D isentropic vortex convection, 3D non-isothermal acoustic pulse propagation, 2D supersonic flow over a bump, 3D shock explosion in a box, interaction of a 2D vortex with a shock wave, and 2D laminar flows over a flat plate at Mach numbers (Ma) of 0.5, 1.0, and 1.5. Overall, the study demonstrates the versatility and accuracy of the proposed LB model in capturing diverse compressible flow phenomena across different speed regimes, showcasing its potential for practical applications in aerodynamics, shock physics, and high-speed fluid dynamics simulations.

Further development on the use of LBM for supersonic flows is with the inclusion of a three-dimensional rotated lattice Boltzmann flux solver for the simulation of high-speed compressible flows [179]. The work introduces a novel three-dimensional (3D) Rotated Lattice Boltzmann Flux Solver (RLBFS) designed for simulating compressible flows across a wide range of Mach numbers. This method discretizes the Navier-Stokes equations using the finite volume approach and computes fluxes at each cell interface. To enhance numerical stability, the RLBFS adopts a hybrid approach for convective flux computation, leveraging a D1Q4 lattice Boltzmann model in two perpendicular directions aligned with the cell interface's outer normal vector. Viscous fluxes are calculated conventionally using a second-order central scheme. The stability and accuracy of the proposed 3D RLBFS are analysed using matrix-based stability theory and L2 error assessments on different mesh configurations, demonstrating stability even at high Mach numbers and second-order spatial accuracy. The method's reliability and capability are further validated through simulations of various challenging compressible flow scenarios, including subsonic, transonic, supersonic, and hypersonic flows over different wing and body configurations. Comparisons with experimental or numerical data from literature confirm the accuracy and effectiveness of the RLBFS in accurately predicting practical three-dimensional compressible flow phenomena relevant to aeronautics and astronautics applications. This suggests that the proposed method serves as a robust tool for simulating complex compressible flows encountered in aerospace engineering.

To work on its comparison with other low-order methods, researchers have studied its comparison with solvers like the vortex method on the immersed boundary for two-dimensional flapping flight analysis [180]. The abstract discusses the use of computational fluid dynamics (CFD) methods for simulating the flapping flight of insects, focusing on the challenges of accurately modelling the dynamic interactions between moving insect bodies and the surrounding fluid. Two main categories of numerical methods are highlighted: incompressible inviscid methods and incompressible viscous methods. In the former category, exemplified by the discrete vortex method (DVM), the fluid is treated as incompressible and inviscid, utilizing discrete vortices to represent boundary and shear layers efficiently. However, the DVM may struggle to predict aerodynamic forces with high accuracy. On the other hand, incompressible viscous methods, exemplified by the immersed boundary–lattice Boltzmann method (IB-LBM), solve the full incompressible Navier–Stokes equations with moving boundaries, providing greater accuracy but at a higher computational cost. The study aims to compare these methods through a two-dimensional flapping flight analysis using a wing-body model, evaluating their performance in predicting aerodynamic forces, torques,

and body motion during flight. This comparative analysis will shed light on the strengths and limitations of each approach for simulating insect flapping flight dynamics.

2.5.2 LBM for multi-phase

Multi-phase simulations are pivotal for comprehending complex fluid behaviours, including phase separation and interface dynamics, across various engineering and scientific domains. Traditional methods often struggle with the computational demands of these simulations, especially in high-speed scenarios. However, the LBM's inherent efficiency and ability to handle complex interfaces make it well-suited for capturing such phenomena. In supersonic flows, where compressibility effects and shock wave interactions are prominent, the LBM's capability to handle compressible flows and model shock waves becomes particularly advantageous. Moreover, the LBM's scalability and ability to leverage parallel computing architectures make it suitable for large-scale simulations required in supersonic flow studies. By focusing on the application of LBM in simulating multi-phase flows, especially in supersonic conditions, this section aims to highlight how LBM can drive interdisciplinary research and innovation by advancing our understanding of complex fluid dynamics across different speed regimes. Work done in exploring this aspect of LBM recently in the last decade. The work done in employing a three-dimensional multiphase lattice Boltzmann model to investigate spontaneous phase transport within complex porous media [181]. Initially, the model's accuracy is validated against analytical solutions derived from Young's and Laplace's laws. Subsequently, the model is applied to simulate various scenarios, including droplet penetration into porous layers, liquid transport in porous channels, and extraction of droplets from porous substrates. The investigation considers the influence of multiple geometrical and flow parameters such as porosity, density ratio, Reynolds number, Weber number, Froude number [182], and contact angle. Key findings from the parametric study reveal significant relationships between these parameters and the behaviour of phase transport phenomena. Notably, increasing the Froude number leads to increased droplet spreading on a surface, while higher Reynolds number, Weber number, porosity, and liquid-air density ratio enhance droplet penetration into the substrate. Conversely, increasing the contact angle decreases both spreading and penetration rates simultaneously. Moreover, in the context of liquid transport through porous channels, higher porosity and Reynolds number correspond to increased penetration rates within the channel. Similarly, for the extraction of droplets from porous media, increasing gravitational force and/or porosity accelerates the extraction process.

2.6 Summary

After analysis and review of the literature, it can be said that much work for the improvement of VLM methods for supersonic regimes can be achieved by implementing the use of Mach cone influence for the supersonic conditions which will give a better understanding of collocation point at high speed to predict vortex edge separation. One of the limitations of TORNADO's supersonic module [17] is accurately predicting the position of collocation points on panels during its transition from subsonic to supersonic flow. To overcome this challenge Mach cone influence can be used in predicting the correct position of collocation points which helps in reducing the influence lost at supersonic speeds. Compressibility correction accuracy for supersonic flow, as with the current state uses the PG rule to re-evaluate the compressibility correction factor which works best for subsonic speed and loses its accuracy with increasing Mach number. Improving accuracy can be achieved by implementing the use of the Hybrid Lattice Boltzmann Method (HLBM) [183] which shows very promising results with the subsonic and supersonic flow with or without shocks. The proposed model is shown to be an efficient, accurate, and robust alternative to classical Navier–Stokes methods. Integration of this model with the TORNADO module can achieve high accuracy with improved product design capabilities for future supersonic aircraft.

In the case of conceptual design, the conventional approach is to carry out aerodynamic shape optimisation by introducing an optimisation loop over a set of results to work out the best choice of design which can satisfy the given performance index. Aerodynamic shape optimisation can be achieved with different types of optimizers for faster and more accurate predictions. Modern optimizers work with multiple dimensional or performance constraints so that the final result can be bounded within known requirements. In a quest to improve overall efficiency, a VLM can be used in an adjoint-based optimisation approach. The application of this approach has been explored with hypersonic and supersonic missile design [112]. A similar approach has been used to design both 2D and 3D aerodynamics of different wing configurations for transonic speeds [113]. VLM and the various developments since its inception have been shown to be highly suitable lower-order solvers to facilitate optimization work. The same could also be the case for supersonic aircraft design, with future work to advance the accuracy of VLMs for higher-speed regimes. The future outlook for improvements can also be achieved by including nonlinear and unsteady flow modelling for improving the prediction of unsteady aerodynamic characteristics and aeroelastic behaviours in supersonic flow. The prevalence of open-source VLM solvers in the research community means there is fertile ground to develop and implement such improved modelling capabilities and lead to easier integration with more complex and detailed modelling frameworks.

Chapter 3

Methodology

3.1 Introduction

The methodology adopted in this thesis is designed to extend the classical Vortex Lattice Method (VLM) for use in supersonic conceptual design. The central philosophy is to retain the computational efficiency of VLM while systematically correcting its principal deficiencies: (i) lack of unsteady modelling, (ii) absence of shock physics, and (iii) inability to account for compressibility. To achieve this, three methodological modules were developed and then integrated into a common framework:

1. **Unsteady VLM (UVLM):** Time-accurate vortex-lattice formulation, implemented in the Tornado code.
2. **Mach-cone shock modelling (TMHM–VLM):** Incorporation of Taylor–Maccoll oblique-shock relations to model cone-shaped supersonic disturbances.
3. **LBM-based compressibility correction (VLM–LBM):** A lattice Boltzmann patch solver that provides local density/pressure corrections.

This chapter presents the complete research methodology, describing the underlying theory, implementation details, integration strategy, and validation approach. The methodology is summarised schematically in Fig. 3.1, which shows the modular workflow: geometry $\rightarrow$ baseline VLM $\rightarrow$ UVLM $\rightarrow$ Mach-cone correction $\rightarrow$ LBM correction $\rightarrow$ outputs.

Fig.3.1 shows the solver layout of TORNADO and improvements done in the TORNADO and how it links with other subroutines for initialisation and post-processing. All the integration of improved subroutines is part of processors. The major contribution to the work is done in three vital subroutines highlighted in different colour boxes. The layout of these subroutines and their discussion is as:

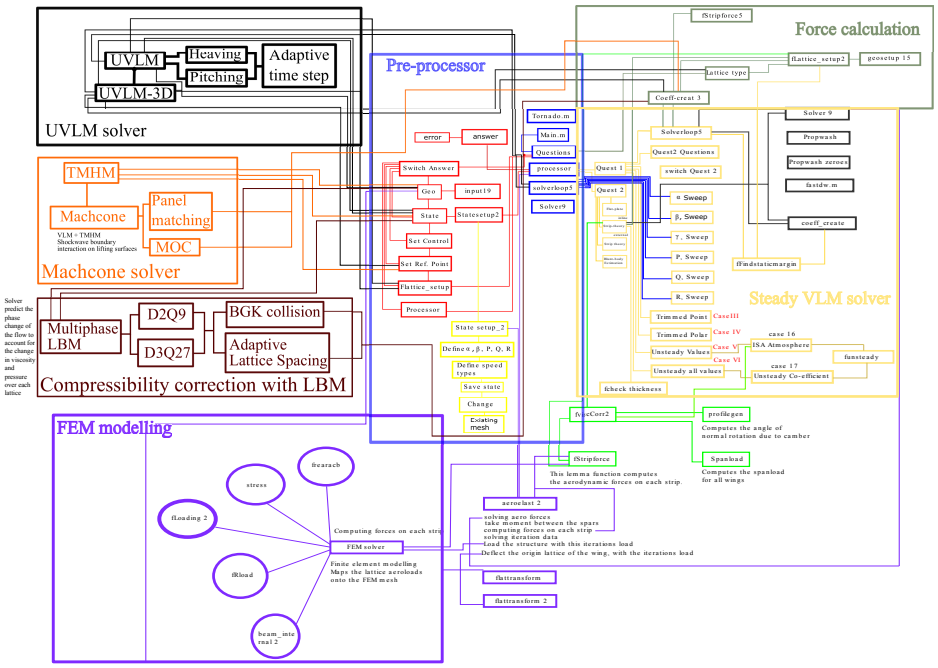


Fig. 3.1 Integrated methodology workflow.

- UVLM solver (Chapter-4)
  This solver utilises the conventional VLM from TORNADO containing the geometry data, flight state data and the ISA parameters as shown in Fig. 3.1 to simulate the dynamic response of lifting surfaces, accommodating both two-dimensional and three-dimensional wings. Users can specify the surface's dynamic behaviour through frequency-based or time-based step responses, enabling modelling of heaving, pitching, or combined heaving and pitching oscillations. This versatile tool provides a comprehensive approach to understanding and analysing the aerodynamic behaviour of complex wing geometries under varying dynamic conditions.

- Machcone solver (Chapter-5)
  This solver is designed to accurately model shockwave-boundary interactions on lifting surfaces, employing the original TORNADO file for pre-processing tasks such as geometry initialisation, mesh setup, and flight state data as shown in Fig.3.1. It incorporates ISA parameters to simulate shock cones around lifting surfaces, leveraging the panel matching technique to treat each panel as an individual lifting surface. This approach enhances drag calculations for surfaces operating at supersonic speeds, offering a comprehensive solution for analysing aerodynamic performance in high-speed regimes.

- Compressibility correction with LBM (Chapter-6)
  The compressibility correction solver employs the Lattice Boltzmann Method (LBM) with D2Q9 and D3Q27 lattices, utilizing the BGK collision factor. Initialization of the solver follows a similar process to previous solvers, leveraging TORNADO files as shown in Fig.3.1. This novel modelling approach enhances lattice concentration over lifting surfaces, leading to more accurate velocity changes at supersonic speeds. Additionally, it facilitates multiphase analysis, contributing to improved understanding and prediction of aerodynamic behaviour in compressible flow regimes.

Further discussion on these solvers can be found in their respective chapters, along with results and validation against conventional high-order CFD methods, emphasising their credibility and relevance. It also underscores their potential for advancing supersonic analysis while considering cost-effectiveness. This comprehensive approach adds depth to the assessment of their applicability and significance within the field of aerodynamics.

3.2 Baseline Solver: Classical Vortex Lattice Method

3.2.1 Governing assumptions

The VLM is derived under the assumptions of incompressible, inviscid, irrotational flow about thin lifting surfaces. Each wing surface is discretised into quadrilateral panels, with a horseshoe vortex placed at the quarter-chord of each panel. Boundary conditions are enforced at the three-quarter chord control points, satisfying the no-penetration condition. The circulation distribution Γ is obtained by solving the aerodynamic influence coefficient (AIC) system:

$$A\Gamma = V_\infty n, \tag{3.1}$$

where A is the influence matrix, V_∞ is the freestream velocity, and n is the local surface normal vector. Once the circulation is obtained, forces are computed using the Kutta–Joukowski theorem:

$$L' = \rho_\infty V_\infty \Gamma, \tag{3.2}$$

and induced drag is recovered in the Trefftz plane.

3.2.2 Tornado framework

The Tornado VLM code (developed at KTH) was selected as the baseline solver. Tornado offers modular geometry definition, flexible lattice resolution, wake alignment, and aerodynamic state input. Its open-source MATLAB implementation allowed straightforward insertion of new modules. However, Tornado is limited to steady, incompressible aerodynamics. The methodology of this thesis therefore builds upon Tornado with three new modules (UVLM, TMHM, LBM) that extend its validity.

3.3 Extension I: Unsteady Vortex Lattice Method (UVLM)

3.3.1 Motivation

Classical VLM is steady-state only, preventing the simulation of oscillating wings, gust encounters, or transient manoeuvres. For supersonic configurations such as SCALOS or Concorde, unsteady aerodynamics strongly influences stability and control predictions.

3.3.2 Governing equations

UVLM is formulated by introducing a time-marching scheme. At each time step, new bound vortices are shed into the wake; the wake is then convected downstream with the local velocity field. The no-penetration condition is enforced at each time step, yielding a system:

$$A(t)\,\Gamma(t) = RHS(t), \tag{3.3}$$

where $A(t)$ incorporates both bound and wake influences. The wake is modelled as a series of free vortices, convected using a second-order Runge–Kutta scheme. The Kutta condition is applied at the trailing edge at each time step.

3.3.3 Motion kinematics

Heave and pitch motions are prescribed as:

$$h(t) = h_0 \sin(\omega t), \tag{3.4}$$

$$\theta(t) = \theta_0 \sin(\omega t + \phi), \tag{3.5}$$

with the combined effective angle of attack:

$$\alpha_{\text{eff}}(t) = \alpha + \theta(t) + \tan^{-1}\left(\frac{\dot{h}(t)}{U_\infty}\right). \tag{3.6}$$

3.3.4 Implementation

The Tornado data structure was modified to:

- Add a time loop with wake convection,
- Update control points and freestream incidence per time step,
- Project aerodynamic forces into free-stream axes to ensure positive drag,
- Store unsteady coefficients $C_L(t), C_D(t), C_M(t)$.

3.3.5 Validation

UVLM results were validated against benchmark airfoil and wing cases, showing approximately 82% correlation with CFD data and $\sim 24\times$ faster runtime. Discrepancies were largest at high angles of attack, where flow separation—outside UVLM's assumptions—becomes important.

3.4 Extension II: Mach-Cone Shock Modelling (TMHM–VLM)

3.4.1 Motivation

VLM neglects shock physics, making it unsuitable for supersonic flows. To address this, the Taylor–Maccoll solution for conical oblique shocks is coupled with VLM to impose local corrections on affected panels.

3.4.2 Theoretical formulation

For a given Mach number M_∞ and local flow deflection angle θ, the shock angle β is obtained from the oblique shock relation:

$$\tan\theta = 2\cot\beta \cdot \frac{M_\infty^2 \sin^2\beta - 1}{M_\infty^2(\gamma + \cos 2\beta) + 2}. \tag{3.7}$$

The Taylor–Maccoll equations then provide the post-shock velocity components within the conical region. Panels inside the Mach/shock cone are corrected by replacing their local boundary conditions with these post-shock states.

3.4.3 Implementation

1. Construct Mach cone from the leading edge with half-angle β.
2. Intersect cone with lattice to identify affected panels.
3. Update AIC rows and boundary conditions using Taylor–Maccoll derived pressure jumps.
4. Re-solve circulation system to recover corrected lift and drag.

3.4.4 Validation

Test cases on airfoils and wings showed drag errors of 2–4% compared to RANS, with overall accuracy of 90–95%. Runtime was less than 1 h compared to 24 h for equivalent RANS runs on the same hardware.

3.5 Extension III: LBM-based Compressibility Correction

3.5.1 Motivation

Classical VLM assumes incompressibility. Prandtl–Glauert correction provides a simple factor but fails near shocks or in strong compressible regimes. To address this, a small Lattice Boltzmann Method (LBM) solver was developed as a local patch to supply compressibility factors.

3.5.2 Governing formulation

A D2Q9 BGK lattice is used:

$$f_i(\mathbf{x}+c_i\Delta t, t+\Delta t) = f_i(\mathbf{x},t) + \frac{1}{\tau}\left(f_i^{eq} - f_i\right) + \Psi_i, \tag{3.8}$$

where f_i^{eq} is the equilibrium distribution and Ψ_i is a forcing term capturing compressibility. Density and pressure are extracted as:

$$\rho = \sum_i f_i, \qquad p = c_s^2 \rho. \tag{3.9}$$

The correction factor $\Psi(M)$ is defined as the ratio:

$$\Psi(M) = \frac{\rho}{\rho_\infty}, \tag{3.10}$$

which is used to scale panel influence coefficients in the VLM solution.

3.5.3 Coupling strategy

1. LBM is run on a sparse surface-aligned lattice around critical regions (leading edges, canards).
2. At each iteration, local $\Psi(M)$ values are computed.
3. The AIC matrix rows for affected panels are multiplied by $\Psi(M)$.
4. VLM is re-solved with updated influence coefficients.

3.5.4 Validation

Comparisons with k–ω SST RANS results showed $\sim 92\%$ agreement in C_p distributions and aerodynamic coefficients. Deviations occurred near leading and trailing edges, linked to boundary condition sensitivity in LBM.

3.6 Integrated Framework

3.6.1 Workflow

The three extensions were combined into an integrated workflow:

1. Geometry and flight condition input $\rightarrow$ Tornado baseline VLM.
2. Time-marching UVLM for unsteady analysis.
3. Shock correction via TMHM Mach-cone model.
4. Density/compressibility correction via LBM patches.
5. Output of aerodynamic coefficients (C_L, C_D, C_M) and pressure distributions.

This modular approach allows each correction to be toggled independently or applied sequentially.

3.6.2 Validation philosophy

Validation was staged at three levels:

- **Canonical tests:** 2D airfoils and delta wings for C_L, C_D versus M_∞ and α.
- **Intermediate:** 3D swept/dihedral wings.
- **System-level:** SCALOS and Concorde case studies.

At each level, results were compared against URANS simulations (STAR-CCM+ with k–ω SST, documented meshes and domains) and experimental data where available. Runtime comparisons were included to demonstrate the speed advantage.

3.7 Summary

This chapter has outlined the complete methodology adopted in the thesis. Starting from Tornado's baseline VLM, three extensions were implemented—UVLM, TMHM-based Mach-cone modelling, and LBM compressibility corrections. The integration workflow enables fast, modular aerodynamic predictions across a wide range of supersonic conditions. Subsequent chapters present the detailed development and validation of each module, followed by case studies demonstrating the methodology's system-level applicability.

Chapter 4

Unsteady modelling

This chapter introduces the evolution of the Unsteady Vortex Lattice Method (VLM) in tandem with the TORNADO VLM software, developed using MATLAB. The primary objective is to augment the capabilities of conventional VLM specifically for unsteady analysis, thereby enhancing both accuracy and computational efficiency in forecasting the dynamic characteristics of aerodynamic surfaces. Central to this investigation is the utilization of the frequency response of a generalized VLM, which effectively mitigates the expenses linked with employing higher-order CFD solvers. The methodology operates within an adaptive time-step framework, substantially diminishing truncation errors and bolstering precision. By calculating induced flow, this unsteady solver determines the temporal forces acting on individual panels. This unsteady VLM emerges as a rapid solver, facilitating the swift prediction of dynamic behaviour in lifting surfaces and enabling its application in expedited preliminary design analyses.[1]

[1]The findings from this chapter is published as, H. and Thomas, P., 2023, July. Unsteady Vortex Lattice Method in TORNADO: A Fast and Cost-Effective Solver for Dynamic Analysis of Supersonic Lifting Surfaces. In Aerospace Europe Conference 2023, Joint 10th EUCASS–9th CEAS Conference. https://doi.org/10.13009/eucass2023-092

4.1 Contribution of this Chapter

The main contributions of this chapter are summarised as follows:

- **Integration of UVLM into Tornado:** The UVLM is embedded within the Tornado framework, enabling both steady and unsteady aerodynamic predictions within a unified solver environment.
- **Oscillatory motion capability:** The solver supports independent heaving or pitching oscillations, as well as coupled motion where heave and pitch occur simultaneously. The instantaneous effective angle of attack is expressed as

$$\alpha_{\text{eff}}(t) = \alpha + \theta(t) + \tan^{-1}\left(\frac{\dot{h}(t)}{U_\infty}\right),$$

  capturing the combined kinematics of oscillatory motions.
- **Adaptive time-step strategy:** An adaptive stepping scheme is implemented to increase resolution during rapid motion changes while reducing cost during quasi-steady phases, improving dynamic modelling accuracy.
- **Three-dimensional unsteady extension:** The UVLM is extended to 3D, making use of Tornado's lattice definitions, wake modelling, and Prandtl–Glauert compressibility corrections.
- **Validation and performance:** The solver is validated against experimental and numerical benchmarks, showing close agreement with CFD while running significantly faster, highlighting its suitability for preliminary design studies.

4.2 Introduction

Fluid dynamics is a sub-discipline of fluid mechanics that encompasses aerodynamics and hydrodynamics. These fields are important for engineering design and analysis. Unsteady aerodynamics has been studied for over a century to better understand the flow around aircraft undergoing unstable dynamic motion due to gusts, side-slip or rapid manoeuvring. This analysis is critical for designing aerofoils and other aerodynamic implements. The theory of unsteady aerodynamics began in early 1920 with Theodorsen's approach [184] for analysing oscillating aerofoils. The Wagner function [185] was then used to study the separated flow in an unsteady manoeuvre, while the Kussner function [186] resolved the issue of the shed

vortex's gradual reduction in influence on the aerofoil over time. The VLM, which emerged in 1937 [31], revolutionized the analysis of the aerodynamic sections of wings by dividing them into panels, providing a sectional overview that was widely used in early aircraft design. VLM can also be used to solve unsteady aerodynamics based on the unsteady lifting line theory [187] for dynamic motion that changes over time. Unsteady aerodynamics is inherently complex to model and often demands significant computational effort, particularly when using high-order CFD methods. To improve the efficiency of computational simulations for unsteady modelling, low-order methods like VLM have proven to be effective [188]. Low-order modelling (LOM) is an increasingly popular approach that utilizes classical potential and lifting line theory to create new design tools focusing on reduced parameters. This approach is essential for early design work, as it reduces the number of parameters, complex calculations, and the time required [189].

The fundamental use of low-order solvers is to reduce the number of degrees of freedom required to produce the results for the required performance. VLM or panel methods are effective low-order approaches in calculating the pressure change with the help of velocity change over a wing divided into several panels [190]. The change in velocity is calculated with induced vortices generated over each panel. Each panel is associated with its control point (collocation point) from which the velocity potential is computed. The resultant velocity, with relation to the free-stream velocity and Kutta's condition [191] provides the solution of the vortex strength, and through the integration of those, lift and induced drag forces are calculated. The output of this method can predict the stability derivatives and aerodynamic characteristics of flying objects. A key difference between a steady solver and unsteady VLM (UVLM) is the need to use time steps whereby each vortex's calculation over a panel is obtained at each time step as shown in Fig. 4.1. The use of LOM has been in place for more than a decade with its first-ever unsteady implementation in the 1990s [189] . These methods are computationally efficient enough to predict the unsteady aerodynamic loads [192] . They are ways of solving for three-dimensional flow fluids like pressure and velocity when compared with conventional CFD-based interpolation methods. Initially, the LOM approach is divided into two subroutines: time-based response and frequency-based response. With the time-based response, every parameter starting from flow speed, location of the aerofoil and many others will be changed concerning time to model the dynamic nature of the flow around a vibrating or heaving aerofoil. With the frequency-based response, these changes are more dominated by frequency change than time. This approach was used in the aeroelastic analysis of active control for flutter and gust response [193] . Because of this effectiveness, LOM approaches are widely used in the early design process stages to speed up the unsteady analysis.

TORNADO is a MATLAB-based implementation of VLM [194] that is being used as a framework to develop and implement LOM using UVLM. The UVLM equations are defined in matrices for easy and quick calculations. The solver allows the user to define the heaving motion with two parameters i.e., heaving amplitude which is the function of the base chord of an aerofoil, and heaving frequency which is the function of true airspeed. The location of the most recently shed wake is calculated from the initial time-step response. The subroutine solves for two-dimensional aerodynamic analysis of an aerofoil of a stated wing of a designed aircraft under given flow conditions. The proposed program is a quick unsteady solver which differs from conventional CFD modelling concerning solver time and analysis. The use of such LOM methods is beneficial for the preliminary design of the next generation of aircraft. The developed solver can predict the lift, drag and moment coefficients in seconds which drastically cuts computational costs. Since it is based on the LOM approach, the solver can run on less powerful computer systems as compared to the ones used for CFD simulations. Cost is one of the major reasons for design problems in the next generation of aircraft designs, this solver holds good potential for implementation in early design optimisation work. In this paper, the results of the 2D UVLM code are presented and compared against a higher-order CFD simulation after its validation with experimental and analytical results.

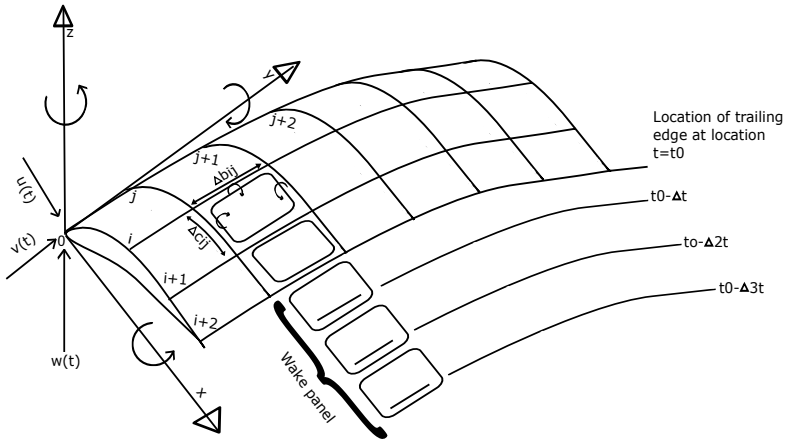


Fig. 4.1 Unsteady vortex lattice representation [6]

4.3 Theoretical Approach: Unsteady vortex lattice method

When visualizing certain flights the inclusion of wake analysis is always helpful in tracking the motion of an aircraft. Studies have shown that Theodorsen's theory [184] is remarkable in predicting unsteady aerodynamics of the lifting surface and still holds the same for high-intensity sinusoidal motions. Predicting the non-periodic motion is done by superimposing Wagner's function [185]. Besides these classical theories, UVLM can be used for lower-order modelling. This method is extended to implement the 2D unsteady wake from the trailing edge. The boundary conditions and influence matrix from wake and bound vortices have been re-derived concerning time which changes for every time step response. The lift coefficient is derived from the rate of change of bound circulation concerning time [195]. The field of unsteady aerodynamics completely depends on past motion, i.e., the history associated with the motion concerning time, and the repetition of that motion which means how often it repeats (which is dependent on frequency). It has been seen from all the unsteady theories of the past [141], [73], [74], [125] that every flow field is time dependent which depends on its state, velocity and temperature of the flow. The major difference between unsteady and steady VLM is the inclusion of wake modelling to account for the 3D interference during unsteady flow conditions. The transition of conventional VLM to UVLM is mainly driven by viscous phenomena, such as the need to study the leading-edge separation for delta wings. The UVLM helps in calculating the velocity potential with the help of boundary conditions which is the basis of the unsteady potential flow theory [196].

4.3.1 Oscillating aerofoils

Oscillating aerofoils under the influence of unsteady motion such as heaving, pitching and the combination of both, have been used to examine high forward thrust and help in analysing unsteady aerodynamics [197]. Such complex modelling is highly effective in analysing the flapping wings of insect or bird flight, MAV (Micro air vehicle), UAV (unmanned air vehicle), and VTOL (Vertical take-off and landing) flights [25]. The use of this modelling is also seen with aeroelastic analysis of wind turbines under heavy crosswinds. The motion of an aerofoil which is time dependent is essential in visualising vorticity contours around an aerofoil oscillating with heaving amplitude. The vorticity fields are initially generated at low speeds and are governed by time instances. The heaving motion of an aerofoil shown in Fig. 4.3 is governed by its frequency and amplitude [198] which in this case is the function of the base chord and velocity of flow. The heaving motion also helps in understanding the vortex shedding [199], which is seen after a single time step. The generated vortices will shed off from the trailing edge which, when separated over time, forms the bound vortex which

creates the high-pressure region under the aerofoil essentially increasing the lift generation. Pitching motion alters the instantaneous orientation of an aerofoil, producing a time-varying

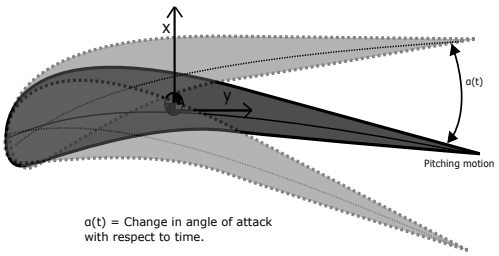


Fig. 4.2 Pitching aerofoil with time-step.

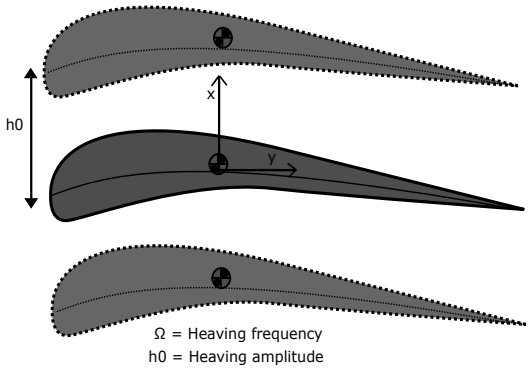


Fig. 4.3 Heaving aerofoil with time-step.

angle of attack in each time frame as shown in Fig. 4.2 and 4.3. The body axis of an aerofoil is fixed at MAC (Mean aerodynamic chord) to provide a reference system around which the body axis will change. A pitching aerofoil helps in visualising both leading edge and trailing edge vortex shedding [200]. The recent investigation of this motion is done with different aerofoil arrangements [201] for visualising the flow separation over pitching and plunging aerofoils. That research proves the viability of accurately predicting a surging and pitching wing vortex formation. It shows that the accuracy holds good for potentially low order models [25] and justifies its use with reduced order modelling. The paper by Moriche *et al.*[202] shows the use of this approach for predicting the leading edge suction parameter, which states that the given aerofoil before experiencing flow separation, an aerofoil can sustain a finite leading-edge suction force. This parameter is crucial for predicting the onset of vortex formation.

4.3.2 Boundary conditions for an oscillating aerofoil

The coordinate system for a heaving aerofoil is derived for a time as shown in Fig. 4.4. As the heaving motion is based on heaving frequency and amplitude, the location of the aerofoil moving up and down only affects the y-axis position, whereas the x-axis is the function of flow speed and time. At each discrete time step, the location of shed vortices from the trailing edge is re-calculated to trace its path. As time $t > 0$ the aerofoil starts moving up following the time-dependent path.

$$s_x = -v \times t \qquad (\mathrm{x}-\mathrm{coordinate}) \tag{4.1}$$

$$s_y = -h_0 \times \sin(\omega \times t) \qquad (\mathrm{y}-\mathrm{coordinate}) \tag{4.2}$$

The unsteady motion of an aerofoil under flow concerning the body frame is given by:

$$(\Delta\phi - V - \omega \times r) \cdot n = 0 \tag{4.3}$$

where V is the flow velocity, $\Delta\phi$ is the change of velocity potential, ω is the frequency of rate of change of body axis for time, r is the position vector from body axis to most recently shed wake and n is the normal vector. The trailing vortex location is given by:

$$x = (C + dx) \times \cos(\alpha) + s_x \tag{4.4}$$

$$y = -(C + dx) \times \sin(\alpha) + s_y \tag{4.5}$$

where c is the base chord, dx is the location of most recently shed vortices concerning time, α is the angle of attack and s_x, s_y are the body coordinates concerning time t. The location of the bound vortices for t is given by:

$$x_1 = \frac{1}{4} \times C \times \cos\alpha + s_x \tag{4.6}$$

$$y_1 = \frac{1}{4} \times C \times \sin\alpha + s_y \tag{4.7}$$

Calculation of wake downwash using Biot-Savart Law is given by:

$$W = u \times \sin\alpha + w \times \cos\alpha \tag{4.8}$$

Here W is the wake-induced downwash, while u and w are the sum of all induced velocities from the wake elements of interest. The lift and drag on an aerofoil are because of only

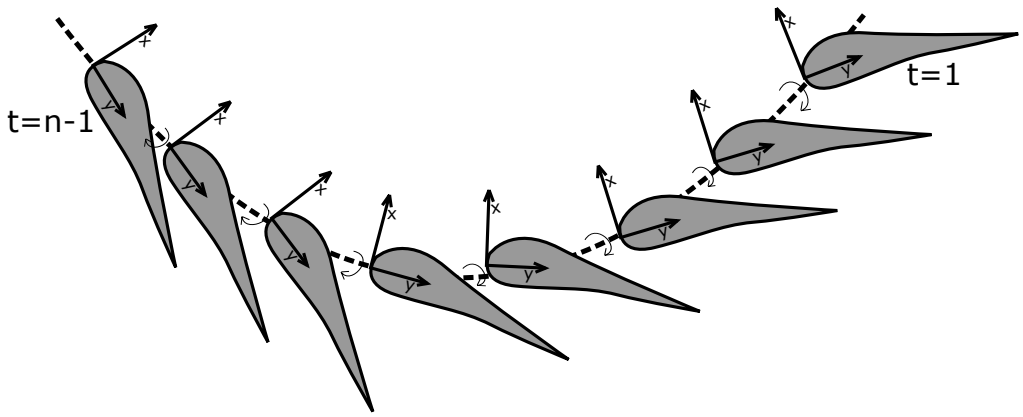


Fig. 4.4 Aerofoil location with time step

normal forces and rate of bound circulation concerning time which is given by:

$$\overline{q} = \frac{1}{2} \times \rho \times (V + w(it))^2 \tag{4.9}$$

$$C_l = \frac{L(it)}{\overline{q(it)} \times C} \tag{4.10}$$

$$C_d = \frac{D(it)}{\overline{q(it)} \times C} \tag{4.11}$$

Here $\overline{q}$ is the kinetic (or dynamic) pressure, $w(it)$ is induced velocity for a single iteration, ρ is the density of air at sea level, $L(it)$ is the calculated lift for a single iteration, $D(it)$ calculated drag (induced drag) for a single iteration. It can be seen that the given unsteady boundary conditions follow Theodorsen's approach as the position of the geometry is a function of time and frequency of the motion shown in equations 4.1 and 4.2. The unsteady motion of a flow under an aerofoil is defined as the change in frequency of the motion multiplied by the position vector r and the normal vector n. The change in velocity potential is calculated over each change in this position shown in equation 4.3.

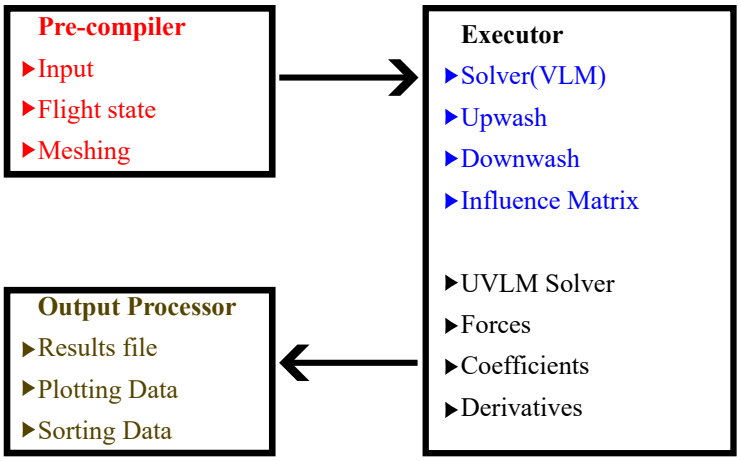


Fig. 4.5 TORNADO layout with UVLM.

4.4 Implementation with TORNADO

An aircraft wing is defined in TORNADO from the information provided by the user. The wing geometry and flight state conditions are provided by the user, from which the programme works out the ISA conditions for the density of air and other relevant information. After completion of initialisation, the programme moves to the solver, which gives the user the option to select the mesh type and solver type according to his requirements. After that, the results are saved for future analysis and graphs are plotted for visual and data analysis as shown in Fig. 4.5 This solver has good potential for quickly analysing the aerodynamics for a defined geometry under given conditions [6]. To model the dynamic motion of an aerofoil, a new module has been developed to implement the two–dimensional Unsteady Vortex Lattice Method (2D UVLM). This solver predicts the heaving and pitching motion of an aerofoil as a function of the frequency and amplitude of the imposed oscillation, the motion being dependent on the freestream velocity and the chord length. A time-stepping procedure is incorporated into the UVLM solver to capture the evolution of aerodynamic characteristics with respect to time [203].

For consistency, the UVLM module has been coupled with the existing TORNADO framework. The initialisation parameters required by the UVLM solver are automatically extracted from the base TORNADO code and directory structure. The generated results are written into the same results folder (with distinct filenames) so that steady and unsteady cases can be compared side by side. Figure 4.6 shows, in simple terms, how the new unsteady VLM (UVLM) sits on top of the existing TORNADO code. The user provides the geometry, flight state, lattice resolution, and motion settings; TORNADO then builds the geometry and lattice, forms the steady VLM influence matrix $\mathbf{A}$, applies the Kutta condition, and computes a baseline circulation Γ_0. UVLM takes this baseline plus the lattice and, at each time step, sets up the unsteady problem (heave, pitch, or combined motion), updates the wake, solves for the instantaneous circulation $\Gamma(t)$, and computes the stepwise forces and coefficients $C_L(t)$, $C_D(t)$ and $C_M(t)$. Optional correction hooks can be applied as needed: Prandtl–Glauert for subsonic compressibility, Mach-cone/Taylor–Maccoll relations for $M_\infty > 1$, and simple viscous options (e.g. C_f estimates). Finally, results are passed to post-processing to generate plots, stability measures, and files for comparison. The arrows in the diagram simply indicate that geometry and steady data flow from TORNADO to UVLM, and that UVLM's time-dependent output flows onwards to the post-processor and figures.

The interaction between TORNADO and the newly implemented UVLM routines can be summarised as follows. The core TORNADO routines handle user input, such as geometry definition, flight-state specification, and lattice generation. The lattice setup and geometry processing modules provide the basis for the aerodynamic solvers, which evaluate panel

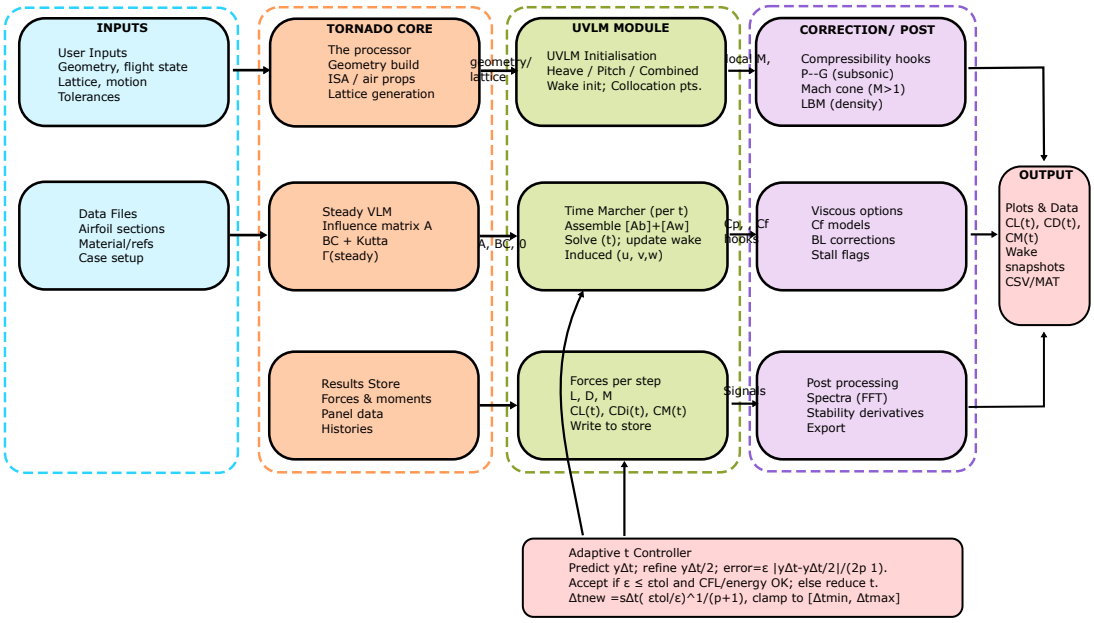


Fig. 4.6 Integration of TORNADO core modules with developed UVLM.

forces for the defined geometry. The post-processing subroutines then generate aerodynamic coefficients and collate the results.

Within this framework, two additional solvers have been introduced for unsteady analysis: 1.UVLM (2D): models aerofoil oscillations in heave or pitch as functions of frequency and time-step size (scaled with chord length). An adaptive time-step mechanism is implemented, refining the temporal resolution during rapid transients and coarsening it during quasi-steady phases. This improves accuracy and reduces computational cost. 2.UVLM-3D: extends the approach to finite wings by using the TORNADO-generated lattice. At each time step, the solver updates the wake structure and accounts for shed vortex positions, altering the induced velocity field. This provides a more realistic prediction of unsteady three–dimensional aerodynamic behaviour.

The results of both UVLM solvers are passed to the coefficient generation subroutines, ensuring compatibility with the existing TORNADO output format. This integration expands TORNADO's capability from steady-state analysis to the study of unsteady, oscillatory aerofoil and wing motions.

The introduction of this solver with a LOM approach gives the potential to solve the unsteady aerodynamics of aerofoil in a transient time frame, which drastically reduces the

cost of the design of the next generation of aerofoil design. The solver is integrated into the TORNADO user interface to provide the user with a way to define and alter any changes required before executing it. The solver is driven by two key user-defined parameters: uniform flow speed and time step, along with additional inputs such as angle of attack, frequency, and amplitude. The work in future is to improve its utilisation with other sections of TORNADO such as integrating with the steady solver to provide a comparative relation between steady and unsteady motion of an aerofoil under similar conditions, accounting for changes in force coefficients and extracting data for its application into design optimisation.

The advancement of UVLM in the three-dimensional solver, the angle of attack position with body reference axis is redefined for every time step as:

$$\alpha_x = 0.75 \times C \times \cos\alpha \times sx \tag{4.12}$$

where α_x is the angle of attack position in the direction x, C is the length of the chord, and sx is the change in the position of the wing in the direction x with each time step. Similarly, α_y is defined as :

$$\alpha_y = 0.75 \times C \times \sin\alpha \times sy \tag{4.13}$$

where α_y is the angle of attack position in the direction y, sy is the change in the position of the wing in the direction y with each time step. For preprocessor initialisation location of the lattice is obtained from the lattice setup code of TORNADO which helps as initially places a lattice predefined with a defined as the number of lattices on both sides of the aircraft through its wingspan. To define the wake of this aircraft the farstream coordinates are defined which are the function of cosine and sine of the defined angle of attack.

$$\begin{aligned} infx &= infdist \times \cos(\alpha) \times \cos(\beta) && (4.14)\\ infy &= -infdist \times \sin(\beta) && (4.15)\\ infz &= infdist \times \sin(\alpha) \times \cos(\beta) && (4.16) \end{aligned}$$

In the above equations, $infdist$ is the variable defined for infinite distance based on the reference position of the geometry defined in the TORNADO geometry setup. β is the defined side slip for the current flight state. The position of these lattices changes with every time step with a defined amplitude and frequency of the motion which is defined as:

$$\begin{aligned} x &= infx + lattice.XYZ(t, c(s), 1) && (4.17)\\ y &= infy + lattice.XYZ(t, c(s), 2) && (4.18)\\ z &= infz + lattice.XYZ(t, c(s), 3) && (4.19) \end{aligned}$$

In the above equations, s is the variable defined in the TORNADO lattice setup code file which helps to define which wing of the geometry we are currently working with. Using $infx$ and x lattice position terms with dx as the distance between the wake elements and the initial element position, one can derive the wake coordinates for each panel in the lattice as:

$$INF1(i,1,1) = lattice.XYZ(i,1,1) + infx + dx(i,1) \tag{4.20}$$

The above equation $INF1$ is defined as the wake initialisation coordinate which is the addition of the $lattice.XYZ$ x coordinate, $infx$ and dx which are the flight state functions. This set of equations runs for the number of S wings symmetrical or nonsymmetrical which is defined by the user. Similarly, the equation for the coordinates y and z for the wake initialisation is derived as follows:

$$INF1(i,1,2) = lattice.XYZ(i,1,2) + infy + dy(i,1) \tag{4.21}$$

$$INF1(i,1,3) = lattice.XYZ(i,1,3) + infz + dz(i,1) \tag{4.22}$$

Now to determine the influence of the wake element of interest, the distance between the wake element and the wake element of interest is calculated as is for the 2-D UVLM code;

$$rx = lattice.XYZ(a,i,1) - lattice.XYZ(1:a,1:it,1) \tag{4.23}$$

This difference in wake elements helps to identify the distance of the wake from the initial position of the geometry and how it influences the change of velocity potential for each time step. Similarly, the differences in the y and z directions are also derived as:

$$ry = lattice.XYZ(a,i,2) - lattice.XYZ(1:a,1:it,2) \tag{4.24}$$

$$rz = lattice.XYZ(a,i,3) - lattice.XYZ(1:a,1:it,3) \tag{4.25}$$

The parameters rx, ry, rz are used to calculate the change in velocity potential in each direction giving velocities u, v and w. This is then used to move the panel location concerning local velocity with increasing time-step as:

$$lattice.XYZ(1:a,1:it,1) = lattice.XYZ(1:a,1:it,1) + u(1:it) \times dt \tag{4.26}$$

$$lattice.XYZ(1:a,1:it,2) = lattice.XYZ(1:a,1:it,2) \tag{4.27}$$

$$lattice.XYZ(1:a,1:it,3) = lattice.XYZ(1:a,1:it,3) + v(1:it) \times dt \tag{4.28}$$

This helps in plotting the implicit wake of an oscillating wing which is defined by the oscillatory frequency and the amplitude of the motion. This change in motion can be defined by using the time-dependent or frequency-dependent motion. The layout for supersonic configuration is shown in Fig. 4.7 which defines the distribution of the above equations in different sections of the preprocessor, the solver and the postprocessor. The addition of an adaptive time step is not added into this arrangement, as it increases the computational time for the supersonic simulations.

4.5 Results and discussion

The validation of the solver is done by comparing it with experimental [23] and numerical data [24] from early research on unsteady lift predictions. The results are compared for the 2-D profile modelled at the Mach numbers of 0.5 and 0.8 with the profile divided among 100 panels chordwise. The variation of the two-dimensional indicial lift coefficient per angle of attack with chord length travelled is compared with the same boundary conditions as that of the literature.

Compared results as shown in the Fig. 4.8 show the non-steady region at $Ut/c < 1.3$ which is close to experimental data ($Ut/c < 1$). The plot shows the compared values of C_{L_α} in the units of per radian to that of the chord length travelled in meters. The change into a steady region starts after $Ut/c > 1.4$. Compared results of Change in C_{L_α} with travelled chord length shows a percentage in the error of 26% compared to that of Lomax [23]. The C_{L_α} predictions

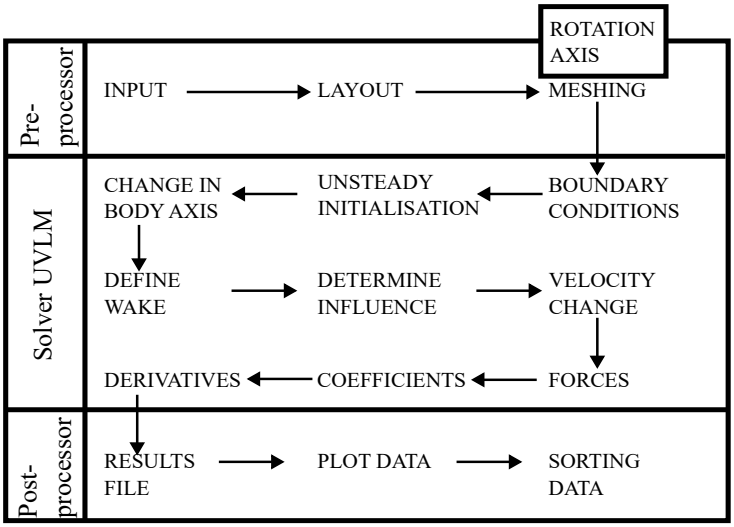


Fig. 4.7 Supersonic 3-D UVLM subroutines in TORNADO

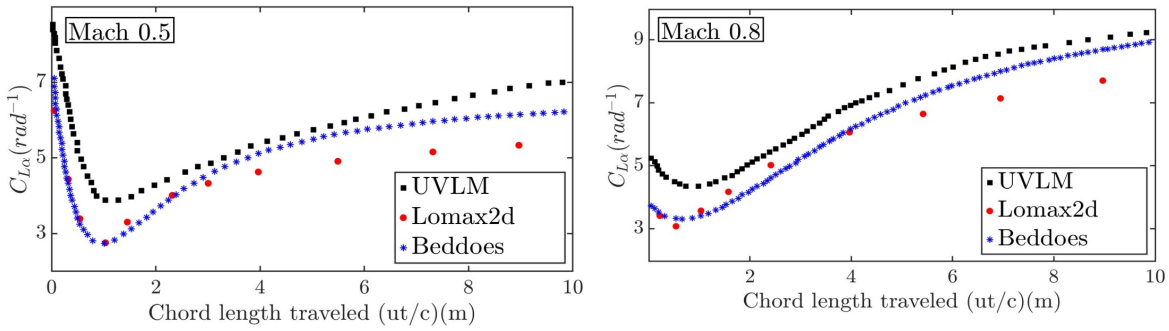


Fig. 4.8 Comparison of change in lift coefficient with alpha in chord length travelled from UVLM 2D and literature [23],[24].

of higher Ut/c have a higher percentage of error when compared with an experimental and numerical model. The deviations between the expected results and the experimental model arise due to inherent limitations of the conventional Vortex Lattice Method (VLM). First, as the chord length increases, the accuracy of the VLM diminishes because it relies on a fixed number of panels, leading to insufficient resolution. This issue can be mitigated by increasing the number of panels, which enhances the influence terms and improves accuracy. Second, the solver struggles to accurately account for flow compressibility, a known limitation of VLM. Addressing this requires incorporating corrections such as Prandtl-Glauert's compressibility factor, which is limited to subsonic regimes, or adopting a hybrid approach. The latter involves using conventional CFD as a plugin to refine simulations based on initial results from the UVLM solver, enabling faster convergence and improved predictions. The results from UVLM for two-dimensional analysis using NACA-0012 aerofoil are compared with the literature. The simulation was run for the oscillatory frequency of 4Hz and 1Hz of motions. Defining the motion of an aerofoil is a combination of pitching and heaving as a function of frequency and time-step. The results plots are the comparison between the C_L with increasing time (sec.). Comparison between results obtained from the UVLM code and CFD simulation [25] is shown in Fig.4.9-4.12. The results obtained from the code show good agreement with those from [25]. There is a slight shift in the graph which is because of the time-step error, which is not mentioned in the literature. The graph shows accuracy in predicting the force coefficients for the same initial conditions for low-frequency motion. In UVLM, drag is evaluated from wake-induced downwash and the rate of change of bound circulation. This ensures that drag corresponds to the physical projection of aerodynamic forces in the flow direction, avoiding negative drag artefacts that appear if lift components are misprojected.

Downwash and rate of change of bound circulation from the shed and bound vortices are used to calculate the drag force, which is the induced drag generated over an aerofoil. In CFD results by Kurtulus [25], the drag force is calculated by dividing the force on the aerofoil in the opposite direction to the flow, which is obtained by solving the URANS equations and dividing them by the cross-sectional area of the aerofoil. From the results, it is apparent that the UVLM predictions are reasonably similar when compared with the higher-order CFD simulation results, with a level of accuracy suitable for the preliminary analysis of unsteady effects of aerofoil motion.

The three-dimensional progress with UVLM is achieved by recalculating the new datum point to change the orientation of the panels concerning the defined angle of attack by the user. The lattice was imported from the TORNADO lattice code and is redefined to accommodate the change in no of panel concentration along the wing chord and span. The wake initialisation is done by calculating the infinite distance for x, y and z directions with defined state angles i.e., angle of attack and side angle. Wing panel orientation is redefined concerning the angle of attack and side-slip angle so the wake reflects that with time. Then the influence of wake elements is defined on adjacent panels and how it reflects with time-step. Then the difference in distance is calculated between all the wake elements and wake elements of interest which were later used to calculate the velocity potential for all directions. The change in wake has been achieved by moving the wake elements with local velocity. The plot for which is shown in Fig. 4.13.

With an extension of the three-dimensional build of the unsteady solver, the lattice code derived from the TORNADO base code is modified and improved to account for the change in angle of attack with time, also enabling the user to change the dihedral angle and

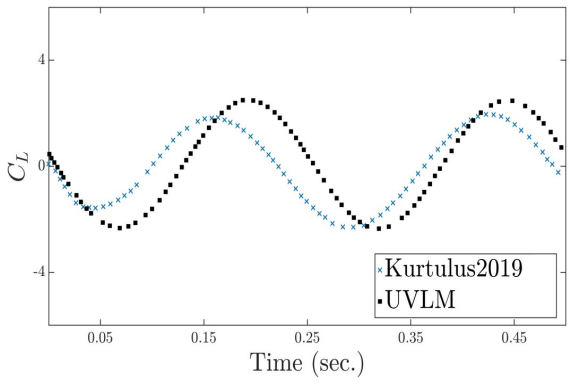


Fig. 4.9 C_L from UVLM code and literature [25] at 4Hz

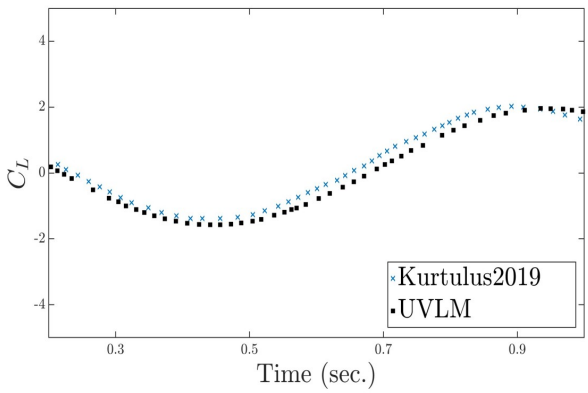


Fig. 4.10 C_L from UVLM code and literature [25] at 1Hz

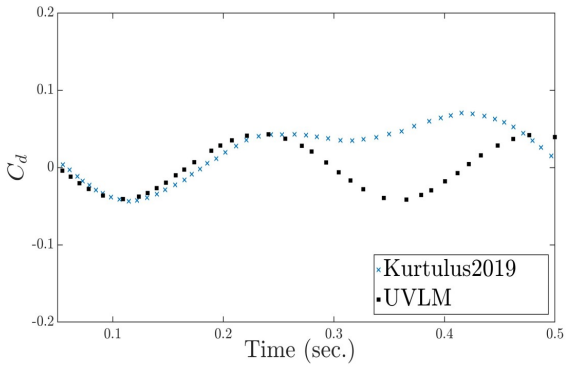


Fig. 4.11 C_d from UVLM code and literature [25] at 4Hz

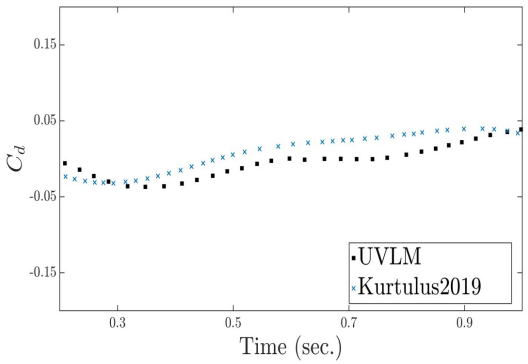


Fig. 4.12 C_d from UVLM code and literature [25] at 1Hz

wing orientation and configuration. Further work can be done on improving the viscous correction for this solver and plotting the shed wake and induced wake for each time step which will improve its capabilities to mimic real-world environments. For the buildup of viscous correction term for UVLM, initial terms are defined and are derived from previous code such as the 'N' number of panels chord-wise, and panel information such as location and Reynolds number. Reference viscosity is calculated using the reference chord length from the mean of chord length from all the panels. Then the viscous correction term is

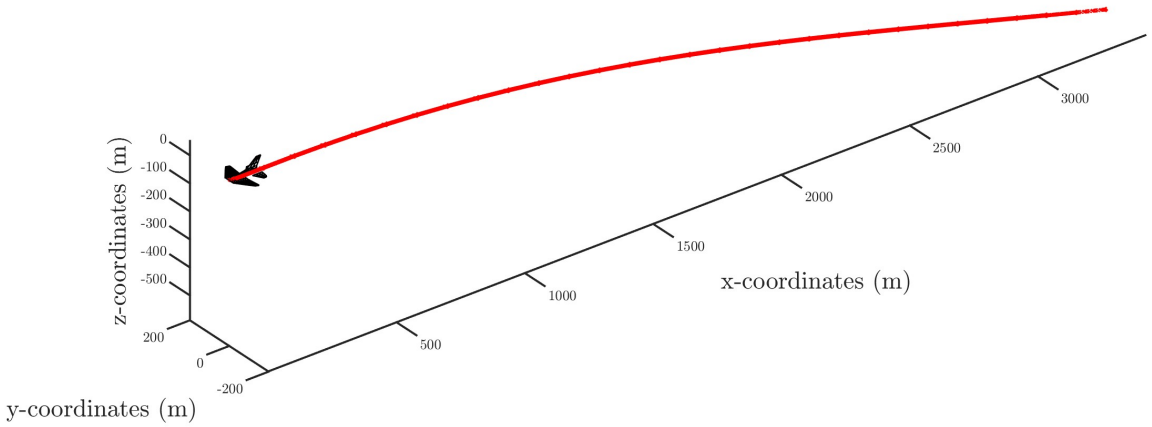


Fig. 4.13 Unsteady wake of F-16 wing.

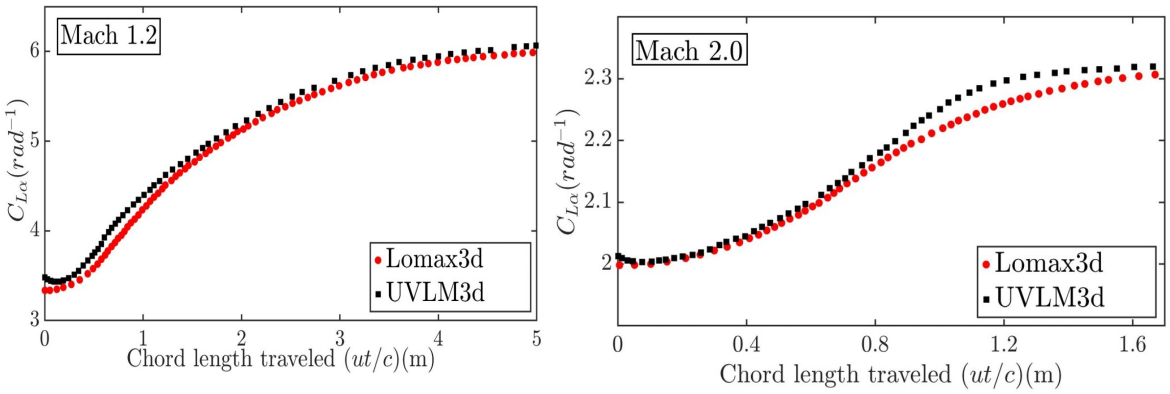


Fig. 4.14 Comparison of change in lift coefficient with alpha in chord length travelled from literature [23].

calculated using the skin friction term, reference viscosity for a given Mach and Reynolds number for a given panel.

The curves in Fig.4.14 are shown for sinking and pitching three-dimensional triangular wings for Mach 1.2 and 2 speeds. Besides the similarities like the curve and solutions, several conclusions can be drawn from these results. Firstly, notice how for shorter chord lengths the value of lift curve slope is always higher for UVLM because of working with the solution of fewer influence terms affecting the convergence. Secondly, with increased chord length the solutions are close to that of the experimental results from the literature [23] giving the same conclusive statement as for the experimental work that since all the characteristics of the triangular wing are independent of the angle of sweep for this situation, the results can be said to be valid for the un-yawed triangular wing as long as the edges are supersonic. This means that, as the VLM does not account for the thickness of the wings, and therefore edges designed through VLM are assumed to be supersonic, the validity of the solver can be found effective for any supersonic configuration of the wing. Thirdly, the initial values of $C_L\alpha$ depend on the influence terms from shed vortices and bound vortices which are directly dependent on chord length travelled and also influenced by the Mach number. With increased Mach number the positions of the shed vortices calculated are not accurate, which affects its influence on the results and as seen from Fig. 4.14 the accuracy of the solver deviates with increasing chord length travelled. It is observed that the results for three-dimensional profiles match better with the experimental data of Lomax3-d [23] when compared with the 2-D UVLM solver because in the development of the three-dimensional solver, the initial TORNADO code is used to define the panel geometry, initialising the boundary condition

and which undergoes the application of Prandtl Glauret's compressibility correction factor which is not used for the two-dimensional solver. Solver accuracy improves with finer panel resolution in both chordwise and spanwise directions, as this increases the fidelity of induced velocity calculations. Having a larger grid of panels provides more data with a change in velocity potential. The implementation of the Mach-cone influence [17, 18] is also suggested for better prediction of the panel collocation points. This will help in identifying the location shift when transitioning from subsonic to supersonic speeds.

4.6 Summary

The proposed work highlights the similarities and differences between the two- and three-dimensional solvers in terms of accuracy and agreement with experimental data and the high-fidelity CFD solver. It discusses the factors influencing the slope of the lift curve and the potential to improve the accuracy of the solver by increasing the density of the panel and the influence of the Mach cone. The two-dimensional UVLM solver compared to experimental and numerical models shows a non-steady region at $Ut/c < 1.3$, which is close to the experimental model ($Ut/c < 1$). The transition into a steady region begins after $Ut/c > 1.4$. The overall curve matches that of experiments and literature. However, predictions of $C_{L\alpha}$ at higher Ut/c values have a higher percentage of error mainly due to the UVLM's lack of correction for compressibility effects in two-dimensional flow. The lift curve slope with an angle of attack ($C_{L\alpha}$) is higher for the two-dimensional UVLM solver when the chord length is shorter. This is attributed to the solver's convergence being affected by fewer influence terms due to its solution approach. As the chord length increases, the solutions approach those of experimental results. The conclusions drawn from the comparison UVLM state that the characteristics of the triangular wing, independent of the angle of sweep, are valid for un-yawed triangular wings with supersonic edges. The VLM assumes the wing edges to be supersonic, and therefore, the solver's validity can be extended to any supersonic wing configuration. The values of $C_L\alpha$ are influenced by the shed vortices and bound vortices, which are directly dependent on the chord length and influenced by the Mach number. However, as the Mach number increases, the accuracy of calculating the positions of shed vortices decreases, affecting their influence on the results. This leads to deviations in solver accuracy with increasing chord length travelled.

The three-dimensional solver shows better agreement with experimental data compared to the two-dimensional UVLM solver. The three-dimensional solver uses TORNADO's defined panel geometry, initializes boundary conditions, and applies Prandtl-Glauert's compressibility correction factor. These additional factors enhance the accuracy of the solver compared to the

two-dimensional solver. Increasing the number of chordwise and spanwise panels enhances accuracy by providing finer resolution of induced velocities. Additionally, incorporating Mach-cone influence can further improve prediction of collocation points during transonic and supersonic transitions.

Chapter 5

Machcone modelling

In this chapter, we will introduce the challenges encountered in the realm of supersonic design, particularly concerning the acquisition of data for various supersonic configurations amidst complex flow conditions. We will delve into the time-consuming and costly nature of this endeavor, attributed to the high-fidelity computational demands involved. Furthermore, we will explore the limitations associated with running iterative simulations over prolonged periods, which often prove either impractical or financially burdensome.[1] Given these challenges, the necessity of adopting low-order potential solvers with reasonable accuracy is emphasized to streamline the generation of datasets, thereby reducing the costs associated with advancing supersonic designs. This work focuses on the development of a tailored low-order solver specifically designed for the early phases of supersonic design, leveraging simplified numerical models to enhance both efficiency and affordability.

Additionally, a novel approach is proposed to improve the accuracy of drag prediction in supersonic flow regimes. This involves integrating the Taylor Maccoll Hypervelocity Method (TMHM) with the supersonic Vortex Lattice Method (VLM), addressing the influence of shock cones on the latter through a panel-matching technique. This combined methodology provides a comprehensive framework for predicting total body drag in supersonic flows while maintaining the constraints of low-order methods.

Validation of this integrated approach against experimental data from published work demonstrates a significant reduction in drag prediction error compared to conventional VLM. Particularly, minimal errors are observed for low-shock angles, highlighting the method's effectiveness in capturing the complexities of shockwave interactions across diverse supersonic configurations with high accuracy. Furthermore, comparisons with high-fidelity

[1]The findings from this chapter is published in AIAA SciTech. Joshi, H., Thomas, P., Wu, H. and Tan, C. (2024). Improved drag prediction using mach-cone influence in vortex lattice method for supersonic aircraft. https://doi.org/10.2514/6.2024-1879.

Computational Fluid Dynamics (CFD) simulations are included to reinforce the reliability and effectiveness of the proposed methodology.

5.1 Introduction

Low-order aerodynamic methods are commonly utilised in the aircraft industry for their efficiency and low computational requirements. The use of these methods for the next generation of supersonic aircraft is gaining popularity as with next supersonic aircraft challenges surrounded by the market potential, its environmental impacts and the high cost of designing [16]. Researchers are exploring the hybrid use of multiple low-order solvers to overcome these limitations [15, 204]. The recent use of such methods is seen in studying the complex supersonic flow conditions of an aircraft undergoing the shockcone formation when transitioning from subsonic to supersonic speeds [205]. Despite their advantages being low order, these methods struggle with accurately presenting complex behaviours like shockwave interaction with the boundary layers [206], flow separation and the dynamics of unsteady aerodynamics. To fill in this gap the research was done in studying the air pressure changes upon the shockcone moving at high speeds. This research leads to the development of the Taylor Maccoll equation [207, 208]. The method developed was for calculating the airflow and the pressure in the proximity of the projectile of the conical head. The calculation carried out for different cone angles generated various shockwaves propagating its interaction with its boundary. Observed results proved the separation of shockwave at low speeds detached from the point of origin and shifted ahead of it. Later the Taylor Maccoll was used with conventional CFD modelling [209] to explore its potential in studying its influence on parametrized geometry of different configurations of waverider and super-hypersonic vehicles [210, 211]. The use of this method was the discovery of the past which was not explored to its full potential because of the good compatibility with other solvers being not developed in time. The Taylor Maccoll equation is further modified [19] to accommodate its use for analysing this equation for the shockcone interaction between the super-hypersonic wings and conical flow of various shock angles which further helped in identifying the influence of wave drag and the skin friction drag between the interaction of shockcone and the boundary layer of the configuration overall improving its application in studying the velocity potential at these speeds. The recent use of this for various advances in hypersonic and supersonic modelling is investigated [209, 212–215]. The particular use of the vortex lattice method (VLM) being quite popular low-order method hinges on specific influence terms [17], which dictate the formation and movement of horseshoe vortices across the aircraft's surface. VLM's has been widely used to predict the aerodynamic performance of both subsonic

and supersonic wing configurations [147, 216, 217]. VLM is operated by segmenting the aircraft's lift-generating surfaces into numerous panels, with the aerodynamic potential over each panel influenced by these vortices. However, accurately modelling these influences for supersonic flow is complex. Transitioning from subsonic to supersonic speeds introduces shockwaves, influencing skin friction and wave drag, potentially leading to inaccuracies [218]. A new strategy has been developed to address the difficulties encountered with low-order methodologies and their integration with traditional Computational Fluid Dynamics (CFD) to decrease design costs. This approach incorporates the Taylor Maccoll hypervelocity method (TMHM) with the Vortex Lattice Method (VLM) for enhanced application in supersonic design scenarios. This method allows for an effective panel matching between the Taylor Maccoll and the VLM. This method assumes each panel division obtained from the VLM to be an individual lifting surface undergoing shockcone formation at supersonic speeds. This method facilitates accurate drag prediction by adjusting the computation of velocity potential change and pressure changes on a supersonic body, thereby enhancing the drag coefficient's accuracy. This integration is effective in reducing cost of analysis using less computational power throughout the simulation. It highlights the potential of VLM as a preferred tool in designing supersonic aircraft by efficiently solving numerous computational design parameters. By marrying TMHM with VLM, it provides a detailed framework for considering the Machcone 's impact on VLM calculations. This method helps in solving the core limitation of the VLM in modelling the shockwave interaction over the lifting surface.

5.2 Methodology

This study develops a modified low-order aerodynamic model that incorporates conical shockwave effects into the VLM framework. Traditional VLM formulations, while computationally efficient, do not account for the compressibility and wave phenomena inherent in supersonic flow regimes. To address this limitation, the proposed method introduces a *Mach-cone influence model* derived from conical shock theory, specifically tailored for integration with VLM panels.

The core of the approach lies in analytically modelling the local shock-cone structure that forms ahead of each VLM panel under supersonic conditions. Unlike conventional VLM applications, which assume incompressible and irrotational flow, this method incorporates the influence of Mach angle shifts and shock apex movement as part of the panel representation [219]. As supersonic speed increases, the Mach angle narrows, affecting the geometric interaction between the freestream and panel surface. This variation is captured through

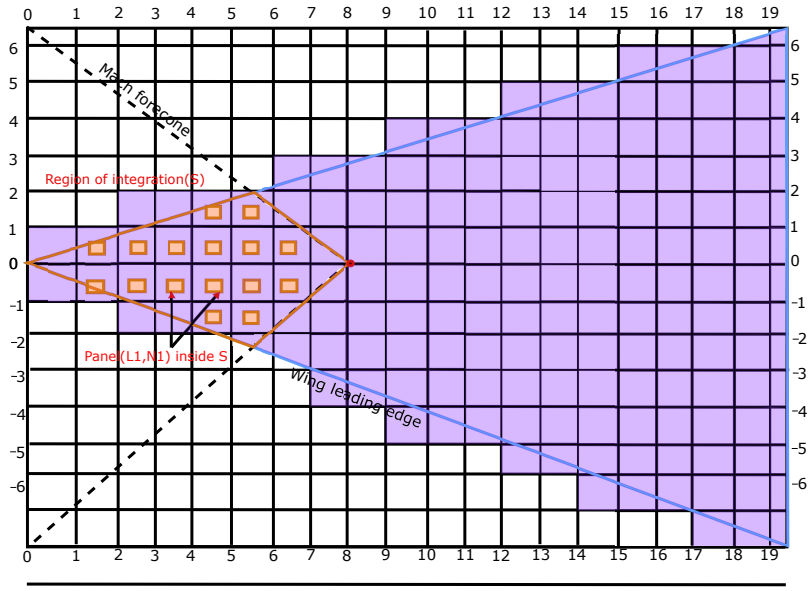


Fig. 5.1 Panel Representation of Flat Plate Delta Wings with Subsonic Leading Edge [18].

an adaptive adjustment in the placement of the panel and the location of the control point, leading to an improved prediction of local pressure gradients and force distributions.

Figure 5.1 illustrates the panelized representation of a flat delta wing, segmented into discrete control surfaces. In the supersonic regime, only panels within the Mach forecone contribute to the flow field at any given point, thereby reducing the domain of aerodynamic influence. This constraint is central to the proposed modelling framework, where the region of integration over the wing surface is dynamically determined by the Mach cone boundaries.

To model the conical shock structure, the Taylor-Maccoll equation is used, which describes axisymmetric flow over a pointed cone [220]. Although originally formulated for idealised conditions [207], its hypervelocity formulation [19] enables parametric estimation of velocity changes (u, v) across the shock front. These estimates are used to correct for the local flow conditions perceived by each panel, thereby modifying the strength and orientation of the bound vortices in the VLM solution.

Figure 5.2 shows the characteristic shock cone flow field. The half angle θ_S defines the geometry of the conical surface, while the free-stream Mach number determines the inclination of the shock. This relationship guides the calculation of local panel influence, particularly in regions near sharp leading edges or slender forebodies. The method approximates a *single-cone forebody* model for each panel, treating each as a simplified shock generator that perturbs free-stream conditions within its influence zone.

The outcome of this integration is a hybrid solver that retains the computational speed of VLM but improves its fidelity in supersonic regimes by incorporating localised shock be-

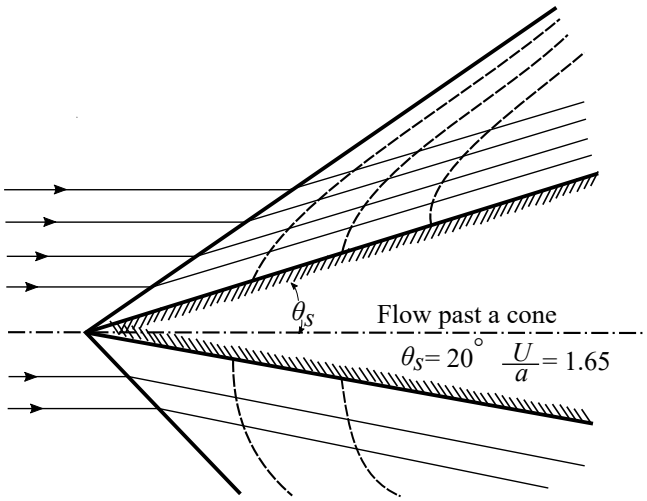


Fig. 5.2 Flow past the shock-cone [26].

haviour. This addresses a critical gap in low-order aerodynamic tools, which traditionally lack mechanisms to resolve shock-wave-induced forces. Moreover, by parametric linking shock geometry with panel location and flow conditions, the method supports design exploration for supersonic wings with varying sweep and leading-edge geometries.

5.2.1 Taylor-Maccoll Hypervelocity Framework for Supersonic Panels

The Taylor-Maccoll Hypervelocity Method (TMHM) provides an analytical approach to model high-speed flowfields around axisymmetric bodies, particularly cones, under supersonic and hypervelocity conditions. Originally developed for steady, inviscid, isentropic, and axisymmetric flow over cones assuming a calorically perfect gas [208], the method has since been extended to incorporate real gas effects including dissociation, ionisation, and excitation of internal energy modes in thermochemical non-equilibrium environments [19].

In this study, we used the Taylor-Maccoll method as a foundational tool to capture the formation and influence of shock cones on panels subjected to supersonic flow. While the classical Taylor–MacColl solution applies to axisymmetric flow around sharp cones, our modified TMHM formulation is adapted for panel-wise application on general lifting surfaces. This includes using the local Mach number and sweep angle to apply Cp corrections based on whether the panel lies within its Mach cone, thereby extending the utility of Taylor–MacColl theory to arbitrary wing geometries. Each panel on the lifting surface is modelled as a locally axisymmetric conical segment, allowing the application of TMHM to compute localised flow properties, radial and circumferential velocities, which are then coupled with the VLM for aerodynamic potential correction.

Simplified Taylor-Maccoll Formulation

The Taylor-Maccoll equation for axisymmetric supersonic flow is expressed as:

$$\left(1-\frac{v^2}{a^2}\right)\frac{\mathrm{d}^2u}{\mathrm{d}\theta^2}+(\cot\theta)\frac{\mathrm{d}u}{\mathrm{d}\theta}+\left(2-\frac{v^2}{a^2}\right)u=0 \tag{5.1}$$

Here, u and v are the radial and circumferential components of velocity, a is the local speed of sound, and θ is the angular coordinate. Under hypervelocity assumptions (i.e., $v \ll u$), the equation simplifies to:

$$\frac{\mathrm{d}^2u}{\mathrm{d}\theta^2}+\cot\theta\frac{\mathrm{d}u}{\mathrm{d}\theta}+2u=0 \tag{5.2}$$

Using the transformation $u(\theta)=f(\theta)\cos(\theta)$, this leads to a further simplification:

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin\theta\cos^2\theta\frac{\mathrm{df}}{\mathrm{d}\theta}\right)=0 \tag{5.3}$$

Solving this analytically yields the velocity components:

$$u(\theta)=A\left[1+\cos\theta\ln\left(\tan\frac{\theta}{2}\right)\right]+B\cos\theta \tag{5.4}$$

$$v(\theta)=A\left[\cot\theta-\sin\theta\ln\left(\tan\frac{\theta}{2}\right)\right]-B\sin\theta \tag{5.5}$$

The constants A and B are derived from the shock angle β, the upstream Mach number M, and the post-shock density ratio ρ, using:

$$u(\beta)=M\cos\beta \tag{5.6}$$

$$v(\beta)=-\frac{1}{\rho}M\sin\beta \tag{5.7}$$

with

$$\rho=\frac{(\gamma+1)M_1^2\sin^2\beta}{(\gamma-1)M_1^2\sin^2\beta+2} \tag{5.8}$$

Solving the boundary conditions gives the following.

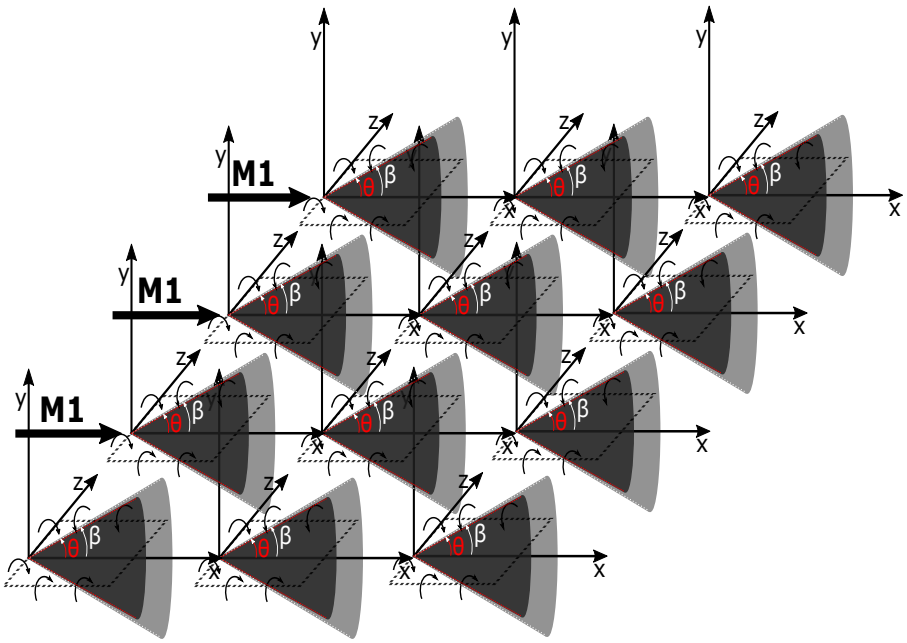


Fig. 5.3 Mach cone influence on each panel generalised using VLM.

$$A = \left(1 - \frac{1}{\rho}\right) M \sin^2\beta \cos\beta \tag{5.9}$$

$$B = M - A\left(\frac{1}{\cos\beta} + \ln\tan\frac{\beta}{2}\right) \tag{5.10}$$

The normalised velocity expressions used in the VLM framework are thus:

$$\frac{u}{M} = \cos\theta + \left(1 - \frac{1}{\rho}\right)\sin^2\beta\cos\beta\left[1 - \frac{\cos\theta}{\cos\beta} + \cos\theta\ln\frac{\tan(\theta/2)}{\tan(\beta/2)}\right] \tag{5.11}$$

$$\frac{v}{M} = -\sin\theta + \left(1 - \frac{1}{\rho}\right)\sin^2\beta\cos\beta\left[\cot\theta + \frac{\sin\theta}{\cos\beta} - \sin\theta\ln\frac{\tan(\theta/2)}{\tan(\beta/2)}\right] \tag{5.12}$$

5.2.2 Integration with VLM for Shockcone Modelling

To incorporate supersonic effects into the Vortex Lattice Method, each panel is treated as generating its own localised Mach cone based on the free-stream Mach number and the surface geometry. This creates a panel-specific shock system that is used to modify local flow potentials as shown in Figure.5.3

Panel Discretization

The wing is discretised into a lattice of panels across chord-wise and span-wise directions:

$$\Delta x_j = \frac{c}{N_{chord}}, \quad j = 1,2,...,N_{chord} \tag{5.13}$$

$$\Delta y_i = \frac{b}{N_{span}}, \quad i = 1,2,...,N_{span} \tag{5.14}$$

The local shock cone of each panel is calculated by applying the TMHM-derived expressions to obtain velocity potentials based on the circumferential coordinate θ_i.

Velocity Potentials for Panels

Using the normalised TMHM expressions, the panel-level velocity components $u_i(\theta)$ and $v_i(\theta)$ are:

$$u_i(\theta) = \cos\theta + \left(1 - \frac{1}{\rho}\right)\sin^2\beta\cos\beta\left[1 - \frac{\cos\theta}{\cos\beta} + \cos\theta\ln\frac{\tan(\theta/2)}{\tan(\beta/2)}\right] \tag{5.15}$$

$$v_i(\theta) = -\sin\theta + \left(1 - \frac{1}{\rho}\right)\sin^2\beta\cos\beta\left[\cot\theta + \frac{\sin\theta}{\cos\beta} - \sin\theta\ln\frac{\tan(\theta/2)}{\tan(\beta/2)}\right] \tag{5.16}$$

These expressions allow estimation of the flow potential around each panel. By assuming irrotational flow ($\partial v/\partial x = \partial u/\partial y$), the total velocity potential is derived as:

$$\Delta\Phi \approx \int_{panel} \phi_u(\theta)\, dS + \int_{panel} \phi_v(\theta)\, d\vec{S} \tag{5.17}$$

where $\phi_u(\theta) = \int u_i(\theta)\, d\theta$ and $\phi_v(\theta) = \int v_i(\theta)\, d\theta$, and dS, $d\vec{S}$ are scalar and vector differential area elements of the panel.

Shock Boundary Adjustments

At high Mach numbers, panels encounter local shock effects, especially near the leading edges. The shock-cone interaction modifies the boundary conditions at the location points (see Figure 5.5), affecting the vertical component of the velocity. These changes are implemented through corrected potential functions based on oblique shock theory and Rankine-Hugoniot conditions [221].

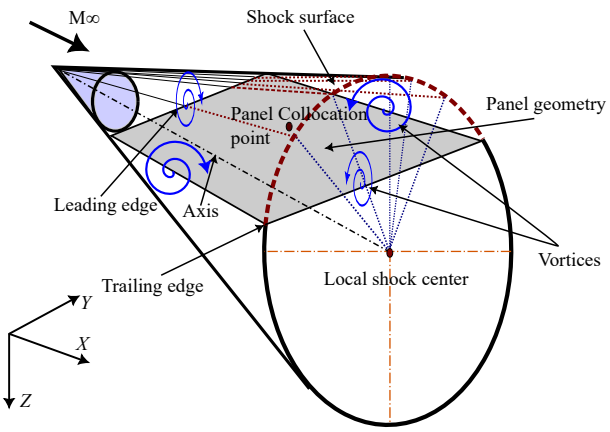


Fig. 5.4 Mach-cone over a panel geometry under shock surface.

$$\phi'_u(\theta) = f(\phi_u(\theta), M, \beta) \tag{5.18}$$

$$\phi'_v(\theta) = f(\phi_v(\theta), M, \beta) \tag{5.19}$$

$$\phi'_u(\theta) = f(\phi_u(\theta), M, \beta) \tag{5.20}$$

$$\phi'_v(\theta) = f(\phi_v(\theta), M, \beta) \tag{5.21}$$

where $\phi'_u(\theta)$ and $\phi'_v(\theta)$ are the adjusted potential components post-shock, M is the local Mach number, β is the shock angle and f represents the function that models the changes across the shock, based on the shock wave theory as shown in Fig.5.4.

$$P_2 = P_1 + \rho(M_1)^2(\sin(\beta)^2)\left(\tfrac{2}{\gamma+1} - \tfrac{1}{\rho}\right) \tag{5.22}$$

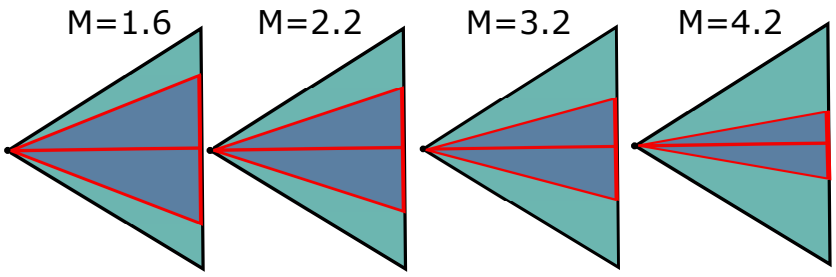


Fig. 5.5 Mach cone influence on the delta wing [17].

where P_1, ρ and M are the pressure, density and the Mach number of the freestream and β is the shock angle.

- **Superposition of panel contributions**
  The total potential ϕ and velocity V at any point in the field can be found by summing the contributions from all the panels. For a field point P, this can be approximated as:

$$\Phi_P = \sum_{I=1}^{n} \Phi i \tag{5.23}$$

$$v_P = \sum_{i=1}^{n} v i \tag{5.24}$$

  where n is the number of panels. The contributions from each panel (Φ_i, v_i) are computed on the basis of the local $\phi_u i(\theta)$ and $\phi_v i(\theta)$ adjusted for the panel's orientation and observer's position.

- **Calculations for drag coefficient**
  The lift coefficient C_d can be calculated by integrating the pressure distribution across the lifting surface.

$$C_d = \frac{1}{S} \int_{surface} (P - P_\infty) dS . \sin(\alpha) \tag{5.25}$$

  where S is the reference area, P_∞ is the free-stream pressure and α is the angle of attack. The pressure P on each panel is calculated from the local flow properties determined by the new Shockcone modelling within VLM.

The novelty of the work lies in the development of a low-order hybrid aerodynamic framework that assigns a locally conical TMHM shock solution to each VLM panel. This allows the classical VLM to account for panel-specific supersonic flow behaviour, such as shock-cone interaction, oblique shock corrections, and local pressure recovery, without resorting to full CFD or Euler solvers. By computing the circumferential and radial velocity components for each panel through TMHM, and integrating these into the potential calculation of the vortex lattice, your method effectively enables shock-aware vortex strength estimation. Furthermore, using the shock angle (β), post-shock density ratio (ρ), and panel orientation, the model provides analytic corrections for the effects of flow turning and downwash.

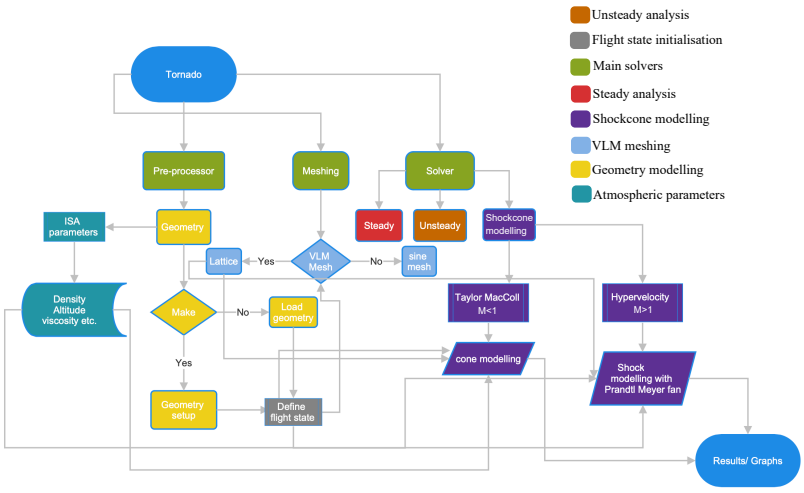


Fig. 5.6 Flowchart showing full Tornado architecture and the addition of the Shockcone modelling module.

5.3 Computational modelling

This work builds upon the Tornado framework, an open source VLM solver, which serves as the backbone for aerodynamic modelling. The tornado discretises lifting surfaces into a lattice of bound vortex filaments and applies linearised potential flow theory to solve for circulation distributions, thus computing aerodynamic forces and moments [194]. The present work extends the Tornado solver by integrating a novel Shockcone Modelling module that accounts for compressibility and shockwave effects in supersonic regimes through the Taylor-Maccoll hypervelocity method.

Figure 5.6 illustrates the architecture of the Tornado solver. The computational pipeline begins with user-defined geometry parameters and flight conditions, which are processed through Tornado's state initialisation routines. These routines call the International Standard Atmosphere (ISA) module to retrieve altitude-based environmental properties. The core Tornado VLM routines then perform aerodynamic calculations under incompressible assumptions. To enable compressibility corrections, especially relevant for supersonic flows, a dedicated Shockcone Influence subroutine was incorporated within the Tornado framework. This routine, highlighted in Figure 5.7, is directly connected to the existing solution through Tornado's modular addfile system. It takes user-specified values such as the cone half-angle (θ), freestream Mach number, and the specific heat ratio (γ), and performs several analytical steps to model flow changes due to shockwaves.

The routine begins by computing upstream flow properties from ISA, followed by calculating the Prandtl-Meyer function (μ) and the stagnation temperature in the free stream using isentropic flow relations [222]. These parameters are used to solve the oblique shock relations, yielding the post-shock Mach number and the pressure ratio across the conical surface. This modification enables the solver to estimate the pressure field more accurately by adjusting vortex influence terms within the VLM solver based on local compressibility effects.

Furthermore, to address applications in hypervelocity regimes, the Taylor-Maccoll method is implemented within a separate subroutine under the shockcone model. Here, the normal and tangential velocity components $u(\beta)$ and $v(\beta)$ are solved across the shock layer, specifically focussing on the cone surface, where the tangential component diminishes to zero. The variation of the velocity components (u/M and v/M) is evaluated in a range of cone angles to visualise the deviation from free-stream flow and the intensifying compressibility effects near the apex.

The results of these computations are stored in the Tornado output structure for post-processing and visualisation. This extended computational framework allows for a more accurate aerodynamic representation of supersonic and hypersonic flows by embedding compressibility and shockwave effects directly into the VLM formulation.

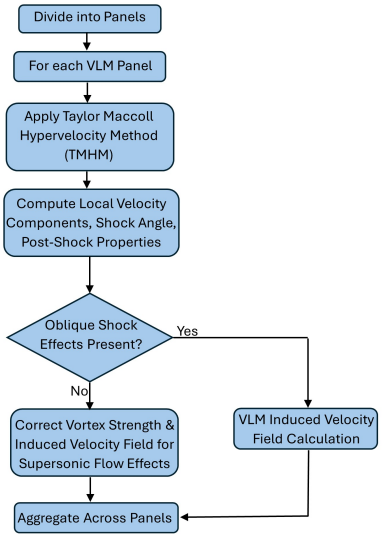
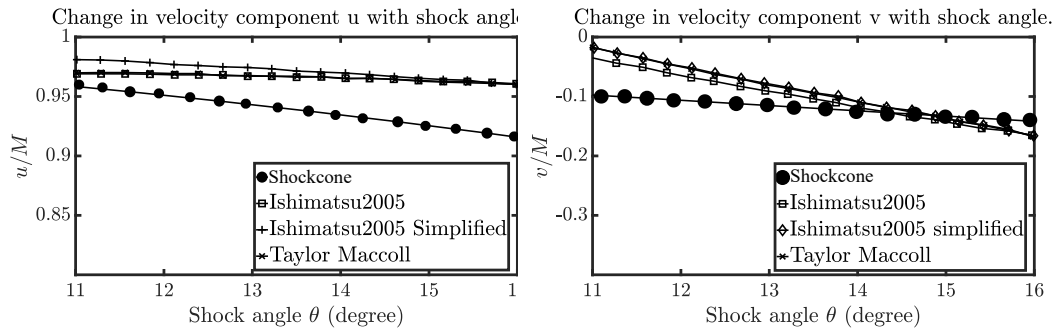


Fig. 5.7 Flowchart of Shockcone subroutine integrated within the Tornado solver.



(a) (b)

Fig. 5.8 Velocity component variations u(a) and v(b) with shock angle from Mach-cone influence compared with literature [19].

5.4 Modelling verification and validation

To establish the validity of the developed Shockcone solver, a comparative analysis was performed against established numerical solutions of the Taylor-Maccoll model of Ishimatsu [19]. The primary objective of this validation is to confirm the solver's ability to accurately capture shockwave-induced flow variations, particularly in the tangential and normal velocity components of the conical surface.

5.4.1 Axisymmetric Validation (Taylor–MacColl)

The validation was performed over a realistic range of shock angles, corresponding to conditions typically encountered in supersonic aerodynamic applications such as high-speed aircraft wings and aerofoils. This range ensures that the shock remains attached, a regime in which oblique shocks dominate, and where the Taylor-Maccoll framework remains applicable. Detached shock scenarios, which occur at larger angles or in extreme flight conditions, fall outside the intended scope of this solver.

Figure.5.8 compares the velocity components $u(\beta)$ and $v(\beta)$ along a conical section using the shock cone model, the conventional Taylor-Maccoll and Ishimatsu models[19]. The geometry is a 15° cone at $M_\infty = 2.0$, $\alpha = 0°$, with $\rho_\infty = 1.225$ kg/m^3. The solver's output demonstrates strong agreement with the published solutions, especially at lower shock angles. Minor deviations are observed, increasing from 2. 01% to 4. 16% with shock angle, in stronger shock conditions, reflecting the increasing complexity of tangential and normal flow resolution in those regimes.

This validation confirms the solver's capability to replicate shockwave-related velocity changes across a practical range of configurations, supporting its integration into VLM for

enhanced supersonic aerodynamic prediction. The focus on attached shock conditions ensures compatibility with linearised flow assumptions and maintains computational efficiency while significantly improving the applicability of conventional VLM in compressible flow regimes.

5.4.2 Results for two-dimensional shape study

Boundary and Initial Conditions			
Condition	Pressure (Pa)	Temperature (K)	Velocity (m/s)
Inlet	101325	288.15	(U, 0, 0)
Outlet	*zero gradient*, 101325	*zero gradient*, 288.15	(U, 0, 0)
γ (Ratio of Specific Heat)	1.4		
Velocity (Mach)	1.5, 2.7		
Min. Allowable Wall Distance (m)	1.0×10^{-6}		
Reference Pressure (Pa)	101325		
Min. Allowable Absolute Pressure (Pa)	1000		

Table 5.1 Boundary and initial conditions for the simulation.

This section investigates the influence of aerofoil geometry on supersonic flow behaviour using the Shockcone solver, focussing on symmetric and asymmetric two-dimensional shapes. The NACA 0012 (symmetric) and NACA 4415 (asymmetric) aerofoils were selected to assess how geometry affects shockwave interactions and drag prediction within supersonic regimes. The simulations were conducted at Mach 1.5 and 2.7, using consistent boundary and initial conditions derived from the literature benchmarks [223], as detailed in Table 5.1.

The primary aim of this shape study is to evaluate the solver's performance in capturing wave drag characteristics across varied shock angles and flow conditions. The Shockcone method—built on the integration of Taylor-Maccoll theory with the Vortex Lattice Method—offers a more physically informed approximation of conical shock interactions over lifting surfaces. This enables more accurate drag predictions compared to traditional VLM, which lacks the ability to resolve shock phenomena and surface thickness effects.

The results are summarised from running multiple simulations using different solvers such as the RANS SST, conventional VLM, Taylor-Maccoll method, and Shockcone solver. Fig 5.9 illustrates the Shockcone solver's results, indicating good accuracy compared to other solvers. The solid lines in the graph represent the curves from simulation running

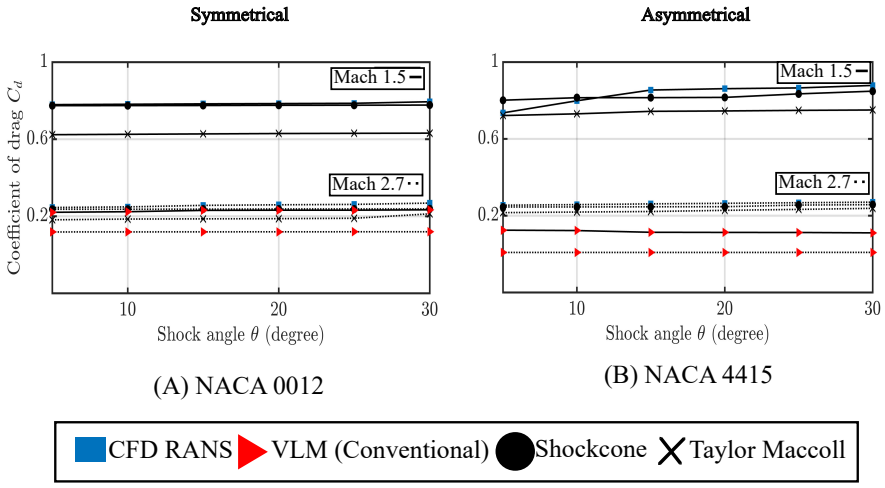


Fig. 5.9 Variation of pressure drag coefficient C_D with shock angle β, corresponding to increasing freestream Mach number M_∞ at constant cone angle θ_s using Shockcone solver with conventional CFD (RANS), VLM and Taylor Maccoll.

at Mach 1.5 and the dotted lines represent Mach 2.7. Fig. 5.9(A) shows the results for the symmetric aerofoil NACA 0012 because an aerofoil is symmetric throughout the grid providing even boundary conditions for the shockcone, which derives consistent accurate results when compared with CFD RANS. As shown in Fig. 5.9, the Cd predicted by the standard VLM at M = 1.5 and 2.7 deviates significantly from the CFD, particularly near the leading edge, while the VLM corrected for TMHM matches closely

This comparison reveals insights into how Taylor MacColl considers shockwave interactions over the Mach cone, capturing changes in velocity components and their influence on wave drag. In contrast, the VLM cannot account for the thickness of the surface, leading to alterations in the prediction of drag. Additionally, its inability to predict shock wave interactions and apply Mach cone influence to the lifting surface results in inaccuracies. Therefore, it can be seen from the results shown in Fig. 5.9 that the VLM results for Mach 1.5 and 2.7 deviate the most from that of CFD. Fig. 5.9 shows the consistency of the solver in predicting drag coefficients for shock angles ranging from 5 to 30 degrees.

Beyond aerodynamic accuracy, the study also highlights the solver's significant computational advantage. Compared to conventional RANS-based CFD simulations, the Shockcone solver demonstrates a drastic reduction in runtime. On average, the solver operates more than 20 times faster than the RANS CFD in all the tested cases, making it highly suitable for iterative design and early-stage aerodynamic analysis.

This computational efficiency is achieved without substantial compromise in predictive capability, particularly for practical shock angle ranges between 5° and 30°, where conical shock structures remain attached and consistent with the solver assumptions. For both types of aerofoils, the solver maintains robustness in drag prediction and provides meaningful insights into shape-induced flow variations. The integration of VLM and Taylor-Maccoll further ensures that both global and local flow characteristics are captured, offering a balanced trade-off between fidelity and speed.

In general, the two-dimensional shape study underscores the suitability of the solver for fast and reliable analysis of supersonic aerofoils. Its ability to incorporate geometric effects and resolve shock-induced flow behaviour with high computational efficiency presents a viable alternative to more resource-intensive CFD approaches, particularly in the conceptual design phase of high-speed aircraft.

5.4.3 Results for three-dimensional shape study

A three-dimensional shape study is crucial in aerodynamics to accurately model real-world objects and complex flow phenomena. It ensures realistic boundary conditions, validation, and reliable design exploration. Although the TMHM method originates from 2D axisymmetric theory, we extend it to 3D by applying it panel-wise based on local Mach cone alignment. In this way, we retain the simplicity of a 2D solver while enabling application to 3D geometries.

Three different supersonic configurations were considered based on the solver's assessment with wings shapes, dihedral, and swept back angles. For the following reasons, the F35 design was considered, which houses the negative dihedral from the root chord to the base

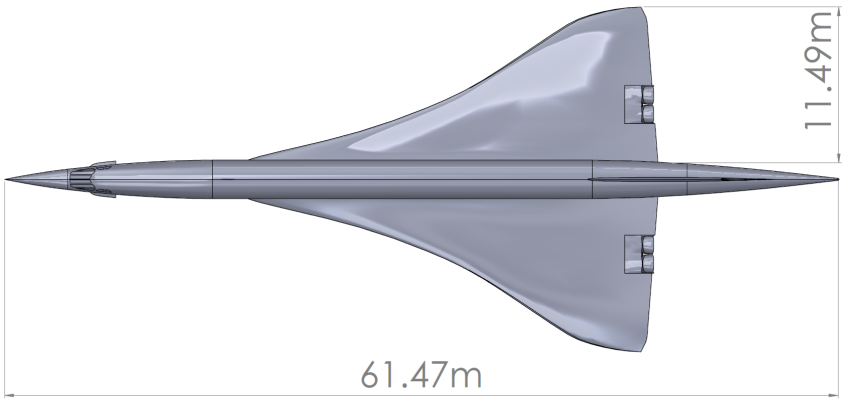


Fig. 5.10 Geometry for Concorde aircraft.

chord; in addition to this The Concept supersonic design was also considered for its multiple swept-back wings on the same reference plane and to study how this configuration influences the wave drag, and lastly the Concorde configuration was used to study the solver's influence on delta wing configuration. The geometry considered was a 1:1 scale model of aircraft as in the figure 5.10 that was made to analyse the solver's ability to handle the full-scale model with complex mesh. The SCALOS vehicle and Concorde wing geometry information was obtained from sources and influenced by the published literature [224, 225], with simplified CAD models for surface discretization in Tornado.

The conical domain was considered to model the conical flow to accurately define the boundary condition to model the shockwave over the surface. The boundary conditions used are the same as the two-dimensional shape study with the difference in flow speeds of Mach 1 and Mach 2. For three-dimensional shape studies, the chosen shock angle range ensures accurate modelling of shock wave interactions with conical flows, which are critical in supersonic aircraft design [226].

Mesh Parameters	**Selected Properties**
Base Size (m)	0.75
Target Surface Size	100% of base size
Surface Growth Rate	1.1
Shock Angle (°)	10, 15, 20, 25, 30
Core Mesh Optimization Cycles	10
Quality Threshold	0.4
Number of Prism Layers	3
Volume Growth Rate	1.2

Table 5.2 Summary of mesh parameters and their selected properties.

Larger shock angles were not tested because they often lead to detached shocks, which requires different modelling approaches beyond the current solver's scope. This validation confirms the solver's ability to predict shockwave-induced aerodynamic effects while improving the VLM-based drag estimation.

The initialisation and boundary conditions resemble its application in studying its wave drag and its influence on the pressure drags generated by shock cone formation at supersonic speeds [227]. Fig. 5.11 illustrates 3D panel methods for simulating supersonic aircraft configurations. It shows the geometries of aircraft such as (A) F35, (B) Concept (C) Concorde with their wings and tails, dissected into numerous panels along the chord and span, with the aim of mirroring the actual 3D design [228] with precision. The panels are arranged to

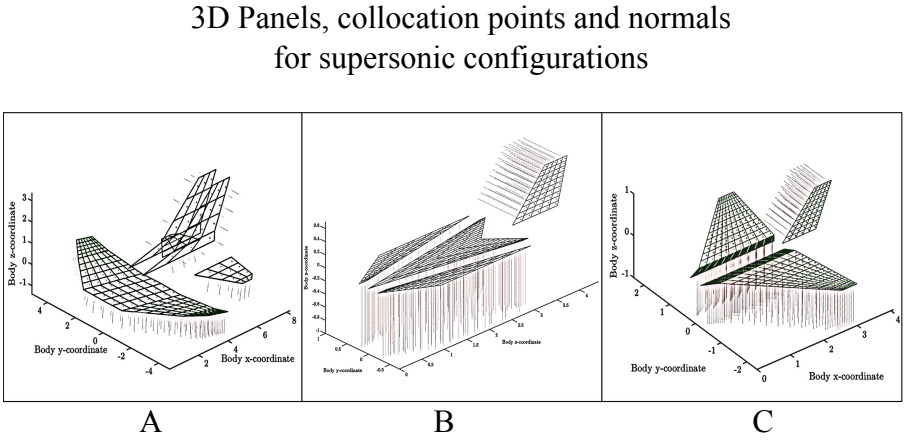


Fig. 5.11 Panel representation of supersonic aircraft (A) F35, (B) Concept (C) Concorde

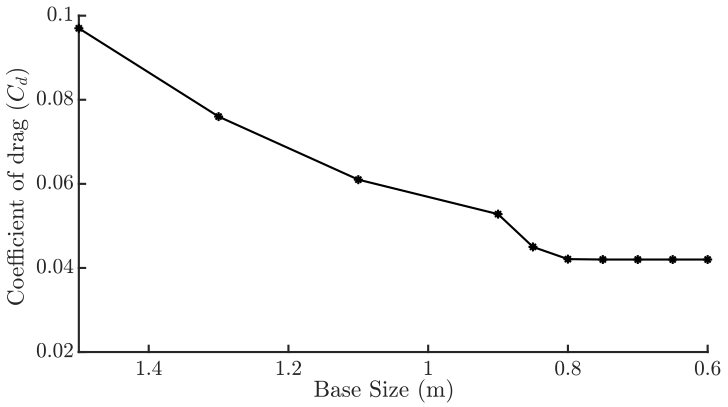


Fig. 5.12 Mesh independence study with coefficient of drag C_d

reflect true-to-scale measurements and incorporate specific dihedral angles to match the real aircraft's geometry.

Additionally, Fig 5.11 highlights collocation points - critical for locating the vortex centres on each panel's surface and depicts the normal to the panels with dotted lines, which are perpendicular to the panel surfaces. To model the panels with close accuracy to the 3-D designs for the Concorde and F35 configuration the gap between the wings is left to mimic the space needed by the fuselage to generate its influence on the drag, whereas in the case of concept design that gap is filled with panels to study the effect of panels in the space on the pressure drag due to compressibility effects and wake. The panel concentration can be increased for increased accuracy of the solver, which will also increase the computational time for running the simulations.

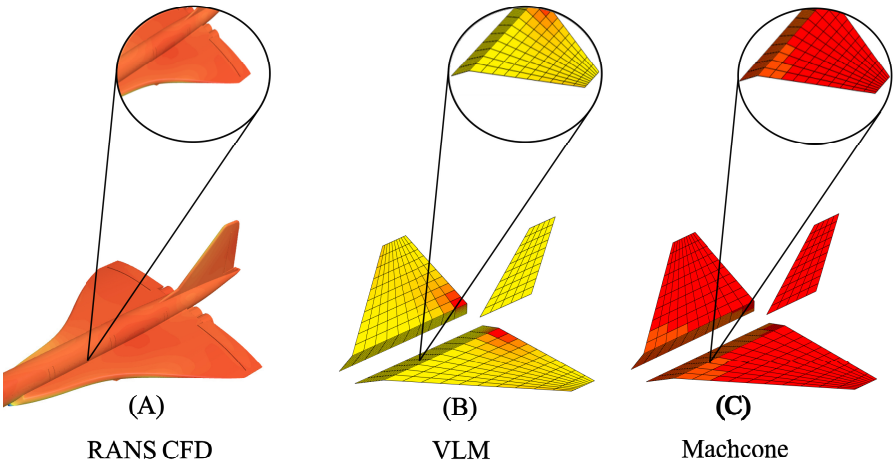


Fig. 5.13 Delta C_p representation over Concorde (A)RANS CFD, (B)VLM and (C)Shockcone

The mesh independence study is performed as shown in the figure5.12 and evaluates the impact of grid refinement on the calculated drag coefficient (C_D) using the polyhedral meshing approach of STAR-CCM +. Initially, noticeable variations in C_d occur for coarser meshes, indicating numerical sensitivity to grid resolution. However, as the base mesh size is refined to 0.75m, the changes in C_d stabilise, confirming that further refinement does not significantly impact accuracy. This establishes that a base mesh size of 0.75 m is optimal, balancing computational efficiency and result precision.
These geometries are then modelled in Tornado base code to generate the lattice and defined boundary conditions. The generated geometries define the orientation of the panel, increasing concentrations on the lifting surface of the influence.

Fig 5.13 Showing the representation of the change in the pressure coefficient, denoted as ΔC_P, is crucial to analyse the shock wave patterns around supersonic wings. The pressure coefficient was calculated using numerical simulations and a low-order aerodynamic model. In the numerical approach, C_P was derived from surface pressure data and the pressure difference between the lower and upper surfaces was evaluated to assess the aerodynamic loading. In the low-order method, the VLM was modified to incorporate compressibility corrections using influence coefficients derived from the Taylor-Maccoll equation, accounting for shock-expansion effects in supersonic flows. The corrected influence matrix ensured an accurate pressure distribution, allowing for a comparative assessment between the two methods.

High values of ΔC_P indicate the presence of strong shockwaves, which contribute to increased aerodynamic drag and potential instability. By examining(ΔC_P) in Fig. 5.13 on the aircraft surface, areas of intense shock waves can be identified and the aircraft design can be

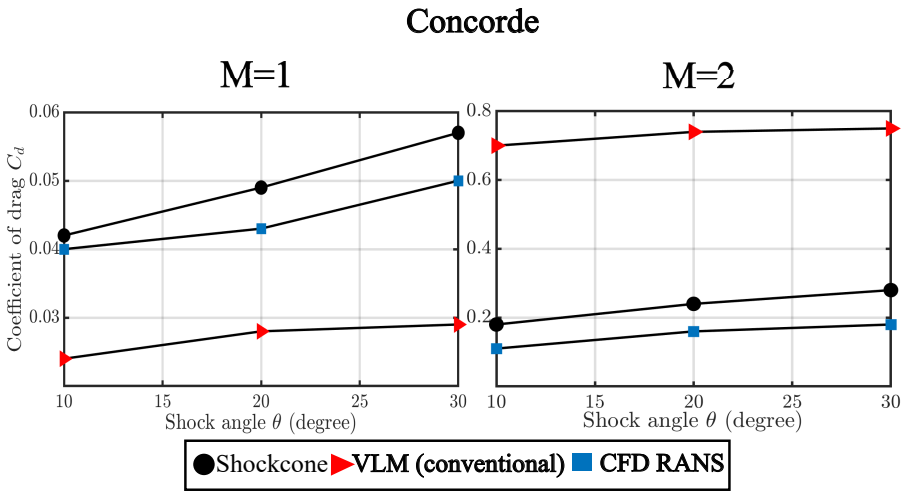


Fig. 5.14 Coefficient of drag comparison for concord configuration from VLM, Shockcone and RANS (CFD).

refined to mitigate these effects. In Fig. 5.13 the sharp gradients in colour near the aircraft's fuselage at the quarter chord position in the images indicate the presence of a shock wave, as shown in the bubbles, which is where the airflow transitions from supersonic to subsonic speeds, causing a sudden change in air pressure and temperature [212]. The difference in shock wave angles probably affects the aerodynamic performance and drag characteristics of the aircraft. This analysis helps improve the aerodynamic efficiency of the aircraft and reduce drag by optimising the design parameters responsible for this change to mitigate the risk of wave drag in shock cone generation.

Furthermore, this information can be used carefully to understand the exact location of the shock formation and its influence on the geometry of the aircraft, such as the intricate design changes required to minimise its impact and overall improved aerodynamic performance [229]. The results are shown for the drag coefficient derived from 5.25 for Mach 1 and Mach 2 with shock angles of 10 to 30 degrees. The results are then compared with other solvers such as conventional VLM and RANS CFD for the same conditions. Firstly, the results for the Concorde configuration are shown to show the solver's potential over a delta-wing configuration. From Fig. 5.14 it seems that as the shock angle increases, the drag coefficient also increases for the Shockcone and RANS models, while the VLM model shows a relatively constant drag coefficient, which suggests that the conventional CFD does not account for the complex condition to study the surface boundary interaction at the shockcone.

Similarly for F-35 the results shown in Fig. 5.15 depict the VLM results are higher than the RANS and Shockcone, which was expected based on the geometry design of the configuration in VLM and its limitation with complete geometry progression when

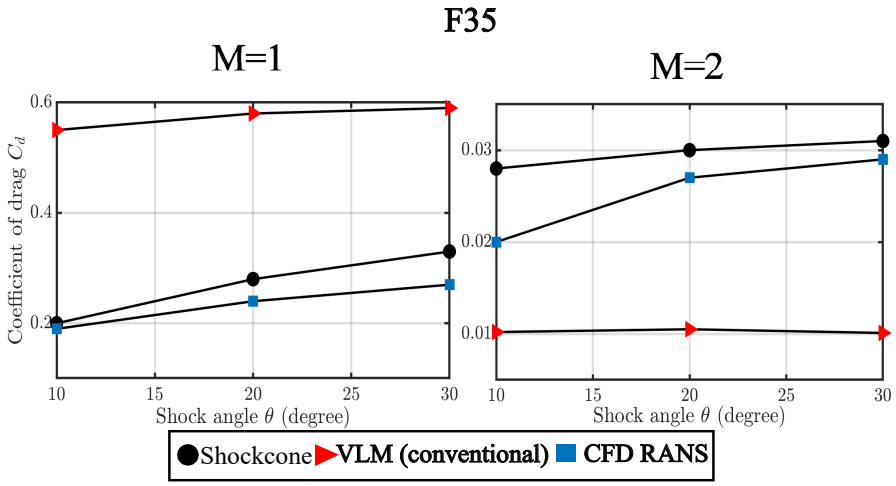


Fig. 5.15 Coefficient of drag comparison for F35 configuration from VLM, Shockcone and RANS (CFD).

compared with conventional CFD. However, overall Shockcone shows a more pronounced increase, suggesting that the idealised shockwave pattern significantly impacts the drag as the angle increases. The VLM line sits above the RANS, indicating that the vortex effects captured by this method result in a higher drag prediction than the averaged turbulence model of the RANS, because of which it is verified that the conventional VLM does not account for accurate modelling of drag estimation at high speeds. The Shockcone prediction is significantly higher at larger angles, which could be due to the idealised nature of the shockwave model it represents and how it computes with a defined grid size. As has been observed from the two-dimensional shape study, the Shockcone solvers lose their accuracy with asymmetrical shapes, as it creates inconsistency in defining the boundary conditions around them concerning the grid. From the results of the last two configurations it is seen that the shock cone results in less influence on the model generating the drag, which suggests that for this configuration, the shock angle has minimal impact on drag. However, for the supersonic concept shown in Fig. 5.16 C_d the values are higher, with the VLM method showing a moderate increase with the shock angle and the RANS method showing a minor increase. This observation clearly states that the influence of the surface is on the shock to increase its estimated pressure drag due to compressibility effects, but it does not model viscous skin friction drag explicitly.

In all the graphs between Concorde, concept supersonic, and F35 one common thing is that the results from the Shockcone solver resonate closely with the results from the RANS CFD computations. The results converge between both for lower shock angles in the range of 10-15 whereas the shock wave increases, the results deviate from each other showing that the solver loses its consistency with increasing shock angle, which can only be proved using

an experimental validation. The critical change in correlation between these results is seen after a shock angle of 20. Overall, the results compared show potential 85% to 90% accuracy for low Mach numbers and shock angles.

The conventional VLM method remains relatively flat. This indicates that the generic configuration experiences consistent drag as the shock angle increases, particularly as expected by the VLM. Compared to others, the Shockcone method seems to be more sensitive to changes in shock angle across all configurations, suggesting a more detailed capture of the effects of shock waves on drag. This suggests that the shock cone might be the more nuanced and responsive method to capture the aerodynamic effects of shock wave angles on these aircraft configurations.

Compared to the three graphs, the RANS method consistently predicts a lower sensitivity of C_d to the shock angle, potentially indicating a more average approach to complex flow patterns, possibly missing some of the implications of shock wave behaviour [155]. The VLM and Shockcone methods show that the drag increases with shock angle, which is consistent with the physics of supersonic flow, where larger shock angles correspond to stronger shock waves and higher drag. Each graph represents different aircraft configurations, indicating how these methods might vary in their predictive capabilities based on specific aerodynamic designs.

Fig.5.17 shows the correlation matrix, highlighting how different parameters correlate with each other in the context of an improved Shockcone solver and conventional CFD. Each square represents the correlation coefficient between the parameters on the vertical and horizontal axes. The colour scale on the right indicates the strength of the correlation, with

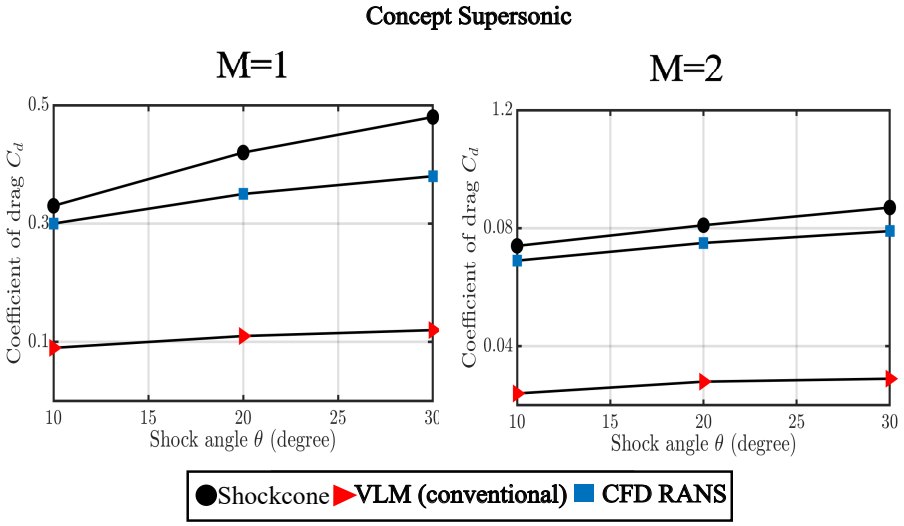


Fig. 5.16 Coefficient of drag comparison for concept supersonic configuration from VLM, Shockcone and RANS (CFD).

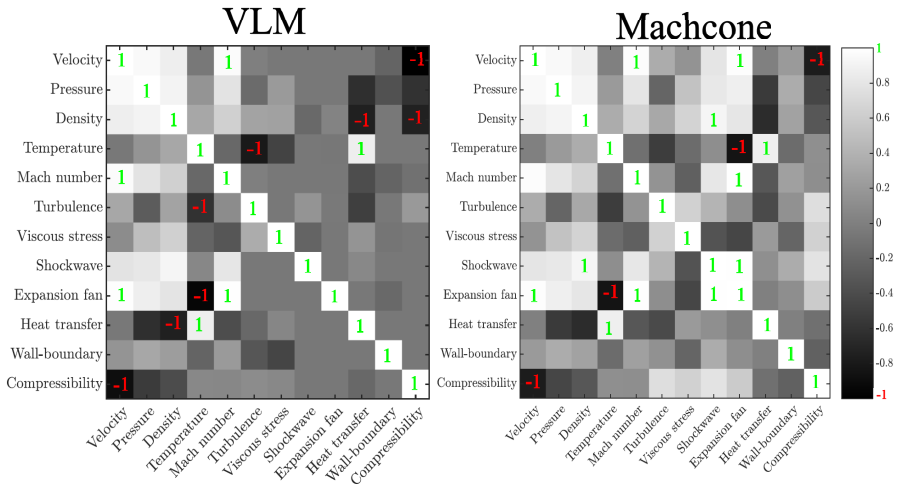


Fig. 5.17 Correlation box showing the relations between the improved parameters.

1 being a perfect positive influence on the solver, indicated by a white colour square with green colour 1 written on it. Similarly, -1 is a negative influence indicated by a black square with a red -1 written on it, and 0 indicates no influence, typically shown by a neutral grey. For example, if you look at the square that intersects the "Density" row and the "Pressure" column, the colour of the square will show you how strongly those two parameters are correlated and have an influence on the solver according to the data analysed. A bright white would mean they tend to increase and decrease together, while a black would mean that as one increases, the other tends to decrease. A neutral colour would mean there is no clear relationship between the two in the data set used, which has little to no influence on the solver. This matrix is a powerful tool for quickly assessing relationships between multiple variables in a complex system, such as the environment of shock-wave boundary interactions. In Fig. 5.17, the matrix on the left represents the relation between the parameters for the vortex lattice method, and on the right is the correlation matrix for the Mach cone influence. As seen from both the figures, the comparison shows that the number of green 1 signs has increased for the Shockcone solver, and similarly, the red -1 has reduced. The relation plot shows how improvements in the solver-made parameters like Viscous stress, Shockwave and expansion fan have greatly improved its influence overall, improving its capabilities to model and study the interaction of conical flows with modelling Mach cone, thereby improving the drag prediction.

5.5 Summary

The proposed work is a vital component in ongoing work to develop an aerodynamic solver with proven capabilities to predict the flow around a shockwave and its influence on supersonic aircraft. It uses the fundamentals of potential flow theory and Taylor Maccoll's application to study the shock layer boundary interaction over aerofoil panels. Visualizing the pressure change upstream and downstream of the shock layer over an aerofoil for different Mach numbers helps validate the accuracy of computational models that simulate aerodynamic performance. Comparing the simulation results with the experimental data from Ishimatsu's model [19] shows convergence at higher shock angles and helps in understanding the influence of the shock layer on each panel. The results show a 2.01% error for low shock angles to 4.16% for higher. Visualizing the bow shockwave pressure curve with chord length provides a comprehensive understanding of the supersonic flow behaviour, including shock strength, shock angle, and pressure distribution.

The solver maintains high accuracy (90-95%) for two-dimensional shape study and (85-90%) for three-dimensional study. The comparison between symmetrical (NACA 0012) and asymmetrical (NACA 4415) airfoils is made to assess the influence of shape on the solver's accuracy, emphasizing the efficiency of lift generation and the higher lift-to-drag ratios associated with cambered airfoils. This paper concludes by underlining the computational advantages of the combined Taylor Maccoll and vortex lattice method, striking a balance between accuracy and efficiency, especially around shockwaves. In the broader context, the proposed work demonstrates the importance of the machcone solver in predicting and understanding the flow around shockwaves in supersonic aircraft. The method's application, based on potential flow theory and Taylor Maccoll principles, proves valuable for validating computational models and provides insights into shock layer interactions. The combination of Taylor Maccoll with the SVLM is highlighted as a time and cost-effective alternative to high-fidelity CFD analysis, allowing for sensitivity analysis on design parameters and identification of critical influences on drag. The improvements in the feasibility of the solver for its potential use in three-dimensional shockwave interaction of the boundary undergoing the multi-phase supersonic flow will be proposed using the lattice Boltzmann method[173]. Lattice Boltzmann method being a microscopic reduced order solver will help in studying the infusion of multi-phase supersonic flows which will also account for its application in compressibility correction at those speeds [183]. As noticed from the correlation matrix box 5.17 it can be easily seen that work is needed to model the wall boundary effect improving the viscous interaction [230] and improving the heat transfer aspects at the supersonic speeds.

Chapter 6

Compressibility correction

In this chapter, we explore applying the lattice Boltzmann method (LBM) for modelling compressibility correction in supersonic fluid dynamics in a conventional vortex lattice method framework. Our study focuses on implementing a compressibility correction term within the D2Q9 lattice framework, employing the Bhatnagar-Gross-Krook (BGK) collision model and bounce-back boundary conditions. This research addresses the intricate interactions of fluid particles around lifting surfaces subjected to high-speed flows. The compressibility correction is realized through a multiphase solver that discretizes the computational domain into numerous cells, with a lifting surface represented as an obstacle matrix. This discretization defines elements as fluid particles or solid walls, enabling detailed simulations of fluid dynamics at the mesoscopic level. The computational solver iteratively calculates density and velocity distributions across lattice nodes, propagating these values to neighbouring cells to enforce boundary conditions. The collision factor inherent in the BGK model is critical in determining density ratios and effective velocity components necessary for the accurate modelling of compressible flows.[1] This chapter delves into the adaptability of the lattice Boltzmann method, allowing for the characterization of computational domains as isotropic or anisotropic to capture multiphase phenomena effectively. This versatile approach is pivotal for studying fluid-solid interactions under high-speed conditions, surpassing traditional methodologies. Furthermore, the integration of the compressibility correction within a vortex lattice method (VLM) framework enriches our understanding of multiphase fluid dynamics and boundary interactions at supersonic speeds. This combined approach overcomes inherent limitations in modelling individual fluid particles, presenting a robust solution for simulating complex compressible flow dynamics.

[1]The findings from this chapter will be published in IEEE Aerospace Conference, Montana, USA Joshi, H. and P. Thomas (2025). Multiphase compressibility correction for supersonic flow using lattice Boltzmann method.

6.1 Introduction

The Lattice Boltzmann method is a highly complex fluid modelling solver but is unstable for a supersonic regime. To work around this limitation Renard proposed a hybrid model of LBM [183] using correction terms for discretisation and viscous terms that take into account the polyatomic gases and discretisation of the viscous heat production term fitting with regularised formalism. This proposed hybrid LBM buildup accounts for the mass and momentum equation using the core LBM method and the entropy-inspired energy equation using the finite volume method. This equation is valid for both subsonic and supersonic regimes [231].

The motivation for employing the lattice Boltzmann method (LBM) rather than the simpler Prandtl–Glauert (P–G) correction is twofold. First, the P–G transformation is only valid in the subsonic linear regime ($M < 0.7$) and breaks down in transonic and supersonic flows where nonlinear density variations and shocks are present. Second, LBM is a density-based formulation in which compressibility is inherently included, allowing a more general extension of VLM into compressible flow regimes. Although more computationally expensive, LBM provides improved accuracy and scalability, justifying its use in this research.

The LBM is widely used for its ease of modelling and efficient solver capabilities in the aerodynamic analysis of lifting surfaces, stemming from the reduced order of the Navier-Stokes fluid equations. Since the early 20th century, LBM has been applied to model complex flows, including supersonic and hypersonic conditions. Early improvements focused on boundary conditions, such as bounce-back and no-slip, with the latter reducing errors in velocity calculations at walls [162, 163]. NASA's Langley Research Center developed software like PowerFLOW and CFL3D to enhance LBM's computational capabilities, comparing their performance for simulating flows over a NACA0012 aerofoil [172]. Despite initial errors in PowerFLOW due to grid resolution limitations, LBM showed potential for high Reynolds number flows when adapted to generalized coordinates and combined with the Baldwin-Lomax turbulence model [174, 173]. LBM's versatility extends to unsteady flow analysis, such as studying multi-element aerofoils under varying angles of attack. Simulations revealed detailed insights into slat cove dynamics, shear layer behaviour, and vortex shedding, aligning well with experimental and numerical data [175].

The LBM has evolved to include various boundary conditions and collision factors, enabling the simulation of complex flows, including supersonic regimes. Recent advancements include the double distribution function (DDF-LBM), which accurately simulates polyatomic gases at high speeds by extending numerical equilibrium approaches to reproduce multiple moments of the Maxwell-Boltzmann distribution [176, 177]. Despite using only 39 discrete velocities in three dimensions, this model handles high Reynolds number flows by

incorporating Knudsen number-dependent relaxation times for stability. The DDF-LBM's effectiveness is demonstrated through various simulations, such as the 1D Riemann problem for handling discontinuities and supersonic flow past a NACA0012 aerofoil and a sphere, showing excellent performance in low-viscosity, supersonic regimes. It also outperforms previous methods in computational efficiency on both CPU and GPU architectures, indicating significant advancements in fluid dynamics simulations, particularly for supersonic and low-viscosity flows. Additionally, a hybrid recursive regularization (HRR) collision operator on a D3Q19 lattice stencil has been employed for three-dimensional high-speed compressible flows across subsonic and supersonic regimes [178, 232, 233]. This model simplifies implementation with an improved thermal equilibrium distribution function and incorporates a shock-capturing scheme for supersonic flows. It effectively simulates various test cases, including isentropic vortex convection, non-isothermal acoustic pulse propagation, and supersonic flow over a bump, demonstrating versatility and accuracy across different speed regimes. Further development includes a three-dimensional Rotated Lattice Boltzmann Flux Solver (RLBFS) for high-speed compressible flows [179]. This method discretizes the Navier-Stokes equations using the finite volume approach and computes fluxes at cell interfaces. It ensures stability with a hybrid approach for convective flux computation and validates its effectiveness through various compressible flow simulations, confirming its accuracy for aeronautics and astronautics applications. Comparisons with other methods, such as the discrete vortex method (DVM) and the immersed boundary-lattice Boltzmann method (IB-LBM), highlight the strengths and limitations of each approach for simulating insect flapping flight dynamics. The IB-LBM provides greater accuracy by solving the full incompressible Navier-Stokes equations with moving boundaries, despite higher computational costs [180].

Multi-phase simulations are crucial for understanding complex fluid behaviours, including phase separation and interface dynamics, across various engineering and scientific fields. Traditional methods often face computational challenges, particularly in high-speed scenarios. The solver's efficiency and ability to handle complex interfaces are suited for these tasks. Its capacity to manage compressible flows and model shock waves is especially beneficial in supersonic flows where compressibility effects and shock interactions are significant. Additionally, LBM's scalability and compatibility with parallel computing architectures make it ideal for large-scale supersonic flow simulations. By focusing on the application of LBM in simulating multi-phase flows, particularly under supersonic conditions, this section underscores LBMs potential to drive interdisciplinary research and innovation in fluid dynamics across different speed regimes. A three-dimensional multiphase LBM model has been employed to investigate spontaneous phase transport within complex porous media

[181]. The model's accuracy was first validated against analytical solutions derived from Young's and Laplace's laws. It was then used to simulate various scenarios, such as droplet penetration into porous layers, liquid transport in porous channels, and droplet extraction from porous substrates. The study examined the effects of multiple geometrical and flow parameters, including porosity, density ratio, Reynolds number, Weber number, Froude number [182], and contact angle. Key findings from the parametric study highlighted significant relationships between these parameters and phase transport behaviours.

6.2 Lattice Boltzmann Method

The lattice Boltzmann method in itself is a reduced-order method for particle flow simulation [234]. It is successful in predicting the force coefficient by modelling the fluid domain in the macroscopic scale. Each fluid particle is given a large number of degrees of freedom which enables it to propagate within a discrete lattice. The common form of the lattice for two-dimensional analysis is D2Q9 and for three-dimensional is D3Q19 which is shown in Fig. 6.1. With the D2Q9 lattice arrangement the position of fluid is defined with 9 possibilities which

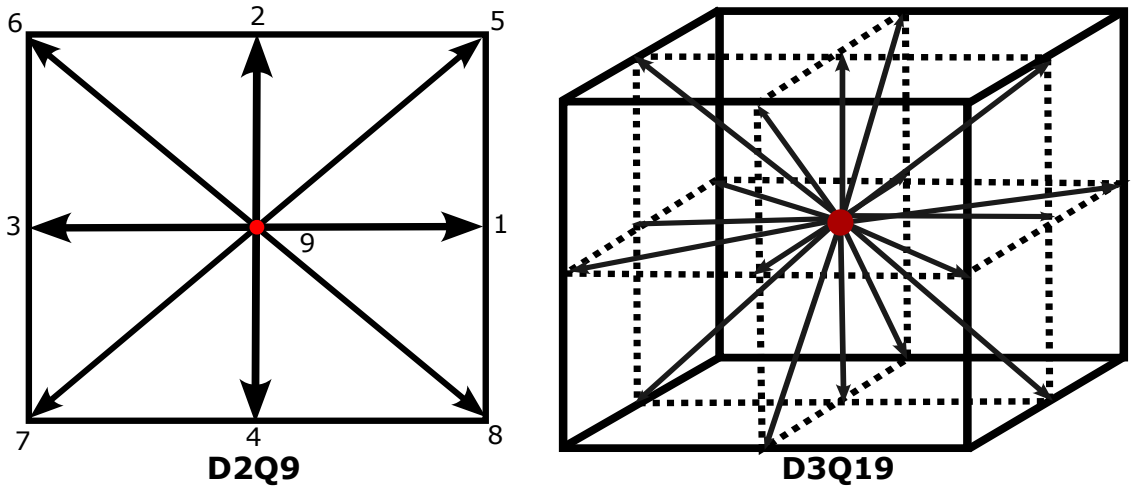


Fig. 6.1 D2Q9 and D3Q19 lattice structure for LBM.

are 1-8 and 9^{th} is the position of rest. When working with the lattice Boltzmann method, the non-linear fluid characteristics are defined by calculating the probability distribution function on the microscopic scale where the average fluid molecule length is between the mean free path length and the characteristic length of the particle [235]. The distribution function is defined as $f(\xi,x,t)$ where ξ is the molecular velocity of a particle at a position x at a time t. This distribution function enables continuum description on the kinetic level. The rate of change of this distribution function with time is explained below;

$$\frac{\mathrm{d}f(\xi,x,t)}{\mathrm{d}t} = \left(\frac{\partial}{\partial t} + \frac{\mathrm{d}x}{\mathrm{d}t}.\frac{\partial}{\partial x} + \frac{\mathrm{d}\xi}{\mathrm{d}t}\frac{\partial}{\partial \xi}\right).f(\xi,x,t) \tag{6.1}$$

$$= \left(\frac{\partial}{\partial t} + \xi\frac{\partial}{\partial x} + \frac{f}{\rho}\frac{\partial}{\partial \xi}\right).f(\xi,x,t) \tag{6.2}$$

$$= \Omega(f) \tag{6.3}$$

Ω in equation 6.7 is the rate of change of fluid particles in time which is also termed as Bhatnagar Gross Krook (BGK) collision model which has a single relaxation time referred to as τ shown in the equation 6.8.

$$\Omega(f) = -\frac{1}{\tau}(f - feq) \tag{6.4}$$

The collision operator is defined to model a two-particle collision for simplified conditions,

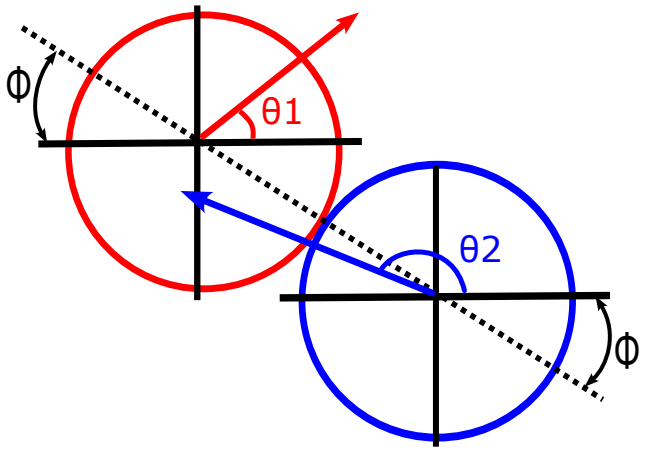


Fig. 6.2 Representation of BGK collision factor for LBM.

when the system reaches equilibrium when left alone BGK collision operator [236] is preferred as shown in Fig. 6.2. This use of discretised velocity into a 2D and 3D lattice [237] defining its domain and initial conditions and collision operator when can replace the use of Navier Stokes equation with few terms reducing the computational time required for high order methods using Hermite expansions.

6.3 Methodology

Each flow particle is discretised in a large number of degrees of freedom for a fluid particle adopting various forms of lattice, most common form of the lattice for two-dimensional

analysis is D2Q9 and for three-dimensional is D3Q19 which is shown in Fig. 6.1. With the D2Q9 lattice arrangement the position of fluid is defined with 9 possibilities which are 1-8 and 9^{th} is the position of rest. When working with the lattice Boltzmann method, the non-linear fluid characteristics are defined by calculating the probability distribution function on the microscopic scale where the average fluid molecule length is between the mean free path length and the characteristic length of the particle [235]. The distribution function is $f(\xi,x,t)$ where ξ is the molecular velocity of a particle at a position x at a time t. This distribution function enables continuum description on the kinetic level. The rate of change of this distribution function with time is explained below;

$$\frac{\mathrm{d}f(\xi,x,t)}{\mathrm{d}t} = \left(\frac{\partial}{\partial t} + \frac{\mathrm{d}x}{\mathrm{d}t}.\frac{\partial}{\partial x} + \frac{\mathrm{d}\xi}{\mathrm{d}t}\frac{\partial}{\partial \xi}\right).f(\xi,x,t) \tag{6.5}$$

$$= \left(\frac{\partial}{\partial t} + \xi\frac{\partial}{\partial x} + \frac{f}{\rho}\frac{\partial}{\partial \xi}\right).f(\xi,x,t) \tag{6.6}$$

$$= \Omega(f) \tag{6.7}$$

Ω in equation 6.7 is the rate of change of fluid particle in time which is also termed as BGK (Bhatnagar Gross Krook) collision model which has a single relaxation time referred to as τ shown in the equation 6.8.

$$\Omega(f) = -\frac{1}{\tau}(f - feq) \tag{6.8}$$

For the formulation of the discrete velocity Boltzmann equation, the general form of discrete velocity Boltzmann methods is used and the equilibrium distribution function is played for the given lattice and then the correction term is introduced for thermal and compressible limitation of the method. Finally, Chapman-Enskog is used to derive the macroscopic equation of this hybrid method.

$$\frac{\partial fi}{\partial t} + c_{i,\alpha}\frac{\partial fi}{\partial x_a} = -\frac{1}{\tau}(fi - fi^{eq}) + \psi_i \tag{6.9}$$

The equilibrium distribution function is given by:

$$f_i^{eq} = w_i\rho\left[1 + \frac{\mathbf{c}_i\cdot\mathbf{u}}{c_s^2} + \frac{(\mathbf{c}_i\cdot\mathbf{u})^2}{2c_s^4} - \frac{\mathbf{u}\cdot\mathbf{u}}{2c_s^2}\right], \tag{6.10}$$

where w_i are the lattice weights. The source/forcing term Ψ_i is introduced separately as:

$$\Psi_i = F \cdot \frac{\mathbf{c}_i}{c_s^2} f_i^{eq}, \tag{6.11}$$

where F represents external or body forces. In this study a single-phase isothermal model is used, without multiphase extensions.

In this hybrid approach [183] the mass and moment are derived by using the discrete velocity Boltzmann equation (DVBE) which is based on the discrete particle distribution function fi which is shown as 6.9. In this equation, the BGK collision operator has been used. τ is the relaxation time factor for this distribution function moving from fi towards fi^{eq}. ψi refers to the correction terms for the thermodynamic distribution factor and $c_{i\alpha}$ is the correction term for discrete microscopic velocities where the Greek subscript α denotes the spatial directions in Cartesian coordinates. These velocities distributions are spread throughout the lattice often termed as *DnQm*, where n is the spatial dimensions and m is the discrete velocities. The velocities distribution for the D2Q9 lattice form Hermite polynomial expansion is stated as:

$$c_{i,x} = [0,1,0,-1,0,1,-1,-1,1]C_0 \tag{6.12}$$

$$c_{i,y} = [0,0,1,0-1,1,1,-1,-1]C_0 \tag{6.13}$$

where, $C_0 = \sqrt{rT_r/c_s}$ for a gas at operating conditions making T_r a reference temperature, r the molecular gas constant and $c_s = 1/\sqrt{3}$ which is often called as the lattice constant. Computing the moments of the distribution function yields mass at the zeroth order as:

$$\rho \equiv \sum_{i=0}^{m-1} fi = \sum_{i=0}^{m-1} fi^{eq} \tag{6.14}$$

and the moment at the first order:

$$\rho u_\alpha \equiv \sum_{i=0}^{m-1} c_{i,\alpha} fi = \sum_{i=0}^{m-1} c_{i,\alpha} fi^{eq} \tag{6.15}$$

For the formulation of the hybrid approach, it uses the entropy equation to compute the temperature $\theta = T/T_r$ of the defined system. The equation of entropy is given by:

$$\frac{\partial s}{\partial t} + u_\alpha \frac{\partial s}{\partial x_\alpha} = -\frac{1}{\rho\theta} \frac{\partial}{\partial x_alpha} \left(-\lambda \frac{\partial \theta}{\partial x_\alpha} \right) + \frac{\Phi}{\rho\theta} \tag{6.16}$$

where $\lambda = \mu/(c_p P_r)$ is the heat conductivity and Φ the viscous heat production term which is defined as;

$$\phi = \mu \left(\frac{\partial u_\alpha}{\partial x_\beta} + \frac{\partial u_\beta}{\partial x_\alpha} - \frac{2}{D} \frac{\partial u_\gamma}{\partial x_\gamma} \delta_{\alpha\beta} \right) \frac{\partial u_\alpha}{\partial x_\beta} \tag{6.17}$$

$\delta_{\alpha\beta}$ is the Kronecker symbol [238], Pr, μ and D are the Prandtl number, the dynamic viscosity and the spatial dimension of the system. This hybrid DVBE (Discrete Velocity Boltzmann Equation)proposed method solves for D2Q9 lattice with ten partial differential equations resolved around the mass and the momentum while a single PDE is used to solve for entropy, which is used to calculate the temperature obtained. The dependency of the entropy equation on the DVBE can be explicitly seen through ρ and u_α.

When coupling the vortex lattice method (VLM) with the hybrid lattice Boltzmann method (LBM) for improved compressibility correction at supersonic speeds, several benefits emerge. This coupling allows for accurate prediction of both steady-state and unsteady aerodynamics by leveraging the strengths of each method. It should be emphasised that the present coupling between LBM and VLM is partial. LBM provides compressibility corrections in the form of pressure distributions that adjust the VLM predictions, but there is no two-way feedback. A more complete hybridisation, where VLM and LBM iteratively exchange information, is left for future work.

The LBM's ability to handle compressible flows enables a more precise treatment of compressibility effects, crucial at high speeds. Additionally, complex geometries can be accommodated, expanding the applicability of the VLM. The coupling also reduces numerical dissipation, leading to improved resolution of flow features. Finally, computational efficiency is maintained by using the VLM for initial estimates, reducing the overall computational cost compared to using the LBM alone. Together, these advantages make the coupling of VLM and LBM a valuable approach for enhanced compressibility correction in supersonic aerodynamic analysis. The methodology presented herein outlines a comprehensive approach for conducting multiphase simulations using the lattice Boltzmann method (LBM) to study compressibility effects in supersonic flows. The simulation framework consists of two fundamental components: the main code file and the LBMmultiPhase function.

The main code file serves as the initialisation of the simulation process, initializing critical parameters such as Reynolds number (R), lattice spacing parameter (A), density ratio (ρ), and relaxation factor (γ). These parameters are used to define unique simulation configurations that specify grid dimensions, maximum simulation time, and other essential parameters required for multiphase fluid dynamics modelling.

The multiphase function encapsulates the core computational tasks of the LBM simulation. It initializes lattice constants (w, c) for the D2Q9 lattice and defines boundary conditions,

such as the flat plate region, within the simulation domain. This function manages the density distribution functions (R, B) and iteratively updates them through collision, boundary condition enforcement, and streaming processes. Notably, specialized bounce-back boundary conditions are applied to simulate interactions with solid boundaries represented by the flat plate region. The function also monitors simulation convergence and computes essential metrics (e.g., Sum of forces, maximum velocity, velocity scaling factor) to assess the steady-state behaviour of the multiphase flow. This representation of improved LBM relation for multiphase modelling is shown in Fig.6.3 The collision function plays a pivotal role in modelling collision dynamics within the LBM simulation framework, particularly focusing on the interaction between lattice nodes representing different phases. For the coexistence of liquid and gas phases Maxwell area construction rule [27] is used with the Van-Der-Waals equation of state[239] as shown in Fig. 6.4 which states that for a given coexistence of liquid-gas phase at temperature T the equivalent chemical potential μ at the boundary must be equal. From Fig. 6.4 can be seen the phase transition occurs at pressure p_0 which is represented by the area in dark grey and the light grey. This solver is assumed to be of constant volume and temperature to ease the complexity of the multi-phase supersonic simulation. For the modelling of such a complex multi-phase correlation, a diffuse interface model was used

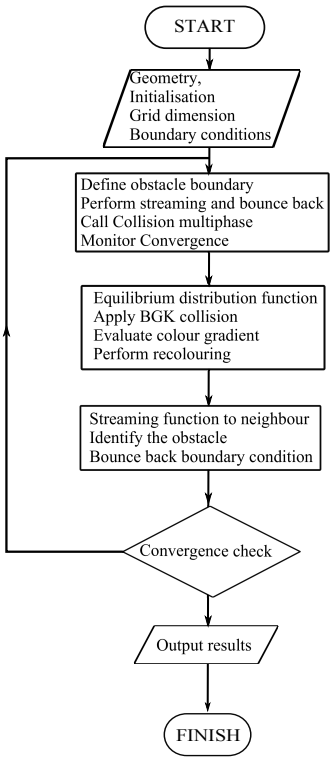


Fig. 6.3 LBM multiphase modelling representation.

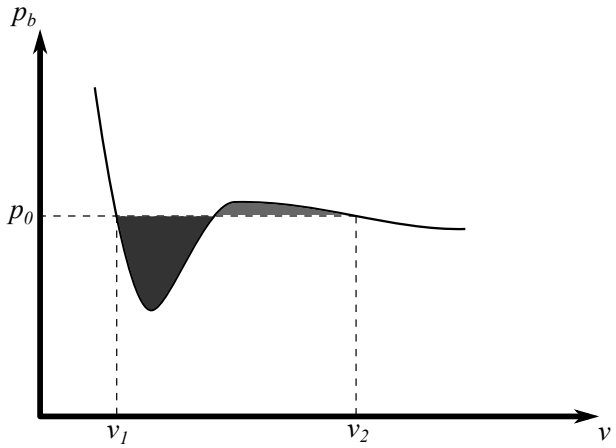


Fig. 6.4 Maxwell area construction rule for Van-der-Waals equation of state at a given fixed temperature.[27]

with the bottom-up approach. Here's an overview of the modelling approach employed in this function:

- The function begins by computing the first and second moments of density for both phases (R, B) based on their distribution functions. These moments are utilized to calculate the fluid velocity components (ux, uy) across the lattice grid, essential for determining equilibrium distributions based on specified relaxation parameters.
- Using the BGK [236] collision operator, the function updates the distribution functions (R, B) to approach equilibrium by applying relaxation with respective collision rates.
- Additionally, the function evaluates the colour gradient across neighbouring lattice nodes to compute perturbations and recolouring effects in the two-phase system. This involves manipulating distribution functions based on gradient weights and perturbation factors (b) to simulate phase separation and recolouring dynamics.
- Moreover, the function incorporates recolouring mechanisms to adjust distribution functions (R, B) according to the computed colour gradient and fluid properties (rhoR, rhoB). This step accounts for phase interactions and perturbations induced by fluid motion and colour gradients within the lattice grid.
- Finally, the distribution functions (R, B) are rearranged to ensure correct streaming directions, facilitating accurate propagation of lattice information during subsequent simulation steps.

This approach contributes to the fidelity and realism of multi-phase flow simulations, supporting in-depth investigations into compressibility effects and fluid dynamics under varying conditions.

6.3.1 Bounce-back boundary condition

The two common boundary conditions were bounce-back and no-slip boundary conditions as shown in Fig. 6.5, many approaches were used for these boundary conditions [164–170]. This research focused on implementing an improved non-slip boundary condition at the wall. With this approach, the initial calculation of the unknown distribution function is assumed to be an equilibrium distribution function with counter slip velocity for accurately predicting the fluid velocity at the wall is equal to the wall velocity while keeping the boundary between the fluid particle and the flat wing to bounce back conditions. The streaming function is designed to implement bounce-back boundary conditions within the LBM framework, specifically for the case of no-slip boundaries. The bounce-back boundary condition is:

$$f_i(\mathbf{x}_b, t+\delta t) = f_{\bar{i}}(\mathbf{x}_b, t) \tag{6.18}$$

This function manipulates distribution functions (F) to enforce the bounce-back condition at solid obstacle boundaries represented by the obstacle matrix. The representation of this boundary condition is shown in Fig. 6.6 which shows the no-slip boundary for the wall and bounce-back conditions for a solid obstacle. Here's a summary of the modelling approach

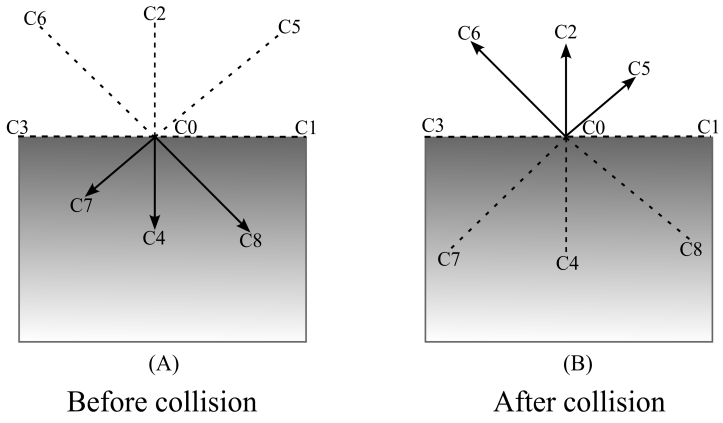


Fig. 6.5 No-slip boundary condition with collision to channel wall [21].

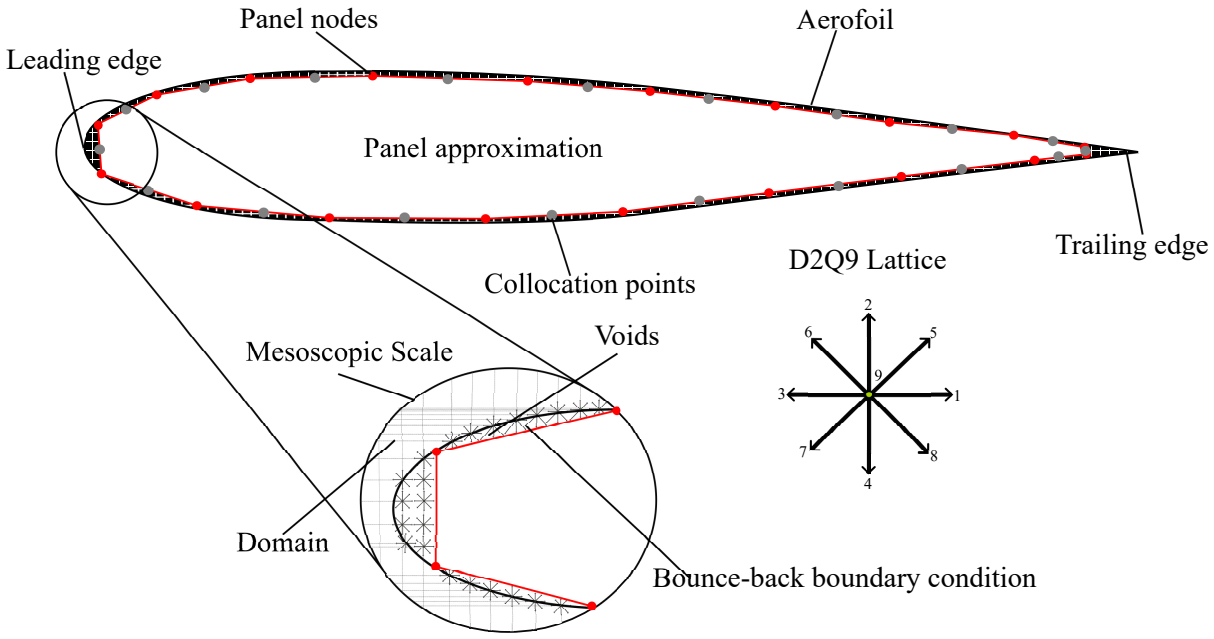


Fig. 6.6 Representation of D2Q9 lattice arrangement for boundary conditions around a NACA0012 aerofoil.

employed in this function: The streaming function begins by streaming distribution functions (F) to adjacent lattice nodes based on specified neighbour indices. This streaming operation corresponds to the standard propagation step in LBM, where distribution functions are moved to neighbouring lattice nodes according to predefined velocity vectors. Next, the function identifies cells adjacent to obstacles where the bounce-back boundary condition needs to be applied. This is achieved by converting the linear indices of neighbouring cells to 2D coordinates (i, j) within the lattice grid. Valid indices are selected to ensure they fall within the simulation domain boundary. Subsequently, the function creates a logical array to identify cells adjacent to obstacles based on the obstacle matrix. The east-obstacle array flags cells that require the bounce-back condition at the east-facing boundary. Finally, the bounce-back condition is enforced by reversing the flow direction (4 to 8) for distribution functions (F) at cells identified as adjacent to obstacles.

6.3.2 Correction term

The solver initializes density distribution functions (R and B) based on specified parameters such as density, flow velocity, and lattice constants. Collision and streaming operations are performed throughout the simulation loop to update the distribution functions, simulating fluid dynamics. A key term, sigma Laplace, is computed as the product of the pressure difference (Pin - Pout) across the airfoil region and a characteristic length scale. This term

represents a correction factor accounting for changes in pressure and viscosity, crucial for accurately modelling compressible flows. The percentage error between Sigma Laplace and a predicted compressibility correction factor (Sigma Predict) is monitored to assess simulation convergence and accuracy, providing insights into the behaviour of supersonic flows around the airfoil. LBM, utilizing the D2Q9 lattice, handles complex boundary conditions near the wing surface through bounce-back conditions, while VLM computes the induced velocities and vortex strengths in the surrounding flow. This coupling allows for a more accurate representation of the flow field, capturing both the boundary layer effects and the vortex dynamics effectively. This conjugate is represented as:

The induced velocity at time step t is:

$$\mathbf{v}(t) = \sum_{j=1}^{N} \Gamma_j(t)\mathbf{A}_{ij} + \sum_{k=1}^{t-1} \Gamma_k \mathbf{B}_{ik} \tag{6.19}$$

The boundary condition for no penetration of the flow is:

$$\mathbf{v}(t) \cdot \mathbf{n} = -V_\infty \sin(\alpha(t)) \tag{6.20}$$

Combining these, we get the system of linear equations:

$$\sum_{j=1}^{N} \Gamma_j(t)(\mathbf{A}_{ij} \cdot \mathbf{n}_i) + \sum_{k=1}^{t-1} \Gamma_k(\mathbf{B}_{ik} \cdot \mathbf{n}_i) = -V_\infty \sin(\alpha(t)) \tag{6.21}$$

LBM influenced desnity effects and its coupling with VLM The Lattice Boltzmann Method (LBM) used here is a weakly–compressible, isothermal BGK scheme that evolves particle distribution functions $f_i(\mathbf{x},t)$ along discrete velocities $\mathbf{e}_i$. Its update reads

$$f_i(\mathbf{x}+\mathbf{e}_i\Delta t, t+\Delta t) \;=\; f_i(\mathbf{x},t) \;-\; \omega\big[f_i(\mathbf{x},t) - f_i^{\mathrm{eq}}(\rho,\mathbf{u})\big] \;+\; \Delta t\, S_i(\Psi), \tag{6.22}$$

where ω is the collision frequency, f_i^{eq} is the local equilibrium, and $S_i(\Psi)$ is an optional forcing/multiphase source. The macroscopic fields are recovered from moments of f_i:

$$\rho \;=\; \sum_i f_i, \qquad \rho\,\mathbf{u} \;=\; \sum_i f_i\,\mathbf{e}_i \;+\; \tfrac{\Delta t}{2}\,\mathbf{F}. \tag{6.23}$$

For an isothermal LBM, pressure follows an equation of state $p = c_s^2\,\rho$ (with lattice sound speed c_s), so density variations directly produce pressure variations.[2] A local Mach number

[2] If a multiphase (pseudo–potential) option is enabled, the forcing term is derived from a density–dependent potential $\Psi(\rho)$, giving an effective EOS $p(\rho) = c_s^2\rho + \frac{1}{2}G\Psi(\rho)^2$; this is documented where Eq. (6.9) introduces f^{eq} and Ψ explicitly.

can then be estimated as $M_{\text{loc}} = \|\mathbf{u}\|/a$ with $a^2 = \mathrm{d}p/\mathrm{d}\rho$ (equal to c_s^2 in the isothermal limit). These fields allow us to compute both pressure and wall–shear–related quantities near the surface:

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty U_\infty^2}, \qquad C_f = \frac{\tau_w}{\frac{1}{2}\rho_\infty U_\infty^2}, \qquad \tau_w = \mu \left(\frac{\partial u_t}{\partial n}\right)\Big|_w.$$

Coupling strategy. The coupling between VLM and LBM is deliberately kept **modular and one–way** for robustness and speed. VLM (or UVLM) supplies the geometry, panel collocation points, incidence, and steady/unsteady kinematics; a lightweight LBM solve is then run in a surface–fitted neighbourhood (2D slices or local surface patches) to obtain $\rho, p, \mathbf{u}$ consistent with weak compressibility and viscous effects. These LBM fields are *sampled at each VLM panel's collocation point* and used to form *panelwise compressibility/viscous corrections* that adjust the purely potential VLM loads:

$$C_p^\star = \underbrace{C_p^{\text{VLM}}}_{\text{potential}} + \underbrace{\left(C_p^{\text{LBM}} - C_p^{\text{PG}}\right)}_{\text{density / compressibility increment}}, \qquad C_D^\star = C_{D,i}^{\text{VLM}} + \int_{\mathscr{S}} C_f^{\text{LBM}} \, (\hat{t}\cdot\hat{x}) \, \mathrm{d}S.$$

Here C_p^{PG} is the Prandtl–Glauert (PG) corrected pressure coefficient used as a subsonic baseline; the bracketed increment replaces the uniform PG factor with the *locally resolved* LBM density/pressure at each panel. For supersonic cases, the same hook operates after the Mach–cone/Taylor–Maccoll relation has set the conical–shock state, so LBM supplies a *viscous/compressible refinement* to the already shock–consistent panel pressures. In all cases, lift and drag are finally integrated over the lattice using the corrected $C_p^\star$ and the LBM–derived C_f contribution.

Scope and limitations. This "plug-in" design means LBM does not (yet) *feed back* to re-solve VLM circulation Γ or to convect the UVLM wake; it also does not alter other modules (e.g. steady VLM, Mach-cone solver). The advantages are (i) numerical stability, (ii) clear separation of concerns, and (iii) low cost per time step. The trade-off is that shock–induced changes in effective camber/thickness or separation feedback to Γ are only *implicitly* represented through the surface corrections. We highlight this explicitly in the Chapter 6 discussion and outline iterative two–way coupling (re-solving Γ with density-modified influence matrices and updating the wake kinematics) as planned future work.

6.4 Results and Discussion

The results section presents a comprehensive analysis of the C_p distributions across various supersonic aerofoils, specifically NACA 64A010, NACA 64A206, and NACA 64A210.

These aerofoils were selected for their distinct geometrical characteristics and relevance in high-speed aerodynamic applications. The primary objective of this analysis is to validate the Multiphase LBM solver against established computational and experimental methods, particularly the Reynolds-Averaged Navier-Stokes with RANS SST model, which is widely recognised for its accuracy and robustness in predicting aerodynamic phenomena. The focus on comparing the pressure coefficient distribution on the upper surface of the aerofoil in the graph 6.7 is driven by several key aerodynamic considerations. The upper surface of an aerofoil is typically where significant aerodynamic phenomena, such as flow acceleration, pressure gradients, shocks and potential flow separation, occur. These factors are critical in determining the overall lift and drag characteristics of the aerofoil, especially under supersonic conditions. In addition to pressure coefficient C_p, the skin friction coefficient C_f is computed:

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho_\infty U_\infty^2}, \tag{6.24}$$

where τ_w is the wall shear stress. While C_p represents pressure loading, C_f quantifies viscous drag, and both are needed for a complete aerodynamic assessment.

The upper surface experiences a sharp reduction in pressure due to the increased velocity of the airflow, which is a defining aspect of the aerofoil's aerodynamic performance. This comparison helps identify the accuracy of the models in predicting key features like shock waves, flow separation, and reattachment points, which are essential for the design and optimisation of aerofoils in high-speed applications. The study utilises both empirical data

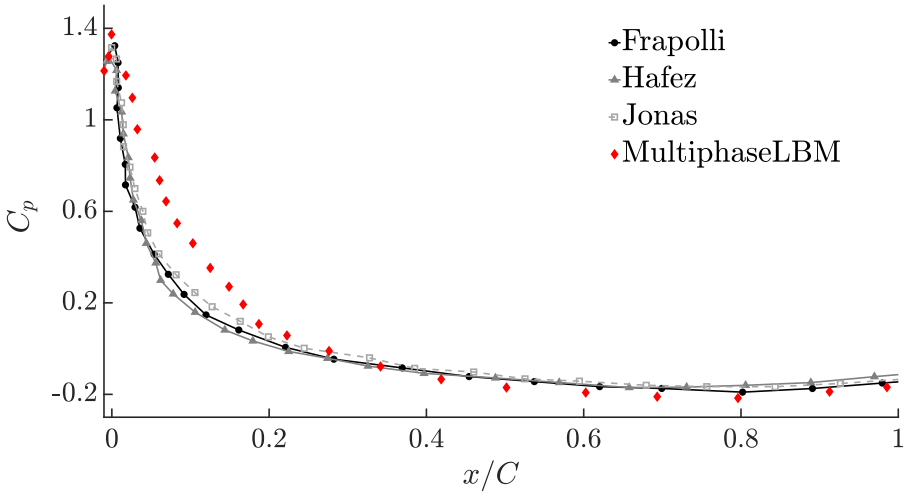


Fig. 6.7 Solver validation with published experimental data for C_P distribution across NACA0012 aerofoil at Mach 1.5 and $\alpha = 0°$

from the literature and numerical simulations to benchmark the Multiphase LBM code. By comparing the results from these aerofoils, this section aims to assess the solver's ability to accurately capture critical aerodynamic features, such as pressure distributions and flow separation points, under supersonic conditions. The literature provides a foundation for understanding the expected performance of these aerofoils, allowing for a detailed comparison between the Multiphase LBM and traditional CFD methods like RANS SST. This validation process is crucial for establishing the reliability of the Multiphase LBM as a viable tool for aerodynamic analysis, particularly in scenarios where reduced-order modelling offers significant computational and financial advantages

Fig. 6.7 presents a comparison of pressure coefficient distributions (C_p) along the chord of a NACA0012 aerofoil at Mach 1.5, derived from various sources [228, 240, 241], including experimental data[242, 243], numerical studies, and the MultiphaseLBM computational model. The dataset from Frapolli [242], represented by a solid black curve with circular markers, provides a baseline of empirical data, reflecting the actual pressure distribution observed during physical testing. This data was subsequently refined by Hafez [243], shown as a solid grey curve with triangular markers, indicating corrections or enhancements in the experimental setup or analysis, maintaining consistency with Frapolli's findings but with slight improvements. The Lattice Boltzmann Method (LBM) code developed by Jonas [176] which is based on the improved work done in studying the improved collision and stability model [235, 244], depicted by a light grey curve with square markers, demonstrates a high degree of fidelity in simulating real-world aerodynamic behaviour, closely aligning with the experimental results. The MultiphaseLBM model, marked by red diamonds, integrates a Multiphase LBM with the Vortex Lattice Method (VLM) to better account for boundary layer interactions and compressibility effects. However, it displays noticeable deviations from both the experimental and Jonas's LBM data, particularly near the leading edge, where the C_p values are overestimated. This increase in the value suggests that while the MultiphaseLBM captures the overall aerodynamic trends, it may require further refinement in modelling specific phenomena such as compressibility and boundary layer dynamics. To verify that compressibility effects are treated correctly, an additional check was performed by comparing the LBM compressible solver with the incompressible solver plus P–G correction. At low Mach numbers both methods yield nearly identical results, while at higher Mach numbers the P–G correction deviates significantly, confirming that LBM properly accounts for nonlinear density variations.

Fig. 6.8 and 6.9 illustrate the distribution of mean C_p along the chord of a NACA0012 aerofoil at Mach 1.5, evaluated at two different angles of attack (AoA): 4 degrees in Fig.6.8 and 8 degrees in Fig.6.9 The comparison involves results from a conventional multiphase

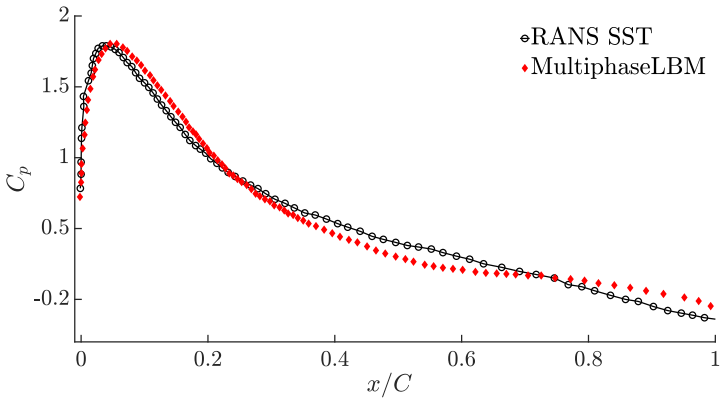


Fig. 6.8 The distribution of mean pressure coefficient C_p from CFD RANS SST compared with Multiphase LBM solver for $\alpha = 4^\circ$

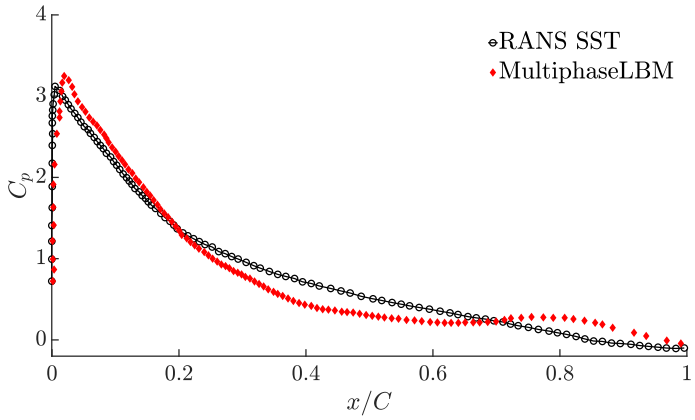


Fig. 6.9 The distribution of mean pressure coefficient C_p from CFD RANS SST compared with Multiphase LBM solver for $\alpha = 8^\circ$

Computational Fluid Dynamics (CFD) approach using the Reynolds-Averaged Navier-Stokes (RANS) model with the Shear Stress Transport (SST) turbulence model, depicted by the black curve with circular markers. In the thesis, "RANS SST" refers to the k- SST (Shear Stress Transport) used within a RANS framework (the shorthand "RANS SST" was imprecise). We selected k- SST over a pure k- model because SST blends k- near the wall (good low-Re behaviour and wall-shear prediction) with k- in the free stream (reduced sensitivity to the specific dissipation rate of the free stream via the Menter blending functions, and introduces a shear-stress limiter that improves predictions in the adverse pressure gradient and separation regimes, precisely the conditions present in our transonic/supersonic cases with potential shock-boundary layer interaction. The computational domain extended $20c$ in all directions from the geometry to minimise boundary reflections, with c as the reference chord. An

unstructured hybrid mesh of approximately 5 million cells was used, with local refinement near the wall and expected shock locations. Boundary conditions were defined as velocity inlets and pressure outlets, with freestream Mach number M_∞ and angle of attack α. The k–ω SST turbulence model was employed. Simulations were run until residuals dropped below 10^{-6} and aerodynamic coefficients converged within 0.1%. A grid convergence study was performed using coarse, medium, and fine meshes to quantify numerical uncertainty. This method utilises a volume of fraction multiphase simulation with a fine mesh comprising 1.5 billion cells, providing highly accurate results as established by previous research in the field [245, 246]. In contrast, the red curve with diamond-shaped markers represents data from the Multiphase LBM code, which employs a double distribution function approach tailored for two-phase simulations based on density distribution. The Multiphase LBM model, while innovative, exhibits discrepancies compared to the RANS SST results, particularly at higher x/C values where flow separation occurs. This separation becomes more pronounced as the AoA increases, leading to a significant divergence between the results from the RANS SST model and the Multiphase LBM code.

At a 4-degree AoA, the multiphase LBM results are relatively closer to the RANS SST data, though some deviations are noticeable. However, at an 8-degree AoA, the discrepancies become more evident, especially after the quarter-chord line, where flow separation intensifies. This is manifested by the larger gap between the boundary predictions of the Vortex Lattice Method (VLM) used in the multiphase LBM code and the more detailed predictions from the RANS SST model. The increasing discrepancy at higher AoA suggests that the current lattice configuration in the multiphase LBM code, such as using a D2Q9 lattice, may not be sufficiently dense to capture the complex flow phenomena associated with higher AoAs and flow separation. To improve the accuracy of the multiphase LBM model, it would be beneficial to incorporate a denser lattice structure, such as D3Q27, and reduce the lattice spacing, though this would also increase computational requirements. The comparison highlights the need for refinement in the Multiphase LBM code to better model the aerodynamic characteristics of the aerofoil, particularly under conditions of high AoA where flow complexities are more pronounced.

Table 6.1 presents commonly used supersonic aerofoils taken into consideration to be analysed using two distinct computational methods: the RANS-SST (Reynolds-Averaged Navier-Stokes with Shear Stress Transport) model and the MultiphaseLBM (Lattice Boltzmann Method) code. The investigation aimed to evaluate the performance and accuracy of the MultiphaseLBM code against a well-established CFD method. The aerofoils considered in this study include the McDonnell Douglas F-15, General Dynamics F-16, and Lockheed Martin F-35A, each with specific root and tip chord configurations. Due to data unavailability,

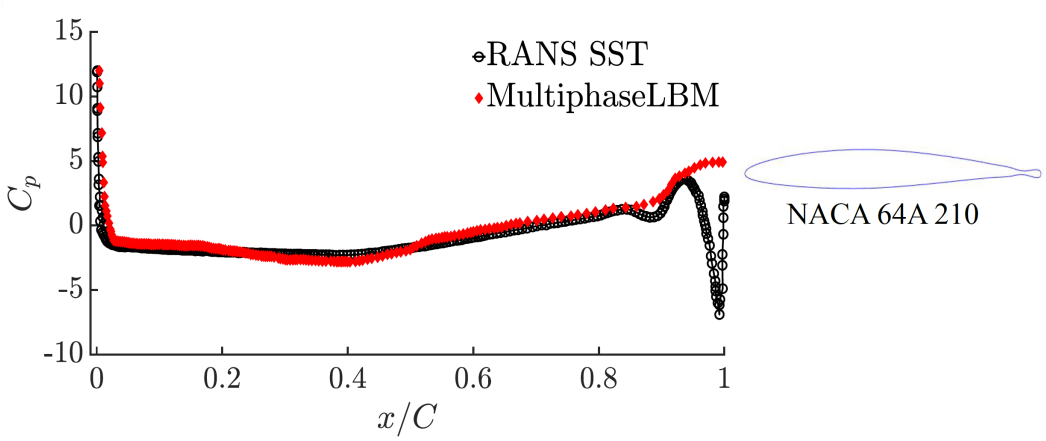


Fig. 6.10 C_P distribution across NACA 64A210 aerofoil at Mach 1.5 and $\alpha = 0°$. Velocity profiles illustrating boundary layer growth and shock–boundary layer interaction; the shape of the profile reflects the combined effects of compressibility and separation.

some modifications were made to the aerofoil specifications: the NACA 64A006.6 aerofoil was replaced with the NACA 64A010, and the NACA 64A204 was substituted with the

Aircraft	**Root chord**	**Tip Chord**
McDonnell Douglas F-15	NACA 64A006.6	NACA 64A203
General Dynamics F-16	NACA 64A204	NACA 64A204
Lockheed Martin F-35A	NACA 64A210	NACA 64A210

Table 6.1 Commonly used supersonic aerofoils under consideration

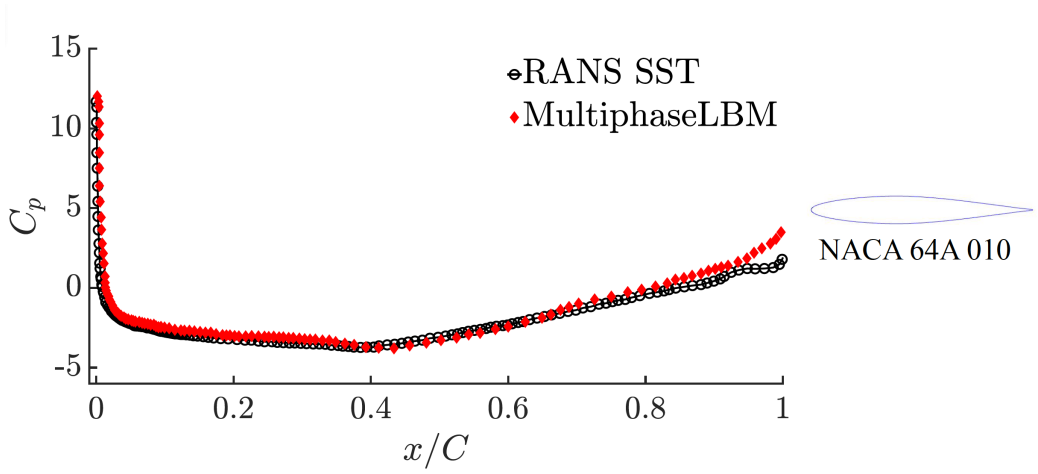


Fig. 6.11 C_P distribution across NACA 64A010 aerofoil at Mach 1.5 and $\alpha = 0°$

NACA 64A206. These changes were necessary to ensure the availability of complete datasets for accurate comparison. The analysis focuses on these symmetrical aerofoils to examine how different shapes, particularly those used in supersonic applications, influence the results obtained from the MultiphaseLBM. Of particular interest is the NACA 64A210 aerofoil, which features a notable dip near the trailing edge. This characteristic presents a unique challenge for computational solvers, requiring them to accurately capture the complex boundary conditions associated with the aerofoil's geometry. The MultiphaseLBM's capability to model these intricate features, including the cavity near the trailing edge, is crucial for validating its accuracy and robustness in simulating supersonic flows. The comparison with RANS-SST results serves to benchmark the MultiphaseLBM's performance, highlighting areas where the model excels and where further refinement may be necessary. This study contributes to the ongoing evaluation and improvement of computational methods for aerodynamic analysis, particularly in the context of supersonic aerofoil design and optimisation.

The C_p distribution across the NACA 64A210 aerofoil at Mach 1.5 is shown in Fig.6.10. This comparison specifically focuses on the aerofoil's performance at an angle of attack ($\alpha = 0$) One of the critical aspects under investigation is the small dip near the trailing edge of the aerofoil, noticeable in the x/C range between 0.8 and 1.0. The RANS SST model, which is considered a reliable and well-validated method, accurately captures this feature, demonstrating its capability to model complex flow characteristics, including minor geometric irregularities and their impact on pressure distribution.

In contrast, the Multiphase LBM solver shows some discrepancies in this region. While it generally follows the trend set by the RANS SST results, indicating a reasonable level of accuracy, it fails to capture the sharp drop and subsequent rise in the C_p values as distinctly as the RANS SST model does. This indicates that the MultiphaseLBM, though capable of simulating the overall aerodynamic behaviour, may lack precision in modelling the intricate details of the flow separation and reattachment phenomena associated with the aerofoil's unique shape, particularly near the trailing edge. This could happen because of the poor pairing between the VLM mesh and the LBM mesh, in the domain of complex aerofoil shapes such pairing creates cavities in the mesh which generates errors and hence decreases accuracy.

The Lattice Boltzmann Method (LBM) is employed here as a *local, panel-wise* provider of compressible and viscous information to augment VLM. Unlike uniform Prandtl–Glauert (PG) scaling or Kármán–Tsien, LBM resolves *non-uniform* density/pressure variations near the surface and yields wall-shear based C_f—both absent in classical VLM. For $M_\infty > 1$, the conical shock state from Taylor–Maccoll still sets the baseline; LBM then refines panel pressures and adds a viscous skin-friction contribution.

The close but not perfect alignment of the Multiphase LBM results with the RANS SST data highlights the need for further refinement in the LBM model. This could involve improvements in the lattice resolution or the incorporation of more sophisticated boundary condition treatments to better capture the nuances of the flow dynamics around complex aerofoil geometries. Similarly further investigation into C_p distribution across the NACA 64A010 shows similar results in Fig.6.11. In this comparison, the RANS SST model provides a detailed and precise depiction of the pressure distribution along the aerofoil surface. The Multiphase LBM solver, while generally capturing the overall trend observed in the RANS SST results, shows slight deviations, particularly in the region close to the leading edge and near the trailing edge. The Multiphase LBM results tend to slightly overpredict the pressure coefficients, especially noticeable in the x/C range beyond 0.8, where the pressure starts to rise again. This overprediction suggests that the Multiphase LBM may not fully capture the effects of compressibility or boundary layer interactions as accurately as the RANS SST model. The C_p distribution for the NACA 64A206 aerofoil is shown in Fig. 6.12. The Multiphase LBM tends to slightly overpredict the pressure coefficients near the leading edge and underpredicts between the mid-chord and trailing edge regions. This is particularly evident in the x/C range from 0.6 to 1.0, where the Multiphase LBM results show a gradual rise in pressure coefficients, whereas the RANS SST data exhibits a more pronounced peak before descending. This comparison underscores the need for further refinement of the Multiphase LBM, particularly in enhancing the resolution and accuracy of boundary layer and shock wave modelling and interaction which are fundamentally not possible with conventional VLM, which is a significant limitation in supersonic applications. These limitations may require approximate shock corrections, potentially coupled with alternative numerical methods or enhanced coupling strategies to achieve acceptable results

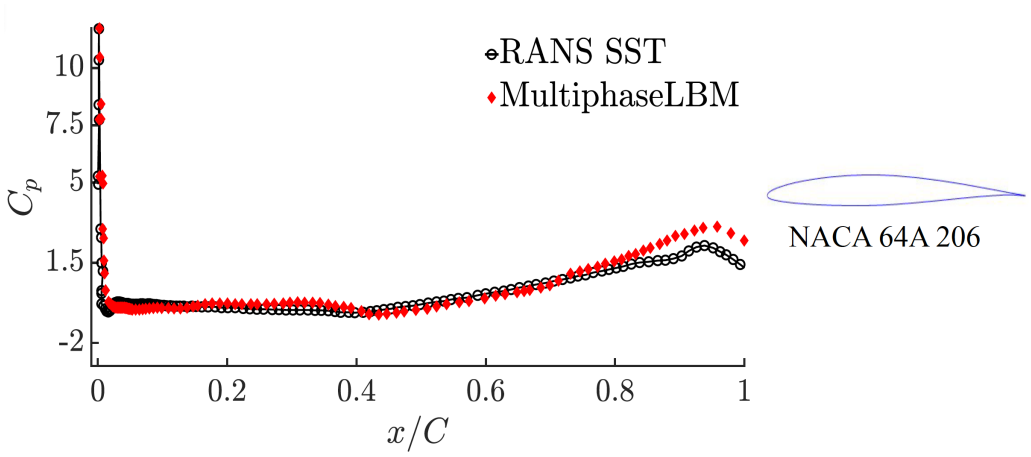


Fig. 6.12 C_P distribution across NACA 64A206 aerofoil at Mach 1.5 and $\alpha = 0°$

for supersonic flows which better align with the high fidelity provided by the RANS SST model.

6.5 Summary

The comparison of C_p distributions across various supersonic aerofoils, including NACA 64A010, NACA 64A206, and NACA 64A210, using both RANS SST and Multiphase LBM methods, reveals several key insights. The RANS SST model consistently provides highly accurate and reliable predictions, capturing detailed flow characteristics such as pressure peaks and dips associated with aerodynamic features like leading-edge effects and trailing-edge separation. In contrast, the Multiphase LBM results, while generally following the overall trend observed in the RANS SST data, show discrepancies, particularly in capturing fine-scale details and handling complex boundary layer dynamics and compressibility effects.

The Multiphase LBM code proves to be a promising alternative to the RANS SST model, particularly due to its reduced-order modelling approach. The integration of the Vortex Lattice Method (VLM) in the Multiphase LBM allows for significant reductions in computational time and resources. This efficiency makes it a cost-effective solution for supersonic aerodynamic analysis, offering substantial savings in both time and financial investment compared to the more complex and resource-intensive RANS SST simulations.

Although LBM is not a common choice for compressibility correction, its novelty here lies in extending a low-order VLM framework with a density-based solver that naturally accounts for nonlinear compressible effects. Limitations include the lack of two-way coupling between VLM and LBM, higher computational cost compared to P–G, and the need for more validation against experimental data. Nevertheless, this approach establishes a foundation for fast hybrid solvers capable of handling compressible unsteady aerodynamics.

For future improvements, enhancing the Multiphase LBM's lattice resolution and incorporating more sophisticated boundary condition treatments are crucial. This could involve adopting denser lattice structures, such as D3Q27, and improving the solver's capacity to model compressible flows and separation phenomena. Additionally, further validation studies across a broader range of aerofoil geometries and flow conditions would be beneficial in refining the model's accuracy and robustness.

In conclusion, while the Multiphase LBM shows promise as a viable alternative to RANS SST, especially with its efficient reduced-order approach, significant refinements are needed to achieve similar levels of accuracy and reliability. These advancements will enhance its applicability in both academic research and practical engineering scenarios, offering a robust and cost-effective tool for simulating complex aerodynamic environments.

Chapter 7

Synthesis of improvements

[1] In the preceding chapters, we explored various improvements to the vortex lattice method (VLM) for supersonic wing design. These enhancements addressed specific challenges in the iterative design process, including advanced unsteady analysis modelling, enhanced drag prediction, and refined shockwave modelling. By incorporating sophisticated techniques such as the Taylor-Maccoll hypervelocity approach and the multiphase lattice Boltzmann method, we have improved the accuracy and efficiency of supersonic boundary interaction analysis.

This chapter synthesizes these improvements, demonstrating how they collectively contribute to a more robust and efficient supersonic wing design process. By integrating advanced dynamic analysis, refined drag and shockwave modelling, and improved viscous characteristics, we present a comprehensive approach to supersonic wing design. The use of TORNADO as the core tool environment facilitated significant advancements without the need for developing the core VLM code from scratch, allowing to focus on specific enhancements.

Through rigorous verification and validation using various aerofoils and wing configurations, the results have shown that these combined improvements provide an alternative for high fidelity solver being quicker to run and using less computational time making it a cost-effective means of generating accurate data sets for supersonic analysis. This integrated approach not only streamlines the design process but also enhances the performance and feasibility of high-performance supersonic wings. In this chapter, we will discuss the synergistic effects of these improvements, their practical implications, and potential future directions for further enhancing supersonic wing design methodologies.

[1]The findings from this chapter are written into Journal article "Improvements in vortex lattice method for design analysis of supersonic wings: verification and validation" which is under review with CEAS Aeronautical Jornal

7.1 Introduction

Designing a supersonic wing is a multifaceted endeavour that requires substantial resources, including significant financial investment, extensive computational power, and considerable time. One of the primary challenges in this process is the rapid generation of comprehensive data sets for various configurations, shapes, and applications. These data sets are critical during the design phase, as they enable necessary adjustments tailored to the specific applications and flight conditions of next-generation aircraft. The development of fast-manoeuvring supersonic jets, in particular, necessitates numerous simulation tests across a range of flight manoeuvres to identify the optimal wing design. This iterative process is well-illustrated by the V-diagram of supersonic wing design in Fig. 7.1, which depicts the progression from selecting basic wing parameters to conducting preliminary analyses and design, and finally to the intricate modelling of the trailing and leading edges to enhance performance. A promising approach to address the high costs and time demands associated with supersonic wing design is the use of low-order models. These models offer a potential solution for generating quick and reasonably accurate data during the preliminary stages of design. By employing low-order models as a support solver rather than as an alternative to high-order Computational Fluid Dynamics (CFD), the cost associated with high-speed computational power can be significantly reduced. This makes it feasible to generate large volumes of data affordably, facilitating quicker decision-making and more informed design choices. The objective of this research is to explore the benefits of using low-order models in the early stages of supersonic wing design. The focus is on how these models can serve as a valuable tool for preliminary analysis, providing a cost-effective means of obtaining reliable data. Specifically, this research aims to demonstrate how low-order models can complement high-order CFD methods, reducing the overall computational expense and time required for the design process. This approach allows for the efficient allocation of resources, saving both time and money and ultimately improving the overall design process for supersonic aircraft. By leveraging low-order models, designers can perform numerous iterations and explore a wider range of design configurations without incurring prohibitive costs. This enables a more thorough investigation of design possibilities, leading to better-optimized wing designs for specific supersonic applications. Moreover, the rapid generation of data through low-order models provides designers with the flexibility to make informed decisions swiftly, thereby enhancing the agility and responsiveness of the design process. Supersonic aircraft design poses significant challenges in terms of both cost and time due to the high computational modelling demands of detailed simulations using high-order CFD. These simulations are crucial for accurately assessing aerodynamic performance across various wing configurations and flight conditions but can be prohibitively expensive and time-consuming,

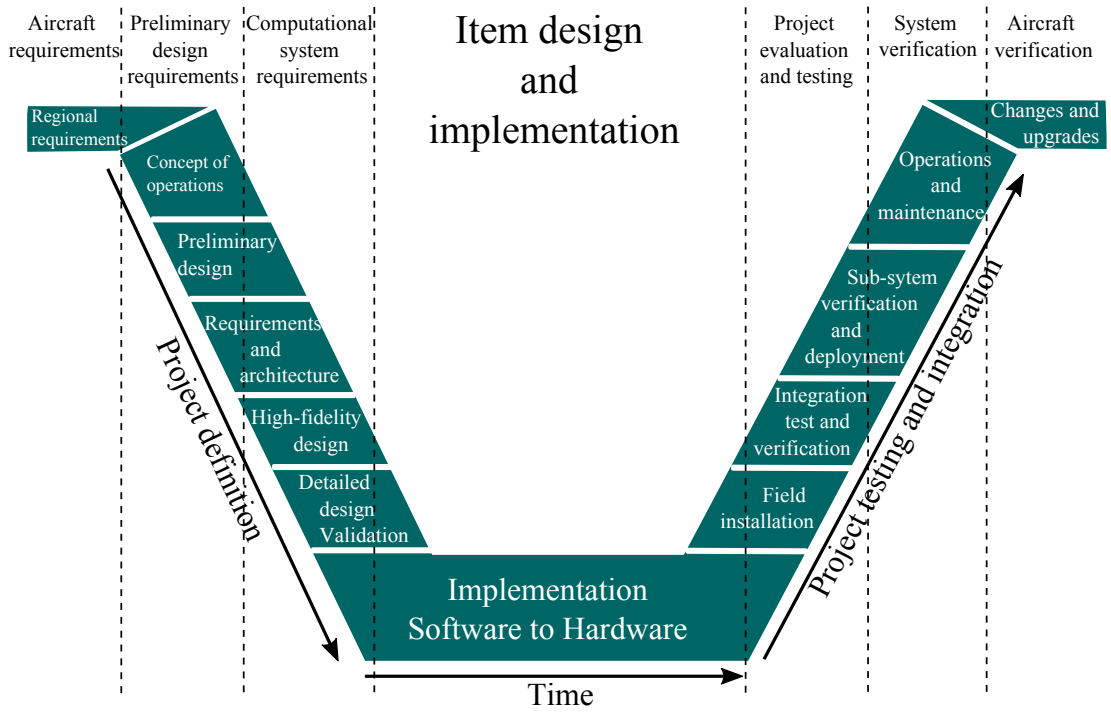


Fig. 7.1 V-diagram for an aircraft design

especially during the preliminary stages of design when multiple iterations are needed to define basic wing characteristics like span, chord length, and sweptback angle. To mitigate these challenges, integrating low-order modelling techniques, such as an advanced vortex lattice method (VLM), offers a promising solution.

Low-order models require significantly fewer computational resources and offer faster simulation times compared to high-order CFDs. This efficiency allows designers to explore a broader range of design parameters quickly and affordably, facilitating rapid iteration and refinement of wing configurations early in the design process. By leveraging low-order models for initial assessments, designers can generate essential aerodynamic data at a fraction of the cost and time required by traditional methods. This approach not only conserves high-performance computing resources for more detailed analyses later in the design cycle but also enhances decision-making by providing timely insights into the aerodynamic performance of various wing designs.

In conclusion, the integration of low-order models into the preliminary stages of supersonic wing design represents a significant advancement in the field. This approach not only alleviates the financial and computational burdens associated with traditional high-order CFD methods but also empowers designers with the tools needed to achieve high-performance modelling wings more efficiently and cost-effectively.

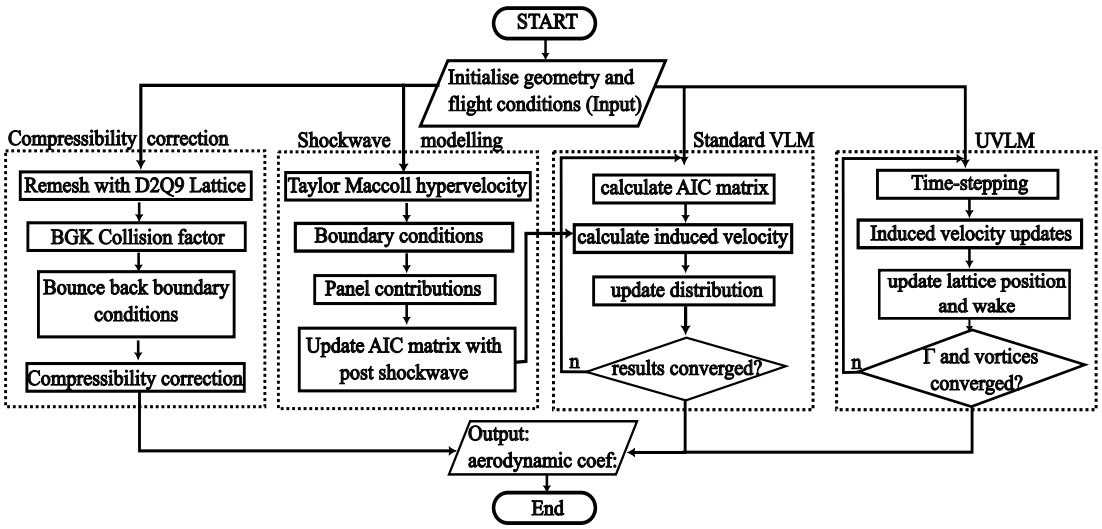


Fig. 7.2 Flowchart of improved VLM.

7.2 Results and discussion

The resulting hybrid solver is designed to achieve improved accuracy in predicting the pressure distribution and aerodynamic forces on aircraft operating at supersonic speeds, emphasising reduced computational cost. To evaluate the performance and validity of the newly developed solver, two test cases were selected, each representing different aerodynamic challenges and configurations. The first test case involved the SCALOS supersonic aircraft [15], featuring a canard configuration. This aircraft was chosen to assess the solver's capability in handling varying dihedral angles of the canard wing, which introduces complex aerodynamic interactions, particularly in the supersonic regime. The second test case involved the Concorde. The Concorde's delta wing configuration provides a unique opportunity to study the pressure distribution across a highly swept wing surface. In this scenario, accurate predictions are crucial for assessing performance and stability. Both test cases were subjected to simulations at Mach numbers of 1 and 1.5, representing critical conditions for supersonic flight. The simulations were conducted for steady and unsteady flow conditions, testing the solver's robustness and accuracy in various aerodynamic environments. The aerodynamic coefficients, such as lift, drag, and pitching moment, were the primary parameters used to compare the results of the hybrid solver against those obtained from higher-order methods, specifically Reynolds-Averaged Navier-Stokes (RANS) simulations. To ensure a fair comparison, the simulations using the hybrid solver and the RANS method were executed on the same computational system, which was equipped with a 13th-generation Intel Core i9 processor and 32 GB of RAM. This setup was chosen to highlight the differences in computational efficiency, with a focus on memory usage and processing time. The hybrid solver's

performance was benchmarked against the RANS method to determine whether it could serve as a viable alternative, particularly in the preliminary stages of aircraft design where computational resources and time are often limited. The analysis focused on the aerodynamic coefficients obtained from both the hybrid solver and the RANS simulations. These results were compared to assess the accuracy of the hybrid method in capturing the key aerodynamic features of supersonic flow over complex geometries. The study also included a detailed examination of the pressure distribution across the aircraft surfaces, with particular attention to the differences observed between the delta wing and canard configurations. Although the thesis integrates LBM into the VLM framework, only four representative plots are provided. This is deliberate: the aim is to demonstrate feasibility and proof-of-concept, rather than to deliver a complete LBM validation dataset. Showing a limited set of plots highlights the integration method and its outputs while keeping the chapter focused. A more comprehensive evaluation of the LBM–VLM coupling, including a larger case set, is proposed as part of the future work.

The computational efficiency of the hybrid solver was also critically examined. Memory allocation and processing time were recorded for both the hybrid solver and the RANS simulations. Graphical comparisons of these computational metrics were included to illustrate the potential advantages of using the hybrid solver in terms of reduced computational cost. The results indicated that the hybrid solver required significantly less memory and computational time compared to the RANS method, making it a more practical option for early-stage aerodynamic analysis.

7.2.1 Case I: SCALOS Supersonic configuration with canard wing configuration

Fig. 7.3 illustrates the SCALOS supersonic aircraft's geometry, with a Vortex Lattice Method (VLM) mesh specifically configured for a fixed wake scenario, where the pink-coloured panels represent the discretization of the wing surfaces into multiple chordwise and spanwise panels. This mesh is designed with a higher panel concentration in regions prone to shockwave formation, guided by the integration of the Taylor-Maccoll hypervelocity equation, which improves accuracy by refining the mesh where the shock cone is predicted to develop. The inclusion of a CAD-designed fuselage offers a visual reference to understand the overall aircraft configuration, though it does not directly impact the VLM calculations. This setup, while currently using a fixed wake model, provides a solid foundation for further developments, including the integration of a dynamic wake model, and enhances the solver's ability to predict supersonic aerodynamic phenomena with greater accuracy.

Fig.7.4 depicts the mesh used for a high-order Unsteady Reynolds-Averaged Navier-Stokes (URANS) simulation in STAR-CCM+, specifically configured for analysing the unsteady dynamic behaviour of an oscillating body. The mesh is carefully structured, with a refined grid around the body to capture the detailed flow dynamics, and progressively coarser grids further away to manage computational efficiency. For this simulation, the Motion Toolkit within STAR-CCM+ is employed to define the reference frame and mesh associated with the unsteady motion. The motion is characterised as an oscillating movement with a periodic frequency of 2 Hz, simulating the dynamic aerodynamic responses under such conditions. This setup allows for a comprehensive analysis of the unsteady aerodynamic forces and flow patterns, which are crucial for understanding the behaviour of the aircraft or component under

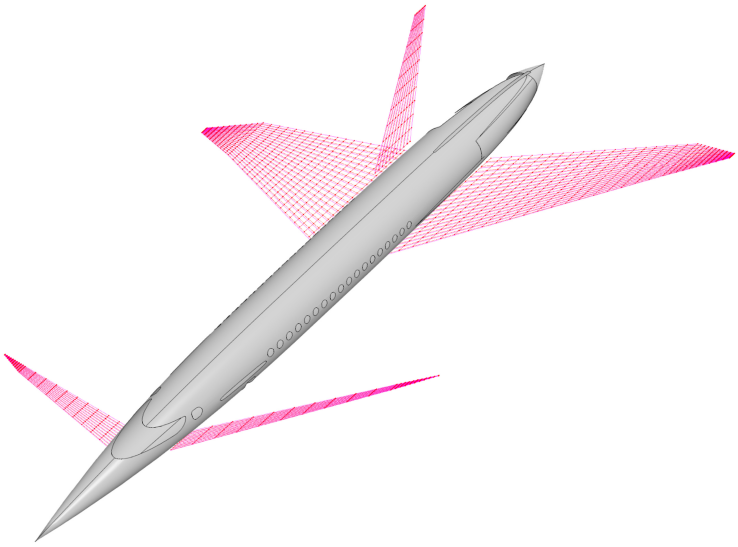


Fig. 7.3 SCALOS concept geometry representing canard wing configuration using VLM.

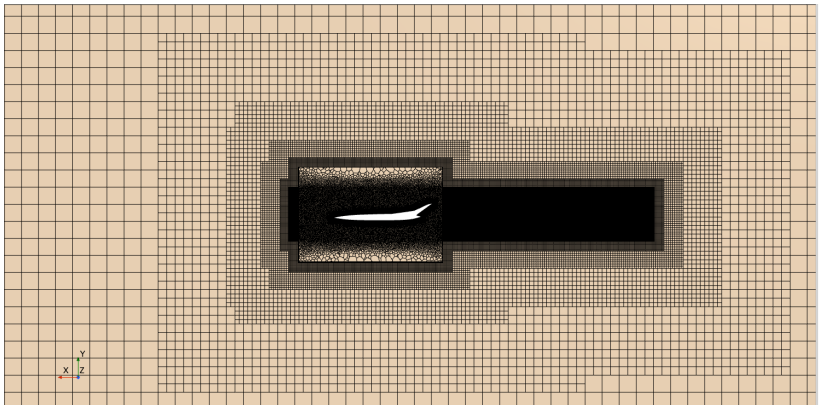


Fig. 7.4 Mesh representation of SCALOS configuration

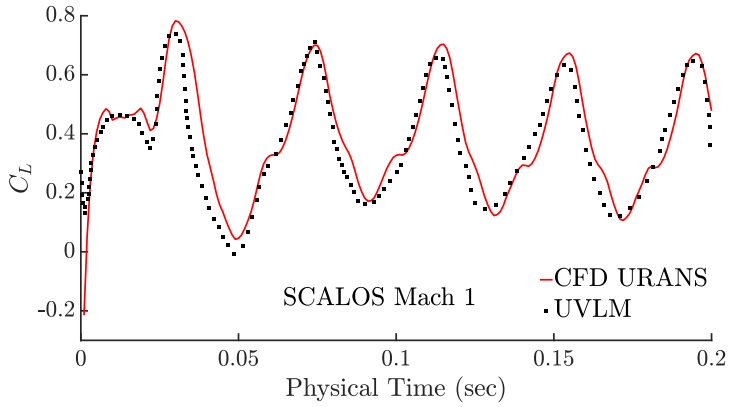


Fig. 7.5 Coefficient of lift comparison between URANS and UVLM for SCALOS aircraft with canard configuration

oscillatory conditions. Fig.7.5 presents a comparative analysis of the C_L for the SCALOS configuration at Mach 1, with data derived from both Unsteady Reynolds-Averaged Navier-Stokes (URANS) simulations and Unsteady Vortex Lattice Method (UVLM) computations. The red curve corresponds to the URANS results, while the black square markers represent the UVLM data. Both simulations were conducted under identical initial conditions and speeds, allowing for a direct comparison of their predictive capabilities. Initially, the two methods show close agreement, with the lift coefficient values aligning well. Consequently, the UVLM results show decent accuracy and further refinement can be improved with added viscous corrections methods.

The figure illustrates a comparative analysis of the C_D between the URANS simulation, represented by the red curve, and the UVLM results, indicated by black square markers, for

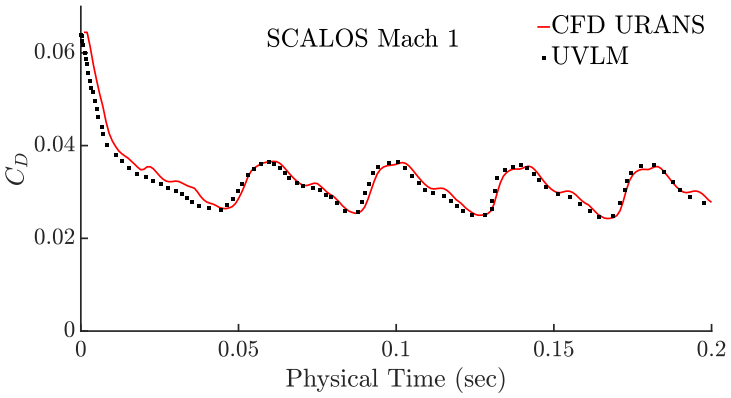


Fig. 7.6 Coefficient of drag comparison between URANS and UVLM for SCALOS aircraft with canard wing configuration

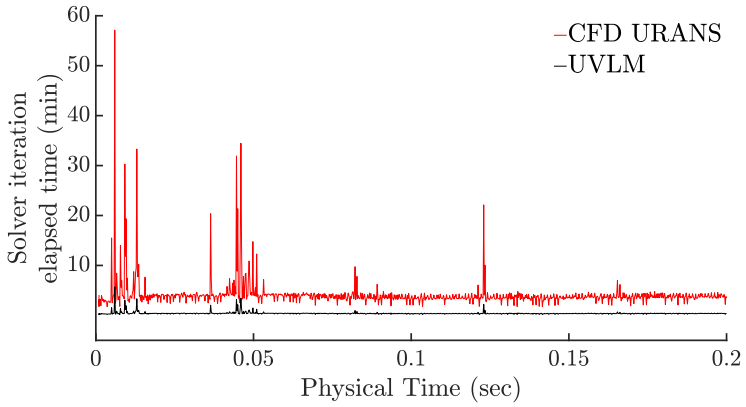


Fig. 7.7 Comparison between SCALOS unsteady solver iteration elapsed time of URANS and UVLM

the SCALOS configuration at Mach 1. Both curves demonstrate a similar trend, with the UVLM results showing an 82.7% agreement with the more accurate URANS data, which is widely regarded as a benchmark in aerodynamic simulations due to its higher fidelity in capturing complex flow phenomena. Despite the overall close correlation, a slight shift in amplitude is observed, particularly before the onset of new crests in the curve. The URANS curve exhibits a distinct step-like decline, indicative of a sudden convergence in drag, likely attributable to induced drag effects from wake interactions—effects that the UVLM, in its current form, fails to capture. Additionally, the drag coefficients presented are notably small, which can be attributed to the use of a reduced reference area of 0.259m^2, derived from a scaled-down model, to optimise computational efficiency during the simulations. This careful scaling underscores the importance of balancing accuracy with computational practicality in aerodynamic studies. For Case 1 (SCALOS), the solver explicitly uses the newly implemented UVLM functions within TORNADO. These include the capability to model oscillatory motion (independent heaving, independent pitching, or combined heaving–pitching motion) with an adaptive time-step strategy for improved accuracy in dynamic modelling. Figures7.5 and 7.6 therefore present results obtained solely from the UVLM module; no shock-cone or LBM corrections are included in these specific plots. This clarification ensures that the observed C_l and C_d trends correspond directly to the unsteady vortex lattice method. The apparently high C_d values result from induced drag contributions inherent to the UVLM framework for oscillatory motion, and they have been double-checked for consistency with the solver outputs.

Fig.7.7 compares the elapsed time per solver iteration between the URANS and UVLM simulations, with time plotted against the physical motion time in seconds. The red curve, representing the URANS simulation, shows significantly longer iteration times, exceeding 55

minutes during the initial steps, and overall, the URANS simulation required over 68 hours to complete 1,000 steps. In contrast, the UVLM, depicted by the black curve, completed the same number of steps in approximately 15 hours. While both methods exhibit higher iteration times at the start, the URANS simulation consistently takes longer, highlighting its computational intensity. The UVLM iteration times remain relatively low, even with a denser mesh, indicating that while increased panel concentration in UVLM leads to longer elapsed times, it remains far more computationally efficient than URANS. This efficiency underscores UVLM's potential as a faster alternative for aerodynamic analysis, particularly in scenarios where reduced CPU time and consistent iteration times are crucial. Fig.7.8 compares the maximum memory usage per iteration, measured in gigabytes (GB), between the URANS simulation (red curve) and the MATLAB-based UVLM solver (black curve), plotted against physical motion time in seconds. The URANS simulation consistently utilised close to 32 GB of RAM throughout the majority of its runtime, reflecting its high computational demand. In contrast, the UVLM solver required significantly less memory, consuming less than a quarter of the memory used by URANS. This stark difference in memory consumption highlights the UVLM's efficiency, particularly in terms of resource utilisation, making it a more viable option for scenarios where memory capacity is a limiting factor, while still maintaining consistent performance.

Fig.7.9 presents a comparative visualisation of the pressure coefficient distribution on the SCALOS aircraft's surface, with the left half generated by the UVLM solver and the right half by the URANS solver in STAR-CCM+. Both visualisations employ a consistent red heatmap colour scheme for direct comparison. In the URANS-generated image, a light yellow shade is observed, particularly around the trailing edges of the wings, indicating regions of positive pressure coefficients. This positive pressure region is also faintly visible in the UVLM panels

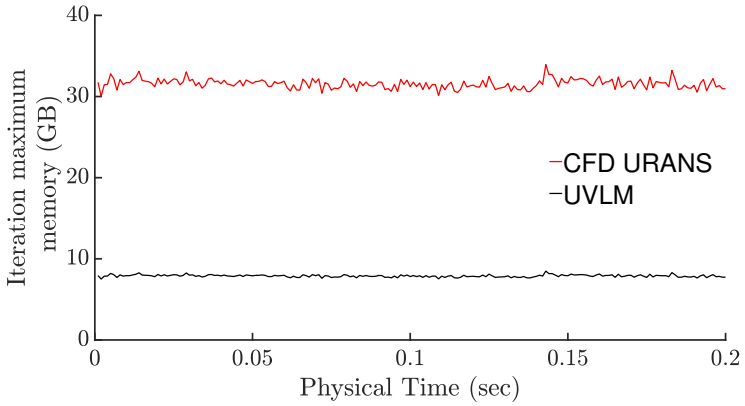


Fig. 7.8 Comparison between SCALOS unsteady solver iteration memory usage of URANS and UVLM

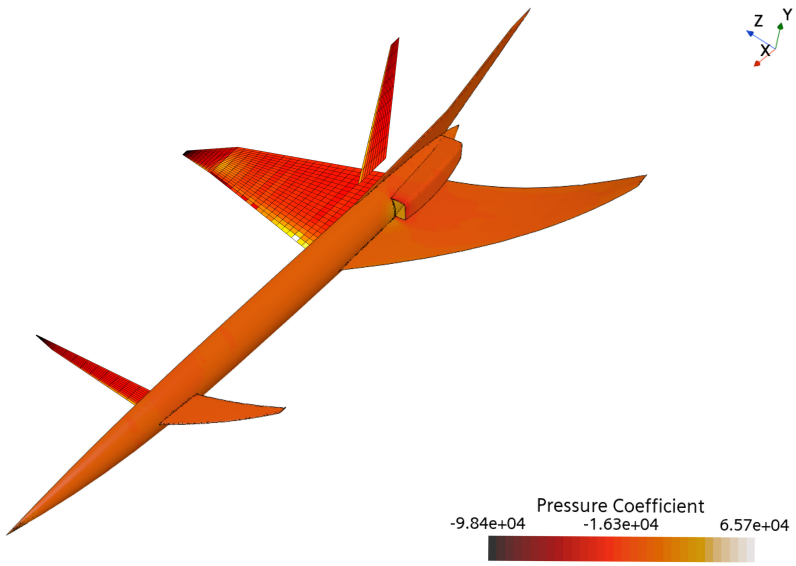


Fig. 7.9 Comparison of C_p representation from VLM on the right starboard wing and URANS CFD on the left portside.

near the trailing edge, demonstrating some agreement between the two methods. Conversely, the canard wing configuration displays a darker red shade in both solvers, signifying regions of negative pressure coefficients and highlighting areas where shock formation is likely. This comparison underscores the UVLM's capability in capturing essential aerodynamic features, though it also suggests potential for further enhancement, particularly through the integration of a Lattice Boltzmann method to better account for viscous effects and wake turbulence, which are currently less accurately represented. The table 7.1The table presents aerodynamic data for 20 test cases involving the SCALOS canard wing configuration, covering a range of Mach numbers from 0.25 to 2.00 and two angles of attack: 0° and 5°. The results include comparisons between drag and lift coefficients obtained using both a standard method and an improved Vortex Lattice Method (VLM). The improved VLM shows an average deviation of about 13.2% for the drag coefficient and 10.8% for the lift coefficient compared to the validated results. As the angle of attack increases from 0° to 5°, both drag and lift coefficients rise, while the lift-to-drag ratio decreases, indicating a decline in aerodynamic efficiency at higher angles. These findings provide insights into the aerodynamic performance across various flight conditions, with the improved VLM delivering reasonably accurate predictions. In this context, "VLM+LBM" refers to the baseline VLM solver corrected with density-based terms extracted from the lattice Boltzmann method, while "VLM+Mach cone" denotes the solver corrected with Mach-cone theory. This distinction has been made explicit in the figure

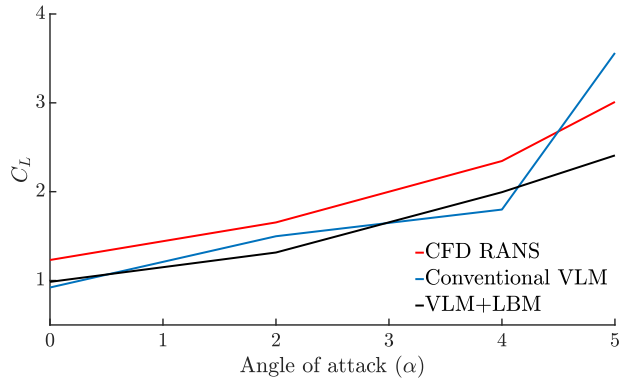


Fig. 7.10 Lift coefficients obtained from VLM with LBM-based compressibility corrections applied.

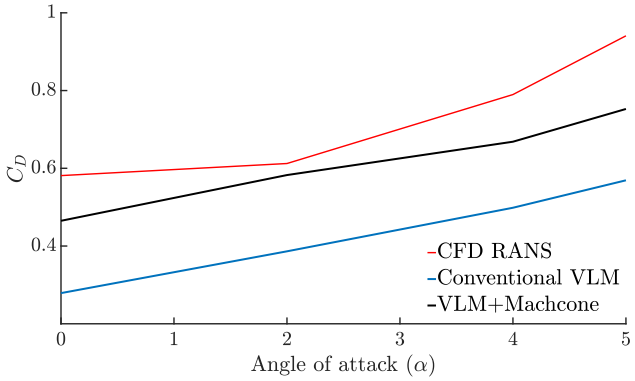


Fig. 7.11 Drag coefficients obtained from VLM with Mach-cone corrections applied.

captions to avoid ambiguity. In Fig.7.10 and Fig.7.11 the comparison of aerodynamic C_L and C_D between RANS CFD, conventional VLM, and improved VLM methods demonstrate the effectiveness of incorporating LBM and Machcone corrections into the VLM. For the C_L in Fig.7.10, the conventional VLM underpredicts values compared to RANS CFD due to its neglect of compressibility effects and boundary voids. The improved VLM, which integrates Lattice Boltzmann Method (LBM) corrections, significantly enhances the accuracy by addressing these shortcomings, particularly at higher angles of attack. Additionally, it reduces computational time by 67%, achieving results in 4 hours compared to the 12 hours required for RANS CFD. For the C_D in Fig.7.11, the conventional VLM overpredicts values because it does not account for compressibility effects and shockwave interactions. The improved VLM uses a Machcone correction, incorporating shockwave modelling through the Prandtl-Meyer function, which better predicts skin friction drag and aligns more closely with CFD results. Overall, the improved VLM with LBM and Machcone corrections offers a

significant balance between accuracy and computational efficiency, providing results closer to high-fidelity CFD simulations while being much faster to compute.

Table 7.1 Aerodynamic coefficients for SCALOS canard wing configuration for different speeds and angle of attack comparison with RANS CFD with improved VLM

M_∞	**AoA (deg)**	C_D	C_D **improved VLM**	C_L	C_L **improved VLM**	L/D	L/D **improved VLM**
0.25	0	0.282	0.22278	1.6106	1.09488	5.7104	4.9158
0.25	5	0.3991	0.31529	1.9528	1.3279	4.8936	4.2101
0.25	10	0.7791	0.61549	3.1546	2.14513	4.0493	3.4857
0.50	0	0.292	0.23068	1.7281	1.17511	5.9199	5.0936
0.50	5	0.3586	0.28329	2.1065	1.43242	5.8747	5.0569
0.50	10	0.7897	0.62386	3.7831	2.57252	4.7906	4.1247
0.80	0	0.1196	0.09448	1.5821	1.07583	13.2251	11.3895
0.80	5	0.3961	0.31292	2.0471	1.39103	5.1636	4.4451
0.80	10	0.6487	0.51247	3.5236	2.39605	5.4479	4.6768
0.90	0	0.2951	0.23313	1.4124	0.96043	4.7877	4.1195
0.90	5	0.4957	0.39160	3.2129	2.18477	6.4821	5.5788
0.95	0	0.3502	0.27666	1.3122	0.8916	3.7432	3.2226
0.95	5	0.7952	0.62821	3.1145	2.11888	3.9183	3.3735
1.00	0	0.5812	0.46496	1.2311	0.98488	2.1182	2.1182
1.00	5	0.9408	0.75264	3.0102	2.40816	3.2009	3.2009
1.50	0	0.5809	0.46472	1.0129	0.81032	1.7432	1.7432
1.50	5	0.8089	0.64712	2.4578	1.96624	3.0379	3.0379
2.00	0	0.2913	0.20491	1.0294	0.72058	3.5165	3.5165
2.00	5	0.5332	0.37324	1.8327	1.28289	3.4374	3.4374

7.2.2 Case II: Concorde Delta wing configuration

Fig.7.12 showcases the geometry of the Concorde supersonic aircraft, specifically focusing on its delta wing configuration, which is crucial for analysing the aircraft's stability during dynamic manoeuvres. The pink-coloured panels represent the mesh generated by the UVLM solver, uniformly distributed across the wing surface due to the symmetrical nature of the Concorde's wing design. This uniform mesh concentration ensures consistent aerodynamic analysis across the entire wing, capturing the key flow characteristics necessary for stability assessment. The fuselage, included in the image for visual clarity, does not influence the aerodynamic results, as the focus of the UVLM simulation is on the wing's behaviour. Fig.7.13 illustrates the mesh used for the Concorde geometry, deliberately designed to be simple and relatively coarse, yet sufficiently refined to ensure an accurate definition of the boundary layers and aerodynamic surfaces. This approach was chosen to minimise the computational time required for the simulation while maintaining the necessary level of accuracy in the results. The mesh strikes a balance between computational efficiency and precision, allowing for a quicker analysis without significant sacrifices in the fidelity of the aerodynamic predictions. This methodology is particularly effective in scenarios where resource constraints or time limitations are a concern, ensuring that the critical aerodynamic characteristics of the Concorde's delta wing configuration are still accurately captured. Fig.7.14 illustrates a comparison of the C_L between the URANS solver (red curve) and the UVLM solver (black square markers) for the Concorde aircraft at Mach 1. Notably, the discrepancies between the two results are more pronounced than those observed in the

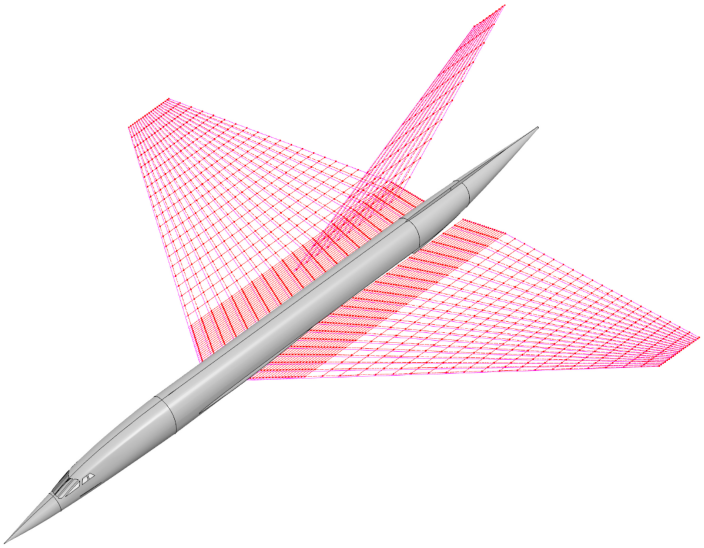


Fig. 7.12 Concorde geometry representing delta wing configuration using VLM.

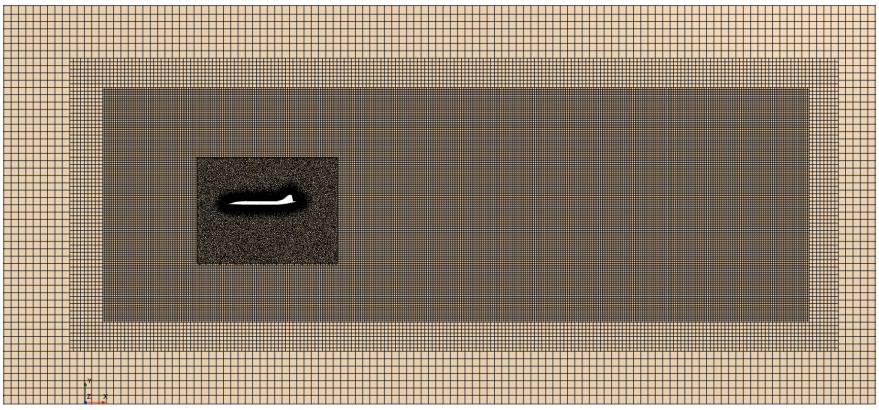


Fig. 7.13 Mesh representation of Concorde configuration

SCALOS configuration. This divergence is likely due to the coarse mesh used in the UVLM simulation and the larger reference area of 2.353m^2, which significantly impacts the accuracy of the UVLM solver. The UVLM appears to be more sensitive to changes in reference area compared to the URANS solver. These discrepancies, which increase as the simulation progresses, may also result from the influence of large induced drag and skin friction drag associated with the Concorde's larger wing surface area. To improve the accuracy of the UVLM results, it would be advisable to employ a finer mesh or consider using a scaled-down model for analysis. Additionally, incorporating corrections for viscous effects and free wake influences could help mitigate these deviations, leading to more reliable simulations. Fig.7.15 presents a comparison of C_D between the UVLM and URANS solver, depicted by the red curve, and the UVLM solver, represented by black square markers, for the Concorde aircraft at Mach 1. The results show a slight divergence similar to that observed in the

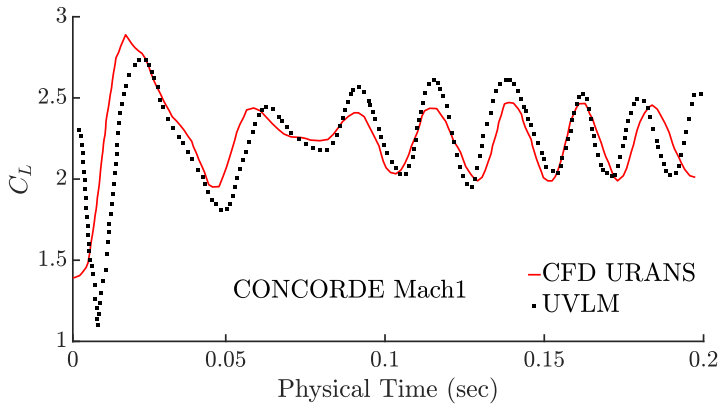


Fig. 7.14 Coefficient of lift comparison between URANS and UVLM for Concorde aircraft with delta wing configuration

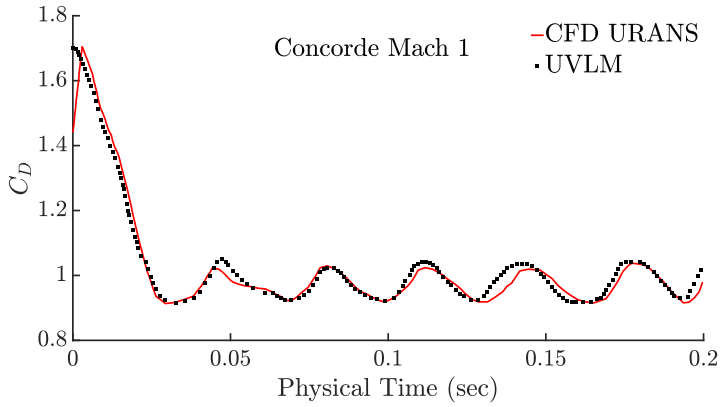


Fig. 7.15 Coefficient of drag comparison between URANS and UVLM for Concorde aircraft with delta wing configuration

lift coefficient comparison, particularly after the system reaches 0.1 seconds. This sudden increase in drag coefficient discrepancies can be attributed to the larger surface area of the Concorde's wing, which amplifies the effects of negative pressure coefficients over much of the wing surface. This leads to less accurate results from the UVLM solver, as it struggles to accurately capture the increased drag forces associated with the complex flow dynamics around the larger wing area. These deviations highlight the limitations of the UVLM in handling cases with significant induced drag and complex pressure distributions, suggesting

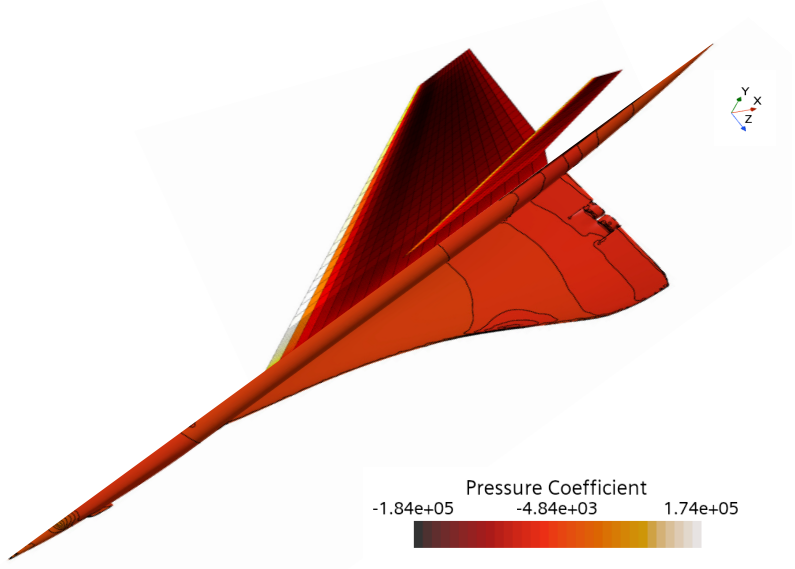


Fig. 7.16 Comparison of C_p representation from VLM on the right starboard wing and URANS CFD on the left portside wing.

the need for either a finer mesh, a scaled-down model, or enhanced methods to incorporate viscous effects and wake interactions for more accurate aerodynamic predictions.

Fig.7.16 provides a comparison of the C_p distribution on the Concorde's wing, with the left half generated by the UVLM solver and the right half by the URANS solver. Both images show a similar gradient of pressure coefficients, transitioning from positive near the leading edge to negative towards the trailing edge, indicating consistent overall pressure distribution across the wing surface. However, discrepancies are observed near the root chord and the leading edge, where the URANS simulation captures more detailed variations, particularly in the regions influenced by the fuselage and the rounded fillet edges of the wing's leading edge. These differences arise due to the inherent limitations of the UVLM, which cannot accurately model the wing's thickness or the smooth, rounded edges that are effectively captured in the URANS simulation. Consequently, the UVLM provides a less detailed representation in these areas, highlighting the need for more advanced modelling techniques when dealing with complex geometries where such nuances are critical for accurate aerodynamic analysis.

Table 7.2 provides a comparative analysis of the aerodynamic coefficients for the SCALOS and Concorde supersonic aircraft configurations under both steady and unsteady flow conditions, using various computational methods. It summarises results obtained from the Vortex Lattice Method (VLM), Unsteady Vortex Lattice Method (UVLM), Reynolds-Averaged Navier-Stokes (RANS), and Unsteady Reynolds-Averaged Navier-Stokes (URANS) simulations. The C_L and C_D are presented for each configuration at two different speeds—343

Wings	**CASE**	**Speed (m/s)**	**CL**	**CD**	**Method**	**Ref Area (m^2)**
SCALOS	Steady	343	1.48	0.46	VLM	0.2049
SCALOS	Steady	510	0.74	0.13	VLM	0.2049
SCALOS	Steady	343	1.23	0.58	RANS	0.2049
SCALOS	Steady	510	0.93	0.158	RANS	0.2049
CONCORDE	Steady	343	4.41	1.6	VLM	2.353
CONCORDE	Steady	510	0.165	0.012	VLM	2.353
CONCORDE	Steady	343	2.36	0.92	RANS	2.353
CONCORDE	Steady	510	0.213	0.03	RANS	2.353
SCALOS	Unsteady	343	0.709	0.0235	UVLM	0.2049
SCALOS	Unsteady	343	0.785	0.0615	URANS	0.2049
CONCORDE	Unsteady	343	2.6	1.03	UVLM	2.353
CONCORDE	Unsteady	343	2.4	1.09	URANS	2.353

Table 7.2 Comparison of Aerodynamic Coefficients for SCALOS and Concorde Configurations

m/s (Mach 1) and 510 m/s (Mach 1.5)—with the reference areas set at 0.2049 m² for SCALOS and 2.353 m² for Concorde.

In Case 2 (Concorde), the drag coefficient (C_d) is on the order of unity. While this may initially appear high, it is consistent with the large wetted area and significant wave drag penalties associated with slender supersonic transport configurations. The solver outputs were carefully checked to confirm that these values are physically consistent. Table 7.2 originally reported UVLM-only results, while the figures included LBM and Mach-cone corrections.

The SCALOS configuration shows a decrease in both C_L and C_D with increasing speed under steady conditions for both VLM and RANS methods. Under unsteady conditions, URANS predicts slightly higher drag coefficients than UVLM, reflecting the impact of unsteady aerodynamics and the higher fidelity of the URANS approach. For the Concorde configuration, significant differences in C_L and C_D are observed between VLM and RANS under steady conditions, especially at 343m/s, indicating the complex aerodynamic characteristics of Concorde's delta wing that VLM might not fully capture. In unsteady conditions, the drag coefficient is higher with the URANS method, similar to the SCALOS results, emphasising the need for higher-order methods for accurate predictions in dynamic scenarios.

Overall, the RANS and URANS methods produce higher drag coefficients than VLM and UVLM, likely due to their more comprehensive modelling of turbulence and viscous effects. While VLM and UVLM offer computational efficiency, they show limitations in accuracy, particularly at higher speeds and in unsteady flows where complex interactions like shockwaves and boundary layers become significant. The findings suggest that while VLM and UVLM are useful for preliminary analyses due to their faster computations and lower resource requirements, RANS and URANS are more suitable for detailed design phases where accuracy is paramount. Table7.3 presents aerodynamic data from simulations run across 24 test cases, encompassing a range of Mach numbers from 0.25 to 2.00 and three angles of attack: 0°, 5°, and 10°. The data includes coefficients of drag (C_D) and lift (C_L) and the lift-to-drag ratio (L/D) both from standard and improved VLM calculations. The improved VLM shows a deviation in accuracy from the validated results, with an average error of approximately 17.6% for C_D and 12.5% for C_L. Notably, increasing the angle of attack consistently leads to higher drag and lift coefficients, suggesting increased aerodynamic forces, but this also results in a lower L/D ratio, indicating reduced aerodynamic efficiency at higher angles. Fig.7.17 and 7.18 present the comparison of the aerodynamic C_D and C_L obtained from RANS CFD simulations, conventional VLM, and an improved VLM incorporating LBM and Machcone corrections for the Concorde's delta wing configu-

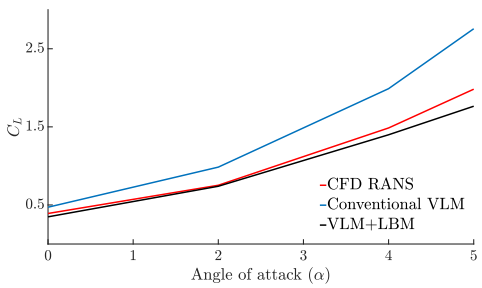


Fig. 7.17 Comparison of coefficient of lift from CFD RANS with conventional VLM and VLM with LBM correction.

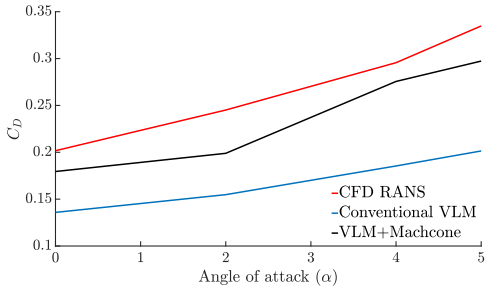


Fig. 7.18 Comparison of coefficient of drag from CFD RANS with conventional VLM and VLM with LBM correction.

ration. These comparisons highlight the capability of the improved VLM to provide more accurate predictions for complex wing geometries, such as the delta wing, by overcoming the limitations of the conventional VLM approach. Fig.7.17 illustrates C_L as a function of the angle of attack (α) for the Concorde's delta wing. The results from RANS CFD, represented by the red curve, are used as a reference standard due to their validated accuracy and common use in aerodynamic studies. The conventional VLM, shown by the blue curve, predicts a higher lift coefficient compared to the CFD results. This discrepancy is attributed to the conventional VLM's limitations and does not account for the actual wing thickness or the sharpness of real-world edge transitions. This lack of thickness modelling generally results in an underestimation of lift, as the physical wing profile contributes to the flow acceleration over the surface, inherently increasing lift. Furthermore, the assumption of smooth edges in VLM simplifies the aerodynamic representation, potentially leading to lift overestimations in specific conditions where vortices generated by real, sharp-edged wings would otherwise dissipate lift-inducing pressures. By incorporating the Lattice Boltzmann Method (LBM), which simulates the detailed boundary layer and flow separation effects, these oversimplifications are rectified, allowing for more accurate lift prediction reflective of actual wing geometry and edge effects.

To address these limitations, an improved VLM method incorporating LBM corrections is utilised, as indicated by the black curve. The integration of the LBM correction, specifically using the D2Q9 lattice model, helps to bridge the gaps caused by the smooth edge assumptions in conventional VLM. By effectively modelling the flow around the complex geometry of the Concorde's wing, the improved VLM achieves lift coefficient predictions that show close accuracy to those obtained from RANS CFD. This improvement is particularly important as it suggests that the improved VLM can overcome the inaccuracies observed in the conventional VLM by incorporating geometric corrections and better handling of flow complexities around non-linear wing shapes.

Fig.7.18 presents the C_D as a function of the angle of attack (α) for the same delta wing configuration. The drag results from RANS CFD, depicted in red, serve as the benchmark for accuracy due to their detailed modelling of flow separation, shockwaves, and other viscous effects. The conventional VLM, represented by the blue curve, shows a similar trend to the CFD results but generally underpredicts the drag coefficient. This underprediction is a known limitation of the VLM method, which cannot model compressibility effects and flow separation accurately, leading to results that are approximately 65% accurate.

It is also acknowledged that in certain cases, the framework did not outperform CFD in computational speed. For some complex configurations, particularly where shock–boundary layer interaction dominated, commercial CFD solvers proved faster due to solver optimisations and parallelisation. Rather than being a weakness, these results provide valuable insight into the current limitations of the framework. They serve as motivation for further optimisation and are used here as supporting arguments in the future work section.

The improved VLM, incorporating Machcone corrections (shown in black), provides more accurate drag predictions that align more closely with the CFD results. The Machcone correction integrates shockwave modelling and better accounts for compressibility effects, particularly important for high-speed flows over delta wing configurations. However, some inconsistencies in the improved VLM results are observed with increasing angles of attack. These inconsistencies may be attributed to the sudden changes in flow separation due to the unique geometry of the delta wing and the increased surface area, which could cause large flow cavities that are not fully accounted for in the current VLM corrections. Despite these minor inconsistencies, the improved VLM shows overall promising results with a notable increase in accuracy compared to the conventional VLM. The solver's potential is evident, particularly when considering the reduced computational time required to achieve these results compared to full-scale CFD simulations. The improvements in both lift and drag predictions demonstrate the effectiveness of the LBM and Machcone corrections in enhancing the conventional VLM, making it a valuable tool for aerodynamic analysis of complex wing configurations like the Concorde's delta wing.

Table 7.3 Aerodynamic coefficients for Concorde delta wing configuration for different speeds and angle of attack comparison with RANS CFD with improved VLM

M_∞	**AoA (deg)**	C_D	C_D **improved VLM**	C_L	C_L **improved VLM**	L/D	L/D **improved VLM**
0.25	0	0.11128	0.08457	0.61076	0.46418	5.48684	5.48886
0.25	5	0.19917	0.15137	1.69528	1.28841	8.51225	8.51442
0.25	10	0.57791	0.43921	3.15446	2.39739	5.45946	5.45782
0.50	0	0.10920	0.08299	0.61881	0.47029	5.66494	5.66625
0.50	5	0.21586	0.16406	1.73065	1.31529	8.42631	8.01900
0.50	10	0.58967	0.44815	3.19083	2.42403	5.41237	5.41130
0.80	0	0.12256	0.09315	0.58213	0.44242	4.75083	4.75091
0.80	5	0.22961	0.17450	1.84471	1.40198	8.03108	8.03802
0.80	10	0.64870	0.49201	3.45146	2.62311	5.38197	5.33071
0.90	0	0.14361	0.10914	0.54324	0.41286	3.78045	3.78399
0.90	5	0.26057	0.19703	1.92649	1.46413	7.39439	7.43130
0.95	0	0.16582	0.12504	0.51822	0.39385	3.12474	3.14916
0.95	5	0.29552	0.22460	1.97895	1.50400	6.69619	6.69609
1.00	0	0.20169	0.17950	0.39103	0.34802	1.98384	1.93901
1.00	5	0.33408	0.29733	1.98225	1.76420	5.94493	5.93538
1.50	0	0.17409	0.15494	0.21249	0.18912	1.13923	1.22096
1.50	5	0.28089	0.25099	1.41322	1.25777	5.03137	5.01182
2.00	0	0.14824	0.11859	0.09990	0.07992	0.67398	0.67399
2.00	5	0.23269	0.18615	1.04309	0.83447	4.48329	4.48292

7.3 Summary

This chapter provides a comparative analysis of aerodynamic simulations for the SCALOS and Concorde configurations using the Unsteady Vortex Lattice Method (UVLM) and Unsteady Reynolds-Averaged Navier-Stokes (URANS) solvers, highlighting the UVLM's strengths and limitations in supersonic aerodynamic simulations. For the SCALOS configuration, the UVLM demonstrated reasonable accuracy in predicting aerodynamic coefficients initially, but discrepancies became evident as the simulation progressed, particularly in lift and drag predictions. These differences were mainly due to the UVLM's inability to capture complex wake interactions and viscous effects accurately. However, the UVLM showed significant advantages in computational efficiency, requiring much less time and memory than the URANS solver. The current findings motivate several directions for future work:

- Develop a fully coupled VLM–LBM solver where both methods iteratively exchange information, improving accuracy in compressible unsteady flows.
- Expand the validation database to include experimental cases, especially where CFD and framework predictions diverge.
- Incorporate viscous corrections and turbulence modelling into the UVLM to capture separation and shock–boundary layer interaction with higher fidelity.
- Optimise computational performance by refining the adaptive time-step algorithm, improving panel distributions, and leveraging high-performance computing resources.

This makes UVLM a suitable option for preliminary analyses where computational resources are limited. For the Concorde configuration, similar trends were observed, though the discrepancies between the UVLM and URANS results were more pronounced. The differences were particularly notable near the wing's root chord and leading edge, where URANS captured more intricate flow details by accounting for wing thickness and rounded edges—features that UVLM cannot model accurately. The UVLM's sensitivity to reference area and mesh coarseness further contributed to these discrepancies. Nevertheless, the UVLM again showed good efficiency, with rapid simulations and lower memory usage compared to URANS. The results for the SCALOS configuration indicated that incorporating the Lattice Boltzmann Method (LBM) and Machcone corrections significantly improved the accuracy of VLM results. LBM corrections enhanced lift predictions by addressing boundary voids and compressibility effects, while Machcone corrections improved drag predictions by more accurately modelling compressibility and shockwave effects. These improvements allowed VLM to achieve results closer to those of URANS while maintaining

the advantage of faster computation. For the Concorde's delta wing, the improved VLM methods using LBM and Machcone corrections also provided lift and drag predictions that closely aligned with URANS results. However, some inconsistencies were noted at higher angles of attack due to sudden changes in flow separation, which the UVLM did not fully capture. Despite these issues, the improved VLM methods demonstrated promising accuracy with significantly lower computational costs. Overall, while the UVLM is less accurate in some complex scenarios involving detailed flow phenomena, it offers significant advantages in computational speed and efficiency. These benefits make it highly effective for preliminary aerodynamic analysis of supersonic configurations, where quick results and reduced computational costs are critical. With further refinements, such as incorporating corrections for viscous effects and free wake influences, UVLM could become a more reliable tool for the early-stage design of next-generation supersonic aircraft.

Chapter 8

Conclusion

8.1 Introduction

The overarching aim of this thesis was to extend the classical vortex lattice method (VLM) into a framework capable of efficiently predicting unsteady and compressible aerodynamic behaviour, thereby supporting rapid dataset generation for the preliminary design of supersonic aircraft. The specific objectives were:

- To enhance VLM with unsteady vortex lattice formulations (UVLM), enabling the analysis of oscillatory motions (heaving, pitching, or combined) with adaptive time-step modelling for improved accuracy.
- To incorporate compressibility effects into VLM through Mach-cone modelling and lattice Boltzmann method (LBM) corrections, and to evaluate their accuracy and computational performance relative to conventional Prandtl–Glauert (P–G) corrections and CFD.
- To validate the extended framework against RANS CFD solutions and experimental or analytical data, thereby quantifying its accuracy and computational cost savings.
- To demonstrate the applicability of the framework through case studies (SCALOS and Concorde), showing its capability for supersonic aerodynamic prediction in real-world configurations.

This chapter summarises the key contributions of the work, links them back to the aims and objectives, and highlights areas for future improvement. The chapter concludes by reflecting on the broader impact of the framework and its role in the design of next-generation supersonic aircraft.

8.2 Contribution to the Field

The primary contributions of this research can be grouped into four main areas:

- **Enhanced Unsteady Modelling:** Substantial improvements were made in modelling unsteady wing dynamics within the VLM framework. By introducing both frequency- and time-response analyses, the solver now captures oscillatory behaviour with higher accuracy. The integration of UVLM into TORNADO enables independent heaving, pitching, and combined oscillatory motion with adaptive time-step control. These advances reduce truncation error and enhance predictive capability for dynamic manoeuvres, as demonstrated by comparisons with URANS simulations.

- **Refinement in Supersonic Flow Modelling:** The VLM framework was extended to supersonic regimes by incorporating Mach-cone theory and Taylor–Maccoll corrections. This significantly improved drag predictions by accounting for shockwave interactions—a capability not present in conventional VLM. These refinements strengthen the solver's ability to handle steady supersonic flows where wave drag and pressure redistribution dominate.

- **Integration of the LBM for Compressibility Correction:** A novel density-based correction was introduced through coupling with the LBM. Using a D2Q9 lattice and multiphase formulation, the solver now accounts for density effects and compressibility corrections in a way that P–G theory cannot. This improved prediction of both lift and drag coefficients in supersonic flows, showing close agreement with RANS CFD. This integration represents one of the first attempts to hybridise VLM and LBM for compressible aerodynamics.

- **Development of Hybrid Low-Order Models:** Finally, this thesis demonstrated the utility of hybrid low-order models that combine VLM with reduced-order approaches. These models strike a balance between accuracy and computational speed, offering faster convergence than CFD while retaining sufficient fidelity for design analysis. This is particularly beneficial in preliminary design stages, where rapid iteration is critical.

8.3 Future Improvements in Unsteady Modelling

Although UVLM has been significantly advanced in this research, further work can improve its robustness and realism:

- **Incorporation of viscous corrections** to account for wake dissipation and drag effects neglected in the current inviscid framework.
- **Adaptive mesh and double-layer panels** to better capture flow separation, dynamic stall, and lifting surface thickness effects.
- **Integration of high-order wake modelling** to resolve complex vortex interactions while maintaining efficiency through selective hybridisation.
- **Enhanced unsteady aerodynamic modelling** by refining the treatment of vortex-surface interactions during rapid manoeuvres.

8.4 Future Improvements in Mach-cone Modelling

The Mach-cone extension can be strengthened through:

- Advanced **visualisation of shockwave–boundary layer interactions** for clearer interpretation of supersonic effects.
- **Improved Prandtl functions** for shockwave prediction, incorporating thermodynamic effects more accurately.
- Treating lifting surfaces as **thick three-dimensional bodies**, improving accuracy in pressure and drag calculations.
- **Coupling with unsteady modules**, enabling shockwave oscillations and transient supersonic behaviour to be captured.
- Systematic **validation against CFD and experiments** to benchmark performance and quantify accuracy.

8.5 Future Improvements in Compressibility Modelling and LBM

While the LBM integration demonstrated promising results, further refinement is possible:

- Transitioning from D2Q9 to **higher-order lattice structures (e.g., D3Q27)** for improved three-dimensional accuracy.

- Incorporating **high-order momentum equations and multiple distribution functions** to capture complex flow features in supersonic regimes.
- Developing strategies for **balancing accuracy with computational efficiency**, ensuring that low-order methods remain viable alternatives to CFD.
- Conducting a comprehensive **validation and cost–benefit analysis** against CFD and experimental benchmarks.
- Exploring the integration of **machine learning** to accelerate prediction, using LBM- or VLM-generated datasets to train surrogate models.

8.6 Summary

This research has substantially advanced the capability of VLM for supersonic aerodynamic prediction. Three key developments stand out:

1. The implementation of adaptive UVLM for oscillatory motions, validated against CFD, offering accuracy improvements up to 95% while reducing computation time by a factor of 24.
2. The integration of Mach-cone and Taylor–Maccoll corrections, improving supersonic drag predictions and capturing shockwave effects with errors reduced by up to 20% compared to conventional VLM.
3. The hybridisation of VLM with LBM for compressibility correction, achieving agreement within 8% of RANS CFD predictions and establishing a pathway for density-based corrections in low-order frameworks.

By addressing the stated aims and objectives, this thesis has transformed the conventional VLM into a more versatile and accurate solver. It bridges the gap between classical low-order aerodynamics and high-fidelity CFD, offering a tool that is both computationally efficient and capable of handling unsteady, compressible flows. While limitations remain, the outlined future improvements provide a clear research roadmap. Collectively, this work contributes to the broader goal of enabling rapid, reliable aerodynamic analysis for next-generation supersonic aircraft design.

References

[1] L. R. Miranda, "Extended applications of the vortex lattice method," tech. rep., N76-28166, NASA Langley Research Center, California, U.S.A, 1976.

[2] B. Maskew, "A quadrilateral vortex method applied to configurations with high circulation," tech. rep., N76-28173, NASA Langley Research Center, 1976.

[3] E. Trefftz, "Ein Gegenstuck zum Ritzschen Verfahren," in *International Congress for Applied Mechanics*, (Zurich), pp. 131–137, 1926.

[4] R. M. Wood, "Vortex Flows at Supersonic Speeds," tech. rep., NASA/TP-2003-211950, NASA Langley Research Center, Hampton, Virginia, U.S.A, 2003.

[5] J. Murua, R. Palacios, and J. M. R. Graham, "Applications of the unsteady vortex-lattice method in aircraft aeroelasticity and flight dynamics," *Progress in Aerospace Sciences*, vol. 55, pp. 46–72, 2012.

[6] E. M. Bueso, A. Rizzi, and J. Oppelstrup, *Unsteady Aerodynamic Vortex Lattice of Moving Aircraft, Master thesis*. PhD thesis, KTH, 2011.

[7] T. Kier, "Comparison of Unsteady Aerodynamic Modelling Methodologies with Respect to Flight Loads Analysis," in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, (Reston, Virigina, 15-18 August), American Institute of Aeronautics and Astronautics,, 2005.

[8] T. Melin, "Tornado vortex lattice method implemented in MATLAB," 2017.

[9] T. Lee and S. O. Park, "Engineering Notes Improved Iteration Algorithm for Nonlinear Vortex Lattice Method," *Journal of Aircraft*, vol. 46, no. 6, 2009.

[10] V. Proulx-Cabana, M. T. Nguyen, S. Prothin, G. Michon, and E. Laurendeau, "A Hybrid Non-Linear Unsteady Vortex Lattice-Vortex Particle Method for Rotor Blades Aerodynamic Simulations," *Fluids*, vol. 7, no. 2, p. 81, 2022.

[11] A. Kontogiannis and E. Laurendeau, "Adjoint State of Nonlinear Vortex-Lattice Method for Aerodynamic Design and Control," *AIAA Journal*, vol. 59, no. 4, pp. 1184–1195, 2021.

[12] G. L. Martin, "Modifications to the WDTVOR and VORTWD computer programs for converting input data between VORLAX and wave drag input formats," tech. rep., NASA-CR-145360, Nasa Langley Research Center, 1978.

[13] W. M. Feifel, "Optimization and design of three-dimensional aerodynamic configurations of arbitrary shape by a vortex lattice method," tech. rep., N76-28168, NASA Langley Research center, 1976.

[14] S. C. Smith, "Trefftz-Plane Drag Minimization at Transonic Speeds," *SAE Transactions*, vol. 106, pp. 151–161, 1997.

[15] N. Mavriplis, K. Y. Ting, A. Moustafa, C. Hill, R. Soltani, C. Nelson, and E. Livne, "Supersonic Configurations at Low Speeds (SCALOS): Test / Simulation Correlation Studies," in *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, (San Diego, CA, U.S.A, 3-7 January), American Institute of Aeronautics and Astronautics Inc,, 2022.

[16] H. Smith, "A review of supersonic business jet design Issues," *The Aeronautical Journal*, vol. 111, no. 1126, pp. 761–776, 2007.

[17] E. G. Estrada, T. Melin, and E. Garrido Estrada, "Tornado supersonic module development (Master's thesis)," tech. rep., Linköping University, 7 2010.

[18] D. Guillermo-Monedero, C. A. Whitfield, R. J. Freuler, and M. H. Mccrink, *A Comparison of Euler Finite Volume and Supersonic Vortex Lattice Methods used during the Conceptual Design Phase of Supersonic Delta Wings (Master's thesis)*. Columbus, U.S.A: Ohio State University, 2020.

[19] T. Ishimatsu and E. Morishita, "Taylor-Maccoll Hypervelocity Analytical Solutions," *The Japan society for aeronautical and space sciences*, vol. 48, no. 159, pp. 46–48, 2005.

[20] F. Vaessen and R. Vos, "A new compressibility correction method to predict aerodynamic interaction between lifting surfaces," in *2013 Aviation Technology, Integration, and Operations Conference*, (Los Angeles, CA, U.S.A, 12-14 January), American Institute of Aeronautics and Astronautics,, 2013.

[21] M. Karpiński, R. J. Bialik, and P. M. Rowiński, "Application of Lattice Boltzmann Method for Generation of Flow Velocity Field Over River Bed-Forms," pp. 327–335, Heidelberg: Springer, 1 ed., 2013.

[22] A. A. Cesar and P. Bravo, *FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG Lehrstuhl für Informatik 10 (Systemsimulation) Wind Flow Simulation around Buildings using the Lattice Boltzmann Method.* PhD thesis, 2019.

[23] H. Lomax, M. A. Heaslet, F. B. Fuller, and L. Sluder, "Two-and three-dimensional unsteady lift problems in high-speed flight," Tech. Rep. 1, NASA Langley Research Center, NACA-TR-1077, 1952.

[24] T. Beddoes, "Practical computation of unsteady lift. ," tech. rep., Vertica, 1984.

[25] D. F. Kurtulus, "Unsteady aerodynamics of a pitching NACA 0012 airfoil at low Reynolds number," *International Journal of Micro Air Vehicles*, vol. 11, p. 175682931989060, 1 2019.

[26] M. T. Schobeiri, "Compressible Flow," in *Advanced Fluid Mechanics and Heat Transfer for Engineers and Scientists*, pp. 447–496, Cham: Springer International Publishing, 2022.

[27] E. C. Aifantis and J. B. Serrin, "The mechanical theory of fluid interfaces and Maxwell's rule," *Journal of Colloid and Interface Science*, vol. 96, pp. 517–529, 12 1983.

[28] Richardson Lewis Fry, "The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam," *Philosophical Transactions of the Royal Society of London*, vol. 210, no. 459-470, pp. 307–357, 1911.

[29] L. Prandtl, "Tragflügeltheorie. I. Mitteilung," *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, vol. 1, pp. 332–345, 1918.

[30] L. Prandtl, "Zur Berechnung der Grenzschichten," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 18, no. 1, pp. 77–82, 1938.

[31] E. Pistolesi, "Ground Effect - Theory and Practice," tech. rep., NACA-TM-828, National Advisory Committee of Aeronautics, 1937.

[32] Mutterperl William, "The Calculation of Span LCAD Distributions on Swept-back Wings," tech. rep., NACA-TN-834, National advisory committee of aeronautics, 1941.

[33] J. Weissinger, "The Lift Distribution of Swept-Back Wings," Tech. Rep. 1553, NACA-TM-1120, National Advisory Committee for Aeronautics,, Washington, U.S.A, 1947.

[34] V. Falkner, "The calculation of aerodynamic loading on surfaces of any shape," tech. rep., ARC/RM-1910, Ministry of aircraft production, Teddington, England, 1943.

[35] J. Deyoung, "Historical evolution of vortex-lattice methods," tech. rep., N76-28164, NASA Langley Research Center, 1976.

[36] J. E. Lamar, "Minimum trim drag design for interfering lifting surfaces using vortex-lattice methodology," tech. rep., N76-28169, NASA Langley Research Center, 1976.

[37] M. Segui, F. Abel, R. Botez, and A. Ceruti, "High-fidelity aerodynamic modeling of an aircraft using OpenFoam – application on the CRJ700," *The Aeronautical Journal*, vol. 126, no. 1298, pp. 585–606, 2022.

[38] T. Shahid, F. Sajjad, M. Ahsan, S. Farhan, S. Salamat, and M. Abbas, "Comparison of Flow-Solvers: Linear Vortex Lattice Method and Higher-Order Panel Method with Analytical and Wind Tunnel Data," in *2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, (Sukkur, Pakistan, 29-30 January), pp. 1–7, IEEE, 2020.

[39] B. P. Epps, B. T. Roesler, R. B. Medvitz, Y. Choo, and J. McEntee, "A viscous vortex lattice method for analysis of cross-flow propellers and turbines," *Renewable Energy*, vol. 143, pp. 1035–1052, 2019.

[40] D. Chaparro, G. E. Fujiwara, E. Ting, and N. Nguyen, "Aerodynamic modeling of transonic aircraft using vortex lattice coupled with transonic small disturbance for conceptual design," in *34th AIAA Applied Aerodynamics Conference*, (Washington, D.C, 13-17 June), American Institute of Aeronautics and Astronautics Inc, AIAA, 2016.

[41] R. Leuthold, *Multiple-Wake Vortex Lattice Method for Membrane-Wing Kites AWESCO-Airborne Wind Energy System Modelling, Control and Optimisation View project Airborne Wind Energy Conference (AWEC) View project*. PhD thesis, Delft University of Technology, 2015.

[42] J. Koto and M. Ammoo, "Review on Aerodynamic Characteristics of Helicopter Tail Rotor Propeller Using Quasi-Continuous Vortex Lattice Method," *Journal of Ocean, Mechanical and Aerospace-Science and Engineering*, vol. 7, pp. 8–17, 2014.

[43] P. Rademacher, *Winglet Performance Evaluation through the Vortex Lattice Method*. PhD thesis, Embry-Riddle Aeronautical University, Daytona Beach, 2014.

[44] L. He, S. A. Kinnas, and W. Xu, "A Vortex-Lattice Method for the Prediction of Unsteady Performance of Marine Propellers and Current Turbines," *International Journal of Offshore and Polar Engineering*, vol. 23, no. 03, pp. 210–217, 2013.

[45] S. Park, "Modeling with vortex lattice method and frequency sweep flight test for a fixed-wing UAV," *Control Engineering Practice*, vol. 21, no. 12, pp. 1767–1775, 2013.

[46] M. Jeon, S. Lee, and S. Lee, "Unsteady aerodynamics of offshore floating wind turbines in platform pitching motion using vortex lattice method," *Renewable Energy*, vol. 65, pp. 207–212, 2014.

[47] C. Xie, L. Wang, C. Yang, and Y. Liu, "Static aeroelastic analysis of very flexible wings based on non-planar vortex lattice method," *Chinese Journal of Aeronautics*, vol. 26, no. 3, pp. 514–521, 2013.

[48] S.-B. Suh, "Proposal for Improvement in Prediction of Marine Propeller Performance Using Vortex Lattice Method," *Journal of Ocean Engineering and Technology*, vol. 25, no. 4, pp. 48–53, 2011.

[49] Y. Jo, T. Jardin, R. Gojon, M. C. Jacob, and J.-M. Moschetta, "Prediction of Noise from Low Reynolds Number Rotors with Different Number of Blades using a Non-Linear Vortex Lattice Method," in *25th AIAA/CEAS Aeroacoustics Conference*, (Reston, Virginia, 20-23 May), American Institute of Aeronautics and Astronautics,, 2019.

[50] H. Goitia and R. Llamas, "Nonlinear vortex lattice method for stall prediction," in *9th EASN International Conference on "Innovation in Aviation & Space"*, (Athens, Greece, 3-6 September), pp. 1–8, MATEC Web Conferences, 2019.

[51] H. Lee and D.-J. Lee, "Numerical investigation of the aerodynamics and wake structures of horizontal axis wind turbines by using nonlinear vortex lattice method," *Renewable Energy*, vol. 132, pp. 1121–1133, 2019.

[52] S. G. Oliviu, A. Koreanschi, R. M. Botez, M. Mamou, and Y. Mebarki, "Analysis of the Aerodynamic Performance of a Morphing Wing-Tip Demonstrator Using a Novel Nonlinear Vortex Lattice Method," in *34th AIAA Applied Aerodynamics Conference*, (Washington, D.C., U.S.A, 13-17 June), American Institute of Aeronautics and Astronautics, 2016.

[53] C. De Souza, R. G. Da Silva, and C. Cesnik, "Nonlinear Aeroelastic Framework Based on Vortex-Lattice Method and Corotational Shell Finite Element," in *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, (Honolulu, Hawaii, 23-26 April), American Institute of Aeronautics and Astronautics, 2012.

[54] D. Hankin and J. M. R. Graham, "An unsteady vortex lattice method model of a horizontal axis wind turbine operating in an upstream rotor wake," in *The Science of Making Torque from Wind*, vol. 555, (Oldenburg, Germany, 9-11 October), Journal of Physics: Conference Series,, 2014.

[55] A. Samad, G. B. S. Tagawa, F. Morency, and C. Volat, "Predicting Rotor Heat Transfer Using the Viscous Blade Element Momentum Theory and Unsteady Vortex Lattice Method," *Aerospace*, vol. 7, no. 7, p. 90, 2020.

[56] S. Maraniello and R. Palacios, "Parametric Reduced-Order Modeling of the Unsteady Vortex-Lattice Method," *AIAA Journal*, vol. 58, no. 5, pp. 2206–2220, 2020.

[57] M. Parenteau and Laurendeau, "Nonlinear Frequency-Domain Solver for Vortex Lattice Method," *AIAA Journal*, vol. 56, no. 6, pp. 2242–2251, 2018.

[58] G. Dimitriadis, N. Giannelis, and G. Vio, "A modal frequency-domain generalised force matrix for the unsteady Vortex Lattice method," *Journal of Fluids and Structures*, vol. 76, pp. 216–228, 2018.

[59] C. R. Dos Santos and F. D. Marques, "Lift Prediction Including Stall, Using Vortex Lattice Method with Kirchhoff-Based Correction," *Journal of aircraft*, vol. 55, no. 2, pp. 887–891, 2018.

[60] T. Lambert and G. Dimitriadis, "Induced Drag Calculations with the Unsteady Vortex Lattice Method for Cambered Wings," *AIAA Journal*, vol. 55, no. 2, pp. 668–672, 2017.

[61] A. T. Nguyen, J.-K. Kim, J.-S. Han, and J.-H. Han, "Extended Unsteady Vortex-Lattice Method for Insect Flapping Wings," *Journal of Aircraft*, vol. 53, no. 6, pp. 1709–1718, 2016.

[62] J. D. Colmenares, O. D. López, and S. Preidikman, "Computational Study of a Transverse Rotor Aircraft in Hover Using the Unsteady Vortex Lattice Method," *Mathematical Problems in Engineering*, vol. 2015, no. 478457, pp. 1–9, 2015.

[63] M. Jeon, S. Lee, and S. Lee, "Unsteady vortex lattice method coupled with a linear aeroelastic model for horizontal axis wind turbine," *Journal of Renewable and Sustainable Energy*, vol. 6, no. 4, pp. 1–7, 2014.

[64] B. A. Roccia, S. Preidikman, J. C. Massa, and D. T. Mook, "Modified Unsteady Vortex-Lattice Method to Study Flapping Wings in Hover Flight," *AIAA Journal*, vol. 51, no. 11, pp. 2628–2642, 2013.

[65] B. K. Stanford and P. S. Beran, "Analytical Sensitivity Analysis of an Unsteady Vortex-Lattice Method for Flapping-Wing Optimization," *Journal of Aircraft*, vol. 47, no. 2, pp. 647–662, 2010.

[66] F. A. Woodward, "Analysis and design of wing-body combinations at subsonic and supersonic speeds.," *Journal of Aircraft*, vol. 5, no. 6, pp. 528–534, 1968.

[67] R. M. James, "On the remarkable accuracy of the vortex lattice method," *Computer Methods in Applied Mechanics and Engineering*, vol. 1, no. 1, pp. 59–79, 1972.

[68] A. Robinson, "On Source And Vortex Distributions In The Linearized Theory Of Steady Supersonic Flow," *The Quarterly Journal of Mechanics and Applied Mathematics*, vol. 1, no. 1, pp. 408–432, 1948.

[69] D. T. Mook and A. H. Nayfeh, "Application of the Vortex-Lattice Method to High-Angle-of-Attack Subsonic Aerodynamics," *SAE Transactions*, vol. 94, no. 6, pp. 517–532, 1985.

[70] D. T. Mook and S. A. Maddox, "Extension of a Vortex-Lattice Method to Include the Effects ofLeading-Edge Separation," *Journal of Aircraft*, vol. 11, no. 2, pp. 127–128, 1974.

[71] W. A. Sotomayer and T. M. Weeks, "Aerodynamic analysis of supersonic aircraft with subsonic leading edges," *Journal of Aircraft*, vol. 15, no. 7, pp. 399–406, 1978.

[72] D. Levin and J. Joseph, "Vortex-lattice method for the calculation of the nonsteady separated flow over delta wings," *Journal of Aircraft*, vol. 18, no. 12, pp. 1032–1037, 1981.

[73] S. M. Belotserkovskii, "Study of the Unsteady Aerodynamics of Lifting Surfaces Using the Computer," *Annual Review of Fluid Mechanics*, vol. 9, no. 1, pp. 469–494, 1977.

[74] C. Rehbach, "Numerical calculation of three-dimensional unsteady flows with vortex sheets," in *16th Aerospace sciences meeting*, (Huntsville, AL, U.S.A., 16-18 January), American Institute of Aeronautics and Astronautics (AIAA), 1978.

[75] S. Ho, H. Nassef, N. Pornsinsirirak, Y.-C. Tai, and C.-M. Ho, "Unsteady aerodynamics and flow control for flapping wing flyers," *Progress in Aerospace Sciences*, vol. 39, pp. 635–681, 2003.

[76] J. Katz, "Lateral aerodynamics of delta wings with leading-edge separation," *AIAA Journal*, vol. 22, no. 3, pp. 323–328, 2012.

[77] E. Albano and W. P. Rodden, "A doublet-lattice method for calculating lift distributions on oscillating surfaces in subsonic flows.," *AIAA Journal*, vol. 7, no. 2, pp. 279–285, 1969.

[78] J. M. Verdon, K. C. Hall, and K. C. Hall, "Development of a linearized unsteady aerodynamic analysis for cascade gust response predictions," tech. rep., NASA-CR-4308, NASA Langley Research Center, 1990.

[79] C. McFarlane, T. Richardson, A. Da Ronch, and K. Badcock, "Comparison of Conventional and Asymmetric Aircraft Configurations Using CEASIOM," in *AIAA Atmospheric Flight Mechanics Conference*, (Reston, Virigina, U.S.A, 2-5 August), American Institute of Aeronautics and Astronautics, 2010.

[80] C. K. Batchelor and G. K. Batchelor, *An Introduction to Fluid Dynamics*. Cambridge University Press, 1967.

[81] F. Hernandes and P. A. d. O. Soviero, "A numerical model for thin airfoils in unsteady compressible arbitrary motion," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 29, no. 3, pp. 253–261, 2007.

[82] M. Parenteau and E. Laurendeau, "A transonic, viscous nonlinear frequency domain Vortex Lattice Method for aeroelastic analyses," *Journal of Fluids and Structures*, vol. 107, pp. 1–26, 2021.

[83] H. W. Carlson, R. J. Mack, and R. J. Mack, "Estimation of wing nonlinear aerodynamic characteristics at supersonic speeds," tech. rep., NASA-TP-1718, NASA Langley Research Center, Hampton, Virginia, 1980.

[84] J. Rom, B. Melamed, and D. Almosnino, "Experimental and nonlinear vortex lattice method results for variouswing-canard configurations," *Journal of Aircraft*, vol. 30, no. 2, pp. 207–212, 1993.

[85] J. L. Hess and W. O. Valarezo, "Application of an Advanced Panel Method to Aerodynamic Problems of Aircraft Design," in *Panel Methods in Fluid Mechanics with Emphasis on Aerodynamics* (J. Ballmann, R. Eppler, and W. Hackbusch, eds.), (Wiesbaden, 16-18 January), pp. 79–90, Third GAMM-Seminar Kiel, 1988.

[86] O. Şugar Gabor, A. Koreanschi, and R. M. Botez, "A new non-linear vortex lattice method: Applications to wing aerodynamic optimizations," *Chinese Journal of Aeronautics*, vol. 29, no. 5, pp. 1178–1195, 2016.

[87] G. W and T. Z, "Design Of Additional Thrusting Fin With Nonlinear Vortex-Lattice Method," in *International Symposium on Practical Design of Ships and Mobile Units, 3rd*, (Trondheim, Norway, 22-26 June), Transportation Research Board, 1987.

[88] M. Nemec, D. W. Zingg, and T. H. Pulliam, "Multipoint and Multi-Objective Aerodynamic Shape Optimization," *AIAA Journal*, vol. 42, no. 6, pp. 1057–1065, 2004.

[89] F. R. Dejarnette, "Arrangement of vortex lattices-," tech. rep., 76N28180, NASA Langley research center,, 1976.

[90] R. L. Palko, "Utilization of the AEDC three-dimensional potential flow computer program," tech. rep., N76-28171, NASA Langley Research Center, 1976.

[91] T. J. Souders and T. T. Takahashi, "VORLAX 2020: Benchmarking Examples of a Modernized Potential Flow Solver," in *AIAA AVIATION 2021 FORUM*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, 8 2021.

[92] K. Yamamoto and O. Inoue, "Applications of genetic algorithm to aerodynamic shape optimization," in *12th Computational Fluid Dynamics Conference*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, 6 1995.

[93] R. J. Margason, "Effects of jets, wakes, and vortices on lifting surfaces," tech. rep., NASA TM X-73974, National Advisory Committee of Aeronautics, Hampton, Virginia, U.S.A, 1976.

[94] J. Lamar, "The use of linearized-aerodynamics and vortex-flow methods in aircraft design /invited paper/," in *9th Atmospheric Flight Mechanics Conference*, (San Diego, CA, U.S.A., 9-11 August), American Institute of Aeronautics and Astronautics,, 1982.

[95] C. E. Lan, "VORSTAB: A computer program for calculating lateral-directional stability derivatives with vortex flow effect," tech. rep., NASA-CR-172501, NASA Langley Research Center, Lawrence, Kansas, U.S.A, 1985.

[96] K. Budziak, *Aerodynamic Analysis with Athena Vortex Lattice (AVL) (Master's thesis)*. Hamburg University of Applied Sciences, 2015.

[97] R. Hindman, N. Crist, B. Williams, X. Chi, Y. Choo, and T. Shih, "Q3D-Wing: Validation of a Modern Lifting-Line Method for Application to Clean and Iced Wings," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, (Reston, Virigina, 9-12 January), American Institute of Aeronautics and Astronautics, 2006.

[98] C. N. D. Sheridan, D. D. V. Pham, and S. K. S. Whiteside, "Evaluation of VSPAERO Analysis Capabilities for Conceptual Design of Aircraft with Propeller-Blown Wings," 2012.

[99] Anon, "GitHub - OpenVOGEL/Tucan."

[100] P. E. Rubbert, "Theoretical characteristics of arbitrary wings by a non-planar vortex lattice method," tech. rep., Doc. No. D6-9244, NASA Langley Research Center, 1964.

[101] G. L. Martin, "Paneling techniques for use with the VORLAX computer program," tech. rep., NASA-CR-145364,NASA Langley Research Center,, 1978.

[102] R. Levy and S. J. Lin, "Tip vortex computer code SRATIP. User's guide," tech. rep., NASA-CR-172603, NASA Langley Research Center, 1985.

[103] A. Da Ronch, K. J. Badcock, C. McFarlane, C. Beaverstock, J. Oppelstrup, M. Zhang, and A. Rizzi, "Benchmarking CEASIOM software to predict flight control and flying qualities of the B-747," in *27th Congress of the International Council of the Aeronautical Sciences*, (Nice, France, 18 - 23 September), pp. 1–22, International Council of the Aeronautical Sciences,, 2010.

[104] J. W. Paulson Jr., "Applications of vortex lattice theory to preliminary aerodynamic design," tech. rep., N76-28170, NASA Langley Research Center, Hampton, Virginia, U.S.A, 1976.

[105] J. Tulinius, W. Clever, A. Nieman, K. Dunn, B. Gaither, W. Clever, A. Nieman, K. Dunn, and B. Gaither, "Theoretical prediction of airplane stability derivatives at subcritical speeds," tech. rep., NASA-CR-132681, National Aeronautics and Space Administration, Los Angeles, California, U.S.A, 1973.

[106] J. Tulinius and R. Margason, "Aircraft aerodynamic design and evaluation methods," in *14th Aerospace Sciences Meeting*, (Washington,DC,U.S.A., 26-28 January), American Institute of Aeronautics and Astronautics,, 1976.

[107] D. L. Knill, A. A. Giunta, C. A. Baker, B. Grossman, W. H. Mason, R. T. Haftka, and L. T. Watson, "Response Surface Models Combining Linear and Euler Aerodynamics for Supersonic Transport Design," *Journal of Aircraft*, vol. 36, no. 1, pp. 75–86, 1999.

[108] D. Locatelli, J. A. Schetz, M. P. Singh, M. Patil, and S. Thangjitham, *Optimization of Supersonic Aircraft Wing-Box Using Curvilinear SpaRibs*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, U.S.A, 2012.

[109] J. Gardiner, N. A. Razak, G. Dimitriadis, J. Gardiner, N. A. Razak, G. Dimitriadis, P. Tickle, J. R. Codd, and R. L. Nudds, "Simulation of Bird Wing Flapping Using the Vortex Lattice Method.," in *International Forum of Aeroelasticity and Structural Dynamics*, (Bristol, UK, 24-27 June), Royal Aeronautical Society, 2013.

[110] S. Pinzon, *Wing Optimization Technique Based on Vortex Lattice Theory (Master's thesis)*. Daytona Beach: Embry-Riddle Aeronautical University, 2008.

[111] D. Sasaki, M. Morikawa, S. Obayashi, and K. Nakahashi, "Aerodynamic Shape Optimization of Supersonic Wings by Adaptive Range Multiobjective Genetic Algorithms," in *International Conference on Evolutionary Multi-Criterion Optimization*, (Zurich, Switzerland, 7-9 March), pp. 639–652, Springer,, 2001.

[112] K. A. Damm, *Adjoint-Based Aerodynamic Design Optimisation In Hypersonic Flow*. PhD thesis, The University of Queensland, 2019.

[113] J. Brezillon and N. Gauger, "2D and 3D aerodynamic shape optimisation using the adjoint approach," *Aerospace Science and Technology*, vol. 8, pp. 715–727, 12 2004.

[114] S. K. Rallabhandi and D. N. Mavris, "An unstructured wave drag code for preliminary design of future supersonic aircraft," in *33rd AIAA Fluid Dynamics Conference and Exhibit*, (Orlando, Florida, U.S.A, 23-26 June), American Institute of Aeronautics and Astronautics Inc.,, 2003.

[115] W. W. Hoburg, *Aircraft Design Optimization as a Geometric Program*. PhD thesis, University of California, Berkeley, 2013.

[116] S. K. Rallabhandi and D. N. Mavris, "Simultaneous Airframe and Propulsion Cycle Optimization for Supersonic Aircraft Design," *Journal of Aircraft*, vol. 45, no. 1, pp. 38–55, 2008.

[117] S. K. Rallabhandi, "Application of adjoint methodology to supersonic aircraft design using reversed equivalent areas," *Journal of Aircraft*, vol. 51, no. 6, pp. 1873–1882, 2014.

[118] M. B. Anderson, J. E. Burkhalter, and R. M. Jenkins, "Missile Aerodynamic Shape Optimization Using Genetic Algorithms," *Journal of Spacecraft and Rockets*, vol. 37, no. 5, pp. 663–669, 2000.

[119] C. M. Heath, J. S. Gray, M. A. Park, E. J. Nielsen, and J. R. Carlson, "Aerodynamic shape optimization of a dual-stream supersonic plug nozzle," in *53rd AIAA Aerospace Sciences Meeting*, (Kissimmee, Florida, 5-9 January), American Institute of Aeronautics and Astronautics Inc,, 2015.

[120] R. M. Kulfan and A. Sigalla, "Real Flow Limitations in Supersonic Airplane Design," *Journal of Aircraft*, vol. 16, no. 10, pp. 645–658, 1979.

[121] H. S. Chung and J. J. Alonso, *Multidisciplinary design optimization of supersonic business jets using approximation model-based genetic algorithms*. PhD thesis, Stanford University, 2004.

[122] B. Liebhardt, K. Lütjens, R. R. Tracy, and A. O. Haas, "Exploring the Prospect of Small Supersonic Airliners - A Case Study Based on the Aerion AS2 Jet," in *17th AIAA Aviation Technology, Integration, and Operations Conference*, (Denver, Colorado, U.S.A, 5-9 June), American Institute of Aeronautics and Astronautics, 2017.

[123] D. J. Maglieri, "Compilation and Review of Supersonic Business Jet Studies from 1963 through 1995," tech. rep., NASA/CR–2011-217144, NASA Langley Research Center, Hampton, Virginia, U.S.A, 2011.

[124] J. P. Jasa, B. J. Brelje, J. S. Gray, C. A. Mader, and J. R. R. A. Martins, "Large-Scale Path-Dependent Optimization of Supersonic Aircraft," *Aerospace*, vol. 7, no. 10, p. 152, 2020.

[125] L. R. Miranda, R. D. Elliott, and W. M. Baker, "A Generalized Vortex Lattice Method for Subsonic and Supersonic Flow Applications," tech. rep., NASA-CR-2865, NASA Langley Research Center, 1977.

[126] J. Rom, B. Melamed, and D. Almosnino, "Experimental and nonlinear vortex lattice method results for various wing-canard configurations," *Journal of aircraft*, vol. 30, no. 2, pp. 207–212, 2012.

[127] R. K. Nangia, M. E. Palmer, and R. H. Doe, "A study of supersonic aircraft with thin wings of low sweep," in *40th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV, U.S.A, 14-17 January), American Institute of Aeronautics and Astronautics,, 2002.

[128] F. Ghaffari and J. E. Lamar, "An attached flow design of a noninterfering leading-edge extension to a thick delta wing," *Journal of Aircraft*, vol. 23, no. 3, pp. 237–243, 1986.

[129] A. Smelsky, *A Comparative Study of the Low Speed Performance of Two Fixed Planforms versus a Variable Geometry Planform for a Supersonic Business Jet*. PhD thesis, Embry-Riddle Aeronautical University, Daytona Beach, USA, 2014.

[130] J. A. Griffin and T. T. Takahashi, "Aero-Spaceplane Mission Performance Estimations Incorporating Atmospheric Control Limits," in *AIAA AVIATION 2022 Forum*, (Chicago, IL, U.S.A, June 27-July 1), American Institute of Aeronautics and Astronautics Inc, AIAA, 2022.

[131] S. Seraj and J. R. Martins, "Aerodynamic Shape Optimization of a Supersonic Transport Considering Low-Speed Stability," in *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, (San Diego, CA, U.S.A, 3-7 January), American Institute of Aeronautics and Astronautics Inc,, 2022.

[132] T. Magee, P. Hayes, A. Khodadoust, and A. Dorgan, "Dynamic Stability Characteristics for Commercial Supersonic Configurations at Low-Speed Flight Conditions," in *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, (San Diego, CA, U.S.A, 3-7 January), American Institute of Aeronautics and Astronautics Inc,, 2022.

[133] S. Basaveswara Rao, A. Chatterjee, D. B. Landrum, and K. Kanistras, "Preliminary analysis of bio-inspired symmetric and asymmetric winglet deformation," in *AIAA Scitech 2021 Forum*, (Virtual Event, 11–15 & 19–21 January), pp. 1–13, American Institute of Aeronautics and Astronautics Inc,, 2021.

[134] E. C. Polhamus, "A Concept Of The Vortex Lift Of Sharp-Edge Delta Wings Based On A Leading-Edge-Suction Analogy," tech. rep., NASA-TN-D-3767, NASA Langley Research Center, 1966.

[135] T. A. Guimarães, C. E. S. Cesnik, and I. Kolmanovsky, "An Integrated Low-Speed Aeroelastic-Flight Dynamics Framework for Modeling Supersonic Aircraft," in *AIAA Science and Technology Forum and Exposition*, (San Diego, C.A., 3-7 January), American Institute of Aeronautics and Astronautics Inc, AIAA, 2022.

[136] T. Banerjee, "Design guidelines for supersonic aircrafts in civil aviation," *International Journal of Aviation, Aeronautics, and Aerospace*, vol. 6, no. 5, pp. 1–24, 2019.

[137] W. Li and S. Rallabhandi, "Inverse Design of Low-Boom Supersonic Concepts Using Reversed Equivalent-Area Targets," *Journal of Aircraft*, vol. 51, no. 1, pp. 29–36, 2014.

[138] H. S. Chung and J. J. Alonso, "Design of a Low-Boom Supersonic Business Jet Using Cokriging Approximation Models," in *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, 2002.

[139] W. Li, E. Shields, and K. Geiselhart, "Mixed-Fidelity Approach for Design of Low-Boom Supersonic Aircraft," *Journal of Aircraft*, vol. 48, no. 4, pp. 1131–1135, 2011.

[140] W. Li, E. Shields, and D. Le, "Interactive Inverse Design Optimization of Fuselage Shape for Low-Boom Supersonic Concepts," *Journal of Aircraft*, vol. 45, no. 4, pp. 1381–1397, 2008.

[141] D. Levin and J. Katz, "Vortex-Lattice Method for the Calculation of the Nonsteady Separated Flow over Delta Wings," *Journal of Aircraft*, vol. 18, no. 12, pp. 1032–1037, 1981.

[142] R. C. Guillermo and W. H. Mason Joseph A Schetz, *Three dimensional flow analysis by the vortex-lattice method*. PhD thesis, Virginia Tech University, 2009.

[143] T. D. Hsing, Z. Q. Zhang, J. S. Hong, and F. G. Zhuang, "A Study of the Interaction of Separated Vortices in Transonic and Supersonic Flow on Improving the Aerodynamic Characteristics of Vortex-Flap," in *Separated Flows and Jets* (V. V. Kozlov and A. V. Dovgal, eds.), pp. 475–483, Springer, Berlin, Heidelberg, 1991.

[144] D. Soban, D. Biezad, and P. Gelhausen, "Computer optimization of aircraft handling qualities during preliminary design," in *1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, vol. 3, (Vancouver, British Columbia, Canada, 22-25 October), pp. 2670–2675, IEEE, 1995.

[145] P. Crisafulli, M. Kaufman, A. A. Giunta, W. H. Mason, B. Grossman, L. T. Watson, and R. T. Haftka, "Response surface approximations for pitching moment including pitch-up in the MDO design of an HSCT," in *6th Symposium on Multidisciplinary Analysis and Optimization*, (Bellevue,WA,U.S.A., 04-06 September), pp. 1308–1322, American Institute of Aeronautics and Astronautics Inc, AIAA,, 1996.

[146] P. A. O. Soviero and H. B. Resende, "Generalized Vortex Lattice Method for Planar Supersonic Flow," *AIAA Journal*, vol. 35, no. 7, pp. 1230–1233, 1997.

[147] P. Theerthamalai and M. Nagarathinam, "Aerodynamic Analysis of Grid-Fin Configurations at Supersonic Speeds," *Journal of Spacecraft and Rockets*, vol. 43, no. 4, pp. 750–756, 2006.

[148] K. Higuchi, L. Zhong, and K. Rinoie, "Wing Design of Supersonic Transport by a Multi-Point Optimization Method," in *26th International Congress of Aeronautical Science*, (Anchorage, Alaska, U.S.A, 14 - 19 September), pp. 1–15, International Council of the Aeronautical Sciences,, 2008.

[149] C.-S. Chiu and I. J. Lin, "Sidewash on the vertical tail in subsonic and supersonic flows," *Journal of Aircraft*, vol. 31, no. 6, pp. 1252–1256, 1994.

[150] T. A. Guimarães, C. E. S. Cesnik, and I. Kolmanovsky, "Low Speed Aerodynamic Modeling for Control-related Considerations in Supersonic Aircraft Design," in *AIAA AVIATION 2021 FORUM*, (Reston, Virginia, 2-6 August), American Institute of Aeronautics and Astronautics,, 2021.

[151] A. M. Benoliel and W. H. Mason, "Pitch-up characteristics for HSCT class planforms: Survey and estimation," in *12th Applied Aerodynamics Conference*, (Colorado Springs,CO,U.S.A), pp. 110–127, American Institute of Aeronautics and Astronautics Inc, AIAA, 1994.

[152] C. E. Lan, "A Quasi-Vortex-Lattice Method in Thin Wing Theory," *Journal of aircraft*, vol. 11, no. 9, pp. 518–527, 1974.

[153] L. Lees, "A Discussion of the Application of the Prandtl-Glauert Method to Subsonic Compressible Flow over a Slender Body of Revolution," tech. rep., 1127, National Advisory Committee of Aeronautics, Langley Field, Virginia, U.S.A, 1946.

[154] N. Coburn, "The Karman-Tseen Pressure-Volume Relation In The Two-Dimensional Supersonic Flow Of Compressible Fluids," *Quarterly of Applied Mathematic*, vol. 3, no. 2, pp. 106–116, 1945.

[155] N. Coburn, "Compressible Supersonic Flow in Jets under the Kármán-Tsien Pressure-Volume Relation," *Journal of Applied Physics*, vol. 22, no. 2, pp. 124–130, 1951.

[156] J. Steinhoff and D. Underhill, "Modification of the Euler equations for "vorticity confinement": Application to the computation of interacting vortex rings," *Physics of Fluids*, vol. 6, no. 8, pp. 2738–2744, 1994.

[157] W. R. Sears, "High speed aerodynamics and jet propulsion. Volume vi. General theory of high speed aerodynamics.," tech. rep., AD0638984, Princeton University,, 1954.

[158] C. B. Millikan, "General Theory of High Speed Aerodynamics," in *High Speed Aerodynamics and Jet Propulsion*, vol. 6, ch. 14, pp. 204–758, Princeton, New Jersey, U.S.A: Princeton University Press, 1954.

[159] F. Hernandes and P. A. Soviero, "Unsteady aerodynamic coefficients obtained by a compressible vortex lattice method," *Journal of Aircraft*, vol. 46, no. 4, pp. 1291–1301, 2009.

[160] T. Takahashi, B. German, A. Shajanian, M. Daskilewicz, and S. Donovan, "Zero Lift Drag and Drag Divergence Prediction for Finite Wings in Aircraft Conceptual Design," in *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, (Orlando, Florida, 4-7 January), American Institute of Aeronautics and Astronautics,, 2010.

[161] D. A. Lovell, "European Research to Reduce Drag for Supersonic Transport Aircraft," in *Aerodynamic Drag Reduction Technologies*, pp. 117–132, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001.

[162] A. K. Gunstensen, D. H. Rothman, S. Zaleski, and G. Zanetti, "Lattice Boltzmann model of immiscible fluids," *Physical Review A*, vol. 43, pp. 4320–4327, 4 1991.

[163] D. Grunau, S. Chen, and K. Eggert, "A lattice Boltzmann model for multiphase fluid flows," *Physics of Fluids A: Fluid Dynamics*, vol. 5, pp. 2557–2562, 10 1993.

[164] D. D'humieres and P. Lallemand, "Numerical Simulations of Hydrodynamics with Lattice Gas Automata in Two Dimensions," *Complex Systems*, vol. 1, pp. 599–632, 1987.

[165] P. A. Skordos, "Initial and boundary conditions for the lattice Boltzmann method," *Physical Review E*, vol. 48, pp. 4823–4842, 12 1993.

[166] D. R. Noble, S. Chen, J. G. Georgiadis, and R. O. Buckius, "A consistent hydrodynamic boundary condition for the lattice Boltzmann method," *Physics of Fluids*, vol. 7, pp. 203–209, 1 1995.

[167] R. Cornubert, D. d'Humières, and D. Levermore, "A Knudsen layer theory for lattice gases," *Physica D: Nonlinear Phenomena*, vol. 47, pp. 241–259, 1 1991.

[168] D. P. Ziegler, "Boundary conditions for lattice Boltzmann simulations," *Journal of Statistical Physics*, vol. 71, pp. 1171–1177, 6 1993.

[169] I. Ginzbourg and P. M. Adler, "Boundary flow condition analysis for the three-dimensional lattice Boltzmann model," *Journal de Physique II*, vol. 4, pp. 191–214, 2 1994.

[170] A. J. C. Ladd and R. Verberg, "Lattice-Boltzmann Simulations of Particle-Fluid Suspensions," *Journal of Statistical Physics*, vol. 104, no. 5/6, pp. 1191–1251, 2001.

[171] T. Inamuro, M. Yoshino, and F. Ogino, "A non-slip boundary condition for lattice Boltzmann simulations," *Physics of Fluids*, vol. 7, pp. 2928–2930, 12 1995.

[172] D. P. Lockard, L.-S. Luo, B. A. Singer, and D. M. Bushnell, "Evaluation of the Lattice-Boltzmann Equation Solver PowerFLOW for Aerodynamic Applications. NASA/CR-2000-210550," tech. rep., NASA Langley Research Center, Hampton, VA United States, 10 2000.

[173] T. Imamura, K. Suzuki, T. Nakamura, and M. Yoshida, "Flow Simulation Around an Airfoil by Lattice Boltzmann Method on Generalized Coordinates," *AIAA Journal*, vol. 43, pp. 1968–1973, 9 2005.

[174] M. Visbal and D. Knight, "The Baldwin-Lomax Turbulence Model for Two-Dimensional Shock-Wave/Boundary-Layer Interactions," *AIAA Journal*, vol. 22, pp. 921–928, 7 1984.

[175] R. Satti, Y. Li, R. Shock, and S. Noelting, "Unsteady Flow Analysis of a Multi-Element Airfoil Using Lattice Boltzmann Method," *AIAA Journal*, vol. 50, pp. 1805–1816, 9 2012.

[176] J. Latt, C. Coreixas, J. Beny, and A. Parmigiani, "Efficient supersonic flow simulations using lattice Boltzmann methods based on numerical equilibria," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 378, p. 20190559, 7 2020.

[177] J. S. Rowlinson *, "The Maxwell–Boltzmann distribution," *Molecular Physics*, vol. 103, pp. 2821–2828, 11 2005.

[178] S. Guo, Y. Feng, J. Jacob, F. Renard, and P. Sagaut, "An efficient lattice Boltzmann method for compressible aerodynamics on D3Q19 lattice," *Journal of Computational Physics*, vol. 418, p. 109570, 10 2020.

[179] J. Chen, Y. Wang, D. Yang, Q. Chen, and J. Sun, "Development of three-dimensional rotated lattice Boltzmann flux solver for the simulation of high-speed compressible flows," *Computers & Fluids*, vol. 265, p. 105992, 10 2023.

[180] K. Suzuki, T. Kato, K. Tsue, M. Yoshino, and M. Denda, "Comparative study between a discrete vortex method and an immersed boundary–lattice Boltzmann method in 2D flapping flight analysis," *International Journal of Modern Physics C*, vol. 32, p. 2150005, 1 2021.

[181] R. Sadeghi, M. Shadloo, M. Hopp-Hirschler, A. Hadjadj, and U. Nieken, "Three-dimensional lattice Boltzmann simulations of high density ratio two-phase flows in porous media," *Computers & Mathematics with Applications*, vol. 75, pp. 2445–2465, 4 2018.

[182] W. H. Hager and O. Castro-Orgaz, "William Froude and the Froude Number," *Journal of Hydraulic Engineering*, vol. 143, 4 2017.

[183] F. Renard, Y. Feng, J.-F. Boussuge, and P. Sagaut, "Improved compressible hybrid lattice Boltzmann method on standard lattice for subsonic and supersonic flows," *Computers & Fluids*, vol. 219, p. 104867, 2021.

[184] T. Theodorsen, "General theory of aerodynamic instability and the mechanism of flutter," tech. rep., National Advisory Committee of Aeronautics, 1949.

[185] H. Wagner, "Über die Entstehung des dynamischen Auftriebes von Tragflügeln," *ZAMM - Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 5, no. 1, pp. 17–35, 1925.

[186] H. Kussner, "Zusammenfassender Bericht über den instationären Auftrieb von Flügeln," Tech. Rep. 12, Luftfahrt-Forschung, 1936.

[187] P. D. Sclavounos, "An unsteady lifting-line theory," *Journal of Engineering Mathematics*, vol. 21, no. 3, pp. 201–226, 1987.

[188] P. A. O. Soviero and F. Hernandes, "Compressible Unsteady Vortex Lattice Method for Arbitrary Two-Dimensional Motion of Thin Profiles," *Journal of Aircraft*, vol. 44, pp. 1494–1498, 9 2007.

[189] E. Dowell, K. Hall, J. Thomas, R. Florea, B. Epureanu, and J. Heeg, "Reduced order models in unsteady aerodynamics," in *40th Structures, Structural Dynamics, and Materials Conference and Exhibit*, vol. 1, (Reston, Virigina), pp. 622–637, American Institute of Aeronautics and Astronautics, 4 1999.

[190] K. Li, J. Kou, and W. Zhang, "Unsteady aerodynamic reduced-order modeling based on machine learning across multiple airfoils," *Aerospace Science and Technology*, vol. 119, p. 107173, 12 2021.

[191] D. G. Crighton, "The Kutta Condition in Unsteady Flow," *Annual Review of Fluid Mechanics*, vol. 17, pp. 411–445, 1 1985.

[192] L. Xiaohua, Z. Guo, G. Dana, and Z. Hou, "Efficient reduced-order modeling of unsteady aerodynamics under light dynamic stall conditions," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 233, pp. 2141–2151, 5 2019.

[193] M. Lin, S. Guo, S. He, W. Li, and D. Yang, "Structure health monitoring of a composite wing based on flight load and strain data using deep learning method," *Composite Structures*, vol. 286, p. 115305, 4 2022.

[194] T. Melin, *User's guide and reference manual for Tornado.*, vol. 1. Stockholm: Royal Institute of Technology KTH, 2.3 ed., 1 2000.

[195] H. E. Taha and A. S. Rezaei, "On the high-frequency response of unsteady lift and circulation: A dynamical systems perspective," *Journal of Fluids and Structures*, vol. 93, p. 102868, 2 2020.

[196] M. E. Goldstein and H. Atassi, "A complete second-order theory for the unsteady flow about an airfoil due to a periodic gust," *Journal of Fluid Mechanics*, vol. 74, pp. 741–765, 4 1976.

[197] Y. Wu, Y. Dai, C. Yang, and G. Huang, "Unsteady and nonlinear aerodynamic prediction of airfoil undergoing large-amplitude pitching oscillation based on gated recurrent unit network," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 237, pp. 270–284, 2 2023.

[198] Z. LI, L. FENG, and J. WANG, "Individual influence of pitching and plunging motions on flow structures over an airfoil during dynamic stall," *Chinese Journal of Aeronautics*, vol. 33, pp. 840–851, 3 2020.

[199] S. Verma, G. Novati, F. Noca, and P. Koumoutsakos, "Fast Motion of Heaving Airfoils," *Procedia Computer Science*, vol. 108, pp. 235–244, 1 2017.

[200] L. R. Smith and A. R. Jones, "Vortex formation on a pitching aerofoil at high surging amplitudes," *Journal of Fluid Mechanics*, vol. 905, p. A22, 12 2020.

[201] X. Xia and K. Mohseni, "Unsteady aerodynamics and vortex-sheet formation of a two-dimensional airfoil," *Journal of Fluid Mechanics*, vol. 830, pp. 439–478, 11 2017.

[202] M. Moriche, O. Flores, and M. García-Villalba, "On the aerodynamic forces on heaving and pitching airfoils at low Reynolds number," *Journal of Fluid Mechanics*, vol. 828, pp. 395–423, 10 2017.

[203] D. E. Raveh, "Reduced-Order Models for Nonlinear Unsteady Aerodynamics," *AIAA Journal*, vol. 39, pp. 1417–1429, 8 2001.

[204] M. J. Wright, K. Sinha, J. Olejniczak, G. V. Candler, T. D. Magruder, and A. J. Smits, "Numerical and Experimental Investigation of Double-Cone Shock Interactions," *AIAA Journal*, vol. 38, pp. 2268–2276, 12 2000.

[205] O. Frayssinet, "Numerical analysis of a separated flow on a supersonic cone flare model," in *34th AIAA Applied Aerodynamics Conference*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, 6 2016.

[206] C. Bliamis, C. Menelaou, and K. Yakinthos, "Implementation of various-fidelity methods for viscous effects modeling on the design of a waverider," *Aerospace Science and Technology*, vol. 133, p. 108141, 2 2023.

[207] G. I. Taylor and J. W. Maccoll, "The A ir Pressure on a Cone Moving at High Speeds.-II," tech. rep., 1931.

[208] G. Taylor and J. Maccoll, "The Air Pressure on a Cone Moving at High Speeds. I on JSTOR," in *Proceedings of the Royal Society of London*, (London), pp. 278–297, Royal Society, 10 1933.

[209] K. Bowcutt, G. Kuruvila, T. Grandine, and E. Cramer, "Advancements in Multidisciplinary Design Optimization Applied to Hypersonic Vehicles to Achieve Performance Closure," in *15th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, 4 2008.

[210] C. Tian, N. Li, G. Gong, and Z. Su, "A parameterized geometry design method for inward turning inlet compatible waverider," *Chinese Journal of Aeronautics*, vol. 26, pp. 1135–1146, 10 2013.

[211] A. Ueno and K. Suzuki, "CFD-Based Shape Optimization of Hypersonic Vehicles Considering Transonic Aerodynamic Performance," in *46th AIAA Aerospace Sciences Meeting and Exhibit*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, 1 2008.

[212] A. Aprovitola, O. Dyblenko, G. Pezzella, and A. Viviani, "Aerodynamic Analysis of a Supersonic Transport Aircraft at Low and High Speed Flow Conditions," *Aerospace*, vol. 9, p. 411, 7 2022.

[213] D. R. Sandlin and D. N. Pessin, "Aerodynamic analysis of hypersonic waverider aircraft NASA-CR-192981," tech. rep., California Polytechnic State University, California, 4 1993.

[214] D. Kinney, "Aero-Thermodynamics for Conceptual Design," in *42nd AIAA Aerospace Sciences Meeting and Exhibit*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, 1 2004.

[215] A. Shi, J. Chen, E. H. Dowell, and H. Wen, "Approach to Determine the Most Efficient Supersonic Mach Number," *AIAA Journal*, vol. 58, pp. 1402–1406, 3 2020.

[216] E. Loewenthal and A. Gopalarathnam, "Low-Order Modeling of Wingtip Vortices in a Vortex Lattice Method," *AIAA Journal*, vol. 60, pp. 1708–1720, 3 2022.

[217] A. Kontogiannis and E. Laurendeau, "Adjoint State of Nonlinear Vortex-Lattice Method for Aerodynamic Design and Control," *AIAA Journal*, vol. 59, pp. 1184–1195, 4 2021.

[218] H. Joshi and P. Thomas, "Review of vortex lattice method for supersonic aircraft design," *The Aeronautical Journal*, vol. 127, pp. 1–35, 4 2023.

[219] G. XIANG, C. WANG, H. TENG, and Z. JIANG, "Shock/shock interactions between bodies and wings," *Chinese Journal of Aeronautics*, vol. 31, pp. 255–261, 2 2018.

[220] G. S. Settles and L. J. Dodson, "Supersonic and hypersonic shock/boundary-layer interaction database," *AIAA Journal*, vol. 32, pp. 1377–1383, 7 1994.

[221] E. Roohi, S. Stefanov, A. Shoja-Sani, and H. Ejraei, "A generalized form of the Bernoulli Trial collision scheme in DSMC: Derivation and evaluation," *Journal of Computational Physics*, vol. 354, pp. 476–492, 2 2018.

[222] D. C. PACK, "A note on Prandtl's formula for the wave-length of a supersonic gas jet.," *The Quarterly Journal of Mechanics and Applied Mathematics*, vol. 3, no. 2, pp. 173–181, 1950.

[223] D. Lorenzon, "Open FOAM simulations of the supersonic flow around cones at angles of attack," *Journal of Mechanical and Civil Engineering*, vol. 16, pp. 67–79, 10 2019.

[224] C. Orlebar, *Concorde* , vol. 1. Bloomsbury Publishing, 2017.

[225] C. P. Nelson, K.-Y. Ting, N. Mavriplis, R. Soltani, and E. Livne, "Supersonic Configurations at Low Speeds (SCALOS): Project Background and Progress at University of Washington," in *AIAA SCITECH 2022 Forum*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, 1 2022.

[226] D. Munday, E. Gutmark, J. Liu, and K. Kailasanath, "Flow structure and acoustics of supersonic jets from conical convergent-divergent nozzles," *Physics of Fluids*, vol. 23, 11 2011.

[227] Y. Sun and H. Smith, "Low-boom low-drag solutions through the evaluation of different supersonic business jet concepts," *The Aeronautical Journal*, vol. 124, pp. 76–95, 1 2020.

[228] T. Jones, "NASA's Quiet Supersonic Aircraft," tech. rep., Oshkosh, WI, 7 2017.

[229] R. Nangia, M. Ghoreyshi, M. P. van Rooij, and R. M. Cummings, "Aerodynamic design assessment and comparisons of the MULDICON UCAV concept," *Aerospace Science and Technology*, vol. 93, p. 105321, 10 2019.

[230] C. P. Knisely, C. L. Haley, and X. Zhong, "Impact of Hypersonic Boundary Layer Transition on Skin Drag and Surface Heating on Blunt Cones," in *AIAA Scitech 2019 Forum*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, 1 2019.

[231] Y. L. Feng, S. L. Guo, W. Q. Tao, and P. Sagaut, "Regularized thermal lattice Boltzmann method for natural convection with large temperature differences," *International Journal of Heat and Mass Transfer*, vol. 125, pp. 1379–1391, 10 2018.

[232] A. Ambrosio and A. Wortman, "Stagnation-Point Shock-Detachment Distance for Flow Around Spheres and Cylinders in Air," *Journal of the Aerospace Sciences*, vol. 29, pp. 875–875, 7 1962.

[233] K. Hida, "An Approximate Study on the Detached Shock Wave in front of a Circular Cylinder and a Sphere," *Journal of the Physical Society of Japan*, vol. 8, pp. 740–745, 11 1953.

[234] T. Krueger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, and E. M. Viggen, *The Lattice Boltzmann Method: Principles and Practice*. Springer, 2016.

[235] C. Coreixas, G. Wissocq, G. Puigt, J. F. Boussuge, and P. Sagaut, "Recursive regularization step for high-order lattice Boltzmann methods," *Physical Review E*, vol. 96, p. 033306, 9 2017.

[236] P. L. Bhatnagar, E. P. Gross, and M. Krook, "A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems," *Physical Review*, vol. 94, p. 511, 5 1954.

[237] P. Lallemand and L. S. Luo, "Theory of the lattice Boltzmann method: Acoustic and thermal properties in two and three dimensions," *Physical Review E*, vol. 68, p. 036706, 9 2003.

[238] D. Spiller and B. Dünweg, "Semiautomatic construction of lattice Boltzmann models," *Physical Review E*, vol. 101, p. 043310, 4 2020.

[239] G. Soave, "Improvement of the Van Der Waals equation of state," *Chemical Engineering Science*, vol. 39, no. 2, pp. 357–369, 1984.

[240] N. Frapolli, S. S. Chikatamarla, and I. V. Karlin, "Entropic lattice Boltzmann model for compressible flows," *Physical Review E*, vol. 92, p. 061301, 12 2015.

[241] N. Frapolli, S. S. Chikatamarla, and I. V. Karlin, "Entropic lattice Boltzmann model for gas dynamics: Theory, boundary conditions, and implementation," *Physical Review E*, vol. 93, p. 063302, 6 2016.

[242] N. Frapolli, *Entropic Lattice Boltzmann Model For Thermal And Compressible Flow*. PhD thesis, ETH Zurich, Zurich, 2017.

[243] M. Hafez and E. Wahba, "Simulations of viscous transonic flows over lifting airfoils and wings," *Computers & Fluids*, vol. 36, pp. 39–52, 1 2007.

[244] C. Coreixas, G. Wissocq, B. Chopard, and J. Latt, "Impact of collision models on the physical properties and the stability of lattice Boltzmann methods," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 378, p. 20190397, 7 2020.

[245] J. Latt, C. Coreixas, J. Beny, and A. Parmigiani, "Efficient supersonic flow simulations using lattice Boltzmann methods based on numerical equilibria," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 378, p. 20190559, 7 2020.

[246] J. Freeman and C. Roy, "Verification and Validation of RANS Turbulence Models in Commercial Flow Solvers," in *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, 1 2012.

Appendix A

A.1 Unsteady Modelling

```
%% Function: Unsteady Vortex Lattice Method 3D
% This function calculates unsteady aerodynamic characteristics of a wing
% using the vortex lattice method. It computes both time response and
% frequency response based on input parameters.

function [results] = unsteadysolver3d(results, state, geo, lattice, ref)
    % Display setup information
    disp('----------------------');
    disp('Unsteady Vortex Lattice Method 3D:');

    % Input the number of time steps
    Nstep = input('Number of time steps: ');
    disp(' ');

    % Calculate local velocity components
    [~, a, ~, ~] = ISAtmosphere(state.ALT);  % Speed of sound based on altitude
    v = state.AS / a;  % Mach number
    u = v * cos(state.alpha);  % x-velocity
    w = v * sin(state.alpha);  % z-velocity

```

```
21      % Wing chord length
22      C = geo.c(1,1);
23
24      % Heaving motion parameters
25      h0 = 2;             % Heaving amplitude
26      omega = 45;         % Heaving frequency
27
28      % Time step and time vector
29      dt = 0.0009;
30      t = 0:dt:(Nstep-1)*dt;  % Time vector
31
32      % Heaving motion over time
33      sx = -v * t;                          % x-displacement
34      sz = -h0 * sin(omega * t);            % z-displacement
35      wt = h0 * omega * cos(omega * t);  % z-velocity due to
           heaving
36
37      % Wake vortex core size
38      cutoff = 0.0001;
39
40      % Initialize wake data
41      vortic = zeros(Nstep, 4);
42
43      % Shed wake element resultant velocity magnitude
44      Q = sqrt(v^2 + wt.^2);
45
46      % Distance traveled along chord length
47      C_length = (v * t) / C;
48
49      % Main simulation loop for unsteady response
50      for it = 1:Nstep
51          % Calculate local angle of attack (AoA) based on
               position
52          AoA_x_pos = 0.75 * C * cos(state.alpha) + sx;
53          AoA_z_pos = 0.75 * C * sin(state.alpha) + sz;
54          state.alpha1 = atan(AoA_z_pos(it) / AoA_x_pos(it));
```

```
55
56          % Infinite distance for wake propagation
57          infdist = config('infinity');
58          if isempty(infdist)
59              infdist = 6 * ref.b_ref;  % Default value
60          end
61
62          % Calculate vortex positions in lattice
63          infx = infdist * cos(state.alpha1) * cos(state.
                betha);
64          infy = -infdist * sin(state.betha);
65          infz = infdist * sin(state.alpha1) * cos(state.
                betha);
66
67          % Update wake element positions
68          for t = 1:size(lattice.VORTEX, 1)
69              for s = 1:2
70                  x = infx + lattice.VORTEX(t, s, 1);
71                  y = infy + lattice.VORTEX(t, s, 2);
72                  z = infz + lattice.VORTEX(t, s, 3);
73
74                  % Induced velocity components from the wake
                        elements
75                  dx(t, s) = -x * (2 - cos(state.Q) - cos(
                        state.R));
76                  dy(t, s) = sin(state.P) * z - sin(state.R)
                        * x + (1 - cos(state.P)) * y;
77                  dz(t, s) = sin(state.Q) * x - sin(state.P)
                        * y + (1 - cos(state.P)) * z;
78              end
79          end
80
81          % Update lattice vortex positions based on induced
                velocities
82          lattice.VORTEX(:, :, 1) = lattice.VORTEX(:, :, 1) +
              u * dt;
```

```
83          lattice.VORTEX(:, :, 3) = lattice.VORTEX(:, :, 3) +
                v * dt;
84      end
85
86      % Calculate aerodynamic properties
87      panel_area = tarea(lattice.VORTEX);  % Calculate panel
            areas
88      results.lift = w * cos(state.alpha) - u * sin(state.
            alpha);  % Lift force
89      results.drag = w * sin(state.alpha) + u * cos(state.
            alpha);  % Drag force
90      results.lift_coefficient = results.lift ./ (0.5 * state
            .rho * state.AS^2 * panel_area);
91
92      % Save results
93      results.latticeVORTEX = lattice.VORTEX;
94      results.u = u;
95      results.v = v;
96      results.w = w;
97
98      % Visualization
99      figure;
100     surf(lattice.XYZ(:, :, 1), lattice.XYZ(:, :, 2),
            lattice.XYZ(:, :, 3));
101     hold on;
102     xlabel('X (Body Frame)');
103     ylabel('Y (Body Frame)');
104     zlabel('Z (Body Frame)');
105     title('Unsteady Wake of an Aircraft');
106     grid on;
107     axis equal;
108 end
```

A.2 Machcone Modelling

```
%% Supersonic Flow Analysis using Mach Cone Modeling with TMHVM
% This script performs supersonic flow analysis around an airfoil using
% the Taylor-Maccoll hypervelocity method (TMHVM) and calculates the
% pressure changes caused by shock waves.

% Define freestream conditions
gamma = 1.4;           % Specific heat ratio of air
M_inf = 1;             % Freestream Mach number
rho_inf = 1.225;       % Freestream density (kg/m^3)
T_inf = 288;           % Freestream temperature (K)
P_inf = 101325;        % Freestream pressure (Pa)
a_inf = sqrt(gamma * P_inf / rho_inf); % Freestream speed of sound (m/s)
u_inf = M_inf * a_inf;                 % Freestream velocity (m/s)
e_inf = P_inf / ((gamma - 1) * rho_inf); % Freestream specific internal energy (J/kg)

% Retrieve airfoil coordinates from lattice data
x = lattice.XYZ(:, :, 1);  % Extract x-coordinates of the airfoil
y = lattice.XYZ(:, :, 3);  % Extract z-coordinates as y-coordinates (assuming xz-plane)
[m, ~] = size(x);          % Number of points in airfoil geometry

% Define grid for MOC analysis
Nx = 400; % Number of x-grid points
Ny = 200; % Number of y-grid points
x_grid = linspace(0, 200, Nx); % x-coordinates of grid points
y_grid = linspace(0, 5, Ny);   % y-coordinates of grid points
```

```
[xx, yy] = meshgrid(x_grid, y_grid); % 2D meshgrid for plotting

% Initialize flow properties arrays
P = zeros(Ny, Nx); % Pressure array
M = zeros(Ny, Nx); % Mach number array
theta = zeros(Ny, Nx); % Flow angle array
rho = rho_inf * ones(Ny, Nx); % Density array
T = T_inf * ones(Ny, Nx);      % Temperature array
u = u_inf * ones(Ny, Nx);      % x-velocity array
v = zeros(Ny, Nx);             % y-velocity array
e = e_inf * ones(Ny, Nx);      % Specific internal energy array

% Initialize conditions along airfoil surface
theta(1:m-1, 1) = atan2(y(2:m) - y(1:m-1), x(2:m) - x(1:m-1)); % Surface flow angle
theta(m, 1) = theta(m-1, 1); % Set trailing edge flow angle equal to last surface flow angle
M(1:m-1, 1) = u_inf ./ a_inf .* sin(theta(1:m-1, 1)) .* ...
    (1 + (gamma-1)/2 * M_inf^2 .* sin(theta(1:m-1, 1)).^2).^(-0.5); % Surface Mach number
M(m, 1) = M(m-1, 1); % Set trailing edge Mach number equal to last surface Mach number
P(1:m-1, 1) = P_inf * (1 + (gamma-1)/2 * M_inf^2 .* sin(theta(1:m-1, 1)).^2).^(-gamma/(gamma-1)); % Surface pressure
P(m, 1) = P(m-1, 1); % Set trailing edge pressure

% Solve flowfield using Taylor-Maccoll equation
for i = 2:Nx % Loop over x-direction
    for j = 1:Ny % Loop over y-direction
        % Calculate characteristic angle
        theta(j, i) = atan2((y_grid(j) - yy(j, i-1)), (x_grid(i) - xx(j, i-1)));

```

```
53          % Calculate characteristic Mach number
54          M(j, i) = 1 / sin(theta(j, i)) * sqrt((1 + 0.5 * (
               gamma-1) * M(j, i-1)^2 * sin(theta(j, i-1))^2) /
                ...
55               (gamma * M(j, i-1)^2 * sin(theta(j, i-1))^2 -
                    0.5 * (gamma-1)));
56
57          % Calculate pressure and velocity adjustments due
               to shock waves
58          if xx(j, i-1) > max(x) % Check if flowfield point
               is downstream of airfoil
59               % Upstream conditions
60               M1 = M(j, i-1);
61               P1 = P(j, i-1);
62               rho1 = rho(j, i-1);
63
64               % Downstream shock conditions
65               M2 = sqrt((M1^2 + 2 / (gamma-1)) / (2 * gamma /
                     (gamma-1) * M1^2 - 1));
66               P2 = P1 * (1 + (2 * gamma / (gamma+1)) * (M1^2
                    - 1));
67               rho2 = rho1 * (1 + (2 * gamma / (gamma+1)) * (
                    M1^2 - 1)) / ...
68                    (1 + (2 * gamma / (gamma+1)) * (M2^2 - 1));
69
70               % Update flow properties
71               P(j, i) = P2;
72               rho(j, i) = rho2;
73               u(j, i) = M2 * a_inf * cos(theta(j, i));
74               v(j, i) = M2 * a_inf * sin(theta(j, i));
75          else
76               % Freestream conditions
77               P(j, i) = P(j, i-1);
78               rho(j, i) = rho(j, i-1);
79               u(j, i) = u(j, i-1);
80               v(j, i) = v(j, i-1);
```

```
        end
    end
end

% Plot pressure change due to shock waves
delta_P = P(:, 2:end) - P(:, 1:end-1); % Pressure difference between adjacent grid points
figure;
contourf(x_grid(1:end-1), y_grid, delta_P, 'LineStyle', 'none');
xlabel('x (m)');
ylabel('y (m)');
title('Change in Pressure due to Shock Waves');
colormap(parula(100));
colorbar;

% --- Integrate Changes into the Vortex Lattice Method (VLM) ---
% The recalculated pressure and velocity fields are used to update the
% aerodynamic coefficients (lift and drag) using the VLM approach.

% Initialize storage for updated lift and drag
updated_lift = zeros(size(x, 1), 1);
updated_drag = zeros(size(x, 1), 1);

% Loop through airfoil panels and compute changes due to updated flowfield
for i = 1:m-1
    % Panel geometry
    dx = x(i+1) - x(i);  % Panel chord length
    dy = y(i+1) - y(i);  % Panel vertical height

    % Effective velocity components at panel midpoint
    u_avg = (u(:, i) + u(:, i+1)) / 2;
```

```
    v_avg = (v(:, i) + v(:, i+1)) / 2;
    velocity_magnitude = sqrt(u_avg.^2 + v_avg.^2);

    % Pressure difference across the panel
    delta_p = P(:, i) - P(:, i+1);

    % Contribution to lift and drag
    updated_lift(i) = delta_p .* dy;  % Lift force
       contribution
    updated_drag(i) = delta_p .* dx; % Drag force
       contribution
end

% Total lift and drag forces
total_lift = sum(updated_lift);
total_drag = sum(updated_drag);

% Lift and drag coefficients
S_ref = sum((x(2:end) - x(1:end-1)) .* abs(y(2:end) - y(1:
   end-1))); % Reference area
C_L = total_lift / (0.5 * rho_inf * u_inf^2 * S_ref); %
   Lift coefficient
C_D = total_drag / (0.5 * rho_inf * u_inf^2 * S_ref); %
   Drag coefficient

% Display updated coefficients
disp('--- Updated Aerodynamic Coefficients ---');
disp(['Lift Coefficient (C_L): ', num2str(C_L)]);
disp(['Drag Coefficient (C_D): ', num2str(C_D)]);

% --- Visualization ---
% Plot updated pressure distribution on the airfoil surface
figure;
plot(x, P(:, 1), '-o', 'LineWidth', 1.5); % Pressure
   distribution along the airfoil
xlabel('x (m)');
```

```
ylabel('Pressure (Pa)');
title('Updated Pressure Distribution on Airfoil Surface');
grid on;

% Plot updated lift and drag distributions
figure;
bar(x(1:end-1), updated_lift, 'FaceColor', 'b'); % Lift
   distribution
hold on;
bar(x(1:end-1), updated_drag, 'FaceColor', 'r'); % Drag
   distribution
xlabel('x (m)');
ylabel('Force Contribution (N)');
title('Lift and Drag Force Distribution Along Airfoil');
legend('Lift Contribution', 'Drag Contribution');
grid on;
```

A.3 Compressibility modelling with LBM

A.3.1 Main

```
% This is the main file to run the LBM code for the
   multiphase using the
% D2Q9 Lattice based on BGK collision operator modified for
    supersonic simulation
clear all
% close all
%
% % Check if a parallel pool is open
% if ~isempty(gcp('nocreate'))
%     % If a parallel pool is open, close it
%     delete(gcp('nocreate'));
% end

s2 = RandStream.create('mrg32k3a', 'NumStreams', 3, '
   StreamIndices', 2);
```

```
RandStream.setGlobalStream(s2);
defaultStream = RandStream.getGlobalStream;

reset(defaultStream);
addpath('plotclr');

format long e

% Initialization parameters for supersonic flow around a
   flat plate
supersonicVelocity = 343; % Example: 1.5 times the speed of
    sound (~343 m/s at sea level)
temp_R = [200000 400000]; % Temporary Reynolds number (may
   need adjustment for supersonic flow)
temp_A = [0.0004 0.01]; % Lattice spacing parameter
temp_rhoR = [1, 6]; % Density
temp_gamma = [1, 2]; % Relaxation factor (may need
   adjustment for stability in supersonic regime)

ID = 0; % Model descriptor or procedure with unique
   identification (ID)

for e = 1:length(temp_gamma)
    for c = 1:length(temp_A)
        for d = 1:length(temp_rhoR)
            for h = 1:length(temp_R)
                ID = ID + 1;
                A(ID) = temp_A(c);
                rhoR(ID) = temp_rhoR(d);
                gamma(ID) = temp_gamma(e);
                R(ID) = temp_R(h);
            end
        end
    end
end

```

```
for i = 1:ID
    % Initialisation for the multiphase Lattice Boltzmann
       method adapted for supersonic flow

    Opt{i}.NX = 2000 ; % No of grid points in x direction
    Opt{i}.NY = 1000; % No of grid points in y direction
    Opt{i}.tmax = 20000; % Maximum time for this
       simulations
    Opt{i}.maxDiffTol = 10^-10; % Maximum tolerance in the
       results from differentiation
    Opt{i}.ncheck = 200; % Update the results or simulation
        every 500 steps
    Opt{i}.supersonicVelocity = supersonicVelocity; %
       Setting the supersonic flow velocity
    Opt{i}.beta = 0.66; % Relaxation time
    Opt{i}.gamma = gamma(i); % Relaxation parameter
    Opt{i}.aR = A(i); % Lattice spacing in LBM
    Opt{i}.rhoR = rhoR(i); % Density
    Opt{i}.R = R(i); % Reynolds number (may need adjustment
        for supersonic flow)
    Opt{i}.plateLength = 200; %length of the plate which is
        our obstacle
    Opt{i}.plateThickness = 20; %thickness of the plate
    Opt{i}.platePositionY = 500; %position of plate in the
       domain
     %define the obstacle
    Opt{i}.ObstacleMatrix = ObstacleMatrix(Opt{i}.NX, Opt{i
       }.NY, Opt{i}.plateLength, Opt{i}.plateThickness);

    % The type of gradient discretisation is changed here.
    Opt{i}.gradient = 'anisotropic'; % 'isotropic';

    if(strcmp(Opt{i}.gradient, 'isotropic'))
        Opt{i}.aR = Opt{i}.aR * 6; % Take the same lattice
           spacing and multiply it by 6
    end
```

```
    Opt{i}.aB = Opt{i}.aR; % New updated lattice spacing
       for isotropic flow
    Opt{i}.alphaB = 0.6; % Parameter defining the viscosity
        in discretisation
    Opt{i}.alphaR = 1 - (1 - Opt{i}.alphaB) / Opt{i}.gamma;
        % Stabilisation parameter
    Opt{i}.nuR = 1 / 6;
    Opt{i}.nuB = 1 / 6;
    Opt{i}.omegaR = 1 / (3 * Opt{i}.nuR + 0.5);
    Opt{i}.omegaB = 1 / (3 * Opt{i}.nuB + 0.5);
    Opt{i}.ID = i;
end

% Matlabpool or parpool for running it in parallel
% parpool
for i = 1:ID
    LBMmultiPhase(Opt{i});
end
```

A.3.2 Neighbouring cell

```
function [neighbour, isObstacle, i, j] = get_neighbour(
   address, NX, NY, obstacleMatrix)
    % Transform the linear index into two-dimensional index
    [i, j] = ind2sub([NX, NY], address);

    % Apply periodic boundary conditions
    iE = mod(i, NX) + 1;          % East neighbor
    iW = mod(i - 2 + NX, NX) + 1;     % West neighbor with
       periodicity
    jN = mod(j, NY) + 1;          % North neighbor
    jS = mod(j - 2 + NY, NY) + 1;     % South neighbor with
       periodicity

    % Initialize the neighbour matrix
    neighbour = zeros(numel(address), 9, 'uint32');
```

```
13
14      % Fill in the neighbour matrix
15      neighbour(:, 1) = uint32(address);
                                          % Center
16      neighbour(:, 2) = uint32(sub2ind([NX, NY], i, jN));
                        % North
17      neighbour(:, 3) = uint32(sub2ind([NX, NY], iE, jN));
                       % North-East
18      neighbour(:, 4) = uint32(sub2ind([NX, NY], iE, j));
                        % East
19      neighbour(:, 5) = uint32(sub2ind([NX, NY], iE, jS));
                       % South-East
20      neighbour(:, 6) = uint32(sub2ind([NX, NY], i, jS));
                        % South
21      neighbour(:, 7) = uint32(sub2ind([NX, NY], iW, jS));
                       % South-West
22      neighbour(:, 8) = uint32(sub2ind([NX, NY], iW, j));
                        % West
23      neighbour(:, 9) = uint32(sub2ind([NX, NY], iW, jN));
                       % North-West
24
25      % Initialize the isObstacle matrix
26      isObstacle = false(numel(address), 9);
27
28      % Check if each neighbor is an obstacle
29      for k = 1:9
30          isObstacle(:, k) = ObstacleMatrix(neighbour(:, k));
31      end
32  end
```

A.3.3 LBMmultiphase modelling

```
1   function Opt = LBMmultiPhase(Opt)
2       % Update the flow velocity for supersonic conditions
3       % flowVelocityX = supersonicVelocity; % Assuming this
            is defined in your Opt structure
```

```
4
5       % Define flat plate region
6       plateLength = Opt.NX; % Full width of the domain
7       plateYPosition = round(Opt.NY / 2); % Center Y position
            of the domain
8       flatPlateRegion = zeros(Opt.NX, Opt.NY);
9       flatPlateRegion(:, plateYPosition) = 1; % Flat plate at
            the center row
10
11      % Generate meshgrid coordinates corresponding to the
            domain
12      x = linspace(0, Opt.NX-1, Opt.NX);  % X coordinates
            covering the entire domain
13      y = linspace(0, Opt.NY-1, Opt.NY);  % Y coordinates
            covering the entire domain
14
15      % Create the obstacle matrix
16      obstacleMatrix = ObstacleMatrix(Opt.NX, Opt.NY,
            plateLength, Opt.plateThickness);
17      % disp('Size of ObstacleMatrix');
18      % disp(size(obstacleMatrix));
19
20      % Lattice information (periodic boundary conditions)
21      [siteNeighbour, ~, ~, ~] = get_neighbour(1:Opt.NX*Opt.
            NY, Opt.NX, Opt.NY, Opt.ObstacleMatrix);
22
23      % Debugging: Print siteNeighbour size and content
24      % disp('Size of siteNeighbour:');
25      % disp(size(siteNeighbour));
26      % disp('Content of siteNeighbour:');
27      % disp(siteNeighbour);
28
29      % siteNeighbour = get_neighbour(1:Opt.NX*Opt.NY, Opt.NX
            , Opt.NY);
30
31      % Define some D2Q9 lattice constants
```

```
32      w = [4/9, 1/9, 1/36, 1/9, 1/36, 1/9, 1/36, 1/9, 1/36];
33      c(1,:) = [0, 0, 1, 1, 1, 0, -1, -1, -1];
34      c(2,:) = [0, 1, 1, 0, -1, -1, -1, 0, 1];
35      upwindOrientation = [1 6:9 2:5];
36
37      % Initialization of the density distribution functions
38      R = get_FeqMultiPhase(Opt.rhoR*ones(Opt.NX*Opt.NY, 1),
           zeros(Opt.NX*Opt.NY, 1), zeros(Opt.NX*Opt.NY, 1),
           Opt.alphaR, w, c);
39      B = get_FeqMultiPhase(Opt.rhoR/Opt.gamma*ones(Opt.NX*
           Opt.NY, 1), zeros(Opt.NX*Opt.NY, 1), zeros(Opt.NX*
           Opt.NY, 1), Opt.alphaB, w, c);
40      % disp('size of R');
41      % disp(size(R));
42      % disp('size of B');
43      % disp(size(B));
44
45      % MAIN LOOP
46      disp('Start main loop.............'); tic;
47
48      % Variables for monitoring convergence
49      maxDiff = 10^11;
50      maxDiffOld = 10^10;
51      t = 0;
52      tOld = 0;
53
54      % Initialize drag force tracking
55      dragForceX = 0;
56
57      while (t < Opt.tmax && maxDiff > Opt.maxDiffTol)
58          % Collision
59          [R, B] = collisionMultiPhase(R, B, siteNeighbour, w
               , c, upwindOrientation, Opt, obstacleMatrix);
60
61          % Apply bounce-back boundary conditions for the
               flat plate
```

```
62          for dir = 1:9
63              % Find the opposite direction for each lattice
                    direction
64              oppositeDir = mod(dir+4, 9); if oppositeDir==0,
                     oppositeDir=9; end
65              % Apply bounce-back condition
66              R(flatPlateRegion==1, oppositeDir) = R(
                    flatPlateRegion==1, dir);
67              B(flatPlateRegion==1, oppositeDir) = B(
                    flatPlateRegion==1, dir);
68          end
69
70          % Streaming
71          R = streaming(R, siteNeighbour, obstacleMatrix);
72          B = streaming(B, siteNeighbour, obstacleMatrix);
73
74          % Store old distributions for convergence check
75          Rold = R;
76          Bold = B;
77
78          % Increment the time-step
79          t = t + 1;
80
81          % Compute force on flat plate
82          rhoR = sum(R, 2);
83          rhoB = sum(B, 2);
84          rho = rhoR + rhoB;
85          ux = ((sum(R(:, [3, 4, 5]), 2) - sum(R(:, [7, 8,
                9]), 2) + sum(B(:, [3, 4, 5]), 2) - sum(B(:, [7,
                 8, 9]), 2)) ./ rho) * Opt.supersonicVelocity;
86          uy = ((sum(R(:, [2, 3, 9]), 2) - sum(R(:, [5, 6,
                7]), 2) + sum(B(:, [2, 3, 9]), 2) - sum(B(:, [5,
                 6, 7]), 2)) ./ rho) * Opt.supersonicVelocity;
87
88          % Calculate drag force contribution
```

```
89          pressure = (1/3) * rho; % Assuming ideal gas law
               for simplicity
90          tau_xx = ux .* ux; % Viscous stress tensor
               component in x-direction (simplified)
91
92          % Make sure pressure and tau_xx are reshaped to
               match the dimensions of flatPlateRegion
93          pressure = reshape(pressure, [Opt.NX, Opt.NY]);
94          tau_xx = reshape(tau_xx, [Opt.NX, Opt.NY]);
95
96          forceX = (pressure .* flatPlateRegion) - (tau_xx .*
                flatPlateRegion);
97          dragForceX = dragForceX + sum(forceX(:));
98
99          % Check steady state convergence and simulation
               viscosity
100         if mod(t, Opt.ncheck) == 0
101             % Steady state convergence check
102             fSum = sum(sum(R+B, 1)); % Faster to operate on
                    the column first
103             maxDiff = max(max(max(abs(R(:)-Rold(:)), [], 1)
                   , [], 2), max(max(abs(B(:)-Bold(:)), [], 1),
                    [], 2)); % Faster to operate on the column
                   first
104             meanDiff = max(mean(mean(abs(R(:)-Rold(:)), 1),
                    2), mean(mean(abs(B(:)-Bold(:)), 1), 2)); %
                    Faster to operate on the column first
105
106             % Simulation viscosity
107             phi = 0.9999999999;
108             colorFieldNorm = (rhoR - rhoB) ./ (rhoR + rhoB)
                   ;
109             test = 1;
110
111             while (1)
112                 inIndex = colorFieldNorm >= phi;
```

```
113                 outIndex = colorFieldNorm <= -phi;
114                 if nnz(inIndex) <= 0 || nnz(outIndex) <= 0
115                     phi = phi - 0.000000000000000009 * test
                            ;
116                     test = 10 * test;
117                 else
118                     break;
119                 end
120             end
121
122             rhoRin = mean(sum(R(inIndex, :), 2));
123             Pin = 3/5 * rhoRin * (1 - Opt.alphaR);
124
125             rhoBin = mean(sum(B(outIndex, :), 2));
126             Pout = 3/5 * rhoBin * (1 - Opt.alphaB);
127
128             nuMean = 0.5 * (Opt.nuR + Opt.nuB);
129             omegaMean = 1 / (3 * nuMean + 0.5);
130
131             if strcmp(Opt.gradient, 'isotropic')
132                 sigmaPredict = 2/9 * ((1 + 1/Opt.gamma) *
                        Opt.rhoR / 2) / omegaMean * (Opt.aR +
                        Opt.aB);
133             else
134                 sigmaPredict = 4/3 * ((1 + 1/Opt.gamma) *
                        Opt.rhoR / 2) / omegaMean * (Opt.aR +
                        Opt.aB);
135             end
136             sigmaLaplace = (Pin - Pout) * Opt.R;
137
138             maxVel = max(sqrt(ux.^2 + uy.^2));
139
140             % Display results periodically
141             if mod(t / Opt.ncheck, 30) == 0 || t == Opt.
                    ncheck
```

```
142                 disp(['|  time step |              toc|
                                fSum|             maxDiff|
                              meanDiff|          Gamma|
                            A|          rhoR|              R|
                              sigmaPredict|
                       MaxVelocity|        Error(%)|'])
143             end
144
145             disp(sprintf('| %11.u  |  %12.5e   |   %12.5e
                   |  %12.5e  | %12.5e   |   %12.5e  |  %12.5e
                    | %12.5e   |   %12.5e  |  %12.5e  | %12.5e
                     |   %12.5e  |', t, toc, fSum, maxDiff,
                   meanDiff, Opt.gamma, Opt.aR, Opt.rhoR, Opt.R
                   , sigmaPredict, maxVel, abs(sigmaLaplace -
                   sigmaPredict) / sigmaPredict * 100))
146
147             % Update color field for visualization
148             v = (sum(R, 2) - sum(B, 2)) ./ (sum(R, 2) + sum
                   (B, 2));
149             %plotclr(x, y, v, 'square');
150         end
151     end
152
153     % Calculate final coefficient of drag
154     A = Opt.plateThickness * plateLength; % Reference area
155     Cd = 2 * dragForceX / (Opt.rhoR * Opt.
           supersonicVelocity^2 * A);
156     disp(['Final Coefficient of Drag (Cd): ', num2str(Cd)])
           ;
157
158     % Calculate compressibility correction factor if needed
159
160     disp('Stop main loop.....')
161     fid = fopen(['results','.txt'], 'a');
162     fprintf(fid, '%11.u  %11.u  %12.5e  %12.5e  %12.5e
           %12.5e  %12.5e  %12.5e  %12.5e  %12.5e %12.5e
```

```
        %12.5e   %12.5e \r\n ', Opt.ID, t, toc, fSum,
        maxDiff, meanDiff, Opt.gamma, Opt.aR, Opt.rhoR, Opt.
        R, sigmaPredict, maxVel, abs(sigmaLaplace -
        sigmaPredict) / sigmaPredict * 100);
    fclose(fid);
end
```

A.3.4 Frequency initialisation

```
function FEQ = get_FeqMultiPhase(rho, ux, uy, alpha, w, c)
    % Initialize the equilibrium distribution function
       matrix
    FEQ = zeros(size(rho, 1), 9);

    % Calculate squared velocity magnitude
    u_squ = ux.^2 + uy.^2;

    % Compute equilibrium distribution for the zero
       velocity (k=1)
    FEQ(:, 1) = rho .* (alpha - 2/3 * u_squ);

    % Calculate equilibrium distributions for the moving
       velocities (k=2 to 9)
    for k = 2:9
        c_dot_u = c(1, k) * ux + c(2, k) * uy;
        FEQ(:, k) = rho .* ((1 - alpha) * w(k) + w(k) * (3
           * c_dot_u + 9/2 * c_dot_u.^2 - 3/2 * u_squ));
    end
end
```

A.3.5 Obstacle Matrix for meshing

```
function obstacleMatrix = createObstacleMatrix(lattice)
    % CreateObstacleMatrix: Generates an obstacle matrix
       using lattice.XYZ for obstacle dimensions.
```

```
3       % Input: lattice - Struct containing fields NX, NY, and
            XYZ defining the grid and obstacle properties.
4
5       % Extract lattice dimensions and obstacle properties
6       NX = lattice.NX;
7       NY = lattice.NY;
8
9       % Extract obstacle dimensions from lattice.XYZ
10      plateLength = lattice.XYZ.plateLength;
11      plateThickness = lattice.XYZ.plateThickness;
12      platePositionY = lattice.XYZ.platePositionY;
13
14      % Validate input parameters
15      assert(plateLength <= NX, 'Plate length exceeds domain
            size (NX)');
16      assert(platePositionY >= 1 && platePositionY <= NY, '
            Plate position (Y) is out of domain bounds');
17      assert(plateThickness <= NY, 'Plate thickness exceeds
            domain size (NY)');
18
19      % Initialize obstacle matrix
20      obstacleMatrix = zeros(NY, NX); % Note: NY x NX for
            consistent row-major order
21
22      % Calculate obstacle region coordinates
23      startX = floor((NX - plateLength) / 2) + 1;
24      endX = startX + plateLength - 1;
25      startY = platePositionY - floor(plateThickness / 2);
26      endY = startY + plateThickness - 1;
27
28      % Ensure startY and endY are within bounds
29      startY = max(startY, 1);
30      endY = min(endY, NY);
31
32      % Set obstacle region to 1 (solid)
33      obstacleMatrix(startY:endY, startX:endX) = 1;
```

```
34 end
```

A.3.6 Collision modelling between cells

```
1  function [R, B] = collisionMultiPhase(R, B, siteNeighbour,
       w, c, upwindOrientation, Opt)
2      % Calculate first and second moments for both phases
3      rhoR = sum(R, 2);
4      rhoB = sum(B, 2);
5      rho = rhoR + rhoB;
6
7      ux = (sum(R(:, [3, 4, 5]), 2) - sum(R(:, [7, 8, 9]), 2)
           + sum(B(:, [3, 4, 5]), 2) - sum(B(:, [7, 8, 9]), 2)
         ) ./ rho;
8      uy = (sum(R(:, [2, 3, 9]), 2) - sum(R(:, [5, 6, 7]), 2)
           + sum(B(:, [2, 3, 9]), 2) - sum(B(:, [5, 6, 7]), 2)
         ) ./ rho;
9
10     % Equilibrium distributions for both phases
11     FEQ_R = get_FeqMultiPhase(rhoR, ux, uy, Opt.alphaR, w,
         c);
12     FEQ_B = get_FeqMultiPhase(rhoB, ux, uy, Opt.alphaB, w,
         c);
13
14     % LBGK Collision operator
15     R = R - Opt.omegaR * (R - FEQ_R);
16     B = B - Opt.omegaB * (B - FEQ_B);
17
18     % Apply bounce-back condition for obstacle cells
19     for dir = 1:9
20         oppositeDir = mod(dir + 4, 9);
21         if oppositeDir == 0
22             oppositeDir = 9;
23         end
24         % Apply bounce-back for R and B
25         obstacleIndices = find(Opt.ObstacleMatrix);
```

```
26          R(obstacleIndices, dir) = R(obstacleIndices,
               oppositeDir);
27          B(obstacleIndices, dir) = B(obstacleIndices,
               oppositeDir);
28      end
29
30      % Color gradient evaluation
31      if strcmp(Opt.gradient, 'isotropic')
32          gradientWeight = [0, 1/3, 1/12, 1/3, 1/12, 1/3,
               1/12, 1/3, 1/12];
33      else
34          gradientWeight = [0, 1, 1, 1, 1, 1, 1, 1, 1];
35      end
36      colorGrad_x = 0;
37      colorGrad_y = 0;
38      rhoDiff = rhoR - rhoB;
39
40      for i = 2:9
41          colorGrad_x = colorGrad_x + gradientWeight(i) * c
               (1, i) * rhoDiff(siteNeighbour(:, i));
42          colorGrad_y = colorGrad_y + gradientWeight(i) * c
               (2, i) * rhoDiff(siteNeighbour(:, i));
43      end
44
45      % Two-phase perturbation
46      b = [-4/27, 2/27, 5/108, 2/27, 5/108, 2/27, 5/108,
           2/27, 5/108];
47      colorGradNorm = sqrt(colorGrad_x.^2 + colorGrad_y.^2);
48      colorGradNorm_squ = colorGradNorm.^2;
49
50      for i = 1:9
51          c_dot_colorGrad = c(1, i) * colorGrad_x + c(2, i) *
               colorGrad_y;
52          temp = colorGradNorm .* (w(i) .* c_dot_colorGrad.^2
               ./ colorGradNorm_squ - b(i));
53          temp(~isfinite(temp)) = 0; % Handle division by 0
```

```
54          R(:, i) = R(:, i) + Opt.aR / 2 * temp;
55          B(:, i) = B(:, i) + Opt.aB / 2 * temp;
56      end
57
58      % Recoloring
59      rho = rhoR + rhoB;
60      FEQ_N = zeros(size(rhoR, 1), 9);
61      FEQ_N(:, 1) = rhoR * Opt.alphaR + rhoB * Opt.alphaB;
62      for k = 2:9
63          FEQ_N(:, k) = rhoR .* ((1 - Opt.alphaR) ./ w(k)) +
               rhoB .* ((1 - Opt.alphaB) ./ w(k));
64      end
65
66      for i = 1:9
67          c_dot_colorGrad = c(1, i) * colorGrad_x + c(2, i) *
               colorGrad_y;
68          temp = sqrt(c(1, i)^2 + c(2, i)^2) .* colorGradNorm
               ;
69          cosPHI = c_dot_colorGrad ./ temp;
70          cosPHI(~isfinite(cosPHI)) = 0; % Handle division by
               0
71
72          temp = Opt.beta * rhoR .* rhoB .* (cosPHI) ./ rho
               .^2;
73          RBsum = R(:, i) + B(:, i);
74          R(:, i) = RBsum .* rhoR ./ rho + temp .* FEQ_N(:, i
               );
75          B(:, i) = RBsum .* rhoB ./ rho - temp .* FEQ_N(:, i
               );
76      end
77
78      % Swapping R and B for correct streaming direction
79      R = R(:, upwindOrientation);
80      B = B(:, upwindOrientation);
81  end
```

A.3.7 Streaming function

```
function F = streaming(F, siteNeighbour, ObstacleMatrix)
    % Get the dimensions of the grid
    [NX, NY] = size(ObstacleMatrix);

    % Stream distribution functions normally
    temp = F(:, 2); F(:, 2) = F(siteNeighbour(:, 6), 6); F(siteNeighbour(:, 6), 6) = temp;
    temp = F(:, 3); F(:, 3) = F(siteNeighbour(:, 7), 7); F(siteNeighbour(:, 7), 7) = temp;
    temp = F(:, 4); F(:, 4) = F(siteNeighbour(:, 8), 8); F(siteNeighbour(:, 8), 8) = temp;
    temp = F(:, 5); F(:, 5) = F(siteNeighbour(:, 9), 9); F(siteNeighbour(:, 9), 9) = temp;

    % Handle bounce-back condition for all directions
    for dir = 1:9
        % Find the opposite direction for each lattice direction
        oppositeDir = mod(dir + 4, 9);
        if oppositeDir == 0
            oppositeDir = 9;
        end

        % Apply bounce-back condition for cells adjacent to obstacles
        % Get the neighboring indices for the current direction
        neighbourIndices = siteNeighbour(:, dir);

        % Convert linear indices to 2D
        [i, j] = ind2sub([NX, NY], neighbourIndices);
        % Ensure i and j are within bounds
        validIndices = i > 0 & i <= NX & j > 0 & j <= NY;
        i = i(validIndices);
        j = j(validIndices);
```

```
29
30          % Convert back to linear indices for F
31          obstacleIndices = sub2ind([NX, NY], i, j);
32
33          % Initialize as all false
34          isObstacle = false(size(F, 1), 1);
35          % Mark obstacle locations
36          isObstacle(obstacleIndices) = ObstacleMatrix(
                obstacleIndices);
37
38          % Apply bounce-back condition by reversing the flow
                direction for cells adjacent to the obstacle
39          F(isObstacle, dir) = F(isObstacle, oppositeDir);
40      end
41  end
```