COMPUTATION OF VISCOUS FLOWS OVER FLAPPING AIRFOILS AND PARALLEL OPTIMIZATION OF FLAPPING PARAMETERS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

MUSTAFA KAYA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

THE DEPARTMENT OF AEROSPACE ENGINEERING

JULY 2003

Approval of the Graduate School of Natural and Applied Sciences.

Prof. Dr. Tayfur Öztürk
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Nafiz Alemdaroğlu
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. İsmail H. Tuncer
Supervisor

Examining Committee Members

Prof. Dr. İ. Sinan Akmandor _____

Prof. Dr. Nafiz Alemdaroğlu _____

Prof. Dr. Zafer Dursunkaya _____

Prof. Dr. Max Platzer _____

Assoc. Prof. Dr. İsmail H. Tuncer _____

ABSTRACT

COMPUTATION OF VISCOUS FLOWS OVER FLAPPING AIRFOILS AND PARALLEL OPTIMIZATION OF FLAPPING PARAMETERS

Kaya, Mustafa

M.S., Department of Aerospace Engineering

Supervisor: Assoc. Prof. Dr. İsmail H. Tuncer

July 2003, 95 pages

Airfoils flapping in pitch and plunge are studied, and the flapping motion parameters are optimized to maximize thrust generation and the efficiency of the thrust generation. Unsteady viscous flowfields over flapping airfoils are computed on overset grids using a Navier-Stokes solver. Computations are performed in parallel using *Parallel Virtual Machine* library routines in a computer cluster. A single flapping airfoil and dual airfoils flapping in a biplane configuration are considered. A gradient based optimization algorithm is employed. The thrust production and the efficiency of the thrust production are optimized with respect to flapping parameters; the plunging and pitching amplitudes, the flapping frequency, and the phase shift between the pitch and plunge motions. It is observed that thrust generation of flapping airfoils strongly depends on the phase shift and high thrust values may be obtained at the expense of reduced efficiency. For a high efficiency in thrust generation, the effective angle of attack of the airfoil is reduced and large scale vortex formations at the leading edge are prevented. At a fixed reduced flapping frequency of 1, a single flapping airfoil in pitch and plunge motion

produces the maximum average thrust coefficient of 1.41 at the plunge amplitude of 1.60, the pitch amplitude of 23.5^o, and the phase shift of 103.4^o whereas the maximum efficiency of 67.5% is obtained at the plunge amplitude of 0.83, the pitch amplitude of 35.5^o and the phase shift of 86.5^o.

Keywords: Flapping Airfoils, Overset Grids, Parallel Processing, Optimization, Reynolds-Averaged Navier-Stokes Equations

ÖZ

ÇIRPAN KANAT KESİTLERİ ÜZERİNDEKİ VİSKOS AKIŞLARIN HESAPLANMASI VE ÇIRPMA PARAMETRELERİNİN PARALEL OLARAK ENİYİLEŞTİRİLMESİ

Kaya, Mustafa

Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü

Tez Yöneticisi: Assoc. Prof. Dr. İsmail H. Tuncer

Temmuz 2003, 95 sayfa

Bu çalışmada, yunuslama ve dalma şeklinde çırpma hareketi yapan kanat kesitleri incelenmiş ve çırpma hareketi parametreleri maksimum itki üretimi ve itki üretiminin verimi için eniyileştirilmiştir. Çırpan kanat kesitleri üzerindeki zamana bağlı viskos akış alanları bir Navier-Stokes çözücü kullanılarak üstüste binen ağ sistemi ile hesaplanmıştır. Hesaplamalar *Parallel Virtual Machine* kitaplık yordamları kullanılarak bir bilgisayar öbeğinde paralel olarak gerçekleştirilmiştir. Tek olarak ve üst üste iki tane yerleştirilmiş çırpan kanat kesitleri incelenmiştir. Gradyant tabanlı eniyileştirme algoritması kullanılmıştır. İtki üretimi ve itki üretiminin verimi çırpma hareketi parametlerine (yunuslama ve dalma hareketlerinin genliği, çırpma frekansı ve yunuslama ile dalma hareketi arasındaki faz farkı) göre eniyileştirilmiştir. Çırpan kanat kesitleri ile itki üretiminin faz farkına oldukça bağlı olduğu ve yüksek itki değerlerinin, düşük verim durumunda elde edilebileceği gözlenmiştir. Yüksek verime, kanat kesitinin etkin hücum açısı azaltıldığında ve hücum kenarında büyük girdap oluşumları engellendiğinde ulaşılmıştır. Değeri 1 olan indirgenmiş çırpma frekansında yunuslama ve dalma hareketi yapan tek bir çırpan kanat

kesiti, 1.60 dalma genliği, 23.5^o yunuslama genliği ve 103.4^o faz farkı değerlerinde, 1.41'lik bir maksimum ortalama itki katsayısı vermiştir. Değeri %67.5 olan maksimum verime ise, 0.83 dalma genliği, 35.5^o yunuslama genliği ve 86.5^o faz farkı değerlerinde ulaşılmıştır.

Anahtar Kelimeler: Çırpan Kanat Kesitleri, Üst Üste Binen Ağ Sistemi, Paralel İşlemler, Eniyileştirme, Reynolds-Ortalamalı Navier-Stokes Denklemleri

to Adem chosen to live in a rose garden

and

to my grandfather visiting me in my dreams

ACKNOWLEDGMENTS

I want to state my thanks to

My thesis supervisor *Assoc. Prof. Dr. İsmail Hakkı Tuncer* for his providing me with a drop of water from the ocean called *CFD*, causing me to have been getting much more thirsty second by second from that moment and of course for his valuable guidance with patience throughout this study, which was an excellent example to me for how an instructor should be.

My parents for smiling in their eyes, when saw their son that they have been bringing up as a gift for *the most beautiful country* in the world, trying to do something and work for that paradise.

My *brother* and *his wife, grandparents, aunts, uncles, cousins, nephews* and *nieces* for their love, admiration, prayers and remembering me always in day and night.

My instructors in the department for their answers to my questions and solutions to my problems, regardless of how silly and annoying they are.

My department for giving me a chance to be a graduate assistantship, which played a very important role during my studies.

My friends; *ones* over the world for carrying my heart with them wherever they go like the sunlight bringing love message of the Sun to the whole Earth, and *ones* in the department:
Aycan, *Bekir*, *Cengiz* and *Erdem* for being always with me to share and to laugh,
Çagatay, *Funda*, *Gizem*, *Gülkız*, *Özgür* and *Özhan* for their peerless friendship,
Demet for giving joy to my heart with her 'hello', for giving light to the day with her eyes, and for giving color to the night with her hair,
Karaca and Ybu for considering me as an elder brother to be respected,
Murat for making me get more intellectual with his knowledge and not hesitating to consult me when he needs help ◡̈ ,
Ömer for being such a friend that no word is able to express how the brotherhood between us has gained immortality,
Tarkan Hoca and *Varlık Hoca* for indicating what I should do to be both successful and cheerful.

and **her** for being heroine of my most beautiful poetries in which I offer her, collecting one by one, the stars in the sky as a bunch of flowers on which land the butterflies coming from the Moon, with pieces of pearl on the wings.

PREFACE

One has been researching the most convenient way, or in academic words, the optimum method of flying in the sky since the first day when men started to think that the birds were not, any more, to be considered as the only living beings able to fly. In spite of the fact that what is done, is nothing but a challenge to the birds, in all the steps and stops of the flying studies, men only imitated the birds. For this reason, I am really thinking if I should add an "unfortunately" in the beginning of my next sentence. This study takes also birds as an example and suggests that, in micro air vehicles, the flapping-wings should be responsible of the thrust.

I think I should finish chattering since each letter here delays a second more your meeting with an interesting aerodynamic story, the heros of which are the *CFD* tools of the present days.

You can report all errors and comments to the following e-mail address:

Mustafa Kaya
mkaya@ae.metu.edu.tr

All comments are welcome.

M.K.

METU, Ankara

July 2003

TABLE OF CONTENTS

LIST OF TABLES

TABLE

LIST OF FIGURES

FIGURES

LIST OF SYMBOLS

ROMAN SYMBOLS

a	Speed of sound
c	Airfoil chord length (reference length)
C_d	Drag coefficient
C_p	Pressure coefficient
C_t	Time averaged thrust coefficient
e	Total energy per unit volume
f	Frequency in Hertz
$\hat{\mathbf{F}}$	Inviscid flux vector in the x-direction
$\hat{\mathbf{G}}$	Inviscid flux vector in the z-direction
h	Plunge position normalized with c
h_0	Plunge amplitude normalized with c
J	Jacobian of the transformation
k	Reduced frequency, $\frac{\omega c}{U_\infty}$
M	Mach Number
p	Pressure
$\hat{\mathbf{Q}}$	Conservative flow variables
Re	Reynolds Number
$\hat{\mathbf{S}}$	Thin layer approximation of the viscous fluxes in ζ-direction
t	Non-dimensional time, $\frac{t' U_\infty}{c}$
t'	Time
T	Temperature
u	Velocity component in x-direction
U_∞	Freestream velocity (reference velocity)
w	Velocity component in z-direction
x, z	Spatial coordinates in Cartesian system
y_0	Mean distance between airfoils in biplane configuration

GREEK SYMBOLS

α	Incidence angle
α_0	Pitch amplitude
η	Propulsive efficiency
γ	Specific heat ratio
δ	Central difference operator
Δ	Forward difference operator
μ	Dynamic viscosity
ρ	Density
ξ, η	Spatial coordinates in computational domain
$\xi_x, \xi_z,$ η_x, η_z	Metrics of transformation
ω	Circular frequency of oscillation, $2\pi f$
ϕ	Phase shift between pitch and plunge oscillation
∇	Backward difference operator

SUBSCRIPTS

i	Node index for x-direction
k	Node index for z-direction
∞	Freestream value
ξ	Derivative with respect to ξ
ζ	Derivative with respect to ζ

SUPERSCRIPTS

k	Optimization step
n	Time step
p	Newtonian sub-iterations

CHAPTER 1

INTRODUCTION

1.1 Introduction

Bird flight has been admired along the history of humans. Folkloric stories about men flying like a bird have been narrated for many centuries. The phrase *flying like a bird* is a fascinating source of imagination for human beings but, from the scientific point of view, the main conclusion to be extracted from this phrase is the fact that the bird flight is realized by flapping wings. Flapping wings are utilized by nearly a million species of insects and 10000 vertebrate species[1]. However, this common utilization of flapping wing in the nature is a concept still waiting to be explored by detailed scientific investigations.

Current research indicates that, based on the high flying performance of birds, insects and many sea creatures, flapping wing propulsion would be more efficient than the conventional propellers if applied to very small scale vehicles, so-called micro-air vehicles (MAVs) [2]. The propulsive efficiency of the flying species with flapping wings are around $65 - 75\%$ [2] whereas the propeller efficiency of a propeller-powered MAV with a fixed wing is about 50% [3, 4]. MAVs with wing spans of less than $15\,cm$, and flight speed of 30 to 60 km/hr are of current interest for military and civilian applications. A typical MAV might carry visual, acoustic, chemical or biological sensors that can be miniaturized so that the entire payload weighs less than 30 grams. These small vehicles considered to fly at very low speeds have a chord Reynolds number

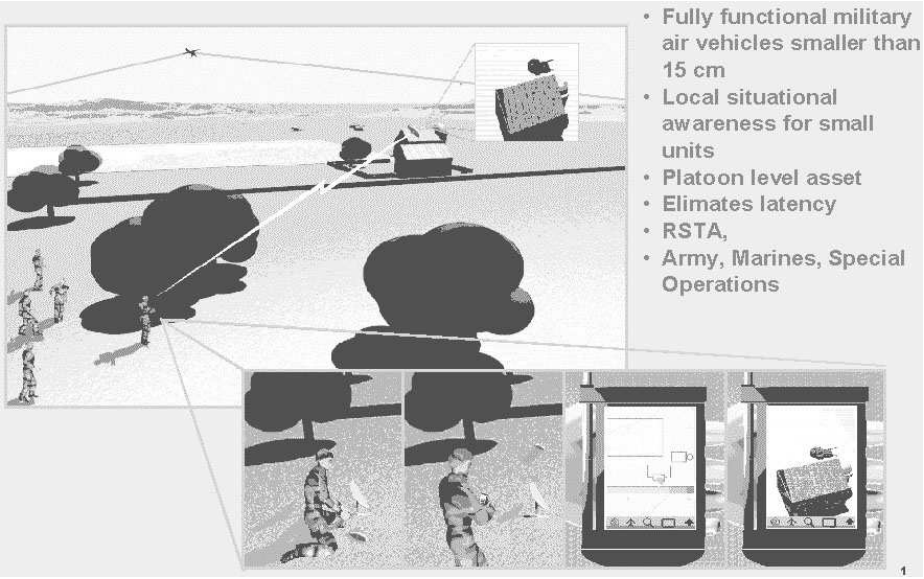


Figure 1.1: A typical MAV mission [5]

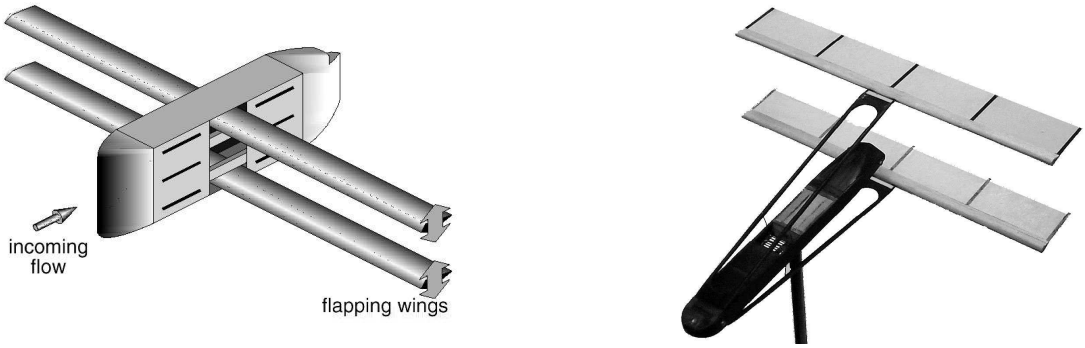


Figure 1.2: Flapping wings (Jones et al. [6])

ranging from 1000 to about 50000. Since the primary mission for MAVs is the surveillance, they are desired to have good maneuverability and sustained flight at low flight speeds. Currently, there is a wide spread research work on finding the most energy efficient airfoil adaptation and wing motion technologies capable of providing the required aerodynamic performance for a MAV flight.

For a long time, complexity of the study and also funding problems have kept the researchers away from investigating flapping motion. There has been neither sufficient effort for the development of numerical flow solvers suitable for the flight regimes at low Reynolds numbers nor enough experimental data to validate the numerical methods. The recent interest in the development of MAVs triggered the start of studies in flapping wing aerodynamics. Recent

experimental and computational studies carried out for development of MAVs, investigated the propulsive characteristics of single and dual flapping airfoils, and shed some light on the relationship among the produced thrust, the efficiency of the thrust production, the amplitude and the frequency of the oscillations, phase shift between the pitch and plunge motions, and the flow Reynolds number. Other important parameters are the camber of the airfoil and the chord change during the flapping motion, which are not considered in this thesis.

The primary objective of this work is to develop a parallel optimization algorithm to maximize the thrust generation and the efficiency of the thrust generation of flapping airfoils, and to asses the computational results by comparing them with recent experimental and numerical studies. To this end, an unsteady parallel Navier-Stokes solver on overset grids and a gradient based optimization algorithm will be employed.

1.2 Experimental Studies on Flapping Airfoils

Historically, design of a propulsion system for air vehicles by using flapping wings was an attractive topic up until the first decade of the last century[1, 7, 8]. In their studies published early in the 20^{th} century, Knoller[9] and Betz[10] are known to be the first to explain the bird's ability to generate a thrust force. They independently observed that a flapping wing generates an effective angle of attack, resulting in a normal-force vector with both lift and thrust components. This is called as *Knoller-Betz effect*. Katzmayr[11] performed some wind tunnel measurements and made the verification of the Knoller-Betz effect. The first explanation of the drag and thrust production of flapping wings is made by von Kármán and Burgers[12], according to the wake vortices as illustrated in Figure 1.3 and Figure 1.4 from Reference [20]. In the case of the drag production, it is observed that there is a row of vortices of clockwise rotation above the symmetry plane, and a row of vortices of counterclockwise rotation below the symmetry plane as shown in Figure 1.3. Note that the flow is from left to right. A velocity (momentum) deficit is induced on the centerline by the vortices, and this is the indication of drag. The vortex street shown in Figure 1.4 is obtained by oscillating the airfoil with a higher plunge amplitude as indicated in the reference. Here, the upper row of vortices rotates

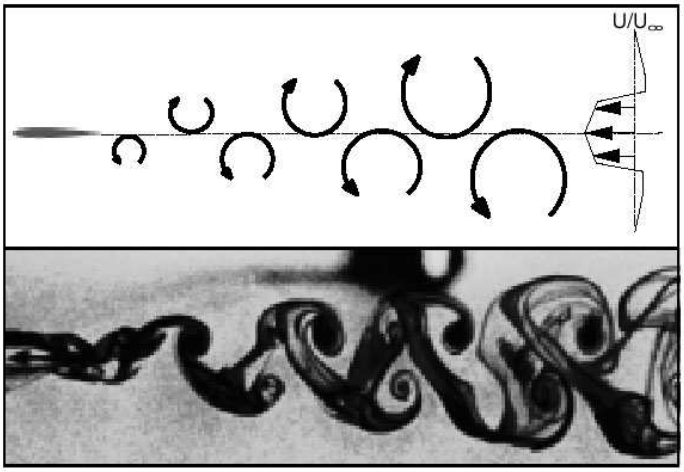


Figure 1.3: Drag indicative vortex street [20]

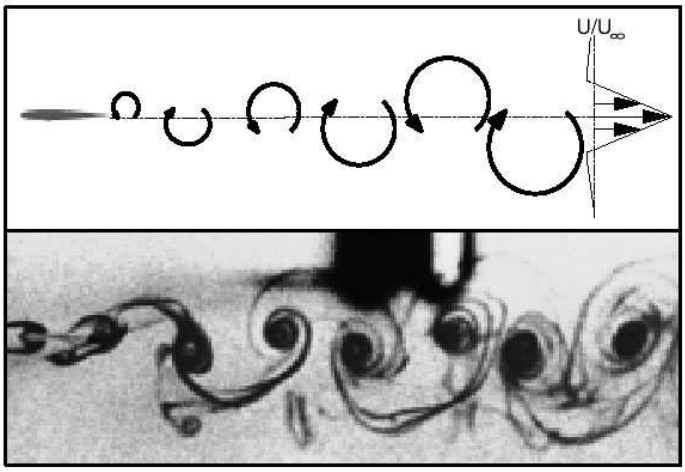


Figure 1.4: Thrust indicative vortex street [20]

counterclockwise, and the lower row rotates clockwise. For this case, the vortices induce a velocity (momentum) surplus on the centerline. This means that the thrust is produced[19].

Applying Theodorsen's (inviscid, incompressible, oscillatory, flat-plate) theory[13], Garrick[14] obtained results for sinusoidally plunging and/or pitching airfoils with a range of flapping frequency. The first experimental verification of Garrick predictions is provided by Silverstein and Joyner[15].

Recognizing the fact that some of the flapping energy is lost in the form of vorticity shed in the wake, Schmidt[16] proposed that a stationary wing be placed in the oscillatory wake of a flapping wing to take back some of the vortical energy lost by the flapping airfoil. Schmidth, as a result of his studies about flapping foils in 1940's and 50's, developed the *wave propeller* and

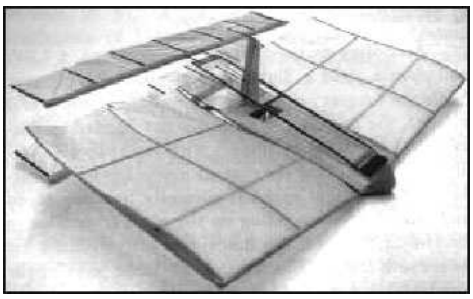


Figure 1.5: *MAV* with flapping wings [25]

demonstrated it on a catamaran boat. In 1982 DeLaurier and Harris[17] obtained experimental measurements of flapping wing propulsion.

More recently, conducted by Lai and Platzer[18] and Jones et al.[21], water tunnel flow visualization experiments on flapping airfoils have provided a considerable amount of information on the wake characteristics of thrust producing flapping airfoils. Anderson et al., in their experiments[22], also observed that the phase angle between pitch and plunge oscillations plays a significant role in maximizing the propulsive efficiency. The experimental studies by Jones et al.[6] and Platzer and Jones[23] exposed that two airfoils arranged in a biplane configuration and oscillating in counter-phase show significant benefits of thrust and propulsive efficiency compared to a single flapping airfoil (Figure 1.2). Jones et al.[24], in their recent numerical and experimental studies, observed the high influence of oscillation frequency and wing aspect ratio on the thrust produced by flapping wings in biplane configuration. Finally, Jones and Platzer[25] developed and flight tested a radio-controlled micro air vehicle propelled by two flapping wings (Figure 1.5).

1.3 Computational Studies on Flapping Airfoils

A precise analysis of flapping wings is not available due to its complexity. However, there have been a number of researchers who provided some progress on the analysis of the simplified problem. The common approach is to model the flapping motion as a sinusoidal oscillation in 2-D and the oscillation of airfoils is defined as a combination of pitch and plunge motions. A wide review of both aeronautical and biological literature about flapping motion is presented in Reference [1] with an inclusion of discussion on the kinematics of flapping wings.

Birnbaum, student of Prandtl, obtained an incompressible flow solution for flapping airfoils[26]. The first notable numerical predictions of the thrust force produced from flapping motion are given by Garrick[14]. In his report published in 1936, he presented results of the problem of airfoils flapping with a range of frequency in sinusoidally defined plunge and/or pitch.

Philips et al.[27] present an unsteady inviscid aerodynamic theory of rigid flapping wings in a forward flight. More recently, Vest and Katz[28] adapted the classical panel method to the flapping flight. Jones and Platzer[19] give also a numerical procedure to obtain the flow solutions of a single flapping airfoil by using an unsteady, two-dimensional panel code coupled with a boundary layer algorithm. The basis of a numerical method for unsteady aerodynamic computation around thin lifting and/or propulsive systems with arbitrary variable geometries, involving the velocity field is presented by Leroy and Devinant[29]. In their publication[30], Sunada and Ellington propose a new method for estimating the aerodynamic forces generated by flapping wings. In this method, the forces are explained by the added mass of vortex wake sheets.

Isogai et al.[31] carried out the Navier-Stokes computations for a single flapping airfoil. They calculated the thrust and the efficiency of the thrust generation for various combinations of the frequency and the phase shift, and gave a detailed analysis of the effects of the dynamic stall phenomena on the behaviors of the thrust and the efficiency of the thrust production. The numerical study of Isogai et al.[32] about aerodynamic performance of a dragonfly clarifies the fundamental mechanism of the hovering flight of a dragonfly. Another flow simulation of an insect is presented by Togashi et al.[33] in which, by solving the Navier-Stokes equations, they give the numerical simulation of the flow over a hornet in a forward flight. Szmelter and Żbikowski[34] propose a model for the aerodynamics of the flapping insect wing to provide insight for the design of flapping wing propulsion system for micro-air vehicles. In their model, flapping motion of an insect is considered to be a special combination of three motions: plunging, pitching and sweeping, which, with the assumption of periodicity, are represented by Fourier series.

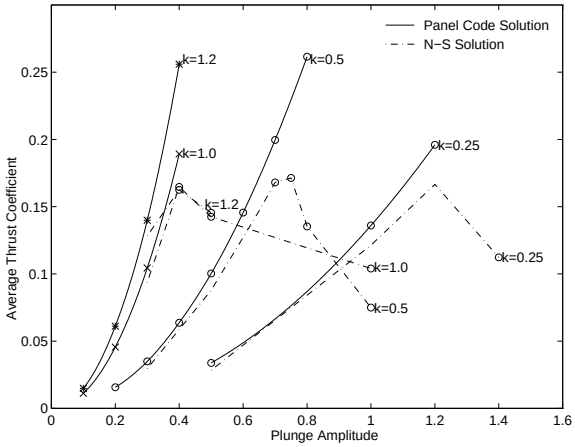


Figure 1.6: Parametric study of thrust [38]

Tuncer et al.[35, 36, 37, 38, 39] have performed the Navier-Stokes computations to explore the effect of flow separation on the thrust and the propulsive efficiency of a single flapping airfoil in combined pitch and plunge oscillations. In Reference [35, 37], the parametric studies of the thrust generation of a single flapping airfoil and a flapping/stationary airfoil combination are given and it is observed that flapping airfoils produce thrust, and the generated thrust and the propulsive efficiency are significantly increased in the case of flapping/stationary airfoil combination in tandem. In Reference [36], the Navier-Stokes solutions of the flows over flapping airfoils are computed by using moving overset grids, and the thrust generation due to flapping motion is observed. Tuncer et al.[38] present the computation of the dynamic stall boundaries of an airfoil flapping in either pure plunge or in combined motion of pitch and plunge. Figure 1.6 shows the average thrust coefficient values plotted versus the plunge amplitude with four different flapping frequencies. In their study, they showed that the dynamic stall is encountered as soon as the effective angle of attack of the flapping airfoil exceeds a specific value. Investigation of the thrust generation of a single airfoil making pure plunge motion and pitch/plunge motion combination is presented in Reference [39] and it is concluded that, under certain conditions, high thrust values with reasonably high propulsive efficiency may be obtained in combined pitch and plunge oscillations. As a continuation of these studies, the present thesis work aims at maximizing the thrust production of flapping airfoils by employing optimization algorithms.

1.4 Definition of the Research Problem

The recent numerical and experimental studies carried out parametrically point out the fact that the thrust production may be optimized in a given range of flow and motion parameters as shown in Figure 1.6 from Tuncer et al.[38]. The figure shows how the thrust of a single flapping airfoil changes with the plunge amplitude and the flapping frequency (denoted by the non-dimensional parameter $k = \frac{2\pi fc}{U_\infty}$). As seen from the figure, for each flapping frequency, the Navier-Stokes (*N-S*) computations predict a different plunge amplitude which maximizes the thrust. The variation of the plunge amplitude which maximizes the thrust for a given frequency implies the importance of the optimization of the flapping motion parameters for maximizing the thrust generation. The variables of the optimization may be the amplitudes of plunge and pitch motions, the flapping frequency and the phase angle between plunging and pitching. The average distance between two airfoils in a biplane configuration plays also an important role in increasing the thrust and the efficiency of the thrust production within the constrains.

The objective of the present study is to compute the unsteady, viscous flowfields over a single flapping airfoil and flapping airfoils in a biplane configuration to optimize the generated thrust and the efficiency of the thrust generation. The unsteady flowfields are computed in parallel using a Navier-Stokes solver on overset grids. For the optimization processes, a gradient based algorithm, steepest ascent method is employed. The gradient of the objective function needed by the optimization algorithm is also evaluated in parallel. The computed flowfields are then analyzed in terms of unsteady variation of aerodynamic loads, distributions of flow variables and unsteady particle traces.

1.4.1 Parallel Processing

Parallel processing can simply be described as a method of using more than one computer for the solution of a single problem. Parallelism is generally preferred if the problem consists of large systems of equations to be solved in large domains as in *CFD* applications. CFD tools, especially, unsteady flow solvers require computers with fast processors to perform a large number of calculations within reasonable time and also quite high memory to process great

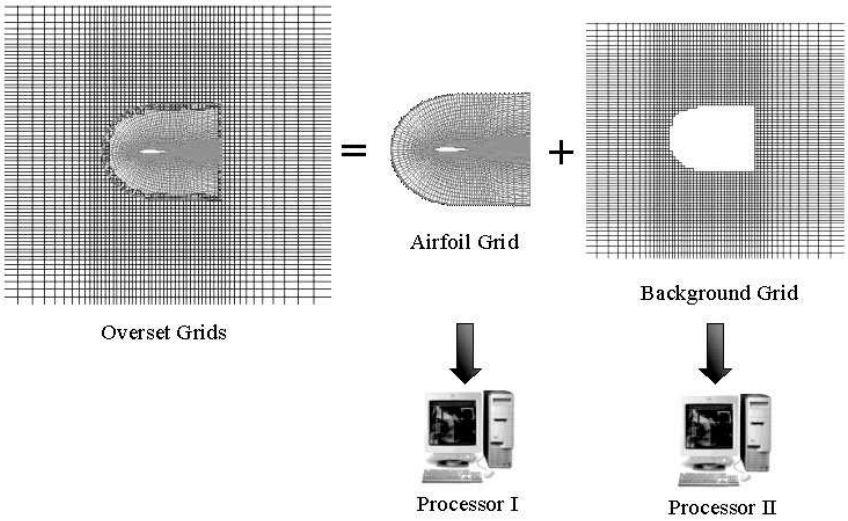


Figure 1.7: Domain decomposition for parallel processing

amount of data. However, there is a limit for a computer to meet the above needs, which is generally unsatisfactory. It is possible to overcome this computer limitation by making use of parallel processing so that the computation time to solve a problem is significantly reduced by having several operations, each being a part of the original computation, performed at the same time.

With the recent decrease in the cost of a computer and its accessories, it has become feasible to deploy a cluster of computers as a parallel computing environment. Therefore, parallelism has become a useful approach for attaining very high computational speeds in the last twenty years.

There are mainly two types of parallel computing environments, which are the *shared-memory* and *distributed-memory* systems. In the shared-memory system, more than one processor share the same memory and may perform computational tasks on the parts of the data structure stored in the common memory by either the compiler directives or with the help of new constructs in the programming language. The individual tasks are then executed simultaneously. Task decomposition is possible if there is no data dependency between the tasks to be decomposed from the original computation. The shared-memory approach is generally employed with multi-processor architectures.

The distributed-memory parallel system has a different topology than the shared-memory system. In the distributed-memory system, each processor has its own private memory and may perform computations on its own data which is a subset of the main task, and communicates with other processors for sharing common data. A networked computer cluster is the most common environment to construct a distributed-memory system. In this thesis, a distributed-memory parallel algorithm based on domain decomposition is implemented in a master-worker paradigm. First, the domain of computation and the given problem to be solved on this domain are decomposed into subdomains and subproblems. Then, the solution on each subdomain is assigned to a processor as shown in Figure 1.7. Finally, the results are combined to give a complete solution to the original problem. All the parallel computations in this thesis are performed in a cluster of dual-processor computers running on `Linux` operating system. Detailed information about the parallel algorithms is given in Chapter 4.

CHAPTER 2

NUMERICAL METHOD

2.1 Introduction

In this chapter, the numerical methods employed to compute the unsteady viscous flowfields around flapping airfoils are described. The unsteady flowfields are computed solving the Reynolds averaged Navier-Stokes equations on overset grids. The computations on each subgrid are performed in parallel. `PVM` message passing library routines are used in the parallel solution algorithm. The computed unsteady flowfields are analyzed in terms of instantaneous distributions of flow variables, the variation of aerodynamic loads in time and unsteady particle traces.

2.2 Navier-Stokes Solver

The finite difference formulation of the Reynolds averaged Navier-Stokes equations are solved on a computational domain discretized with an overset grid system. The overset grid system consists of `C`-type airfoil and Cartesian background grids, as shown in Figure 2.1.

The strong conservation-law form of the 2-D, thin-layer, Reynolds averaged Navier-Stokes equations is solved on each subgrid. The governing equations in a curvilinear coordinate system (ξ, ζ), are given as follows:

$$\partial_t \hat{\mathbf{Q}} + \partial_\xi \hat{\mathbf{F}} + \partial_\zeta \hat{\mathbf{G}} = Re^{-1} \partial_\zeta \hat{\mathbf{S}} \tag{2.1}$$

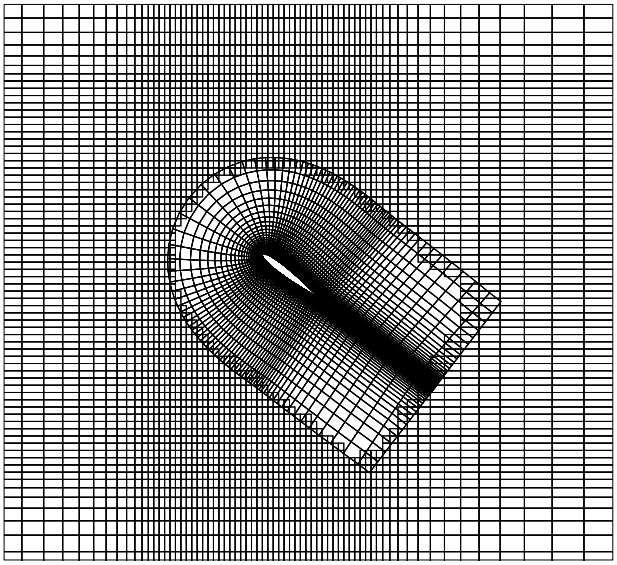


Figure 2.1: Overset grid system

where $\hat{\mathbf{Q}}$ is the vector of conservative variables, $\frac{1}{J}(\rho, \rho u, \rho w, e)$, $\hat{\mathbf{F}}$ and $\hat{\mathbf{G}}$ are the convective flux vectors, and $\hat{\mathbf{S}}$ is the thin layer approximation of the viscous fluxes in the ζ direction normal to the airfoil surface. The convected fluxes are evaluated using the third order Osher's[40, 41] upwind biased flux difference splitting scheme.

$$\hat{\mathbf{F}} = \frac{1}{J}\begin{pmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho w U + \xi_z p \\ (e+p)U - \xi_t p \end{pmatrix} \qquad \hat{\mathbf{G}} = \frac{1}{J}\begin{pmatrix} \rho W \\ \rho u W + \zeta_x p \\ \rho w W + \zeta_z p \\ (e+p)W - \zeta_t p \end{pmatrix} \tag{2.2}$$

$$\hat{\mathbf{S}} = \frac{1}{J}\begin{pmatrix} 0 \\ \mu m_1 u_\zeta + \frac{\mu}{3} m_2 \zeta_x \\ \mu m_1 w_\zeta + \frac{\mu}{3} m_2 \zeta_z \\ \mu m_1 m_3 + \frac{\mu}{3} m_2 m_4 \end{pmatrix} \tag{2.3}$$

where,

$$\begin{aligned} m_1 &= {\zeta_x}^2 + {\zeta_z}^2 & m_2 &= \zeta_x u_\zeta + \zeta_z w_\zeta \\ m_3 &= \tfrac{u^2+w^2}{2} + (\gamma-1)^{-1} Pr^{-1} \tfrac{\partial a^2}{\partial \zeta} & m_4 &= \zeta_x u + \zeta_z w \end{aligned} \tag{2.4}$$

and U and W are the contravariant velocity components given by;

$$U = u\xi_x + w\xi_z + \xi_t \qquad W = u\zeta_x + w\zeta_z + \zeta_t \tag{2.5}$$

x and z are the spatial coordinates in Cartesian system and J is the metric Jacobian given as;

$$J^{-1} = x_\xi z_\zeta - x_\zeta z_\xi \tag{2.6}$$

The pressure is related to density and total energy through the equation of state for an ideal gas:

$$p = (\gamma - 1)[e - \rho(u^2 + w^2)/2] \tag{2.7}$$

In the equations 2.2 to 2.7, all lengths are non-dimensionalized by the airfoil chord length c. ρ is the density non-dimensionalized by the freestream density ρ_∞; u and w are the Cartesian velocity components in the physical domain, which are non-dimensionalized by the freestream speed of sound a_∞; e is the total energy per unit volume non-dimensionalized by $\rho_\infty a_\infty^2$. Re is the Reynolds number, Pr is the Prandtl number and γ is the specific heat ratio.

Turbulent flows may be computed employing the *Baldwin-Lomax*[42] or *Spalart-Allmaras*[43] turbulence models with the assumption of fully turbulent flow.

2.2.1 Evaluation of Thrust and Propulsive Efficiency

For a harmonically flapping airfoil, thrust (negative drag) production shows a periodic behavior as shown in Figure 2.2. Thrust of a flapping airfoil is therefore defined in terms of an average value over a flapping period. Along the unsteady flow computations, the periodic behavior of the solution is established as the average thrust over a flapping period converges to a constant value within a given accuracy. Definitions of the average thrust coefficient and the propulsive efficiency are given as follows:

$$C_t = -\frac{1}{t_2 - t_1}\int_{t_1}^{t_2} C_d \, dt \tag{2.8}$$

$$\eta = \frac{C_t \, U_\infty}{\dot{W}} \tag{2.9}$$

where $\dot{W}$ is the input power required to maintain the flapping motion and defined as:

$$\dot{W} = \frac{1}{t_2 - t_1}\int_{t_1}^{t_2} (C_l \, V_{plunge} + C_m \, \omega_{pitch}) \, dt \tag{2.10}$$

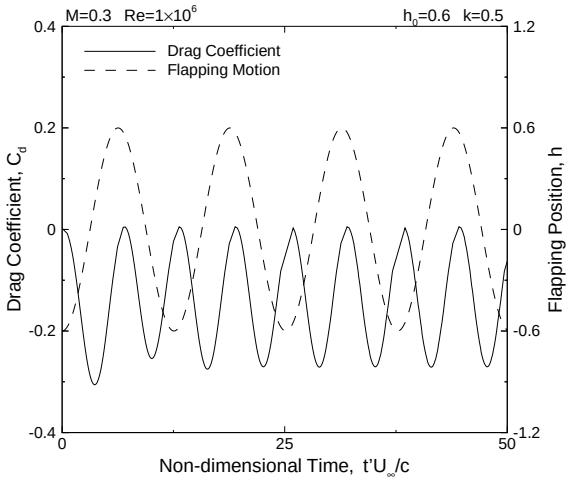


Figure 2.2: Drag history along flapping periods

t_1 and t_2 are the start and end points of the last flapping period, respectively. C_d and C_l are the drag and lift coefficients, respectively. C_m is the pitching moment coefficient about the pitching center. Note that, in this thesis work, viscous forces are neglected in the calculation of C_l, C_d and C_m. V_{plunge} is the plunge velocity of the airfoil and ω_{pitch} is the angular velocity of the airfoil due to the pitching motion.

2.2.2 Turbulence Models

Turbulent flow computations are performed using the Baldwin-Lomax and Spalart-Allmaras models. First one is an algebraic turbulence model and the second one is a one-equation turbulence model. The formulations of both models and the closure coefficients used in this thesis are given in the following sections.

2.2.2.1 Baldwin-Lomax Model

The Baldwin-Lomax model[42] is based on the local flow properties and does not require the calculation of some boundary layer properties like the boundary layer thickness or the velocity thickness. However, since the model requires the directional normal to the body, the turbulence computations are only performed on the airfoil grid(s), that is, the flow in the background grid is assumed to be laminar when using the Baldwin-Lomax model. The eddy-viscosity formulations for the inner and outer viscous layers, used in this thesis, are as follows:

Inner Layer:

$$\nu_{T_i} = l_{mix}^2 |\omega| \tag{2.11}$$

$$l_{mix} = \kappa y \left[1 - e^{-\frac{y}{A_o^+}\left(\sqrt{\rho\frac{|\omega|}{\mu}}\right)_{y=0}} \right] \tag{2.12}$$

l_{mix} is the length scale of turbulence and y is the distance normal to the solid body.

Outer Layer

$$\nu_{T_o} = \alpha C_{cp} F_{wake} F_{Kleb}(y\ ;\ y_{max}/C_{kleb}) \tag{2.13}$$

$$F_{wake} = \min[y_{max}F_{max}\ ;\ C_{wk}y_{max}U_{dif}^2/F_{max}] \tag{2.14}$$

$$F_{max} = \frac{1}{\kappa}\left[\max_y\left(l_{mix}|\omega|\right)\right] \tag{2.15}$$

$$F_{Kleb} = \left[1 + 5.5\left(C_{Kleb}\frac{y}{y_{max}}\right)^6\right]^{-1} \tag{2.16}$$

where y_{max} is the value of y at which $l_{mix}|\omega|$ is maximum.

Closure Coefficients

$$\begin{array}{lll} \kappa = 0.40 & \alpha = 0.0168 & A_o^+ = 26 \\ C_{cp} = 1.6 & C_{Kleb} = 0.3 & C_{wk} = 0.25 \end{array} \tag{2.17}$$

The function F_{Kleb} is Klebanoff's intermittency function and $|\omega|$ is the magnitude of the vorticity vector.

$$|\omega| = \left|\frac{\partial w}{\partial x} - \frac{\partial u}{\partial z}\right| \tag{2.18}$$

and U_{dif} is defined as

$$U_{dif} = \left(\sqrt{u^2+w^2}\right)_{max} - \left(\sqrt{u^2+w^2}\right)_{y=y_{max}} \tag{2.19}$$

The turbulent eddy viscosity values switches from the inner formulation to the outer formulation along the boundary layer at the location where $\nu_{T_i} > \nu_{T_o}$.

2.2.2.2 Spalart-Allmaras Model

The Spalart-Allmaras model[43] is compatible with any type of grid structure and 2-D or 3-D Navier-Stokes solvers. Although the model allows the transition point localization, in this thesis work, all the computed flowfields are assumed to be fully turbulent when the Spalart-Allmaras

model is used. Eqn. 2.21 is solved by an implicit algorithm based on a second order finite difference method. The hole points in the background grid are excluded from the solution (See Section 2.5.1.2). In this thesis work, the following formulations of the model are used:

Kinematic Eddy Viscosity

$$\nu_T = \tilde{\nu} f_{v1} \tag{2.20}$$

Eddy Viscosity Equation

$$\frac{\partial \tilde{\nu}}{\partial t} + U_j \frac{\partial \tilde{\nu}}{\partial x_j} = c_{b1}\tilde{S}\tilde{\nu} - c_{\omega 1} f_\omega f_{v3} \left(\frac{\tilde{\nu}}{d}\right)^2 + \frac{1}{\sigma}\frac{\partial}{\partial x_k}\left[(\nu + \tilde{\nu})\frac{\partial \tilde{\nu}}{\partial x_k}\right] + \frac{c_{b2}}{\sigma}\frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k} \tag{2.21}$$

d in the equation is the distance from the field point to the closest body surface point.

Closure Coefficients and Auxiliary Relations

$$c_{b1} = 0.1355 \qquad c_{b2} = 0.622 \qquad c_{v1} = 7.1 \qquad \sigma = 2/3 \tag{2.22}$$

$$c_{\omega 1} = \tfrac{c_{b1}}{\kappa^2} + \tfrac{1+c_{b2}}{\sigma} \qquad c_{\omega 2} = 0.3 \qquad c_{\omega 3} = 2 \qquad c_{\omega 4} = \kappa^2 \tfrac{1+c_{b2}}{\sigma c_{b1}} \qquad \kappa = 0.41 \tag{2.23}$$

$$f_{v1} = \tfrac{\chi^3}{\chi^3 + c_{v1}^3} \qquad f_{v4} = \tfrac{\chi}{1+\chi f_{v1}} \qquad f_{v3} = \tfrac{f_{v4}+c_{\omega 4}}{1+c_{\omega 4}} \tag{2.24}$$

$$f_\omega = \left[\tfrac{g^{-6}+c_{\omega 3}^{-6}}{1+c_{\omega 3}^{-6}}\right]^{-1/6} \qquad g = r + c_{\omega 2}(r^6 - r) \qquad r = \tfrac{\tilde{\nu} f_{v4}}{\tilde{S}\kappa^2 d^2} \qquad \tilde{S} = \left|\tfrac{\partial w}{\partial x} - \tfrac{\partial u}{\partial z}\right| \tag{2.25}$$

2.3 Numerical Algorithm

The discretized equations are solved by an approximately factored, implicit algorithm[44]. The scheme is given as,

$$\begin{aligned} &\left[I + h_\xi\left(\nabla_\xi^b \tilde{A}_{i,k}^+ + \triangle_\xi^f \tilde{A}_{i,k}^-\right)\right]^p \times \left[I + h_\zeta\left(\nabla_\zeta^b \tilde{B}_{i,k}^+ + \triangle_\zeta^f \tilde{B}_{i,k}^- - Re^{-1}\delta_\zeta \tilde{M}_{i,k}\right)\right]^p \\ &\times\left(\hat{Q}_{i,k}^{p+1} - \hat{Q}_{i,k}^p\right) = -\left(\hat{Q}_{i,k}^p - \hat{Q}_{i,k}^n\right) - h_\xi\left(\hat{F}_{i+1/2,k}^p - \hat{F}_{i-1/2,k}^p\right) \\ &-h_\zeta\left(\hat{G}_{i,k+1/2}^p - \hat{G}_{i,k-1/2}^p\right) + Re^{-1}h_\zeta\left(\hat{S}_{i,k+1/2}^p - \hat{S}_{i,k-1/2}^p\right) \end{aligned} \tag{2.26}$$

In this equation, $h_\xi = \triangle^f t/\triangle^f \xi$ and $h_\zeta = \triangle^f t/\triangle^f \zeta$, where t is the computational time; $\tilde{A}^\pm = \partial\hat{F}/\partial\hat{Q}$, $\tilde{B}^\pm = \partial\hat{G}/\partial\hat{Q}$ and $\tilde{M} = \partial\hat{S}/\partial\hat{Q}$ are the flux Jacobian matrices and $\triangle^f$, ∇^b and δ are the forward, backward and central difference operators, respectively. The superscript ^{n} denotes the time step and ^{p} refers to Newtonian subiterations within each time step. $\hat{F}$ and $\hat{G}$ are the numerical inviscid fluxes, which are evaluated using Osher's third-order-accurate

upwinding scheme[40, 41]. The inviscid flux Jacobian matrices, $\tilde{A}$ and $\tilde{B}$ on the left hand side are evaluated by Steger-Warming[45] flux-vector splitting. The viscous fluxes $\hat{S}$ are computed by second-order-accurate central differences.

The hole(s) in the background grid formed by the airfoil grid(s) are excluded from the computations by an *i-blanking* algorithm in each time step of the solution. Again in each time step, the conservative flow variables are interpolated at the intergrid boundaries formed by the overset grids. The details of the intergrid boundary point localization and the interpolation of the flow variables are given in Section 2.5.1.1, and the details of the exclusion of the hole grid points are given in Section 2.5.1.2.

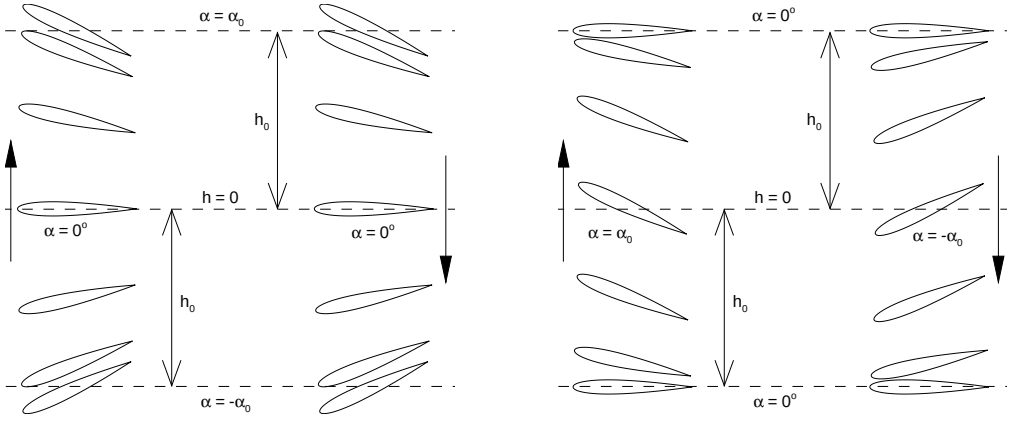
2.4 Overset Grids

The computational domain is discretized with overset grids. `C`-type grid around the airfoil(s) is overset onto a Cartesian background grid (Figure 2.1). The flapping motion of the airfoil is imposed by moving the airfoil and the computational grid around it over the background grid. The flapping motion of a single airfoil in plunge, h, and pitch, α, is defined by:

$$\begin{aligned} h &= -h_0 \cos(\omega t) \\ \alpha &= -\alpha_0 \cos(\omega t + \phi) \end{aligned} \tag{2.27}$$

In Eqn. 2.27, h_0 and α_0 are the amplitudes of the plunge and pitch motions, respectively. ω is the circular frequency of the flapping motion and ϕ is the phase shift between plunging and pitching. Figure 2.3 illustrates the combined pitch and plunge motion of a single airfoil with the phase shifts, $\phi = 0^o$ and $\phi = 90^o$. For the case of two airfoils in a biplane configuration, the flapping motion is given in counter-phase.

Complex geometries and irregular boundaries are currently being discretized by using the block-structured grid techniques[47, 48], in which one patches a number of structured grid blocks together with enforced continuity across the interfaces. All the inherent advantages of structured grids are carried by the block-structured grids, but, since all the grid blocks are to be regenerated at each time step of solution, they are not much convenient for the unsteady flow problems consisting of moving boundaries. The unstructured grids[49, 50] are also currently be-



(a) Phase shift $\phi = 0^o$ (b) Phase shift $\phi = 90^o$

Figure 2.3: Flapping motion of a single airfoil

ing proposed to treat irregular boundaries. However, like in the case of block-structured grids, when solving an unsteady flow problem with moving boundaries, the whole computational domain needs to be regenerated at each time step of solution. Moreover, algebraic turbulence models require directional normal to the solid surface, which leads to new problems in unstructured grid topologies[50].

An alternative to handle the problems with complex geometries with moving boundaries is to use overset grids. In overset grid models, individual grid blocks are overset on top of each other and free to move with respect to each other. For this set of grid, localization of intergrid boundary points and interpolation of flow variables at these points are needed to be performed and, in this thesis, this is achieved through a directional search algorithm. The directional search algorithm used[36, 46] in this study interpolates the flow variables as a part of the localization of intergrid boundary points. In fact, the method employed is similar to the well-known Chimera/Pegsus[51] approach. However, the difference between the two approaches is the way how the overlapping boundaries are localized and the flow variables are interpolated.

In Pegsus, several stages are needed to localize the boundary points and interpolate the flow variables. The first step is to test all the grid points on the base grid if they are a certain distance away from a temporary origin on the overlapping grid. Then, all the grid points within the specified distance are again checked to determine if they are inside or outside of the

overlapping grid region. In the next stage, the closest points falling outside of the overlapping boundary are marked as the boundary points on the base grid. The last step is to localize the interpolation cells on the base grid, into which a boundary point of the overlapping grid falls, and then to store the corresponding bilinear interpolation data in an external file to be used by the flow solver. The Pegsus code has to be integrated into the flow solver for unsteady flows and also to be executed at every time step.

In the present method, the localization of the boundary points and the interpolation of the flow variables are integrated into a directional search algorithm placed in the main solver. Since the boundary points are localized sequentially and no external data is stored, and the interpolation of the flow variables is an integral part of the localization process, this approach to set intergrid boundary conditions is quite efficient and robust.

2.5 Boundary Conditions

The boundary conditions applied in the Navier-Stokes solver consist of numerical and physical boundary conditions. The numerical boundary conditions are applied at the intergrid boundaries and the physical boundary conditions are applied at the farfield inflow and outflow boundaries, and on the airfoil surface.

2.5.1 Numerical Boundary Conditions

The intergrid boundaries consist of the outer boundaries of the airfoil grid and the hole boundaries of the background grid. The conservative flow variables (and the turbulence variable if required) are interpolated from the neighbouring donor subgrid at each time step by the localization of the boundary points (Section 2.5.1.1). There is another intergrid boundary if it is the case of biplane configuration. For the parallel solution of this configuration, the Cartesian background grid is decomposed into two overlapping parts. The overlapping buffer is a single fringe from the symmetry line of the airfoils. Therefore, during the parallel computation, a data interchange between the lower and upper parts of the background grid is performed at each step of solution.

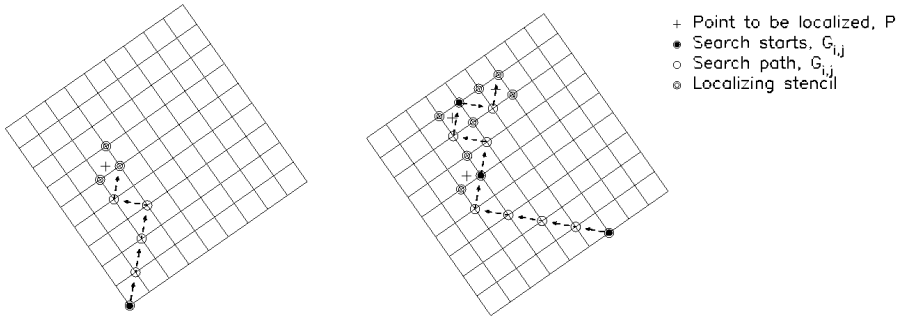


Figure 2.4: Localization of a point and sequential localization points

2.5.1.1 Localization of Boundary Points

The intergrid boundary points on the overset subgrids are localized on the donor grids with a directional search algorithm[36]. A schematic chart of the algorithm is shown in Figure 2.4 and the boundary points to be localized can be seen in Figure 2.5.

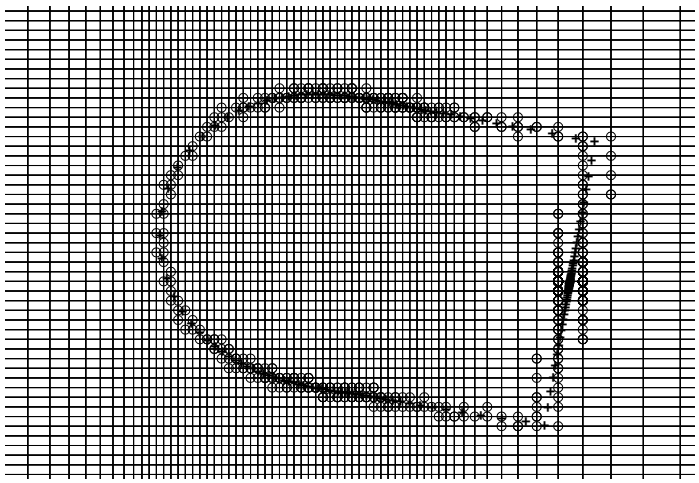
In Figure 2.4, + sign defines the point P to be localized on the Cartesian grid in terms of three closest points forming a triangle around the point P. The search process starts from an arbitrary point $G_{i,j}$ and then it is checked whether $G_{i,j}$, $G_{i\pm1,j}$ and $G_{i,j\pm1}$ constitute a triangle or not. The correct sign $\pm$ is chosen according to the search direction. The search direction is based on the geometry gradients on the donor grid. The test whether the point P falls into the triangle is based on the representation of a convex surface defined by three points. Since any point in a convex triangular surface may uniquely be represented by three real numbers provided that they are all positive. The solution of the linear equations 2.29 through 2.30 in terms of α, β and γ determines if point P falls inside the triangle.

$$x_p = \alpha x_{i,j} + \beta x_{i\pm1,j} + \gamma x_{i,j\pm1} \tag{2.28}$$

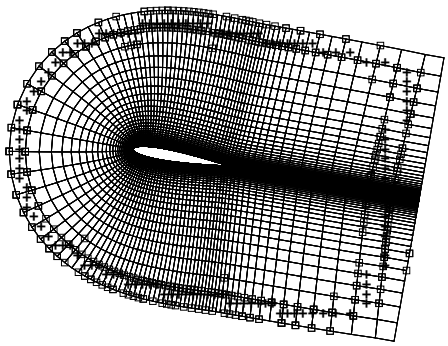
$$y_p = \alpha y_{i,j} + \beta y_{i\pm1,j} + \gamma y_{i,j\pm1} \tag{2.29}$$

$$1 = \alpha + \beta + \gamma \tag{2.30}$$

Here, x_p, y_p are the coordinates of the point to be localized, P. The condition for the point P to lie in the triangle stencil, is that α, β and γ are all less than or equal to one and they are all positive. If this condition is not satisfied, then the search location $G_{i,j}$ is advanced diagonally as shown by the arrows and the circles in the Figure 2.4.



(a) Localization of the airfoil grid boundary points on the background grid



(b) Localization of the hole boundary points on the airfoil grid

Figure 2.5: Boundary points to be localized

The search direction is determined by using the *quadrant walking* algorithm[36] which is considered to be the most robust one. In the quadrant walking algorithm, the search direction is based on the local quadrant where the boundary points falls. As shown in Figure 2.4 $G_{i,j}$ advances along the diagonal of the quadrant P falls into. The quadrant into which point P falls is determined by the dot products of the vector $\overline{P - G_{i,j}}$ with vectors $\overline{G_{i\pm1,j} - G_{i,j}}$ and $\overline{G_{i,j\pm1} - G_{i,j}}$.

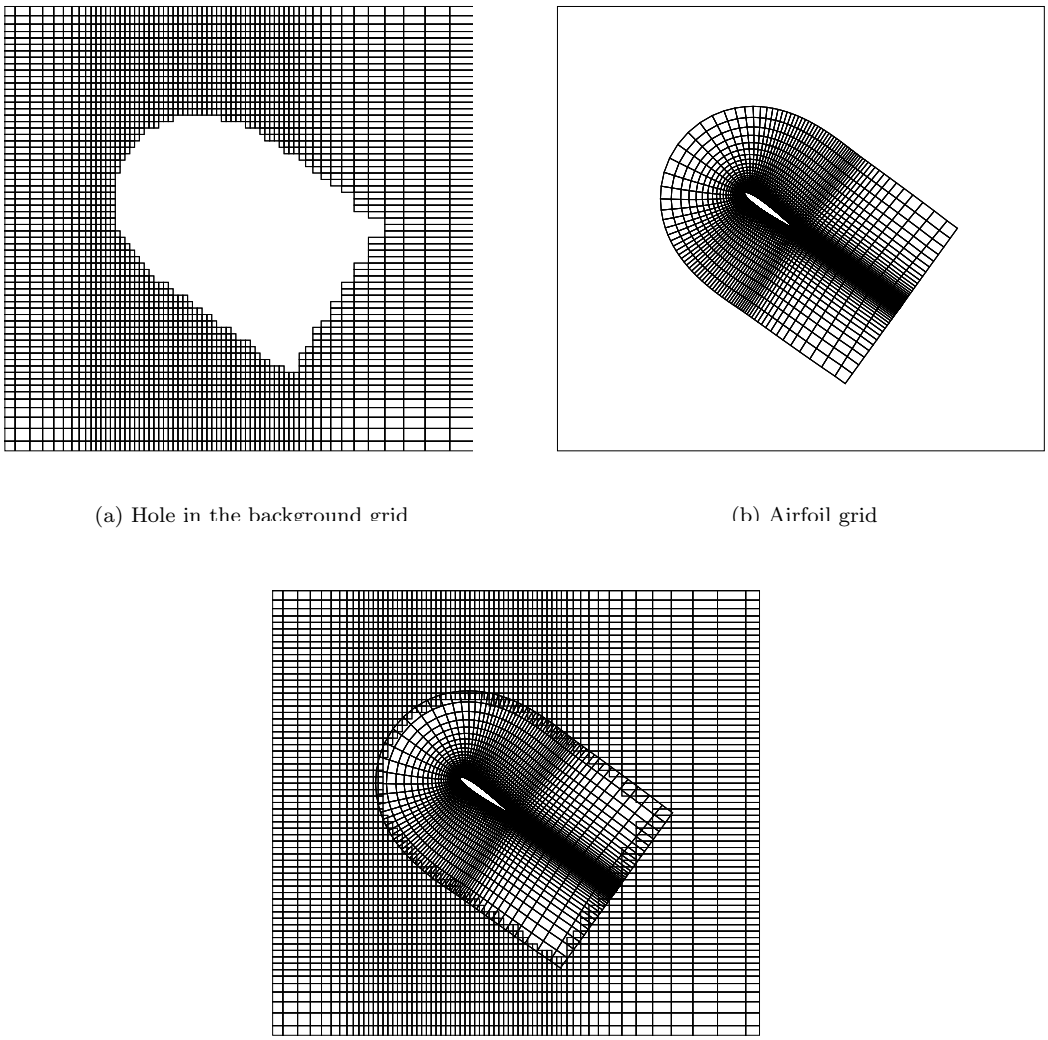
Finally, the triangular stencil into which point P falls is localized as depicted by the circles at its vertices. Then, the coefficients α, β and γ are determined, which give the geometric interpolation weights at point P. Therefore, having the interpolation coefficients ready, now the flow variables (and turbulence variable) at point P can be interpolated from the donor grid with the following equation:

$$Q_p = \alpha Q_{i,j} + \beta Q_{i\pm1,j} + \gamma Q_{i,j\pm1} \tag{2.31}$$

The sequential localization of several points is also shown in Figure 2.4. As seen, for the next consecutive point, the search process starts from the same $G_{i,j}$ grid point which localizes the previous point.

2.5.1.2 Hole Cutting

The localization of the boundary points (Figure 2.5) is not only performed to interpolate the values of the flow variables along the airfoil grid boundary but also to set the values of the flow variables along the boundary of the hole in the background grid, which is cut by the airfoil grid (Figure 2.6). Once the outer boundary points of the airfoil grid are localized in the background grid as explained in Section 2.5.1.1, three consecutive tasks are executed. First, the values of the conservative flow variables on the boundary points of the airfoil grid are interpolated from the background grid. Second task is to localize the boundary points of the hole in the background grid. This is done by marching along the direction normal to the localized airfoil grid boundary with some step size from the airfoil grid boundary localized in the background grid . Marching step size is set to be double fringe, that is, two-size of local cell in the background grid. This causes an overlapping buffer between the localized airfoil grid boundary and the boundary of the hole generated in the background grid. The overlapping buffer is required and important since it reduces the number of the points which are both interpolated (in the receiver grid) and also element of a localizing triangle stencil (in the donor grid). Then, after this last localization process, the flow variables on the boundary points of the hole are interpolated from the airfoil grid. Figure 2.5(a) shows the localization of the points in the background grid along the boundary of the airfoil grid and Figure 2.5(b) shows the localization of the points in the airfoil



(a) Hole in the background grid

(b) Airfoil grid

(c) Overset grids

Figure 2.6: Background and airfoil grids with hole in the background grid

grid along the boundary of the hole generated in the background grid. Note that, due to the overlapping buffer, the boundary of the hole does not coincide with the airfoil grid boundary.

Finally, the determination of all the background grid points falling in the hole is required. This is achieved by a very simple algorithm. Since the boundary of the hole is known, one horizontal and one vertical, totally two marching processes are performed inside of the region determined by the hole boundary. During these two marching processes, all the points in the track are marked. Therefore, a hole is generated in the background grid (Figure 2.6). This hole is excluded from the flow computation in each time step of computation. This exclusion

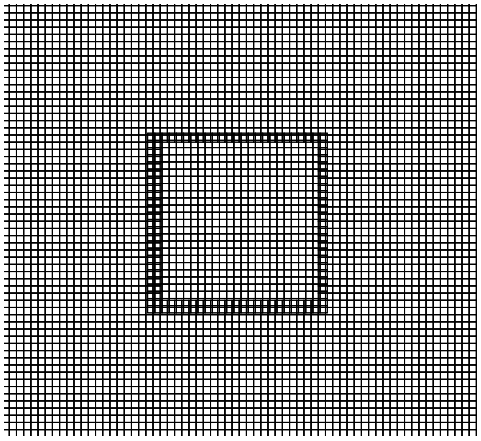


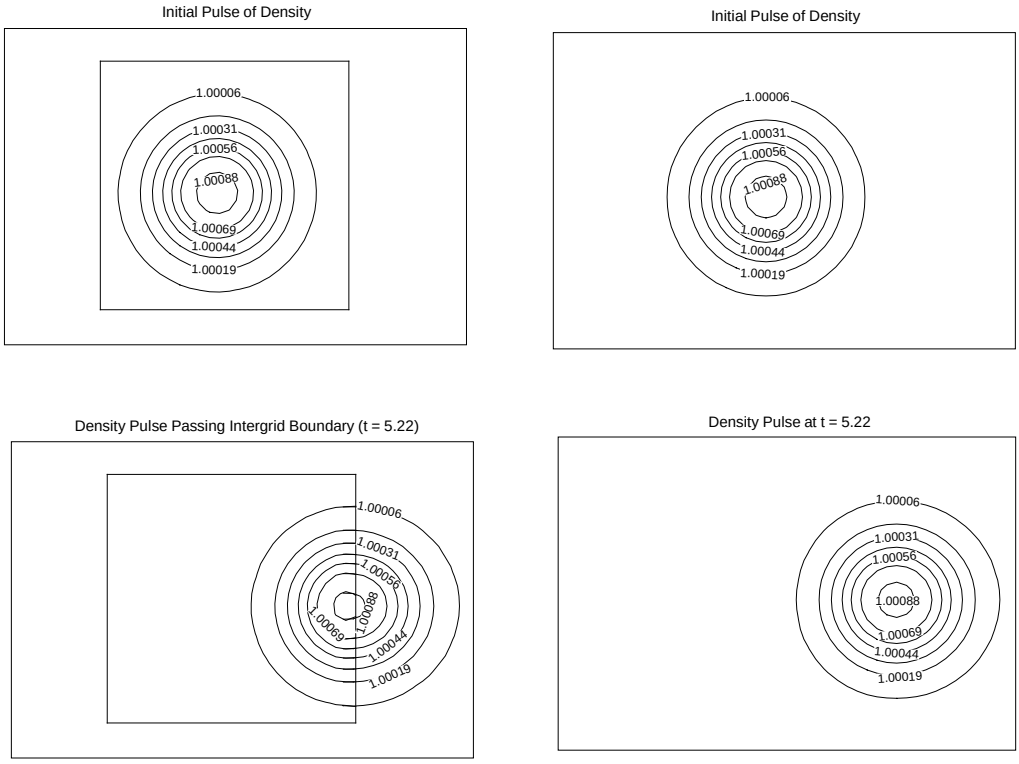
Figure 2.7: Cartesian overset grids used for Gaussian pulses

called as *i-blanking* is a method in which the corresponding coefficient matrices are replaced by an identity matrix and the right hand side vector is set to zero. Essentially, the idea behind of this approach is not to carry a computation for the flow variables on the hole points during the time integration in each step.

2.5.1.3 Accuracy of the Intergrid Boundary Continuity

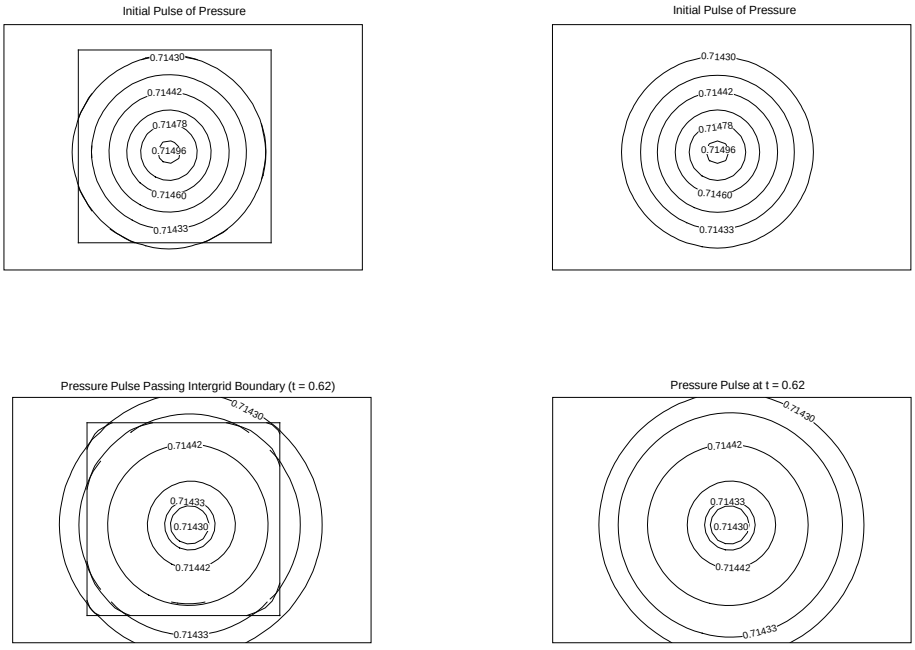
The accuracy of the overset grid computations in parallel is established by Tuncer[36]. Also in this study, it is further demonstrated by employing the method for a single flapping airfoil case, and comparing the solution to that of the computation obtained with a single grid on a serial computer. It should be noted that, based on the grid sensitivity study in Reference [38], Tuncer et al. showed that the results computed using the Navier-Stokes solver employed in this thesis work are not sensitive to the grid size.

First, an investigation of density and pressure Gaussian pulse perturbations on an uniform flow is performed to show the accuracy of the variable continuity through the intergrid boundaries. Figure 2.8 shows the convection of a Gaussian pulse density perturbation and the convection and the diffusion of a Gaussian pulse pressure perturbation is shown in Figure 2.9. The initial Gaussian pulse of density given to the uniform flow at $M = 0.25$ in Figure 2.8 is $\rho = \rho_\infty(1 + \frac{2^{-(x^2+y^2)/4c^2}}{1000})$ and the initial Gaussian pulse of pressure given to the uniform flow at $M = 0.25$ in Figure 2.9 is $p = p_\infty(1 + \frac{2^{-(x^2+y^2)/4c^2}}{1000})$. c is the non-dimensionalizing parameter of the spatial variables. t in the figures is the non-dimensionalized time. The overset grid system



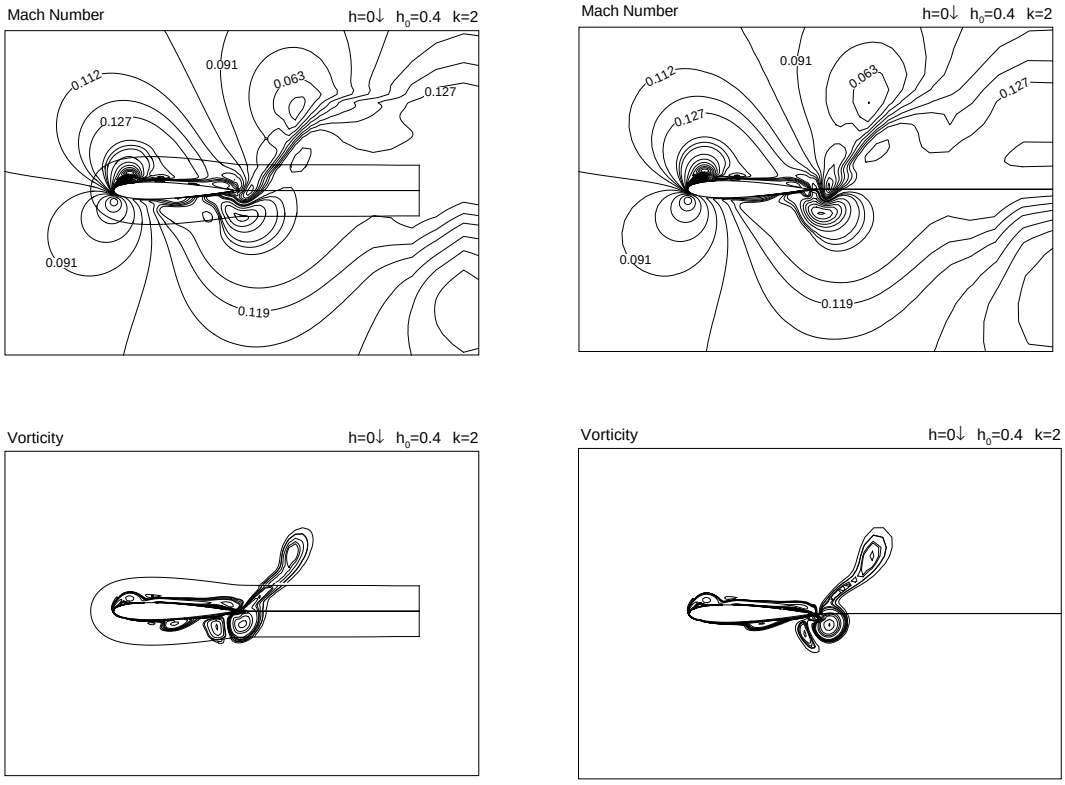
(a) Overset grids (b) Single grid

Figure 2.8: Gaussian pulse of density



(a) Overset grids (b) Single grid

Figure 2.9: Gaussian pulse of pressure



(a) Overset grids

(b) Single grid

Figure 2.10: Instantaneous flowfield

used in this investigation consists of two Cartesian grids which are of 101×101 and 26×26 size (Figure 2.7). The single grid is the background grid of the overset grid system.

The accuracy of the interpolation of the flow variables at the intergrid boundaries can be seen from Fig. 2.8 and Fig. 2.9. Each pulse is convected without almost any loss, being a good proof not only for the interpolation accuracy at the intergrid boundaries but also for the numerical accuracy of the solver. An excellent agreement is observed between the single grid and overset grid solutions, which is a good sign for the reliability of the computations on overset grid system.

Computed by using both overset and single grid systems, instantaneous flowfield of an unsteady flow over a single flapping NACA0014 airfoil in plunge motion is shown in Figure 2.10. The flowfield is computed laminar at $M = 0.1$, $Re = 1 \cdot 10^4$, $k = 2$ and $h_0 = 0.4$. In the overset grid system, the airfoil and the background grids are of 141×31 and 135×157 size, respectively.

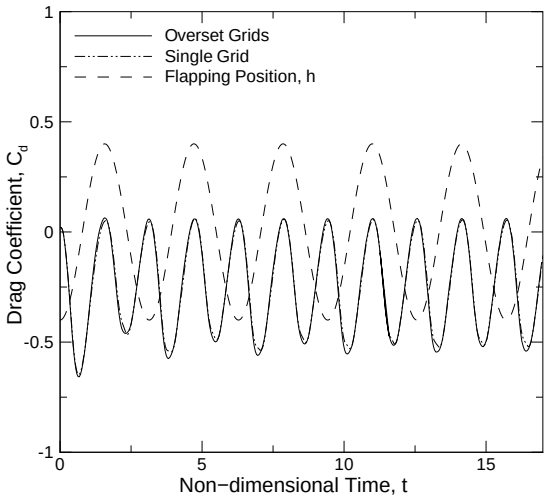


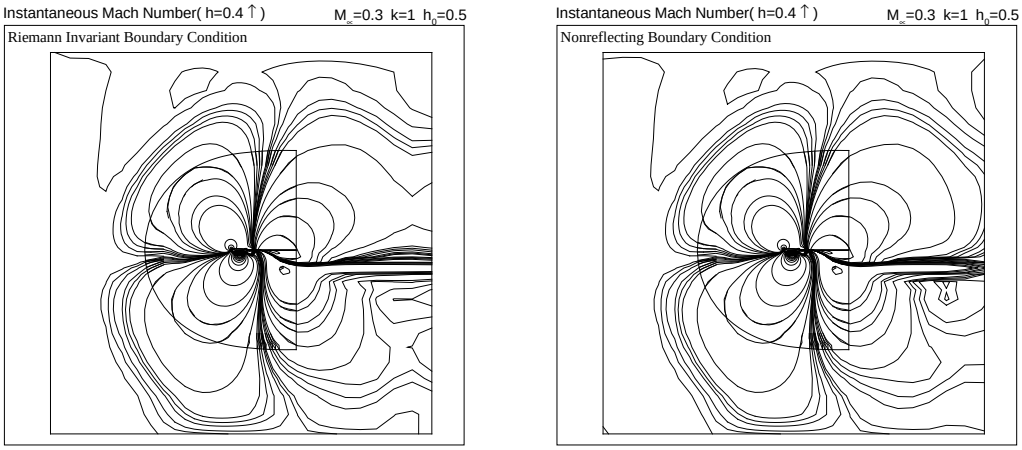
Figure 2.11: Unsteady drag coefficient of a single flapping airfoil

The outer boundary of the airfoil grid is placed about 0.25 chord length distance away from the airfoil surface. The single grid is obtained by further extending the airfoil grid used in the overset grid solution outward, and is of 181×81 size. The time history of the unsteady drag coefficient predicted by the present method is compared to the single grid solution in Figure 2.11. As seen from Figure 2.10 and 2.11, the solution on the overset grid system is as accurate as the solution on the single grid. In Figure 2.10, the Mach number isolines in the case of the overset grid system continue smoothly from one grid to another as in the single grid system case, indicating that the intergrid boundary conditions provide an accurate solution.

2.5.2 Physical Boundary Conditions

Since the formulation of the Navier-Stokes solver is based on an inertial frame of reference, for the flapping motion of the airfoils, the airfoil grids are traversed over the stationary background grid. Therefore, on the airfoil surface, the instantaneous flow velocity is set equal to the local surface velocity prescribed by the oscillatory motion (Eqn. 2.27), and so that the no-slip boundary condition is applied. Besides, on the airfoil surface, the density and the pressure gradients are set to zero.

At the farfield inflow and outflow boundaries, the flow variables may be evaluated using



(a) Riemann invariant B.C.'s

(b) Nonreflecting B.C.'s

Figure 2.12: Instantaneous Mach number

either zeroth-order Riemann invariant extrapolation or more expensive nonreflecting boundary conditions. Comparison of these two boundary conditions apparently showed that the Riemann extrapolation method is adequate when the farfield boundaries are sufficiently far away from the airfoil.

2.5.2.1 Nonreflecting Boundary Conditions

The objective in formulating nonreflecting boundary conditions is to prevent nonphysical reflections at the inflow and outflow boundaries.

The formulation of the nonreflecting boundary conditions for the flow solver used in this thesis work is adapted from the analysis in Reference [52]. The given analysis is based on the wave-like solutions for the unsteady, linearized, two-dimensional Euler equations governing a uniform flow under small perturbations. Non-trivial solutions show that, caused by the perturbations, one upstream running wave and three downstream running waves are generated. Thus, to establish the correct nonreflecting boundary conditions at the inflow and outflow boundaries, these waves are forced to march only in their travelling direction. That is, at the inflow boundary, while the upstream running wave is allowed to leave the domain, the downstream running waves are prevented from crossing the inflow boundary. Similarly, at the

outflow boundary, only the downstream running waves are let to leave the domain.

As a result of his analysis[52], Giles proposes the following *quasi-one dimensional* inflow and outflow boundary conditions:

- *Fourth-order, two-dimensional, unsteady, inflow boundary conditions*

$$\frac{\partial}{\partial t}\begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} + \begin{pmatrix} v & 0 & 0 & 0 \\ 0 & v & \frac{a+u}{2} & \frac{a-u}{2} \\ 0 & \frac{a-u}{2} & v & 0 \end{pmatrix} \times \frac{\partial}{\partial y}\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = 0 \tag{2.32}$$

- *Second-order, two-dimensional, unsteady, outflow boundary conditions*

$$\frac{\partial c_4}{\partial t} + \begin{bmatrix} 0 & u & 0 & v \end{bmatrix} \frac{\partial}{\partial y}\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = 0 \tag{2.33}$$

where,

$$\begin{aligned} c_1 &= p' - a^2\rho' & \rho' &= \rho - \rho_e \\ c_2 &= \rho a v' & u' &= u - u_e \\ c_3 &= p' + \rho a u' & v' &= v - v_e \\ c_4 &= p' - \rho a u' & p' &= p - p_e \end{aligned} \tag{2.34}$$

and ρ, u, v, p, a are the flow variables at the boundary to be solved while ρ_e, u_e, v_e, p_e are the extrapolated flow variables at the boundary. Physically, c_1, c_2, c_3 and c_4 are the amplitudes of the four characteristic waves.

Computed by using different farfield boundary conditions, instantaneous flowfield solutions of a single flapping airfoil making plunge motion are shown in Figure 2.12. Flapping parameters are shown in the figure. As seen from the figure, in the case of Riemann boundary conditions solution, Mach number isolines flatten out on the outflow boundary. This is due to the zeroth order extrapolation characteristic of the Riemann invariant boundary conditions. However, the overall flowfield agrees well with the solution computed by using nonreflecting boundary conditions.

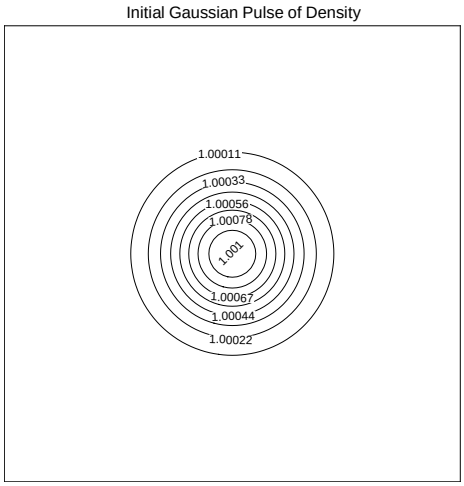
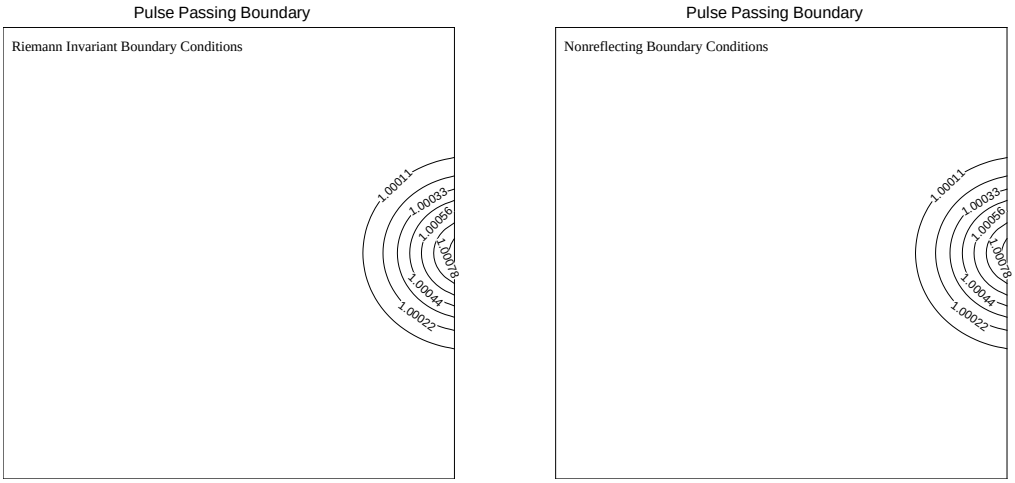


Figure 2.13: Initial density contours

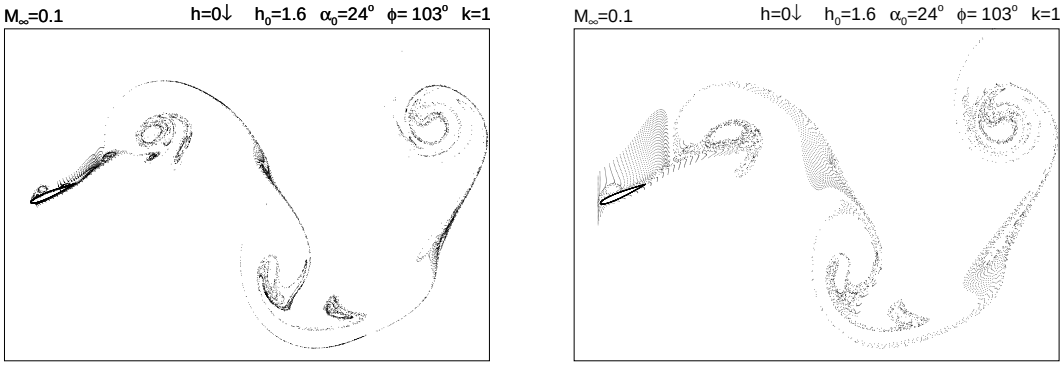


(a) Riemann invariant B.C.'s

(b) Nonreflecting B.C.'s

Figure 2.14: Density contours when the pulse is passing the boundary

Figure 2.13 shows the initial density contours due to a gaussian pulse of density $\rho = \rho_\infty(1 + \frac{2^{-(x^2+y^2)/25c^2}}{1000})$ given to a uniform flow at $M = 0.25$ and Figure 2.14 shows density contours at the time when the pulse is passing the outer grid boundary. c is the non-dimensionalizing parameter of the spatial variables. The grid used in this numerical experiment is a 101×101 size Cartesian grid, which is the background grid of the overset grid system shown in Figure 2.7. Center of the initial Gaussian pulse is located $10c$ distance away from the outer grid boundary. During the unsteady computation, the pulse is convected with the freestream velocity without



(a) Shedding from the airfoil grid

(b) Shedding from a vertical line

Figure 2.15: Particle traces

any loss and passes the outer boundary causing no nonphysical reflections. In both methods, the values of the conservative variables on the boundary are computed with excellent accuracy. After some time, the pulse leaves the domain yielding a uniform flow as it was before.

2.6 Particle Traces

Particle traces are obtained by a simple and efficient integration of the particle pathlines within the flow solver as the unsteady flowfield is computed. In this method, the particles may be released anywhere in the flowfield at certain intervals. The particles are then localized in the computational grid and convected with the local velocity at every particle integration time step defined by the user. The particles may be localized across the intergrid boundaries. The search algorithm used to localize the particles is the same search algorithm described in Section 2.5.1.1.

Figure 2.15 shows the instantaneous particle traces of a flapping airfoil in combined pitch and plunge motion with the flapping parameters shown in the figure. In Figure 2.15(a), the particles originated near the leading edge from a normal line on the airfoil grid are shown. Figure 2.15(b) gives the particle traces in the case of shedding the particles from a vertical line placed one tenth chord length distance away from the leading edge.

CHAPTER 3

OPTIMIZATION

3.1 Introduction

The recent experimental and numerical studies investigating the flapping motion of single or dual airfoils in tandem show that the thrust generation and the propulsive efficiency of flapping airfoils are closely connected to the flapping motion and the flow parameters. The major parameters are the unsteady flapping frequency, the amplitude of the pitch and plunge motions, the phase shift between them, and the air speed. It is apparent that to maximize the thrust and/or propulsive efficiency of a flapping airfoil, an optimization of all the above parameters is needed. The optimization tool employed in this work is the *steepest ascent* method, which is a gradient based algorithm. This chapter presents how the steepest ascent method is applied for an optimization process to maximize the thrust and/or the propulsive efficiency of flapping airfoils.

The objective function to be maximized may be a linear combination of thrust and propulsive efficiency. The thrust (negative drag) of a harmonically flapping airfoil has a periodic behavior (Figure 3.1). Thrust production of a flapping airfoil is therefore defined as an average value over a period of the flapping motion. Similarly, the propulsive efficiency is also defined in the average sense. The definitions of the average thrust coefficient and the propulsive efficiency are given in Eqn. 2.9. Required by the gradient based optimization algorithm employed, the gradient of the

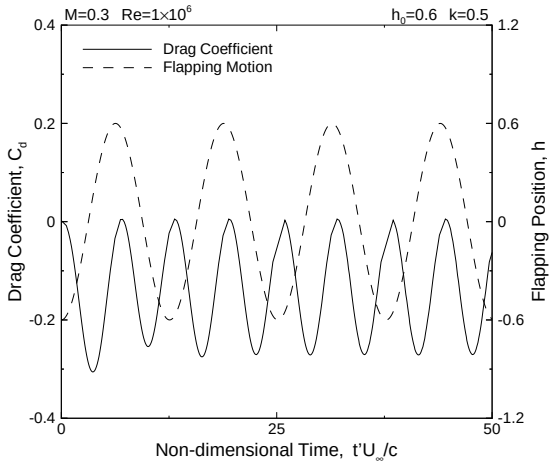


Figure 3.1: Drag history along flapping periods (Figure 2.2 is repeated)

objective function (linear combination of thrust and propulsive efficiency) is evaluated once a periodic flow behavior is established. In the optimization studies, the optimization variables are taken to be the variables defining the flapping motion, that is, pitching and plunging amplitudes, the flapping frequency and the phase shift between the pitch and plunge motions.

3.2 Steepest Ascent Method

The maximization of an objective function can be iteratively performed by the *steepest ascent* method. The method is based on marching, by a succession of steps, along the local gradient direction of the objective function. The idea is originated from the fact that the gradient of a function gives the direction along which the function has the maximum rate of change. The following formulations for a maximization process with steepest ascent method are extracted from the derivations in Reference [53].

3.2.1 Maximization Process

In the optimization space, for the vector of optimization variables, $\vec{V}$ having four components (h_0, k, α_0, ϕ), the maximization process seeks to find an ascent direction $\vec{D}$ and a positive step size ε in which to change $\vec{V}$ such that

$$O(\vec{V} + \varepsilon\vec{D}) \geq O(\vec{V}) \tag{3.1}$$

This process is repeated until the objective function O reaches to its maximum value. Note that the step size, ε is analogous to the relaxation parameter used in the iterative matrix inversion method.

3.2.2 Steepest Ascent Direction

An ascent direction $\vec{D}$ of the optimization variables can be determined as follows. The second order Taylor series expansion of the objective function O about $\vec{V}$ is written as

$$O(\vec{V} + \varepsilon\vec{D}) = O(\vec{V}) + \varepsilon\vec{D} \cdot \vec{\nabla}O(\vec{V}) + o(\varepsilon^2) \tag{3.2}$$

$\vec{\nabla}O$ is the gradient of O and given by

$$\vec{\nabla}O = \frac{\partial O}{\partial h_0}\hat{h_0} + \frac{\partial O}{\partial \alpha_0}\hat{\alpha_0} + \frac{\partial O}{\partial k}\hat{k} + \frac{\partial O}{\partial \phi}\hat{\phi} \tag{3.3}$$

where the vector variables with $\hat{}$ denotes the positive unit direction along the corresponding variable coordinate. Eqn. (3.2) clearly shows that if

$$\vec{D} = \frac{\vec{\nabla}O}{|\,\vec{\nabla}O\,|} \tag{3.4}$$

then Eqn. (3.1) is satisfied. The ascent direction $\vec{D}$ given in Eqn. 3.4 is the steepest ascent direction along which the objective function is locally increasing at the fastest rate[54]. Note that the equality in Eqn. (3.1) occurs when $\vec{\nabla}O = 0$, at the maximum value of the objective function. To obtain the ascent direction $\vec{D}$, the gradient of O must be evaluated. In this thesis, evaluation of $\vec{\nabla}O$ (Eqn. 3.3) is done by using the first order forward finite difference formulation. A point to be noted is that Eqn. (3.4) is not the unique solution for Eqn. (3.1).

3.2.3 Step Size

Once the steepest ascent direction has been determined, now, the step size ε is to be evaluated. A simple choice is to take ε = constant. Another approach, as done in this thesis, is to do a line search with respect to ε along the steepest ascent direction. In other words, to find ε such that $|\,\vec{\nabla}O(\vec{V} + \varepsilon\vec{D})\,|^2$ is a minimum. That is,

$$\frac{\partial\,|\,\vec{\nabla}O(\vec{V} + \varepsilon\vec{D})\,|^2}{\partial\varepsilon} = 0 \tag{3.5}$$

Reference [53] gives the following approximate solution of Eqn. 3.5 for the step size, ε,

$$\varepsilon = -\frac{\vec{\nabla}O \cdot (\vec{\nabla}\vec{\nabla}O)\vec{D}}{(\vec{\nabla}\vec{\nabla}O)\vec{D} \cdot (\vec{\nabla}\vec{\nabla}O)\vec{D}} \tag{3.6}$$

where $(\vec{\nabla}\vec{\nabla}O)$ is a symmetric matrix and is often referred to as the Hessian. The computation of the Hessian is expensive and the cost is proportional to the number of the optimization variables. Assuming that, in the next optimization step, a point in the neighborhood of the maximum will be reached, the following approximation may be made not to pay high prices for the Hessian:

$$\vec{\nabla}O \cdot (\vec{\nabla}\vec{\nabla}O)\vec{D} \approx -|\,\vec{\nabla}O\,| \cdot |\,(\vec{\nabla}\vec{\nabla}O)\vec{D}\,| \tag{3.7}$$

A minus sign appears in the above approximation due to the Hessian being a negative definite matrix and, the vectors $\vec{\nabla}O$ and $\vec{D}$ being aligned in the same direction. Substituting this approximation into Eqn. (3.6), the step size is found as,

$$\varepsilon = \frac{|\,\vec{\nabla}O\,|}{|\,(\vec{\nabla}\vec{\nabla}O)\vec{D}\,|} \tag{3.8}$$

Finally $|\,(\vec{\nabla}\vec{\nabla}O)\vec{D}\,|$ is suggested to be evaluated with the finite differences as follows:

$$|\,(\vec{\nabla}\vec{\nabla}O)\vec{D}\,|^k = \lambda\frac{|\,\vec{\nabla}O^k - \vec{\nabla}O^{k-1}\,|}{\varepsilon^{k-1}} \tag{3.9}$$

where ^{k} denotes the optimization step and λ is the relaxation parameter between two successive optimization steps, $^{k-1}$ and ^{k}. An expression for λ is given as,

$$\lambda = \frac{|\,\vec{\nabla}O^k - \vec{\nabla}O^{k-1}\,|}{|\,\vec{\nabla}O^{k-1}\,|} \tag{3.10}$$

However, considering again that, in the next optimization step, a point in the neighborhood of the maximum will be reached, that is, $|\,\vec{\nabla}O^k\,| \approx 0$, then λ may be approximated to be equal to the unity according to Eqn. 3.10. Therefore, Eqn. 3.9 turns out to be:

$$|\,(\vec{\nabla}\vec{\nabla}O)\vec{D}\,|^k = \frac{|\,\vec{\nabla}O^k - \vec{\nabla}O^{k-1}\,|}{\varepsilon^{k-1}} \tag{3.11}$$

3.3 Multi-Objective Optimization

For the multi-objective optimization problems, an effective approach is to employ the *Weighted-Sum* method. In this method, the multi-objective optimization problem is reduced to a single-objective optimization problem by defining a multi-objective function as a linear combination of

objective functions. In this study, the multi-objective function is defined as a linear combination of the normalized thrust and propulsive efficiency values:

$$O(\vec{V}) = (1-\beta)\cdot T + \beta \cdot E \tag{3.12}$$

where T and E are the *scaled normalized form* of the average thrust coefficient and the propulsive efficiency, respectively. β is the weight of the normalized propulsive efficiency in the linear combination and provided by the user. Note that $\beta = 0$ sets the objective function to a normalized thrust. Normalization of the average thrust coefficient and the propulsive efficiency is performed at each optimization step as shown below:

$$\begin{aligned} T &= \frac{C_t}{C_{tN}} \\ E &= \frac{\eta}{\eta_N} \end{aligned} \tag{3.13}$$

where,

$$\begin{aligned} C_{tN} &= C_t + |\Delta C_t| \\ \eta_N &= \eta + |\Delta \eta| \end{aligned} \tag{3.14}$$

The normalizing parameters, C_{tN} and η_N reduce C_t and η values approximately into the same order of magnitude. The values of C_{tN} and η_N are not fixed and updated in each optimization step. ΔC_t and $\Delta\eta$ are the total changes in the values of C_t and η, respectively, in each optimization step due to the perturbation of optimization variables.

Once the average thrust coefficient and propulsive efficiency values are normalized in each optimization step with the definition given by Eqn. (3.14), then the objective function $O(\vec{V})$ is evaluated to apply the process explained in Section 3.2 to find the flapping motion conditions, $\vec{V}$ maximizing $O(\vec{V})$ with the given weight β.

CHAPTER 4

PARALLEL PROCESSING

4.1 Introduction

Parallel processing is the method of dividing a large computing task into smaller independent tasks and executing them simultaneously in parallel, rather than sequentially in serial. Having a physical processing speed limit, serial computers of today and the future may not be sufficient to meet the requirements of the present *CFD* applications where the solution of large system of partial differential equations is needed for numerical simulations. Parallel computers may be employed to overcome this limitation of a single computer so that the computation time is significantly reduced by performing several operations simultaneously.

Parallel processing is currently realized by the *shared-memory* and *distributed-memory* systems. In the shared-memory systems, the processors share the same common memory and the compiler or the programming language decomposes the computation into smaller tasks and execute them in parallel using the available processors in a multi-processor architecture, provided that there is no data dependence between the tasks. If there is a data dependence, tasks run in the same order as in the sequential execution. In the distributed-memory systems (Figure 4.1), each processor has its own memory and solves a sub-problem with its own private data decomposed from the full problem, and also communicates with other processors for required data sharing. The computer programming in distributed-memory and shared-memory systems

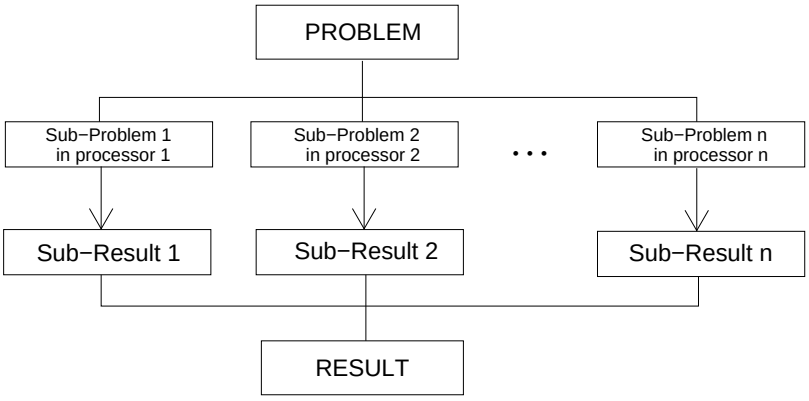


Figure 4.1: Parallel solution of a problem with distributed-memory approach

is provided by the following means of extension[55]:

- *Library Routines*: In addition to the standard libraries available to the sequential language, a set of new library functions is added to support the parallelism and the communication between the processors. Examples of such libraries include `PVM` and `MPI` message passing libraries, which provide the user to employ distributed-memory systems in computer clusters, and the `POSIX Pthreads` multi-threading library for shared-memory systems in multi-processor architectures.

- *New Constructs*: The programming language is extended with some new constructs to support the parallelism. An example is the aggregated array operations in `Fortran 90`. This approach supports the shared-memory systems.

- *Compiler Directives*: The programming language stays the same, but formatted comments, called compiler directives are added. This approach of parallelization is employed mostly for the shared-memory systems.

The library approach is currently the most widely used one, since it is easy to implement. Here, all the parallelism and the interaction functionalities are realized by a set of library routines to integrate into a sequential computer code written with either the `Fortran` or `C` programming languages. As a consequence, there is no need for a new compiler. However,

without the compiler's support, the user is deprived of the compile-time analysis, the error checking and the optimization.

In this study, PVM message passing library routines for the Fortran language are used in parallel programming. Although the MPI message passing library has been developed recently and considered to be the standard parallel programming tool, PVM is still the most popular software platform for parallel processing[56]. Apart from PVM and MPI, the message passing environments include also the P4, Express and Linda software packages.

The author believes that PVM has the advantage of being employed in the heterogenous computer systems, and has grown to be quite stable. In this work, PVM version 3.4.4 is used. PVM is also tolerant to a larger number of nodes (processors), since even though one (or more) task fails running, other tasks will continue while MPI exits the program when an MPI error is encountered.

It will be beneficial to the parallel processing community if PVM and MPI eventually merge into a single, standard library[55].

4.2 Parallel Programming

The parallel programming algorithms developed in this study are based on the distributed-memory concept in the form of domain decomposition. The computational domain is first decomposed into subdomains. Then, the computation in each subdomain is assigned to a processor in the computer cluster and the interprocess communication is realized by the PVM library routines.

There are some important issues related to parallel processing in a distributed-memory system, such as domain decomposition, load balancing and the performance criteria, speed-up, which influences the efficiency of the parallel computations.

4.2.1 Domain Decomposition

Partitioning of data and the computational tasks among multiple processors is denoted as *domain decomposition*. In the present work, the computational domain, which is discretized by

overset grids, is decomposed into its subgrids first, and the solution on each subgrid is assigned to a processor (Figure 4.2). Note that overset grids lend themselves to a natural domain decomposition. When computing the flowfield around dual airfoils in a biplane configuration, the background grid is also partitioned into two subgrids in the crossflow direction (Figure 4.2).

4.2.2 Load Balancing

Maintaining uniform computational activity on each processor is known as *load balancing*. It is desirable that each processor work on the same amount of data to minimize the waiting time of a processor for the data coming from another processor. A simple approach is to determine how to decompose the domain, that is, how to partition the data, before assigning subproblems to the processors. This is called *static load balancing*. However, this technique may not provide a complete load balancing if the given problem consists of some factors/constraints. For example, the nature of the governing equations may change during the computation or the number of the governing equations may have different values in each processor. Moreover, the characteristics of the computational subdomains, such as discretization, may be altered during the computation due to adaptations. A remedy is to provide *dynamic load balancing* on each processor during computation, which is usually difficult to be implemented.

4.2.3 Speed-Up

Another important issue is the efficiency of the parallel computations. The true measure of the performance is simply the wall clock time needed for the completion of the computations. A criteria called *speed-up* shows how fast the parallel code is, compared to its serial counterpart (Eqn. 4.1). A good algorithm is one for which this ratio is larger.

$$\text{Speed–Up} = \frac{\text{CPU time for 1 serial processor}}{\text{CPU time for } n \text{ processors in parallel}} \tag{4.1}$$

4.3 PVM, the Parallel Programming Library

In this thesis, `PVM` is used as the parallel programming tool. It is a software developed by the Heterogeneous Network Project (Oak Ridge National Laboratory, the University of Tennessee

and Emory University) initiated in 1989 to generate heterogeneous network-based parallel computers. It currently supports the `Fortran`, `C`, `C++`, and `Java` languages. The most common operating system where `PVM` is executed is `Unix`. It has also been implemented on non-`Unix` platforms such as `Windows NT` and `Windows 9X`.

A user can construct a computer cluster, a set of connected computers with a single processor or multi processors. The user can then assign a computation task to each processor by using `PVM`. The communication between the processors for sharing data is called as the *interprocess communication*. `PVM` provides all the required library routines to support the interprocess communication and some other functions.

4.4 Computing Environment

The computer cluster available for the parallel processing studies consists of 10 computers with dual `Pentium-III` processors of 700 MHz speed and 512 MB memory. The communication between the processors is provided by a 100 MB ethernet local network with a 100 Mbps switch. The computers run under the `Linux` operating system.

4.5 Structure of the Parallel Codes

The computational tool used in this thesis includes a Navier-Stokes flow solver and an optimization software. The computations with both of the codes are performed in parallel and their algorithms are based on the parallelization technique in a distributed-memory system. Both parallel algorithms are very similar to each other in the way of task sharing. The details of the structure of the codes are given in the following sections.

4.5.1 Parallel Algorithm for the Flow Solver

A coarse parallel algorithm based on domain decomposition is implemented in a master-worker paradigm. As mentioned in Section 4.2.1, the computational domain which is discretized by overset grids as shown in Figure 4.3, is partitioned into subdomains first, and the solution on each grid block is assigned to a different processor. The assignment is done by the *master*

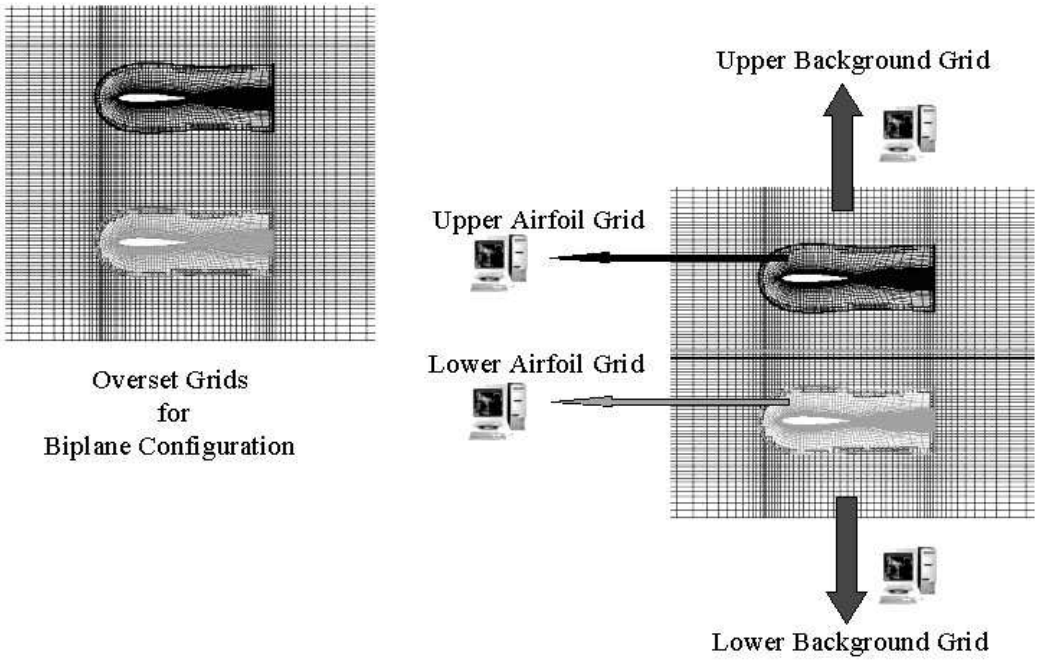


Figure 4.2: Domain decomposition

program. Giving the assignment to a processor, *master* sends also to this processor an initial solution as well as the necessary information for the computation. Each processor, which executes the *worker* program, computes the flow on either a background or an airfoil grid block, and communicates with other *worker*s for data sharing. Combining the results coming from *worker*s is finally performed by *master*.

As an example for grid partitioning, the overset grid system for a single airfoil is shown in Figure 4.3 and 4.4. As seen from the figures, the region where the airfoil grid is to be located on the background grid is blanked out. This hole region is not involved in the flow computation on the background grid. Necessary information for the hole localization and the intergrid boundary conditions for the Cartesian background grid, are supplied by the airfoil grid block, and in the same way, necessary intergrid boundary conditions for the airfoil grid which is a `C`-type structured grid, are supplied by the background grid block, through `send-receive` operations of `PVM`. This data interchange between the grid blocks are performed in each time step of computation. In the biplane configuration, since the background grid is also partitioned to improve the static load balancing, there is one more intergrid boundary, which is between

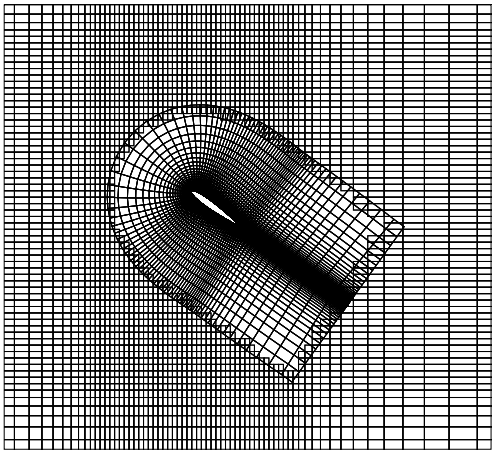


Figure 4.3: Overset grids

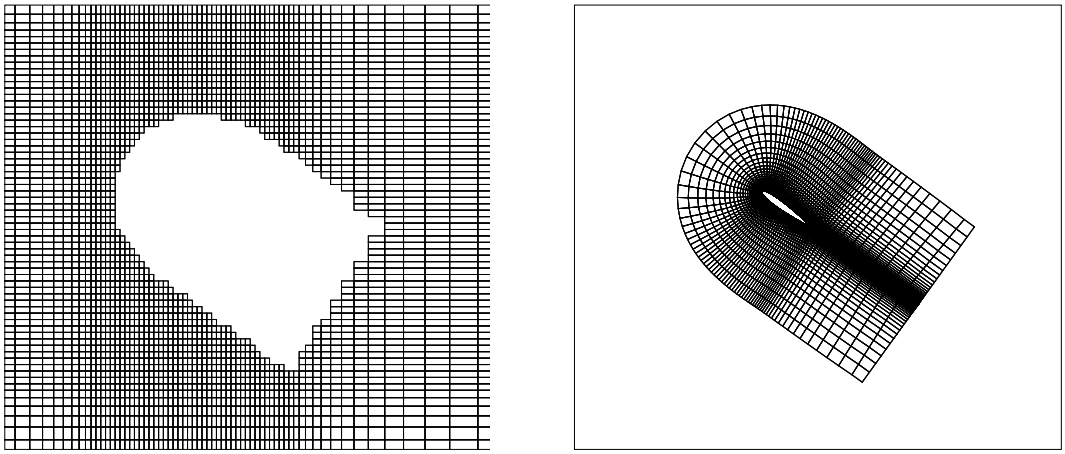


Figure 4.4: Background and airfoil grids (Figure 2.6 is repeated)

two overlapped Cartesian subgrids partitioned from the original background grid (Figure 4.2). Therefore, another data interchange between *worker*s solving the flow in the upper and lower part of the background grid is provided.

Figure 4.5 shows a brief algorithm of the parallel code used in the flow solver. First, the main code (*master*) assigns tasks to processors corresponding to the number of airfoils. For example, for a single flapping airfoil computation, then two processors are needed; one for computing the flow on the background grid and one for computing on the airfoil grid. Again, for the case of flapping airfoils in biplane configuration, a total of four *worker*s are spawned in four processors. After the task assignment is done, *worker*s start to compute the flow on the

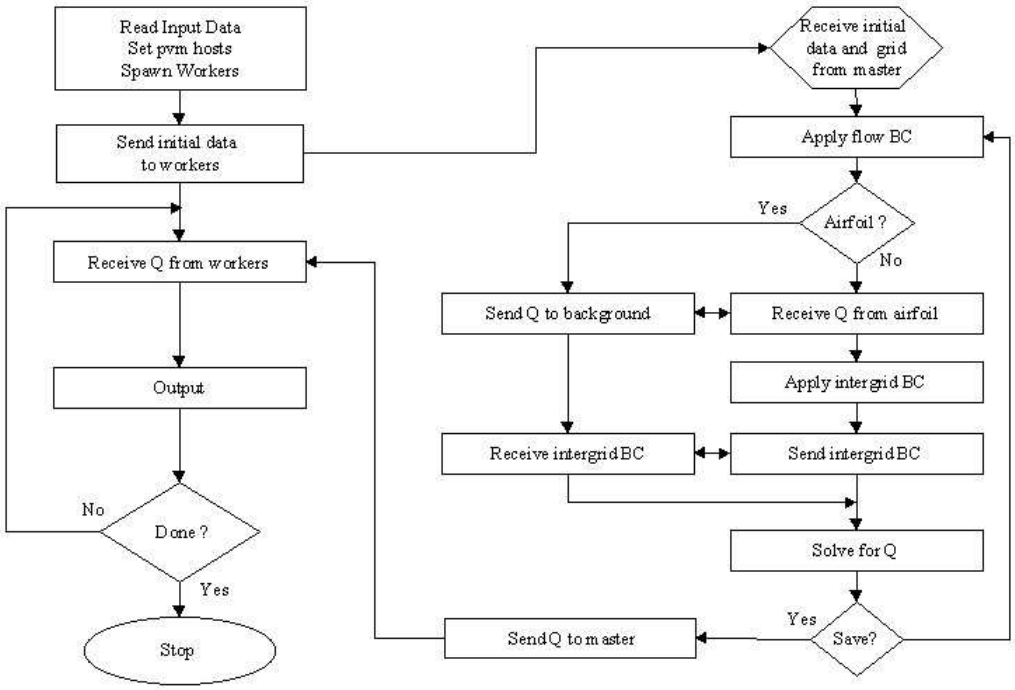


Figure 4.5: Flow chart of the parallel code for the flow solver

grid block assigned to them. Necessary data interchange for the intergrid boundary conditions is carried out between *worker*s in each time step of computation. If it is time to stop or to output, each *worker* sends its sub-result to *master* which combines all the sub-results.

4.5.2 Parallel Algorithm for the Optimization Processes

The parallel algorithm used for the optimization processes is similar to the master-worker paradigm employed in the flow solver. The optimization tool used is constructed of a gradient based optimization algorithm. Consequently, in each step of optimization, the gradient of the objective function is required to be calculated. The evaluation of the gradient, the definition of which is given in Eqn. 3.3, is done using the first order forward finite difference formulation.

In the algorithm, first, the *master* program spawns the *worker* programs to be executed in parallel for computing the flow with perturbed optimization variables. The number of the processors are set according to the number of the optimization variables. As an example, for an optimization process of a single flapping airfoil with a single optimization variable, four processors (two groups of two processors) are spawned by *master*. The first group computes the

unperturbed flow and the second one computes the perturbed flow in terms of the optimization variable. If it is the case of an optimization process of three variables, and again for a single flapping airfoil, then, eight processors (four groups of two processors) are charged. In the biplane configuration in which there are four subdomains decomposed from the overset grids, the number of the processors is doubled. Then, at the end of an unsteady flow computation in parallel as explained in the previous section, each *worker* group sends its objective function value to *master*. Finally, *master* calculates the gradient of the objective function using the first order forward finite difference formulation, and then determines the new flapping motion parameters for the next optimization step. This procedure continues until the gradient of the objective function vanishes within a given accuracy.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Introduction

Unsteady laminar and turbulent flows over flapping airfoils are computed in parallel by using a Navier-Stokes solver on overset grid systems. Validation and preliminary solutions are first obtained for a single flapping airfoil. Two airfoils in a biplane configuration flapping in counter-phase are next considered. Finally, optimization studies of flapping motion parameters are performed. The flows are mostly computed laminar at $Re = 10000$. In the turbulent flow computations, the Baldwin-Lomax and Spalart-Allamaras turbulence models are used.

Unsteady flow solutions are computed as airfoils undergo the periodic flapping motion in plunge and, in pitch and plunge together. The unsteady computations are carried out until a periodic flow solution is obtained. The computed flowfields are analyzed in terms of aerodynamic loads, instantaneous distributions of pressure and flow variables, and unsteady particle traces.

The parallel computation of flowfields is carried out by decomposing the overset grid system into subdomains, each of which is assigned to a processor. Parallel processing is performed in a cluster of computers with dual `Pentium-III` processors of 700 MHz speed, 512 MB memory and `Linux` operating system. The computers are connected to a 100 Mbps switched ethernet network. `PVM` library routines are used for the interprocess communication.

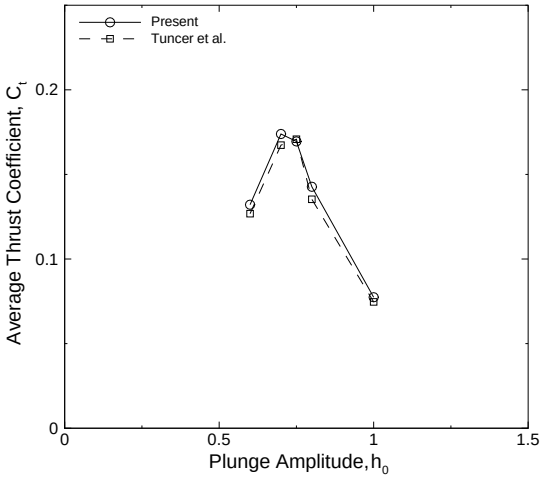


Figure 5.1: Comparison of thrust coefficient with Reference [38]

5.2 Validation Studies

In order to assess the numerical accuracy of the computed flowfields of the present method, that is, the parallel computation with the overset grid system, the flow solutions are compared with the serial single grid solutions performed earlier by Tuncer et al., [38, 39].

Reference [38] gives the computed thrust coefficients for a NACA0012 airfoil undergoing a pure plunge oscillation at various amplitudes and reduced frequencies. The computations are carried out at a freestream Mach Number of $M = 0.3$ and Reynolds Number of $Re = 1 \cdot 10^6$. The flow is assumed to be fully turbulent and the Baldwin-Lomax model is used. A comparison with these solutions is shown in Figure 5.1 for the reduced frequency, $k = \frac{2\pi f c}{U_\infty} = 0.5$. f is the flapping frequency. As seen, the present results agree well with the Reference values.

Table 5.1 shows the thrust coefficients and the efficiencies computed for a NACA0012 airfoil undergoing combined plunge and pitch oscillation with different reduced frequencies and phase angles. The left matrix is from Reference [39]. The pitch oscillation is about the mid-chord. Flowfields are computed at $M = 0.3$ and $Re = 1 \cdot 10^5$. The flow is again assumed to be fully turbulent and solved with the Baldwin-Lomax turbulence model. Although it is observed that the difference between the solutions may differ as much as 10% in cases with low thrust values, the thrust coefficients and the efficiencies given in Table 5.1 are in good agreement.

Table 5.1: Comparison of thrust coefficient and efficiency with Reference [39]

$h_0 = 1, \alpha_0 = 10^o$			
k	ϕ	C_t	η
0.3	90	0.072	0.86
0.3	30	0.116	0.70
1.0	90	0.446	0.25
1.0	30	0.211	0.11

Tuncer et al.[39]

$h_0 = 1, \alpha_0 = 10^o$			
k	ϕ	C_t	η
0.3	90	0.068	0.81
0.3	30	0.108	0.70
1.0	90	0.437	0.23
1.0	30	0.184	0.10

Present

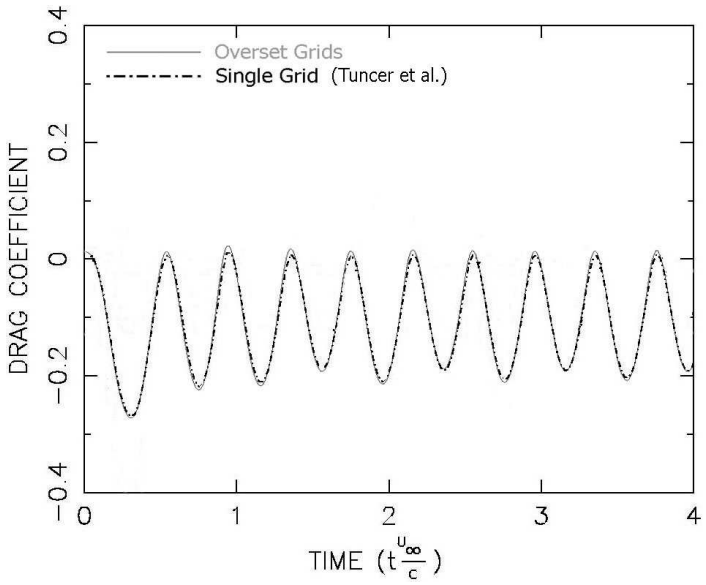


Figure 5.2: Comparison of drag coefficient history with Reference [39]

Figure 5.2 shows the drag coefficient history of a plunging NACA0012 airfoil at a high reduced frequency, $k = 7.85$ with a plunge amplitude of $h_0 = 0.075$. The solution of this flow is given in Reference [39]. The unsteady flow is computed at $M = 0.3$ and $Re = 2 \cdot 10^4$. The eddy-viscosity field is computed by using the Baldwin-Lomax turbulence model. As observed from the comparison in Figure 5.2, agreement between the two solutions indicates again the accuracy of the present parallel overset grid solutions.

5.3 Preliminary Studies

Preliminary studies are carried out to asses the effect of the compressibility and the turbulence models for a single flapping airfoil in pure plunge. For the compressibility analysis, the flow computations are performed at the freestream Mach numbers of $M = 0.1$ and $M = 0.3$. In the

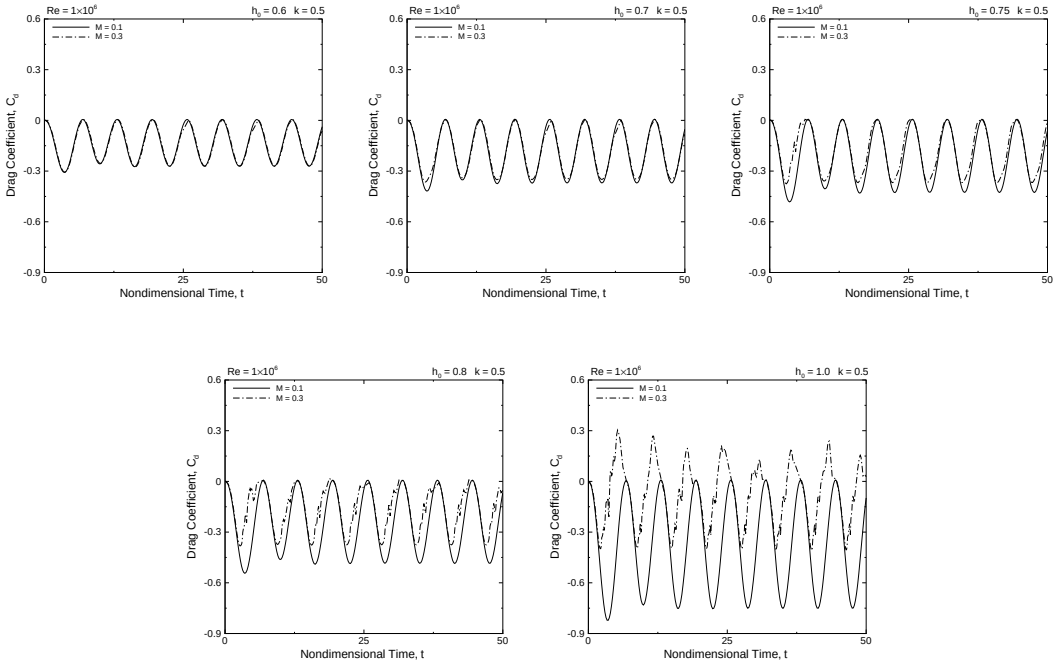


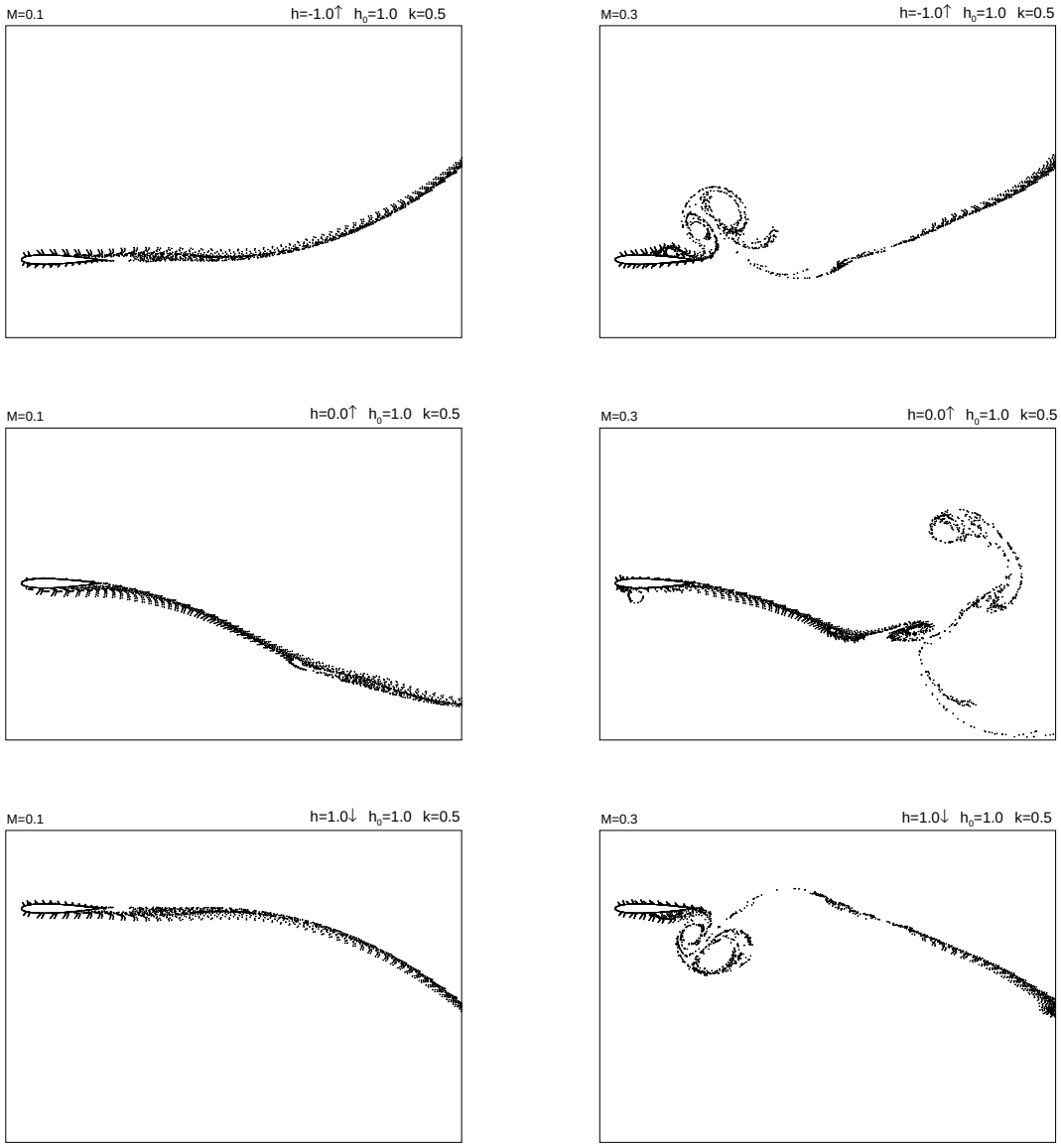
Figure 5.3: Drag coefficient history for $M = 0.1$ and $M = 0.3$

turbulent flow cases, the solutions with the Baldwin-Lomax and Spalart-Allmaras models are compared. Throughout these preliminary investigations, the Reynolds number and the reduced frequency of the flapping motion is set to be $Re = 1 \cdot 10^6$ and $k = 0.5$, respectively.

5.3.1 Effect of Mach Number

The effect of the freestream Mach number on the thrust generation is investigated to explore how the thrust generation and the unsteady flowfield change under the influence of the fluid compressibility. The flowfields around the harmonically plunging `NACA`0012 airfoil are computed at Mach numbers, $M = 0.1$ and $M = 0.3$. The flow is assumed to be fully turbulent and the Baldwin-Lomax model is used.

The drag coefficient histories for the range of $h_0 = 0.6, 0.7, 0.75, 0.8$ and 1.0 are given in Figure 5.3. As seen from the figure, at the plunge amplitudes, $h_0 = 0.6, 0.7$, the drag coefficient histories do not differ significantly. However, in the $M = 0.3$ case, starting from $h_0 = 0.75$, it is observed that the local Mach number exceeds one in a small packet around the leading edge of the airfoil. At high plunge amplitudes, especially at $h_0 = 1.0$, the drag coefficient histories



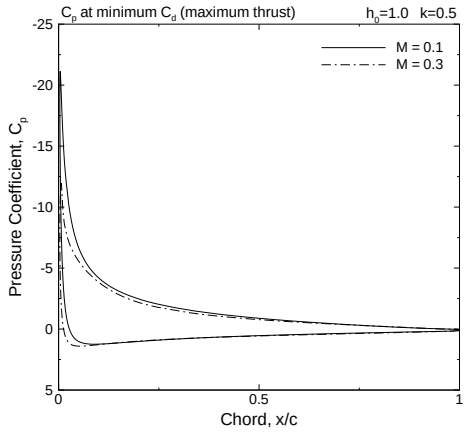
(a) Mach number, $M = 0.1$

(b) Mach number, $M = 0.3$

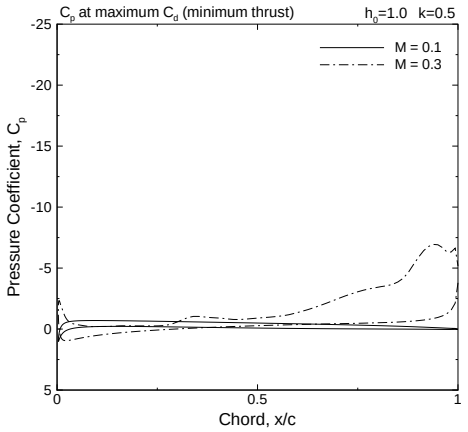
Figure 5.4: Particle traces for $h_0 = 1.0$

show significant variation. It is also observed that the steady periodic solution is not established in the $h_0 = 1.0$ case.

The instantaneous particle traces at three plunge positions for the $h_0 = 1.0$ case are given for both $M = 0.1$ and $M = 0.3$ in Figure 5.4. As seen, in the $M = 0.1$ case, the flow is attached at the leading edge, while in the $M = 0.3$ case, a flow separation at the leading edge and large vortex formations are present. Figure 5.5 compares the instantaneous pressure distributions

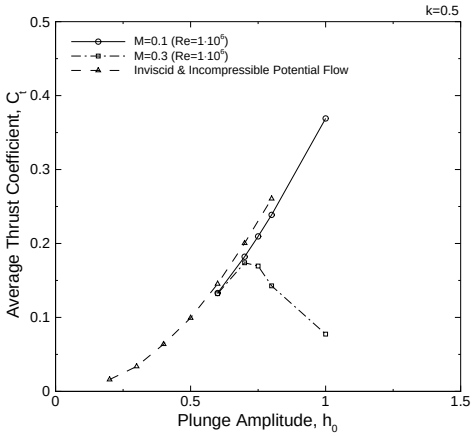


(a) Minimum drag instant

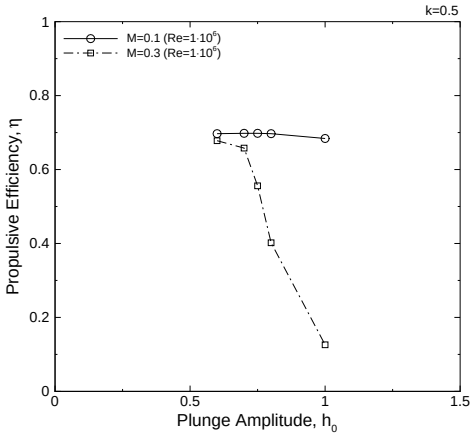


(b) Maximum drag instant

Figure 5.5: Instantaneous pressure distribution for $h_0 = 1.0$



(a) Average thrust coefficient



(b) Propulsive efficiency

Figure 5.6: Thrust and efficiency for $M = 0.1$ and $M = 0.3$

along the airfoil chord at the instant of minimum and maximum drag values for the $h_0 = 1.0$ case. As seen from Figure 5.5(a), the leading edge suction is about 75% greater in the $M = 0.1$ case than that of the $M = 0.3$ case. It is apparent that the flow separation at the leading edge is shock induced. Once the flow separates, the leading edge vortex forms and is convected downstream. The suction induced by the travelling vortex is observed in Figure 5.5(b).

The thrust and propulsive efficiency predictions for the whole plunge amplitude range are shown in Figure 5.6. The potential flow solution in Figure 5.6(a) is from Reference [38]. In the

figure, it is seen that, at low plunge amplitudes ($h_0 \leq 0.7$), the Navier-Stokes predictions for the $M = 0.1$ and $M = 0.3$ cases, and the incompressible potential flow solutions are in agreement. Beyond $h_0 = 0.7$, the thrust value decreases for the $M = 0.3$ case, while it still increases for the $M = 0.1$ case. Similarly, the propulsive efficiency also drops significantly as the plunge amplitude increases beyond 0.7 due to the leading edge vortex formation and subsequent loss of the suction pressure at the leading edge of the airfoil (Figure 5.6(b)).

It may be concluded that the thrust generation is more effective and more efficient at low Mach numbers. As the Mach number increases, the suction field at the leading edge collapses and the flow separates forming leading edge vortices, which in turn reduces the thrust.

5.3.2 Effect of Turbulence Models

In order to asses the influence of the turbulence on the flapping airfoil aerodynamics, numerical studies are carried out by using the Baldwin-Lomax (BL) and Spalart-Allmaras (SA) turbulence models. The BL model is an algebraic model and the SA is a one-equation model. A detailed comparison of turbulence models for the unsteady flows is available in Reference [57].

The unsteady flows over a single flapping `NACA`0012 airfoil in plunge are computed at $M = 0.1$ and $Re = 1 \cdot 10^6$. The reduced frequency is $k = 0.5$, and the non-dimensional plunge amplitudes are chosen to be $h_0 = 0.6, 0.7, 0.75, 0.8$ and 1.0.

Figure 5.7 shows the time history of the drag coefficients for the range of plunge amplitudes. As seen, the drag coefficient histories given by both models are in good agreement except for the $h_0 = 1.0$ case. Figure 5.8 compares the particle traces obtained for the $h_0 = 1.0$ case. In the SA solution, the leading edge vortex formations followed by a flow separation are observed, while in the BL solution, the flow at the leading edge stays attached. This difference is mainly due to the BL model delaying the flow separation. The instantaneous pressure distributions at the instant of minimum and maximum drag values for the $h_0 = 1.0$ case is shown in Figure 5.9. At the instant of maximum thrust, the leading edge suction is slightly greater in the SA solution than that of the BL solution. It may be concluded that the SA model predicts slightly smaller eddy-viscosity, which leads to an earlier flow separation than that of the BL model.

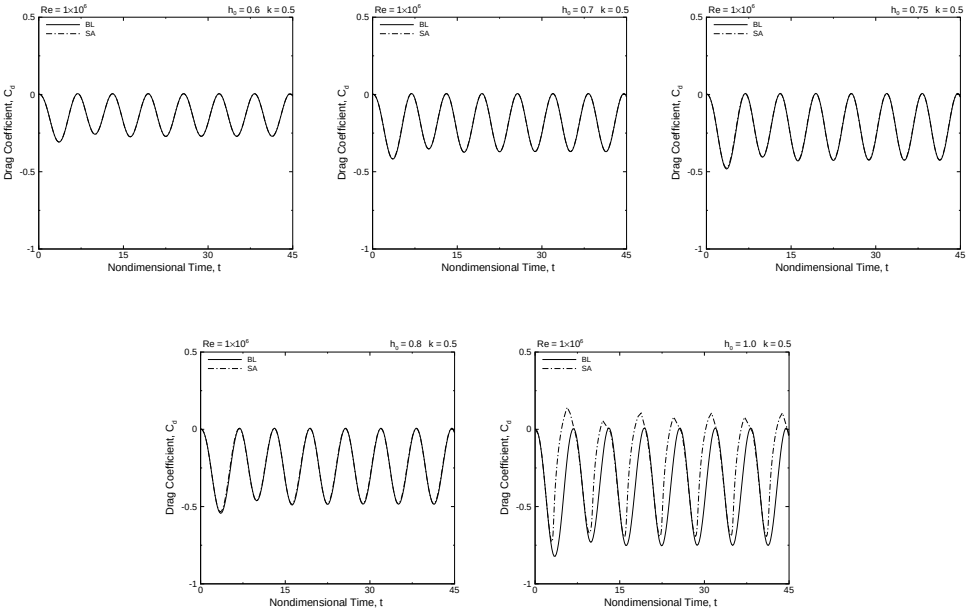
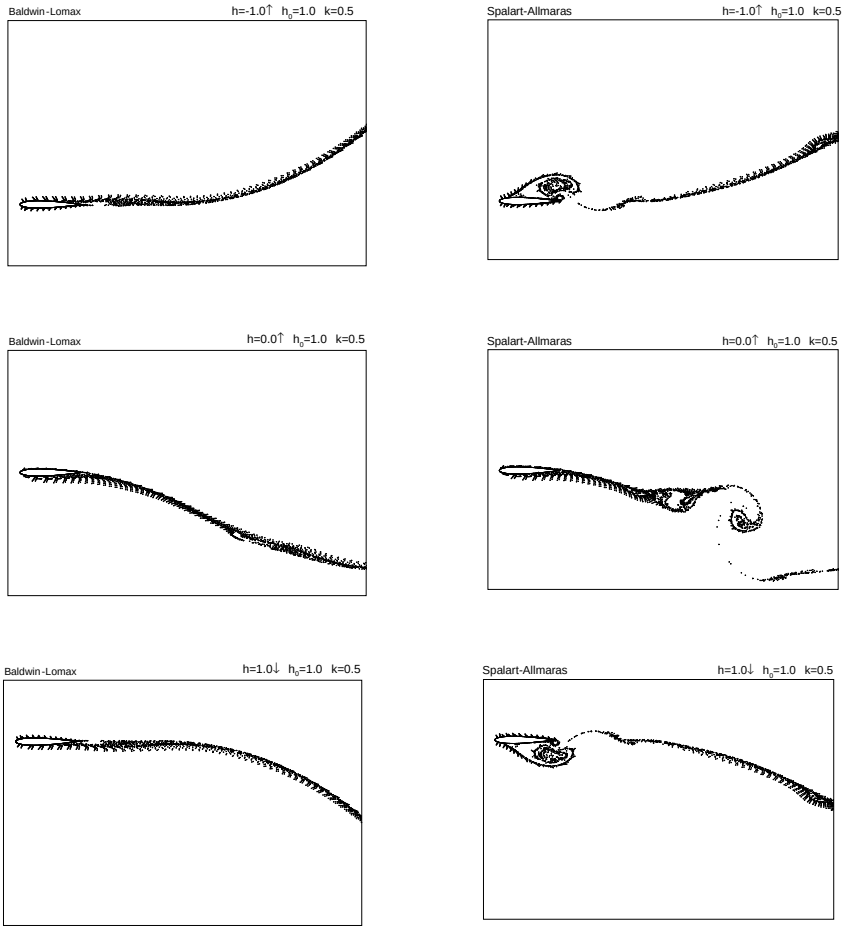
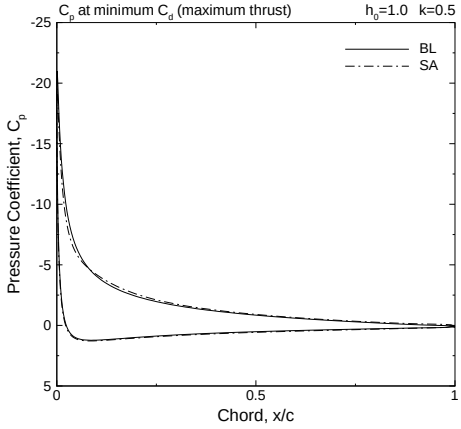


Figure 5.7: Drag coefficient history for *BL* and *SA* turbulence models

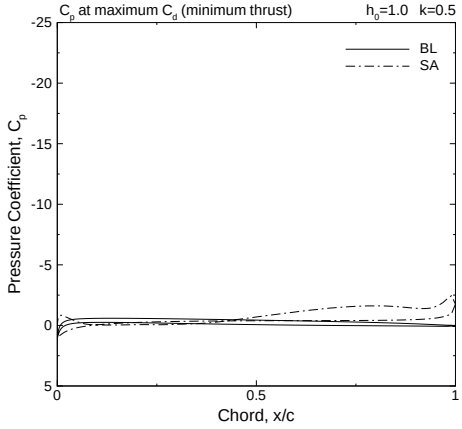


(a) Baldwin-Lomax Model (b) Spalart-Allmaras Model

Figure 5.8: Particle traces for $h_0 = 1.0$

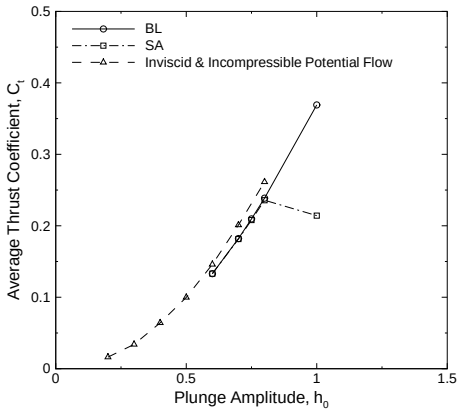


(a) Minimum drag instant

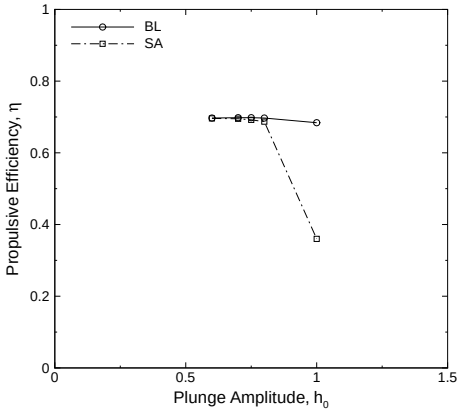


(b) Maximum drag instant

Figure 5.9: Instantaneous pressure distribution for $h_0 = 1.0$



(a) Average thrust coefficient



(b) Propulsive efficiency

Figure 5.10: Thrust and efficiency for *BL* and *SA*

Figure 5.10(a) shows the average thrust coefficients computed by the *BL* and *SA* models, and by the potential flow solution (panel code). The potential flow solution in the figure is from Reference [38]. The propulsive efficiency values are compared in Figure 5.10(b). Except for the plunge amplitude, $h_0 = 1.0$, the *BL* and *SA* models give the same predictions of the thrust and efficiency values. At $h_0 = 1.0$, the flow separation in the flowfield computed by the *SA* model is responsible for the reduction in the thrust and the propulsive efficiency.

It is concluded that, for low plunge amplitudes, unsteady flow solutions computed using the

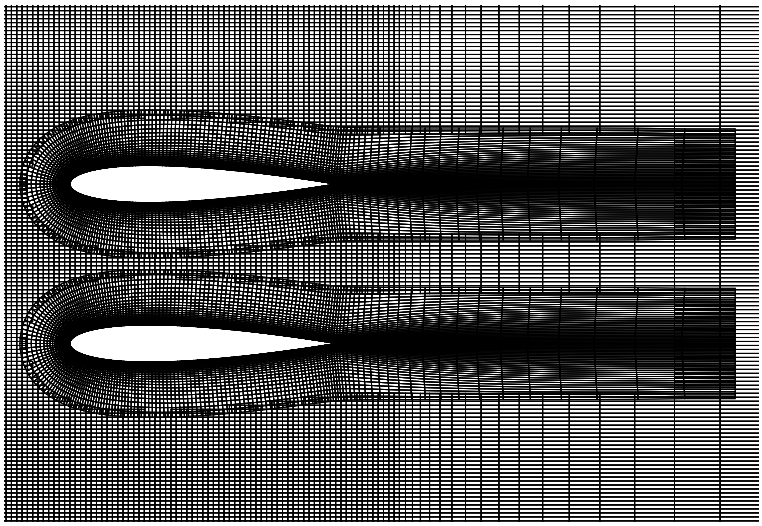


Figure 5.11: Overset grid system for a biplane configuration

BL and *SA* turbulence models are similar. As the plunge amplitude increases, the flowfields computed by both methods apparently give slightly different eddy-viscosity predictions in the flowfield, which affects the unsteady aerodynamic characteristics significantly.

5.4 Biplane Configuration

Based on the experimental studies[6], the thrust generation of flapping airfoils is enhanced in biplane configuration as the airfoils undergo an opposed flapping motion. In the earlier studies with flapping airfoils[35, 39], the flapping motion may be implemented either by locally deforming the grid around moving airfoils or employing overset grids. The grid deformation may impose restrictions on the flapping amplitude, or introduce inaccuracies due to the reduced grid quality. Overset grids is therefore an alternative to impose the flapping motion, and is employed in this study. This section presents the unsteady flow computations for a biplane configuration as the airfoils undergo pure plunge, and combined pitch and plunge oscillations in counter-phase.

5.4.1 Pure Plunge Motion

In the following unsteady flow solutions for a biplane configuration of `NACA`0014 airfoils, the mean distance between the airfoils is set to $y_0 = 1.4$ and the plunge amplitude, to $h_0 = 0.4$ as in

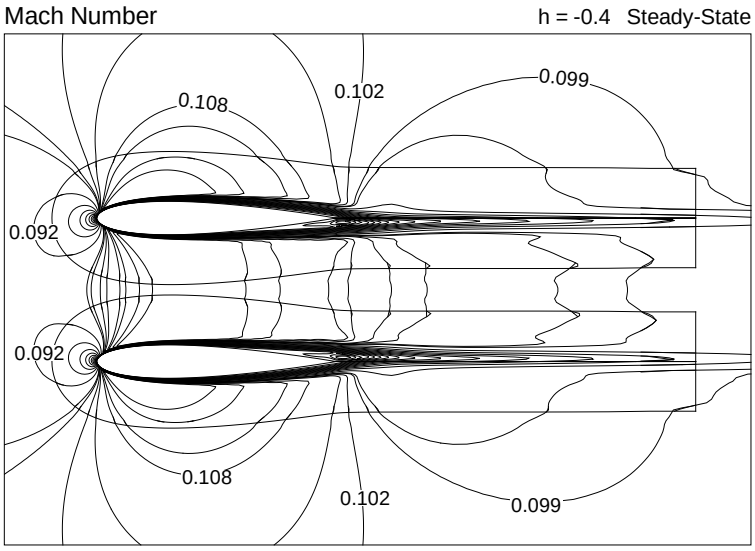
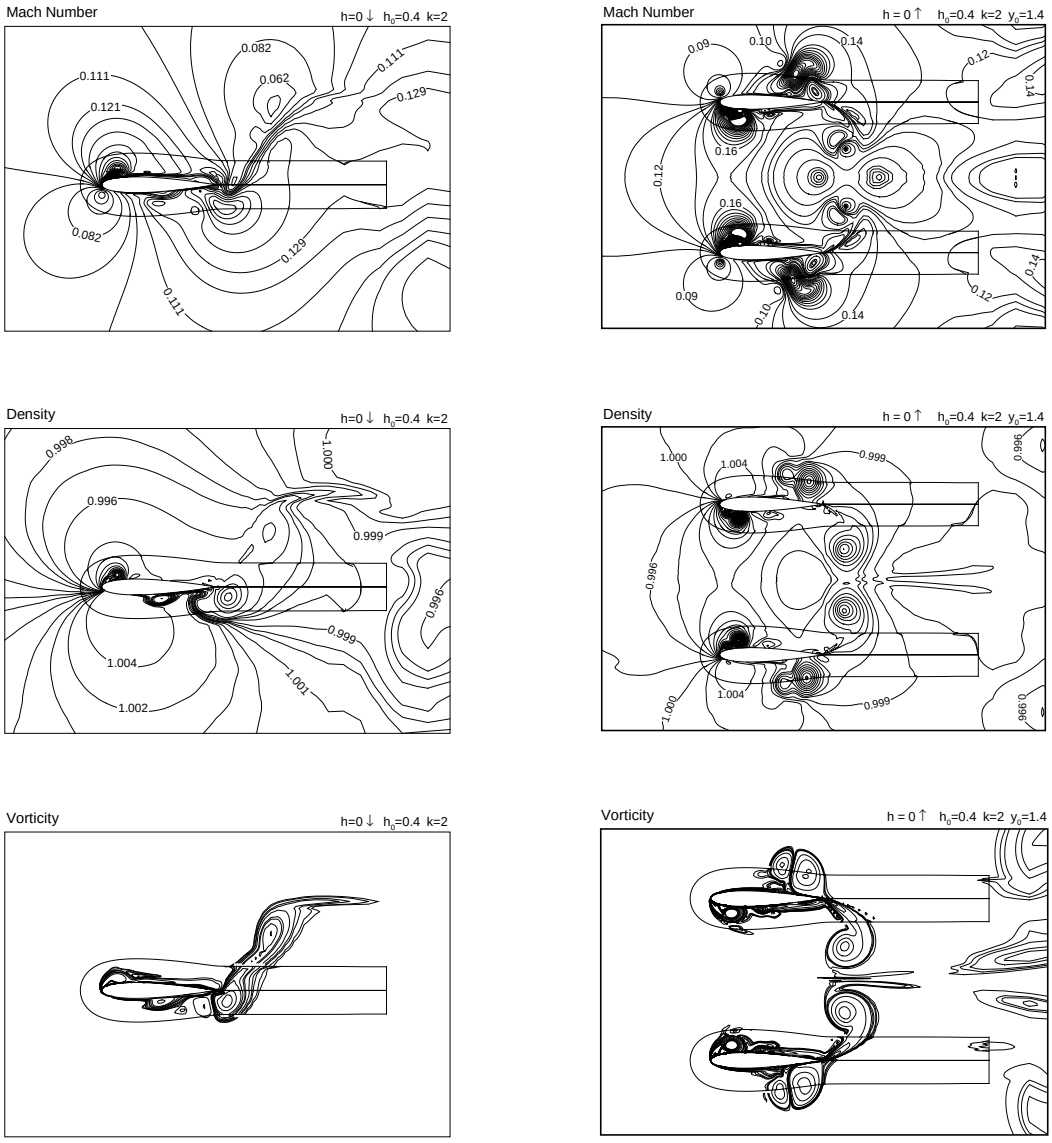


Figure 5.12: Steady flowfield

the experimental studies performed by Jones et al.[6]. The overset grid system shown in Figure 5.11 is used. The airfoil and background grids are of 141×31 and 135×262 size, respectively. Although the flowfield is symmetric about the middle plane in the crossflow direction, the full domain is discretized to avoid the application of the numerical symmetry condition, and to asses the accuracy of the computations. In the parallel computations, the background grid is partitioned into two at the symmetry plane. The computational domain is then decomposed into a total of four subgrids. This procedure is explained in more detail in Chapter 4.

The flow is computed at $M = 0.1$ and $Re = 1 \cdot 10^4$ assuming laminar flow. The steady flowfield is first computed to provide the starting solution for the unsteady flow (Figure 5.12). It is noted that the boundary layers grow considerably thick at this low Reynolds number laminar flow. The unsteady flow solutions for a range of reduced frequency values are computed for about 5 periods of the flapping motion, in which periodic flow conditions are established. A typical parallel computation, which is distributed over 4 processors, takes about 20 `CPU` hours.

Figure 5.13 shows the computed flowfield at the mean plunge position for $k = 2$. The particle traces, which originate from the leading edge of the airfoils, are shown in Figure 5.14 at three plunge positions. The figures reveal the presence of the large leading and trailing edge vortices forming and convecting downstream, similar to the single airfoil case. However, the



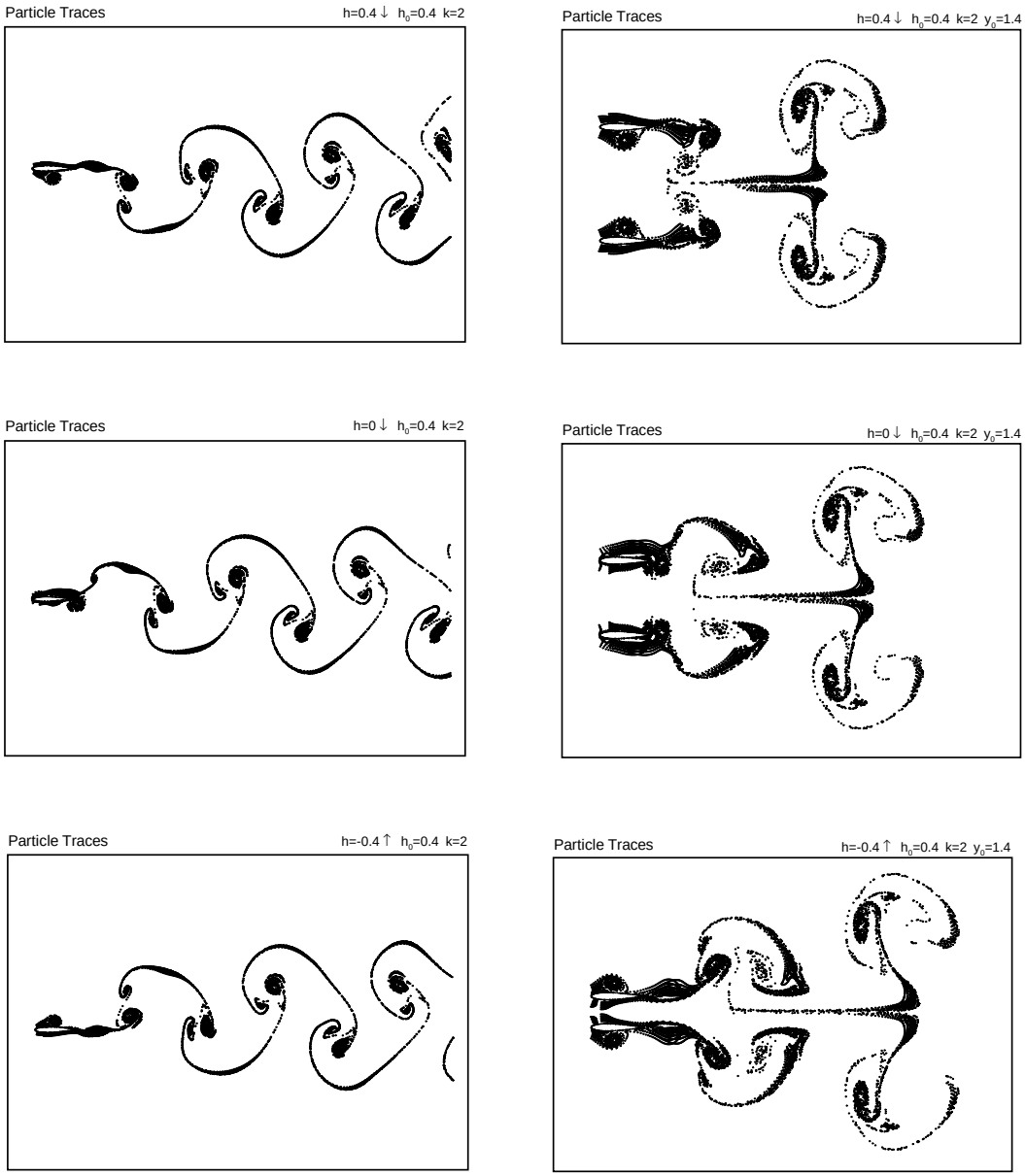
(a) Single Airfoil

(b) Biplane Configuration

Figure 5.13: Instantaneous unsteady flowfield

flow in the wake shows a considerably different profile due to the interaction of the airfoil wakes. It is also noted that the computed flowfield is symmetric about the mid-plane. The instantaneous plunge positions and directions given in the figures are with respect to the upper airfoil. The lower airfoil plunges in the opposite direction.

The time variation of the drag coefficient and its comparison to that of the single airfoil (Figure 5.15) show that the unsteady thrust production is enhanced in the biplane configuration.



(a) Single airfoil (b) Biplane configuration

Figure 5.14: Particle traces

The instantaneous thrust value in biplane configuration is considerably larger than that of the single airfoil case as the airfoils plunge toward each other at $h \approx 0 \downarrow$ for the upper airfoil ($h \approx 0 \uparrow$ for the lower airfoil). The additional thrust is attributed to the larger suction at the leading edge of the airfoils due to the ground effect as shown in the pressure distribution (Figure 5.16). Note that the variation of drag coefficients for the upper and lower airfoils in the biplane configuration do overlap, showing perfect symmetry in the solution.

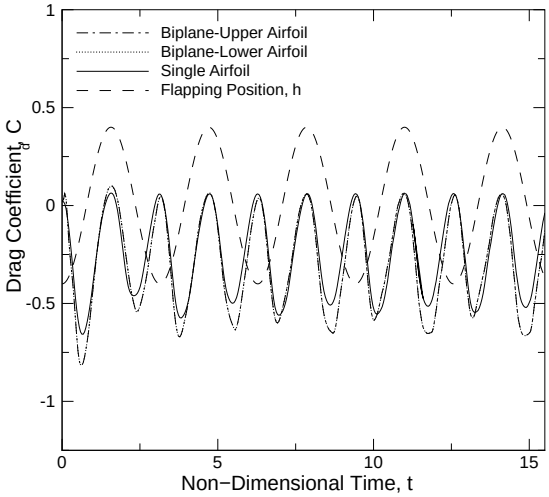


Figure 5.15: Time history of drag coefficient ($h_0 = 0.4, k = 2$)

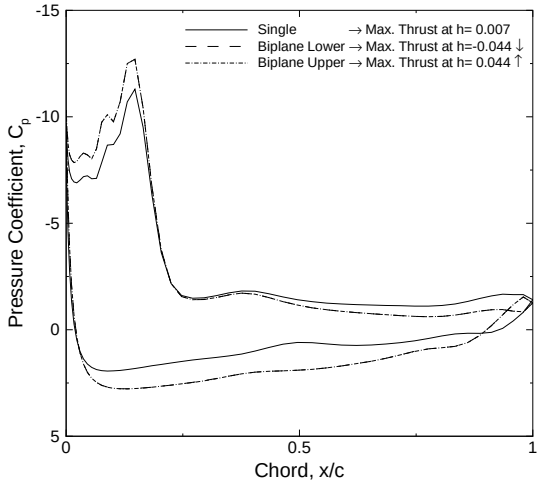


Figure 5.16: Pressure coefficient distribution at maximum thrust ($h_0 = 0.4, k = 2$)

In Figure 5.17, the average thrust coefficients obtained for a range of reduced frequency values, $k = 0.5 - 3$, are also compared to that of the single flapping airfoil, the experimental data and the panel code[6]. To facilitate a direct comparison with the experimental data, average thrust coefficient, here, is defined as $C_t = -(C_d - C_{d,steady})$. As shown the thrust enhancement due to biplane configuration over the single flapping airfoil becomes more pronounced at frequencies higher than $k = 2$, where the enhancement is about 20%. At $k = 0.5$, the thrust coefficient in the biplane configuration is even slightly less that of single flapping airfoil. The present numerical predictions also compare well with the experimental data, especially at high

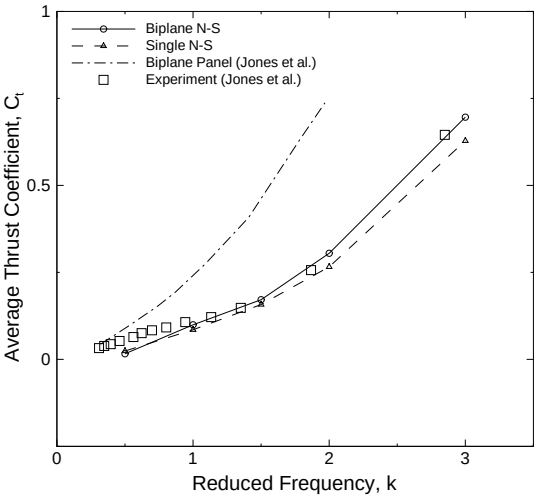


Figure 5.17: Average thrust coefficient

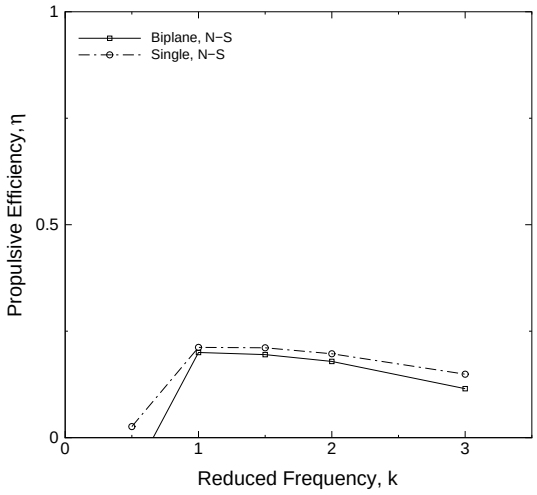
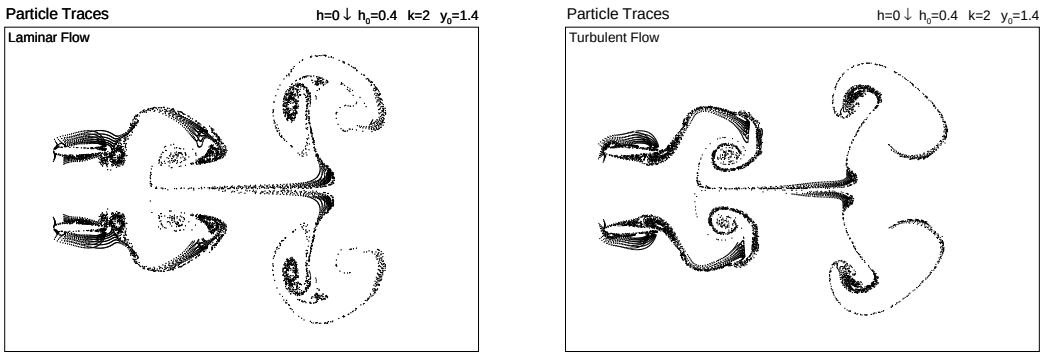


Figure 5.18: Propulsive efficiency

reduced frequencies. However, the experimental data show a higher thrust at low frequencies. As expected the panel code solution, which assumes a fully attached flow over airfoils, overpredicts the average thrust considerably.

The computed propulsive efficiencies of the biplane configuration are compared to those of the single airfoil in Figure 5.18. In contrast to the thrust enhancement seen in Figure 5.17, it is observed that the propulsive efficiency is lower in the biplane configuration. It appears that slightly more power is required to impose the flapping motion (against the lift) in biplane configuration due to the ground effect.



(a) Laminar flow (b) Turbulent flow

Figure 5.19: Particle traces

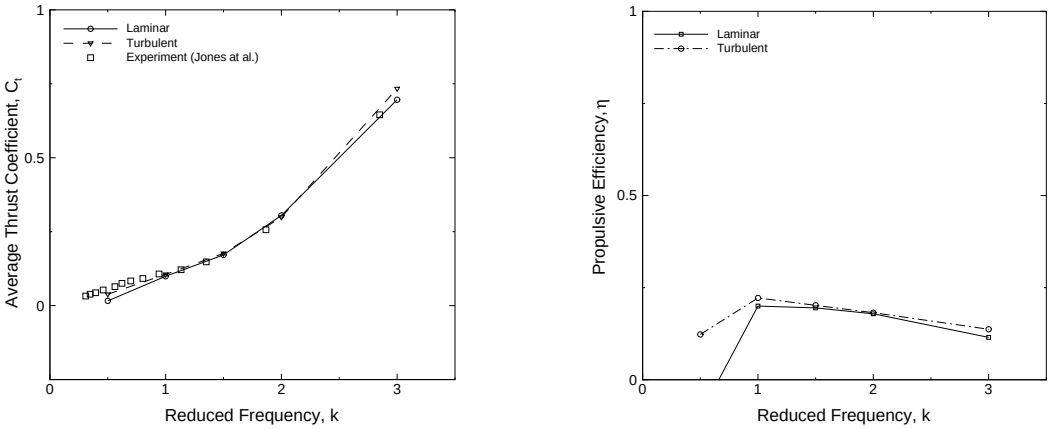


Figure 5.20: Average thrust coefficient and propulsive efficiency

The effect of turbulence on the thrust generation is next investigated. Reynolds number and Mach number are again, $Re = 1 \cdot 10^4$ and $M = 0.1$, respectively. The turbulence model used is the Spalart-Allmaras model. In Figure 5.19, the particle traces in the case of turbulent flow are compared to the particle traces in laminar flow. Average thrust coefficients and propulsive efficiencies are given in Figure 5.20 and compared to those of laminar flow predictions and experiment[6]. Since the turbulence in the flow delays the flow separation, a slight increase in the thrust value is observed. As expected, the turbulent flow solutions predict higher efficiency for all frequencies.

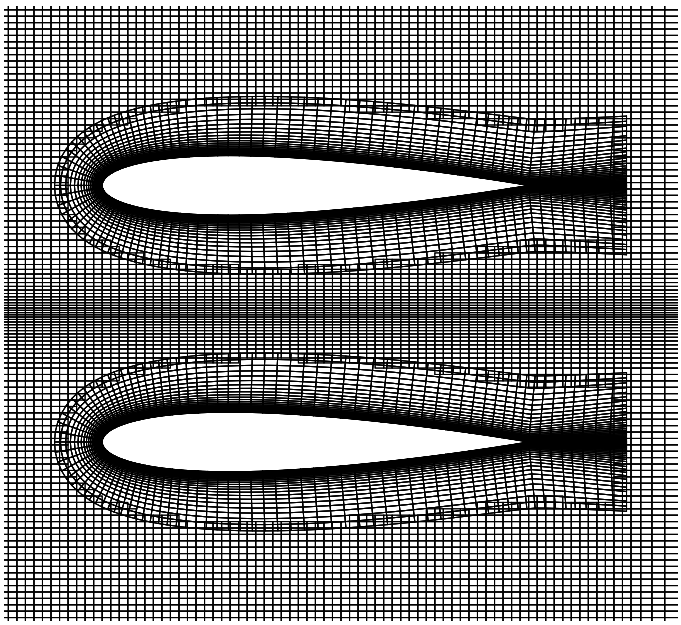


Figure 5.21: Overset grid system for combined pitch and plunge motion

5.4.2 Combined Pitch and Plunge Motion

In the combined pitch and plunge motion of the airfoils in biplane configuration, the distance between the moving airfoils has to be considered more carefully because of the in-phase pitch motion with the plunge motion. Due to the constraint on the distance between the symmetry line of the airfoils and the outer boundary of the airfoil grids, an airfoil grid of smaller size than the one presented in Section 5.4.1 is used in this study. Figure 5.21 shows the airfoil and background grids, which are of 117×27 and 135×276 size, respectively. The unsteady, laminar and turbulent flowfields are computed at $M = 0.1, Re = 1 \cdot 10^4$ for $y_0 = 1.4$, $h_0 = 0.4$, $k = 0.5, 1.0$, $\alpha_0 = 5^o, 10^o$ and a range of phase angles between $\phi = 0^o - 180^o$.

Figures 5.22 and 5.23 give the flowfields for $k = 0.5$, $\phi = 120^o$, and $\alpha_0 = 5^o$ and $\alpha_0 = 10^o$, respectively, at the plunge position, $h = 0$. The flowfields computed for $k = 1.0$ with all the other flapping parameters remaining the same are shown in Figures 5.24 and 5.25. Figure 5.26 gives the time history of the drag coefficient for $k = 0.5$ and $k = 1.0$. In Figure 5.27, the computed average thrust coefficients are compared to those of the pure plunge motion (i.e., $\alpha_0 = 0^o$) studied in Section 5.4.1 for $k = 0.5$ and $k = 1$. The average thrust coefficient is again defined as $C_t = -(C_d - C_{d,steady})$ for a direct comparison of the results in Section 5.4.1.

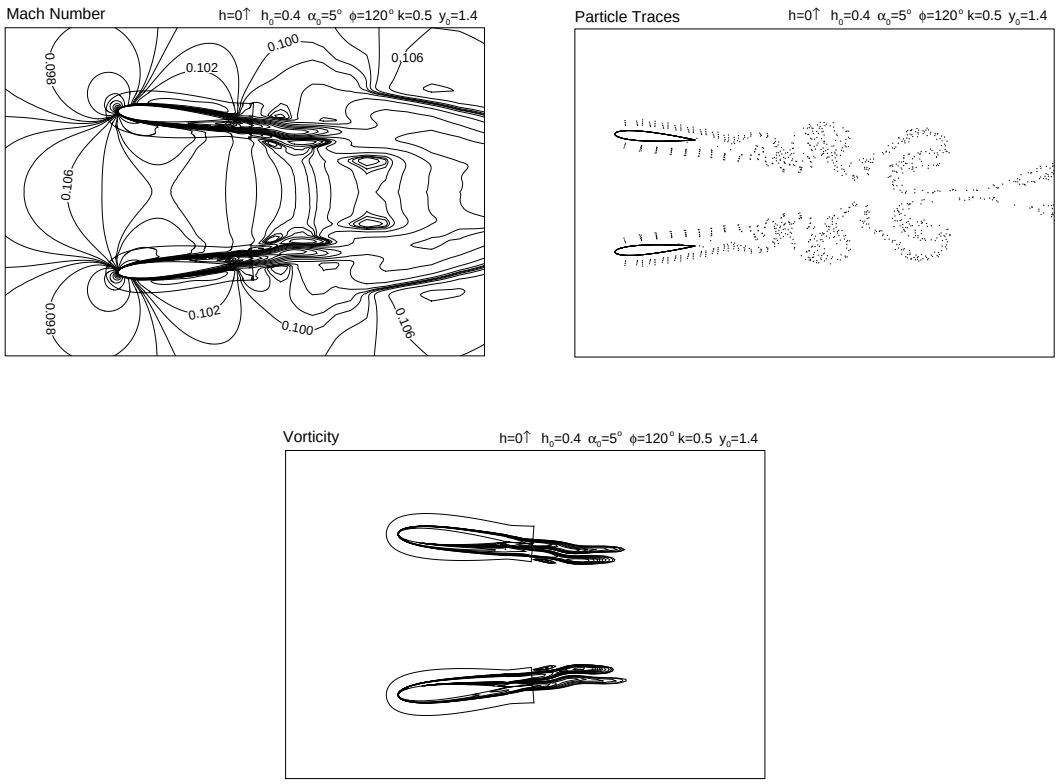


Figure 5.22: Instantaneous flowfield at $k = 0.5, \alpha_0 = 5^o, \phi = 120^o$

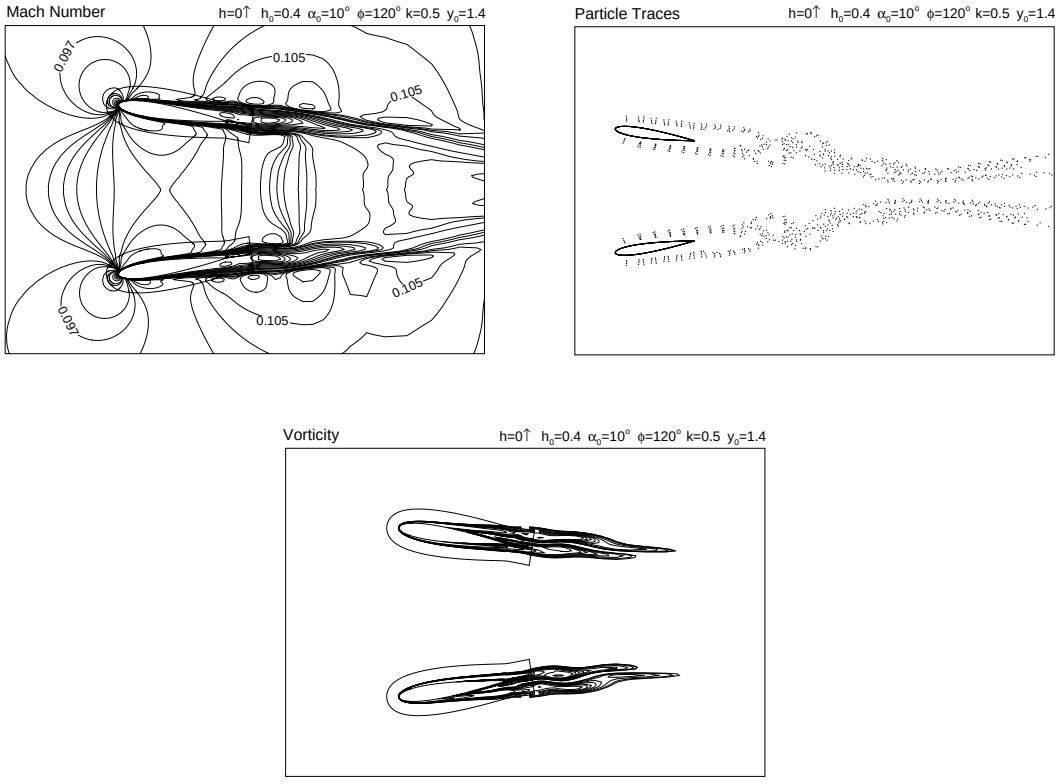


Figure 5.23: Instantaneous flowfield at $k = 0.5, \alpha_0 = 10^o, \phi = 120^o$

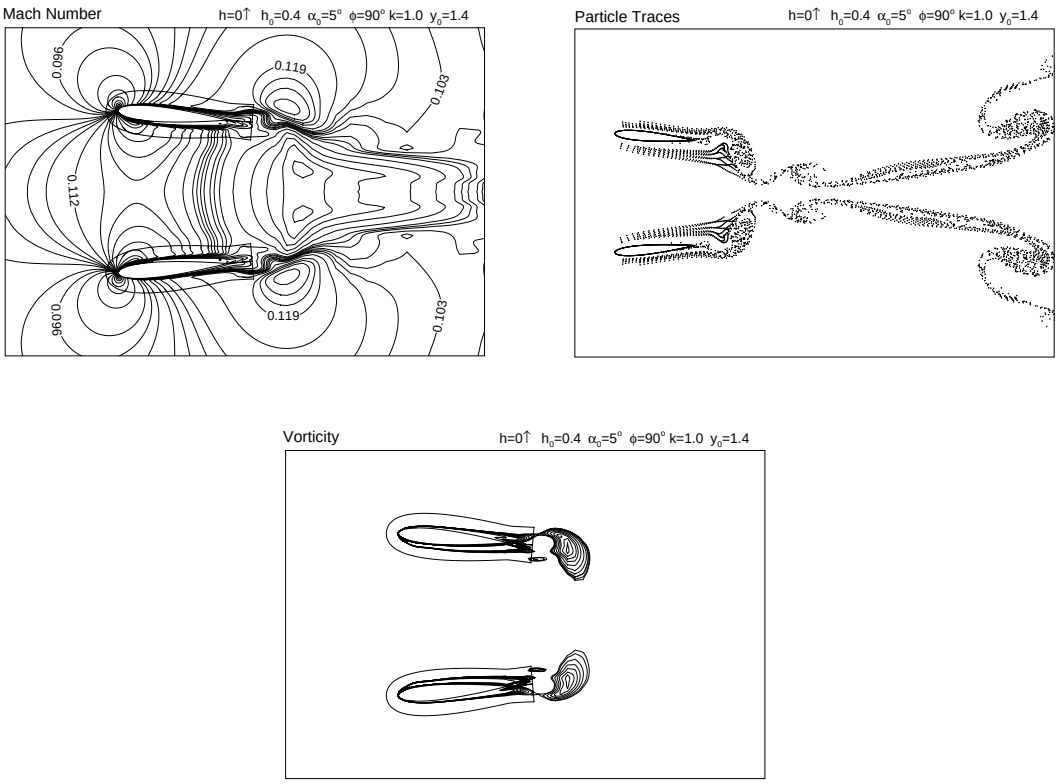


Figure 5.24: Instantaneous flowfield at $k = 1.0, \alpha_0 = 5^o, \phi = 90^o$

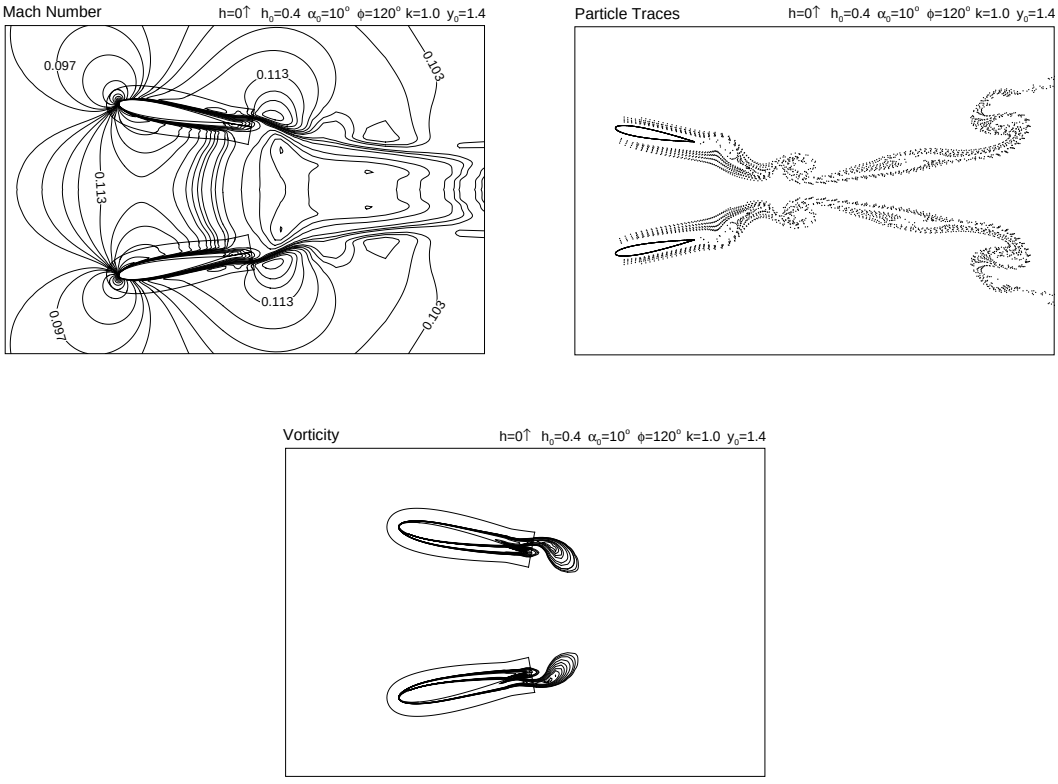
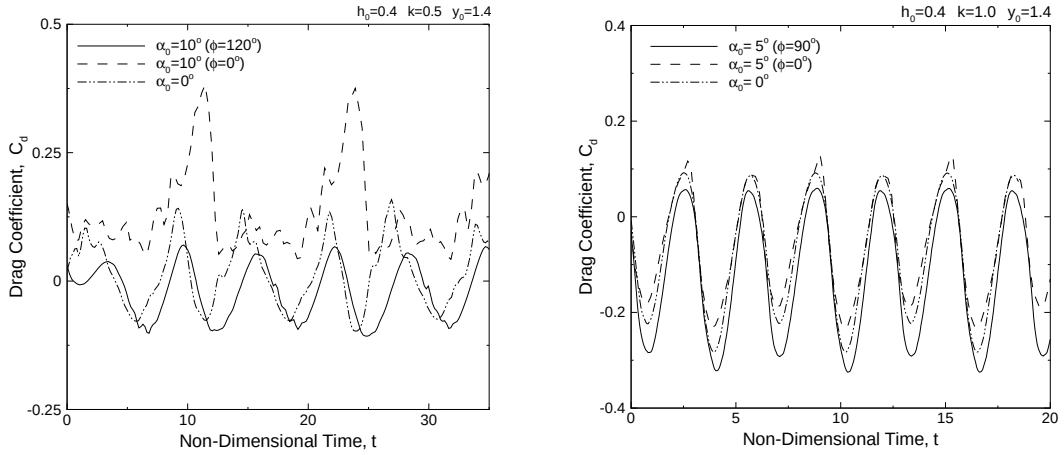


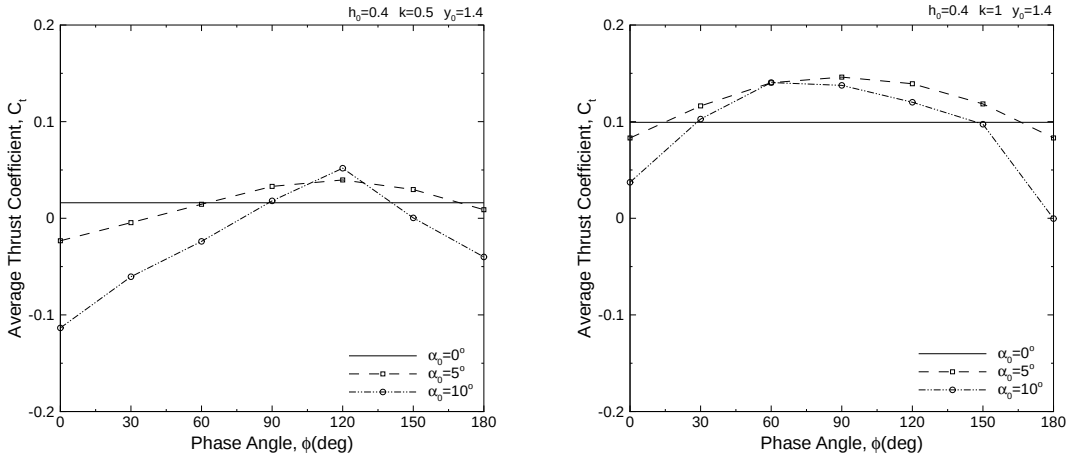
Figure 5.25: Instantaneous flowfield at $k = 1.0, \alpha_0 = 10^o, \phi = 90^o$



(a) Reduced frequency, $k = 0.5$

(b) Reduced frequency, $k = 1.0$

Figure 5.26: Drag history of biplane configuration at combined pitch and plunge motion

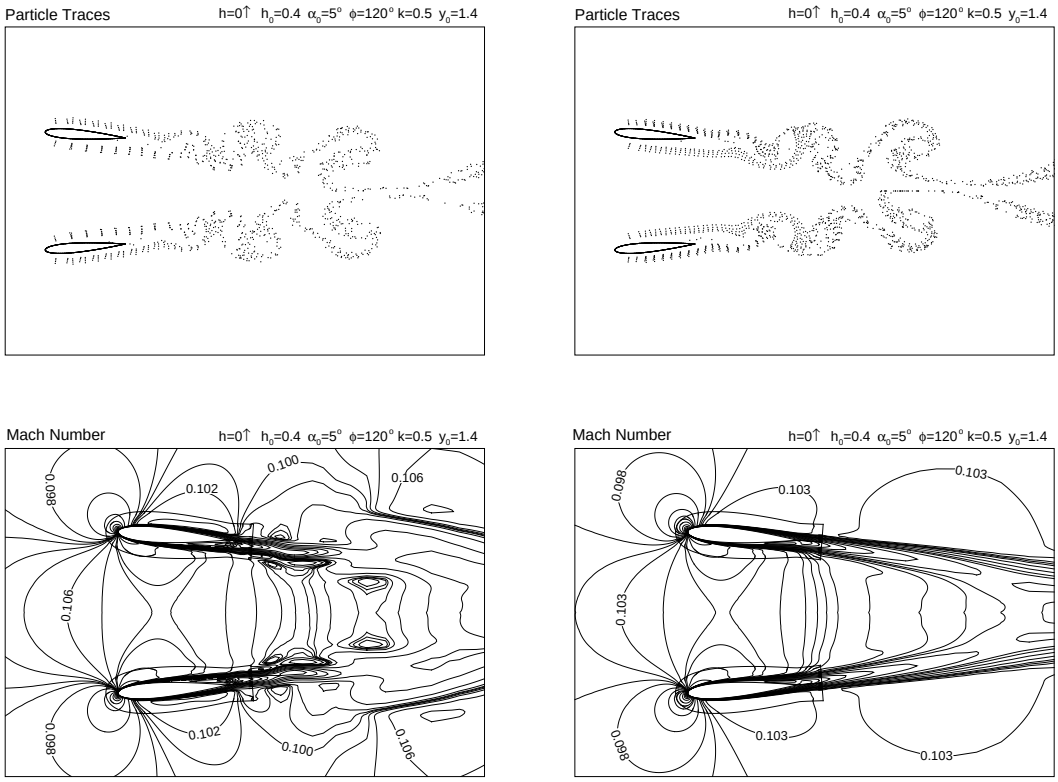


(a) Reduced frequency, $k = 0.5$

(b) Reduced frequency, $k = 1.0$

Figure 5.27: Thrust of biplane configuration at combined pitch and plunge motion

The results clearly show that the combined pitch and plunge oscillations of airfoils in the biplane configuration enhances the thrust significantly. The enhancement of the average thrust is more than 100% near $\phi = 120^o$ when the frequency is, $k = 0.5$ and about 50% at $\phi = 90^o$ when the frequency is $k = 1$. These preliminary results show that the thrust produced by flapping airfoils may further be enhanced for certain phase angles between the plunge and pitch motions. It is also noted that for some phase angles flapping airfoils do not produce thrust at all.



(a) Laminar flow

(b) Turbulent flow

Figure 5.28: Instantaneous flowfield for $k = 0.5, \alpha_0 = 5^o, \phi = 120^o$

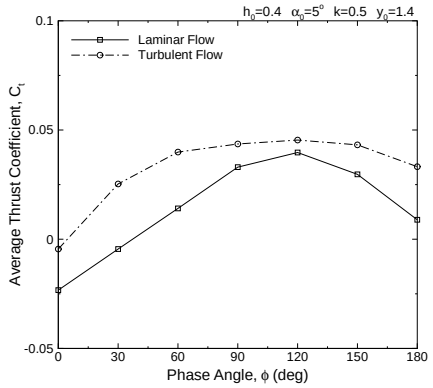


Figure 5.29: Thrust coefficient for combined pitch and plunge motion (Laminar and Turbulent)

Finally, turbulent flow computations are carried out at $\alpha_0 = 5^0$, $k = 0.5$, $Re = 1 \cdot 10^4$ and $M = 0.1$. Figure 5.28 shows the instantaneous flowfields at the plunge position $h = 0$ computed with the laminar and turbulent flow assumptions. The thrust comparison is given in Figure 5.29. Significant enhancement of the thrust generation along the phase angle range is observed.

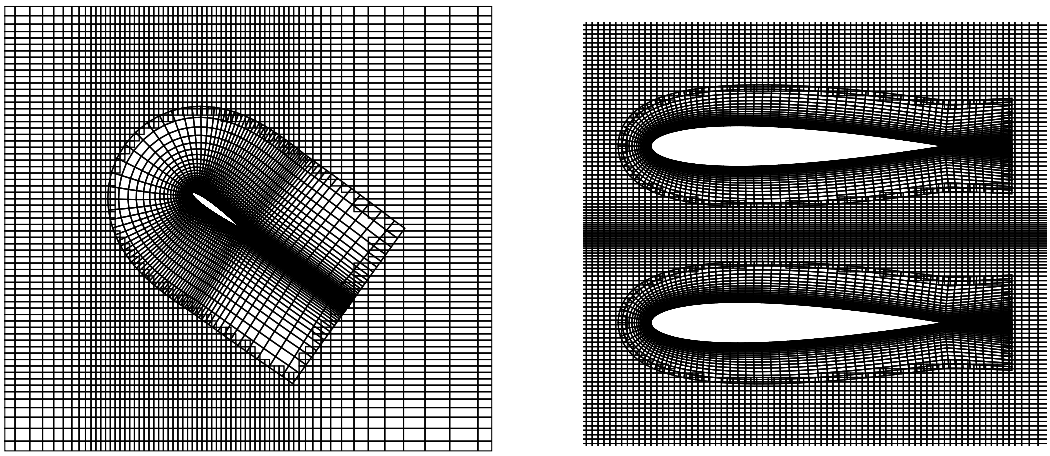


Figure 5.30: Overset grids used in optimization

5.5 Optimization Studies

The numerical optimization algorithm described in Chapter 3 is implemented for the flapping motion parameters to maximize the thrust and/or the propulsive efficiency of flapping airfoils. The optimization algorithm is based on the steepest ascent direction of the objective function, which is either the thrust or the linear combination (See Section 3.3) of the thrust and the propulsive efficiency.

The computational and experimental findings show that the thrust generation and the propulsive efficiency of flapping airfoils are closely connected to the flapping motion and the flow parameters, such as the unsteady flapping velocity, the frequency and the amplitude of the pitch and plunge motions, the phase shift between them, and the air speed. It is apparent that to maximize the thrust and/or the propulsive efficiency of a flapping airfoil, an optimization of all the above variables is needed. In this study, the optimization variables are taken to be the plunge and pitch amplitudes, the flapping frequency and the phase shift between the pitch and plunge motions. Figure 5.30 shows the overset grid systems used in the optimization studies with the single airfoil and the dual airfoils in a biplane configuration.

The components of the gradient vector of the objective function are evaluated in parallel after an unsteady flowfield computation for each optimization variable. A typical parallel computation for an optimization process takes about $20 - 30$ hours of wall clock. It should

Table 5.2: Thrust optimization cases and variables

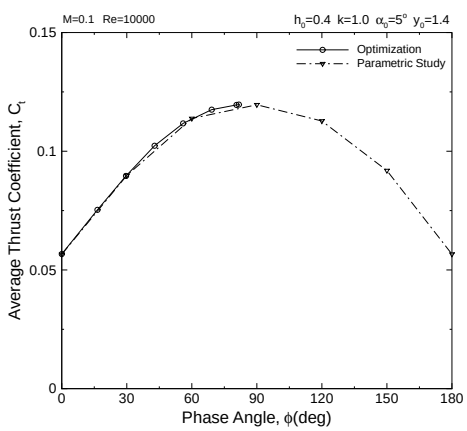
Case	Re	M	k	h_0	α_0	ϕ
1	$1 \cdot 10^4$	0.1	1.0	0.4	5.0^o	V
2	$1 \cdot 10^6$ Tur.	0.3	0.5	V	0.0	0.0
3	$1 \cdot 10^4$	0.1	1.0	V	0.0	0.0
4	$1 \cdot 10^4$	0.1	V	V	0.0	0.0
5	$2 \cdot 10^4$	0.2	V	V	0.0	0.0
6	$3 \cdot 10^4$	0.3	V	V	0.0	0.0
7	$1 \cdot 10^4$	0.1	1.0	V	V	V
8	$2 \cdot 10^4$	0.2	1.0	V	V	V
9	$3 \cdot 10^4$	0.3	1.0	V	V	V
10	$1 \cdot 10^4$ Tur.	0.1	1.0	V	V	V

be noted that the total duration of an optimization process does not depend on the number of optimization variables since the components of the gradient vector for each optimization variable are calculated in parallel.

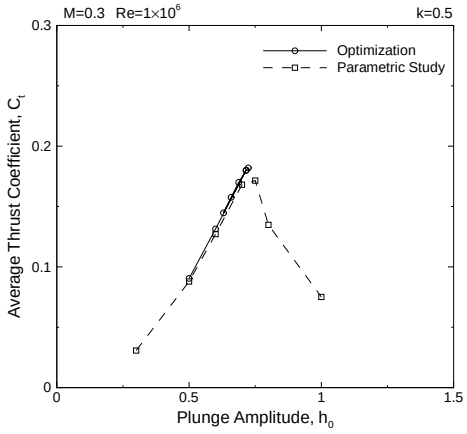
5.6 Maximization of Thrust Generation

Optimization studies are performed to maximize the thrust generation of a single flapping NACA0012 airfoil and dual flapping NACA0014 airfoils in a biplane configuration. Table 5.2 summarizes the optimization cases studied. V denotes the optimization variable used in the case study. All the cases except Case 1 are the single airfoil cases. In Cases 2 and 10, flows are assumed to be turbulent and, Baldwin-Lomax and Spalart-Allmaras turbulence models are employed respectively.

Figure 5.31(a) (Case 1) compares the result of an optimization study for a biplane configuration with a parametric study done earlier (Section 5.4.2). In this case, the optimization variable is the phase angle between the plunge and pitch motions. As shown, the optimization algorithm works well. The phase angle which maximizes the thrust is reached along the optimization steps. In Case 2, the plunge amplitude, h_0 is the only optimization variable. Optimization steps are given in Figure 5.31(b), and compared with the parametric study presented in Figure 1.6. The plunge amplitude, $h_0 = 0.72$, that maximizes the thrust is similarly approached rapidly.



(a) Dual airfoils in a biplane configuration in pitch and plunge (ϕ is optimized)



(b) Single airfoil in plunge (h_0 is optimized)

Figure 5.31: Comparison of optimization results with parametric studies

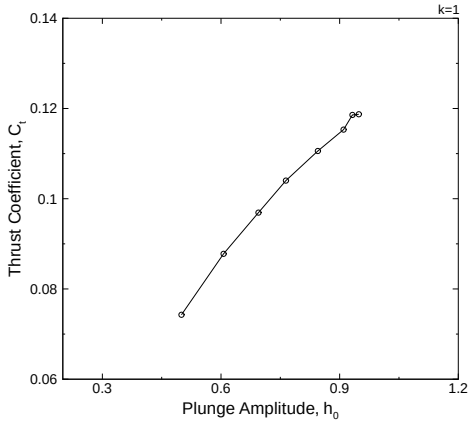
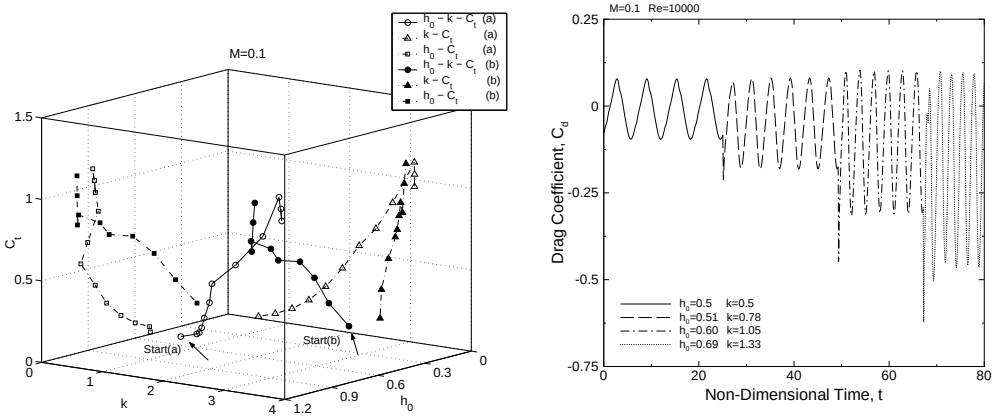


Figure 5.32: Thrust optimization with a single parameter

In the following optimization cases, low Reynolds number, laminar flows are considered to match the Reynolds number of the experimental studies performed by Jones et al.[6]. In Case 3, another single variable optimization, for h_0, is considered. As seen from Figure 5.32, the optimization process starts from $h_0 = 0.5$ and rapidly converges to $h_0 = 0.95$ where the thrust is maximized.

In Case 4, a two-variable optimization is performed for h_0 and k at $M = 0.1$. Two sets of starting conditions for the optimization variables are employed for the optimization process: In the first case, $h_{0,start} = 0.5$ and $k_{start} = 0.5$, and in the second case, $h_{0,start} = 0.2$ and $k_{start} = 2.5$. The optimization steps are shown in Figure 5.33(a). Two optimization processes



(a) Optimization steps

(b) Unsteady drag (-thrust) coefficient

Figure 5.33: Optimization Case: 4

started from

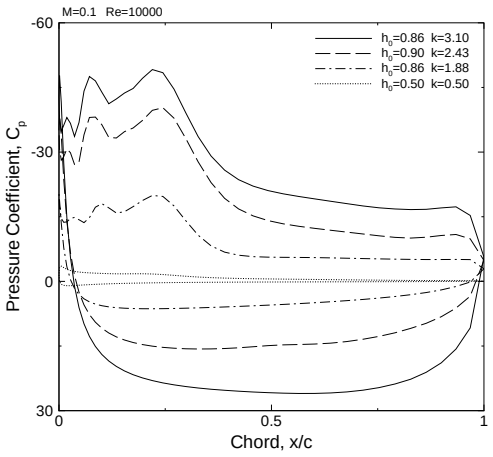


Figure 5.34: Instantaneous pressure distributions at maximum thrust, Case 4

two different initial conditions are shown to converge about the same k and h_0 values. At this low Mach number flow, very high values of thrust are computed as the k and h_0 values are increased along the optimization steps. The variation of the unsteady drag (negative thrust) coefficient along a few optimization steps is shown in Figure 5.33(b). As the plunge amplitude and frequency values are incremented along the optimization steps, the unsteady computations are carried out for a few periods of the flapping motion until a periodic behavior is obtained. The instantaneous surface pressure distributions and Mach number contours at the minimum drag (maximum thrust) positions along the optimization steps are given in Figures 5.34 and

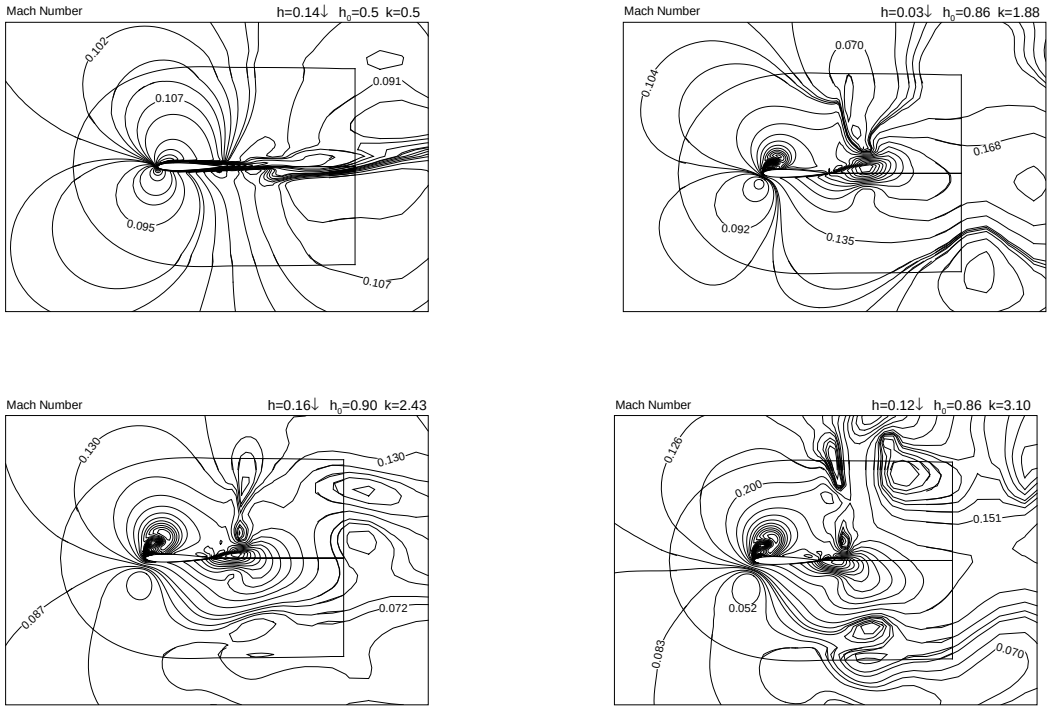
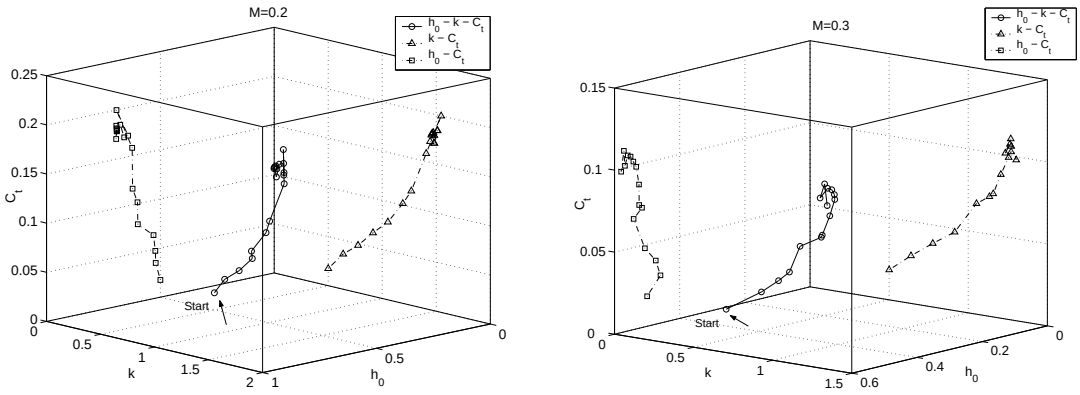


Figure 5.35: Instantaneous Mach number distributions at maximum thrust, Case 4

5.35, respectively. As observed, the maximum thrust, which is mostly due to the high suction induced at the leading edge, is produced as a strong vortex forms at the leading edge of the airfoil.

The optimization run in Case 4 is next repeated for $M = 0.2$ and $M = 0.3$ as shown in Figure 5.36. A similar trend is observed. However, as expected, the maximum average thrust values computed are significantly smaller than those of the $M = 0.1$ case, and they are reached at relatively low k and h_0 values (Figure 5.37), which is in agreement with the results given earlier in Section 5.3.1. The pressure distributions at the instants of maximum thrust for Cases $4-6$ are compared in Figure 5.38. It is again noted that compressibility plays a significant role in thrust production. It appears that at the low Mach number flow case, $M = 0.1$, high flow velocities at the leading edge, due to the high flapping frequency and the plunge amplitude, can be sustained without a massive flow separation, producing large suction pressures. Whereas, at higher Mach numbers, the rapid flow separation at the leading edge appears to be responsible for the significantly reduced suction, and the low average thrust values obtained at the relatively smaller flapping frequency and amplitudes.



(a) Mach number, $M = 0.2$ (b) Mach number, $M = 0.3$

Figure 5.36: Optimization Cases: 5 and 6

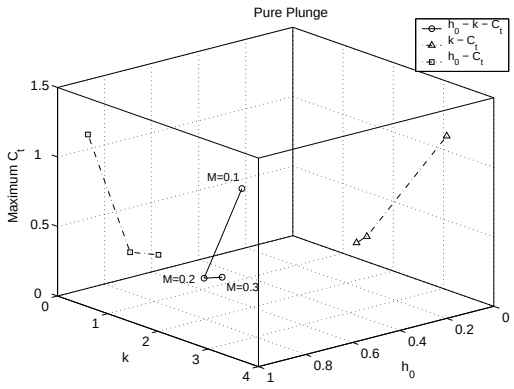


Figure 5.37: k and h_0 values for maximum thrust in Cases $4-6$

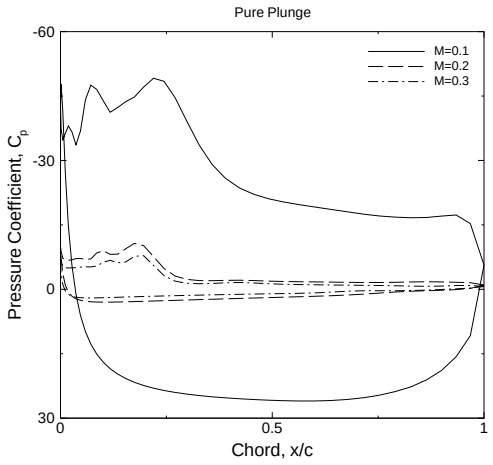
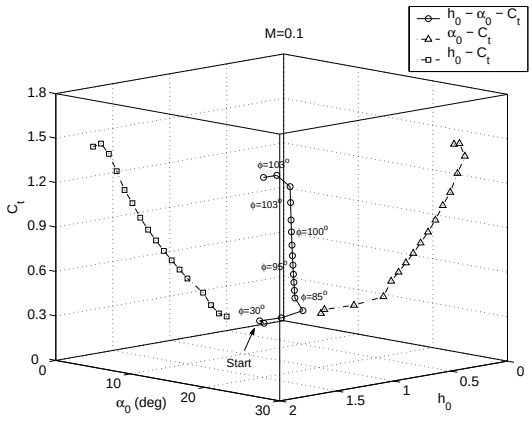
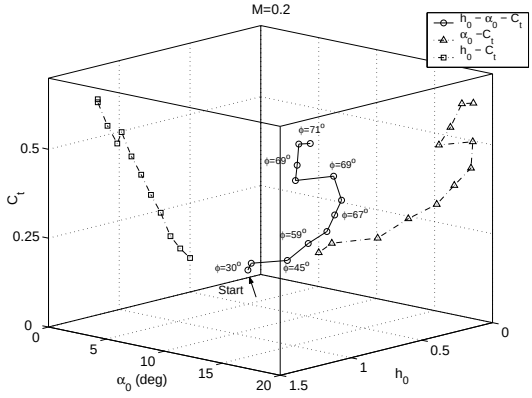


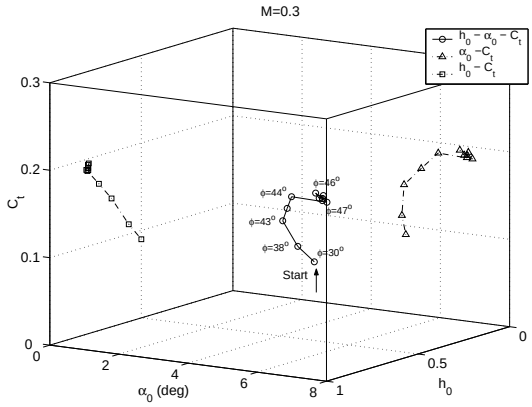
Figure 5.38: Pressure distributions at the instant of maximum thrust in Cases $4-6$



(a) Mach number, $M = 0.1$

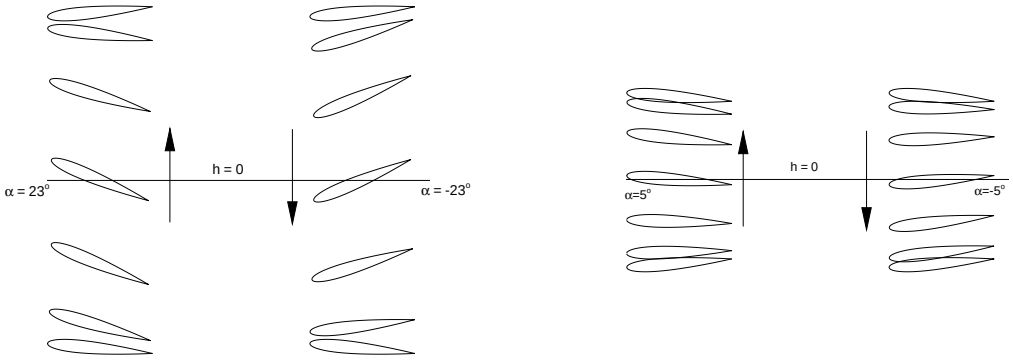


(b) Mach number, $M = 0.2$



(c) Mach number, $M = 0.3$

Figure 5.39: Optimization Cases: $7 - 9$



(a) Case 7: $M = 0.1$, $h_0 = 1.60$, $\alpha_0 = 23.5^o$, $\phi = 103.4^o$

(b) Case 9: $M = 0.3$, $h_0 = 0.79$, $\alpha_0 = 6.8^o$, $\phi = 46.9^o$

Figure 5.40: Optimum combined pitch and plunge motions for Cases 7 and 9

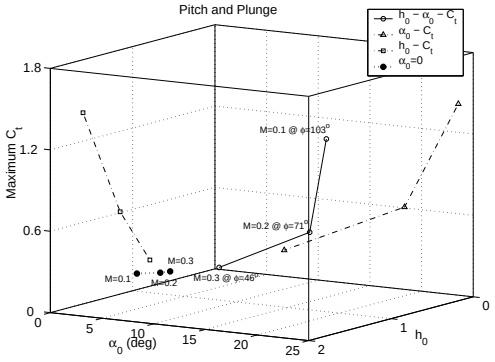


Figure 5.41: h_0, α_0 and ϕ values for maximum thrust in Cases $7-9$

It has long been recognized[22] that when a flapping airfoil undergoes a combined plunge and pitching motion with a proper phase angle between them, the propulsive efficiency is improved. In optimization Cases $7-9$, a combined plunge and pitching motion is considered at a fixed frequency of $k = 1.0$, and the plunge and pitch amplitudes and the phase angle are taken as optimization variables. Optimization results are given in Figure 5.39 for the same set of Mach numbers. As observed, in all the cases, the average thrust is augmented for the optimized values of the pitch and plunge amplitudes and the phase angle. The augmentation is quite significant at $M = 0.2$ and $M = 0.3$. The maximum thrust coefficients at this fixed frequency of $k = 1$ is about twice the maximum values obtained in Cases 5 and 6, where airfoils flap only in plunge. In addition, it should be noted that as the Mach number decreases, the pitch and

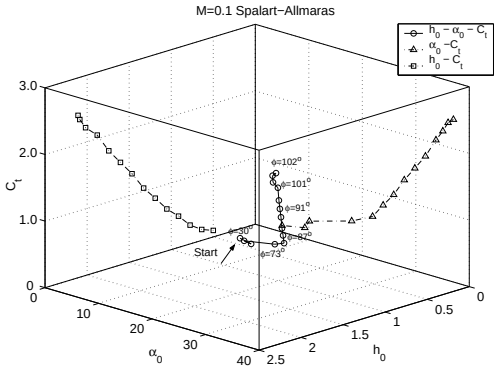


Figure 5.42: Optimization for a turbulent flow, Case 10

Table 5.3: Optimization results

Case	k	h_0	α_0 (deg)	ϕ (deg)	C_t
1	–	–	–	81.9	0.12
2	–	0.72	–	–	0.18
3	–	0.95	–	–	0.12
4a	3.07	0.86	–	–	1.03
4b	2.93	0.97	–	–	1.09
5	1.55	0.69	–	–	0.20
6	1.27	0.57	–	–	0.11
7	–	1.60	23.6	103.4	1.41
8	–	1.15	18.5	71.0	0.61
9	–	0.81	5.9	46.5	0.20
10	–	2.11	37.0	102.3	2.44

plunge amplitudes and the phase angle are decreased for the maximum thrust. The optimum combined pitch and plunge motions of the airfoil for Cases 7 and 9 are shown in Figure 5.40. The optimum motion reduces the effective angle of attack for a more streamlined flow. In Figure 5.41, the maximum average thrust values computed in Cases $7-9$ are compared against that of the airfoil flapping in pure plunge at the same k and h_0 values. The comparison again shows the significant augmentation of thrust production under the influence of the pitching motion.

In Case 10, the same optimization variables as in Case 7 are employed, but the flow is assumed to be fully turbulent. The optimization steps are given in Figure 5.42. It is apparent that as the flow separation is delayed due to the turbulence, higher pitching and plunge amplitudes, and higher thrust values are obtained.

All the results of the optimization cases are given in Table 5.3. It is concluded that thrust generation of a flapping airfoil strongly depend on the flapping frequency and the phase angle

Table 5.4: Optimization cases and starting conditions

Case	β	Re	M	h_0	α_0 (deg)	ϕ (deg)
1	0.0	$1 \cdot 10^4$	0.1	$V : 0.5$	0	0
2	0.25	$1 \cdot 10^4$	0.1	$V : 0.5$	0	0
3	0.5	$1 \cdot 10^4$	0.1	$V : 0.5$	0	0
4	0.75	$1 \cdot 10^4$	0.1	$V : 0.5$	0	0
5	1.0	$1 \cdot 10^4$	0.1	$V : 0.5$	0	0
6	0.0	$1 \cdot 10^4$	0.1	$V : 0.5$	$V : 5$	$V : 30$
7	0.5	$1 \cdot 10^4$	0.1	$V : 0.5$	$V : 5$	$V : 30$
8	1.0	$1 \cdot 10^4$	0.1	$V : 0.5$	$V : 5$	$V : 30$
9	0.0	$1 \cdot 10^4$	0.1	$V : 0.5$	$V : 25$	$V : 60$
10	0.0	$1 \cdot 10^4$	0.1	$V : 1.0$	$V : 5$	$V : 60$
11	0.0	$1 \cdot 10^4$	0.1	$V : 1.0$	$V : 25$	$V : 90$
12	1.0	$1 \cdot 10^4$	0.1	$V : 0.5$	$V : 25$	$V : 60$
13	1.0	$1 \cdot 10^4$	0.1	$V : 1.0$	$V : 5$	$V : 60$
14	1.0	$1 \cdot 10^4$	0.1	$V : 1.0$	$V : 25$	$V : 90$
15	0.0	$2 \cdot 10^4$	0.2	$V : 0.5$	$V : 5$	$V : 30$
16	0.0	$1 \cdot 10^4$	0.2	$V : 0.5$	$V : 5$	$V : 30$
17	0.0	$1 \cdot 10^4$ Tur.	0.1	$V : 0.5$	$V : 5$	$V : 30$

between the plunge and pitch motions. For a fixed flapping frequency, the maximum thrust is produced at a proper phase angle which prevents massive flow separation at the leading edge. The phase angle is about 100^o for $M = 0.1$, giving a maximum thrust of 1.41 at $h_0 = 1.60$ and $\alpha_0 = 23.6^o$.

5.7 Maximization of a Linear Combination of Thrust and Propulsive Efficiency

In the previous section, the objective function was taken to be the average thrust. In this section, a linear combination of the propulsive efficiency and thrust production for a flapping airfoil is taken to be the objective function. The objective function in terms of average thrust and propulsive efficiency is now defined as:

$$O(C_t, \eta) = (1 - \beta)\frac{C_t}{C_{tN}} + \beta\frac{\eta}{\eta_N} \tag{5.1}$$

where β is the weight of the propulsive efficiency in the linear combination. Note that $\beta = 0$ sets the objective function to a normalized thrust coefficient. C_{tN} and η_N vary along the optimization steps. The definitions of C_{tN} and η_N are given in Section 3.3.

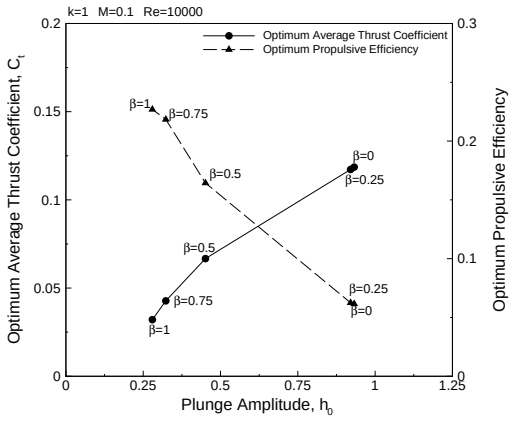


Figure 5.43: Optimization for thrust and efficiency, Cases $1-5$

The optimization studies are performed for a single NACA0012 airfoil flapping in pure plunge motion and combined plunging and pitching. The reduced frequency of the oscillatory motion is fixed at $k = 1$. The optimization variables are the pitch and plunge amplitudes and the phase shift between the pitch and plunge motions. Table 5.4 summarizes the optimization cases studied. V denotes the optimization variable used in the case study. The number next to V is the starting condition for the corresponding optimization variable. All the flows are assumed to be laminar, and computed at $Re = 1 \cdot 10^4$ and $M = 0.1$ except for Cases $15-17$. In Cases 15 and 16, the flow is assumed to be laminar with the freestream Mach number of $M = 0.2$ and the Reynolds numbers of $Re = 2 \cdot 10^4$ and $Re = 1 \cdot 10^4$, respectively. In the last case, the flow is computed turbulent and the Spalart-Allamaras turbulence model is used.

In Cases $1-5$, the only optimization variable is the plunge amplitude, h_0, and β is incremented from 0.0 to 1.0. The results for the set of β values are given in Figure 5.43. When $\beta = 0$, the maximum thrust of $C_t = 0.12$ and the minimum efficiency value of $\eta = 6\%$ are obtained for $h_0 = 0.95$. As β is increased, the maximum efficiency is obtained for lower plunge amplitudes at the expense of reduced thrust figures. For $\beta = 1$, $\eta = 23\%$, $C_t = 0.03$ at $h_0 = 0.28$.

In Case 6, where $\beta = 0$, the average thrust coefficient is maximized with respect to three optimization variables, h_0, α_0 and ϕ. The instantaneous variation of the unsteady drag (negative thrust) coefficient along a few optimization steps is shown in Figure 5.44. As the optimization variables are incremented along the optimization steps, unsteady computations are carried out for a few periods of the flapping motion until a periodic behavior is obtained. The variation of

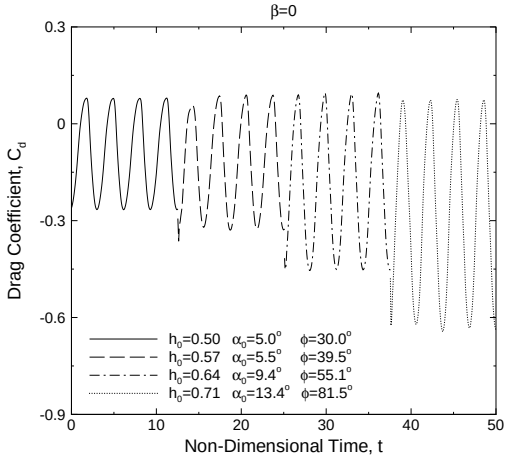


Figure 5.44: C_d history along optimization steps, Case 6

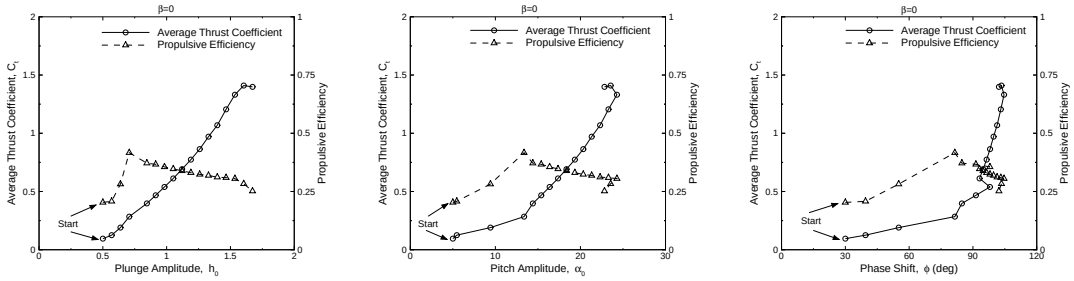


Figure 5.45: Maximization of thrust coefficient ($\beta = 0$), Case 6

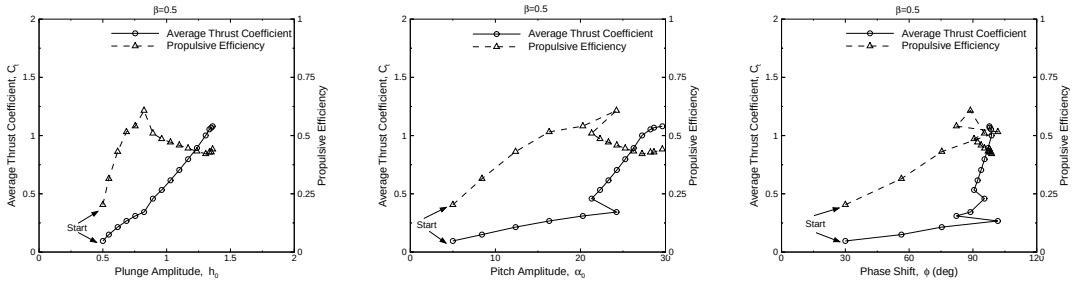


Figure 5.46: Maximization of propulsive efficiency and thrust coefficient ($\beta = 0.5$), Case 7

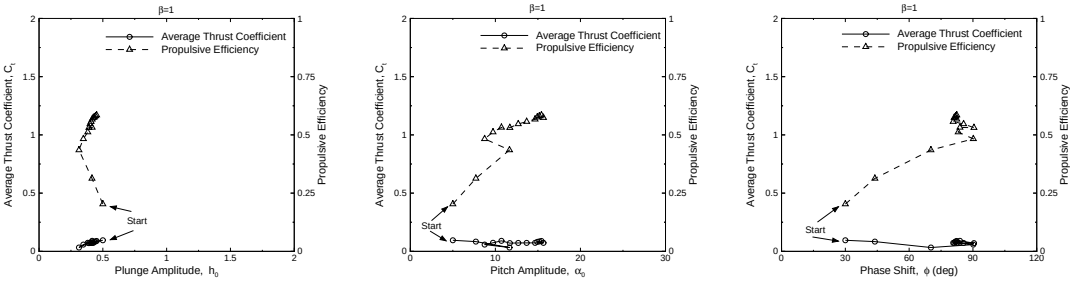
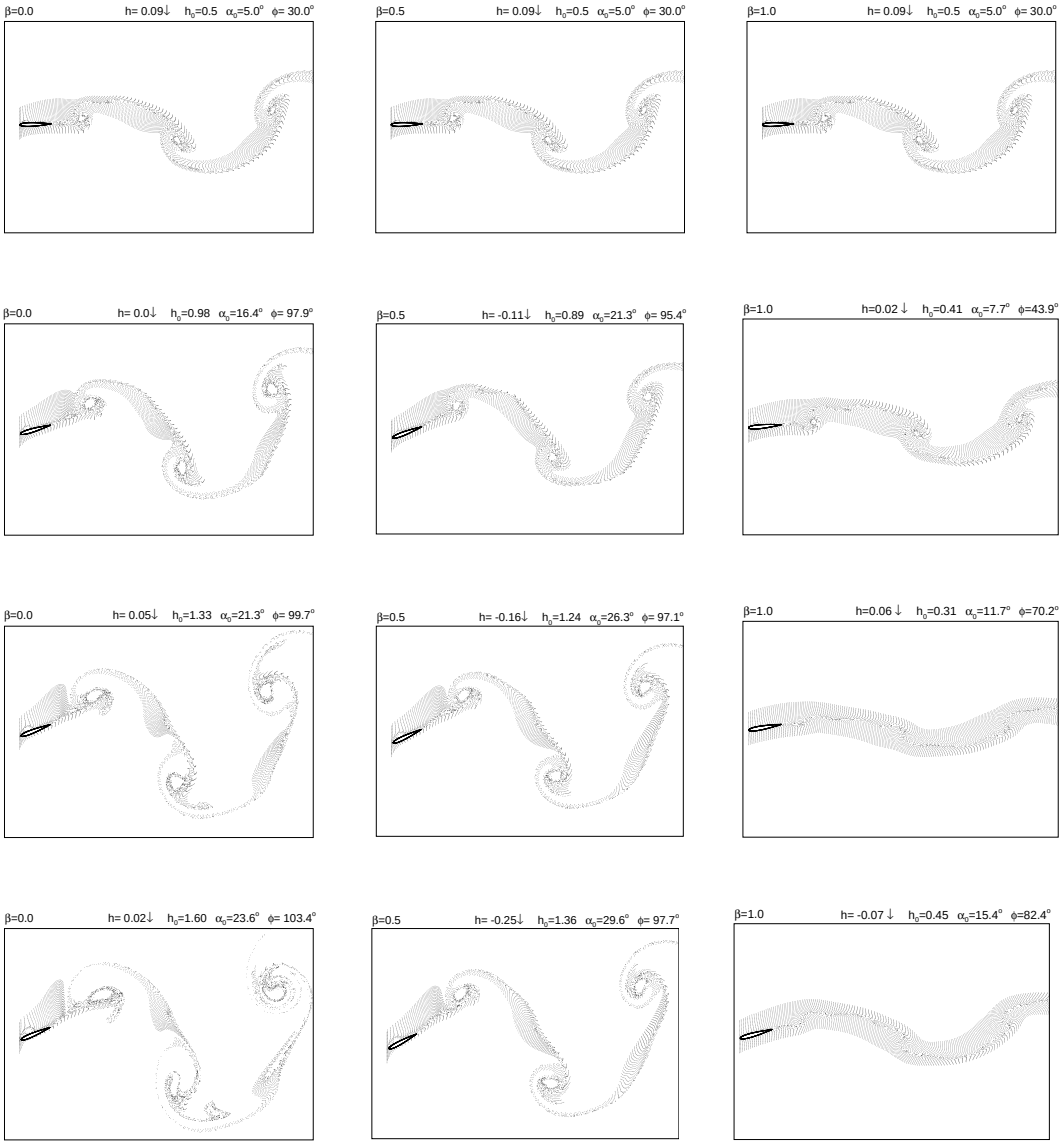


Figure 5.47: Maximization of propulsive efficiency ($\beta = 1$), Case 8



(a) Case 6 (b) Case 7 (c) Case 8

Figure 5.48: Particle traces at the instant of maximum thrust along optimization steps for Cases $6-8$

the average thrust coefficient and the propulsive efficiency with respect to the optimization variables are given in Figure 5.45. As shown, as the optimization variables are incremented along the gradient of the objective function, the average thrust coefficient increases gradually, and a maximum value of 1.41 is reached at $h_0 = 1.60$, $\alpha_0 = 23.5^o$ and $\phi = 103.4^o$. The corresponding propulsive efficiency is 28.3%. When compared to Case 1, it is observed that the combined pitch and plunge motion for maximum thrust augments the thrust by about 10 folds, and the propulsive efficiency by about 5 folds.

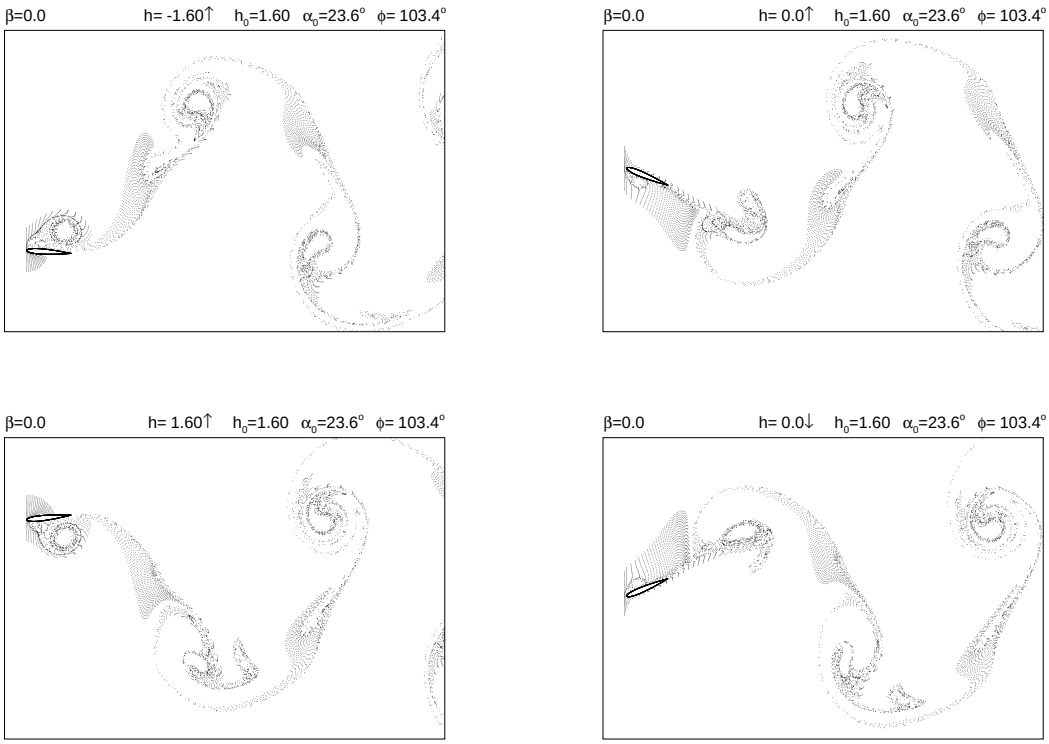


Figure 5.49: Instantaneous particle traces along a period of optimized flapping motion, Case 6

Optimization Cases 7 and 8 are shown in Figure 5.46 and 5.47, respectively. In Case 7, where $\beta = 0.5$, the average thrust and the propulsive efficiency have equal weights in the objective function. As a result, the efficiency is improved at the expense of average thrust. It is observed that the higher efficiency is achieved at a lower plunge amplitude, $h_0 = 1.36$, and a higher pitch amplitude, $\alpha_0 = 29.6^o$. The phase shift slightly drops to 97.8^o. In Case 8, the propulsive efficiency is maximized at low pitch and plunge amplitudes with the corresponding low thrust coefficient. It is apparent that the propulsive efficiency and thrust production are inversely proportional.

The unsteady flowfields along the optimization steps are investigated with unsteady particle traces for Cases $6-8$. The particles are emitted at a given frequency along a straight line in the vicinity of the leading edge of the airfoil, and traced in time. In Figure 5.48, the instantaneous particle traces at the instant of maximum thrust (minimum drag) in a flapping period are given along the optimization steps. It is observed that in Case 6, the leading edge vortex formation is promoted along the optimization steps. The maximum instantaneous thrust occurs at about

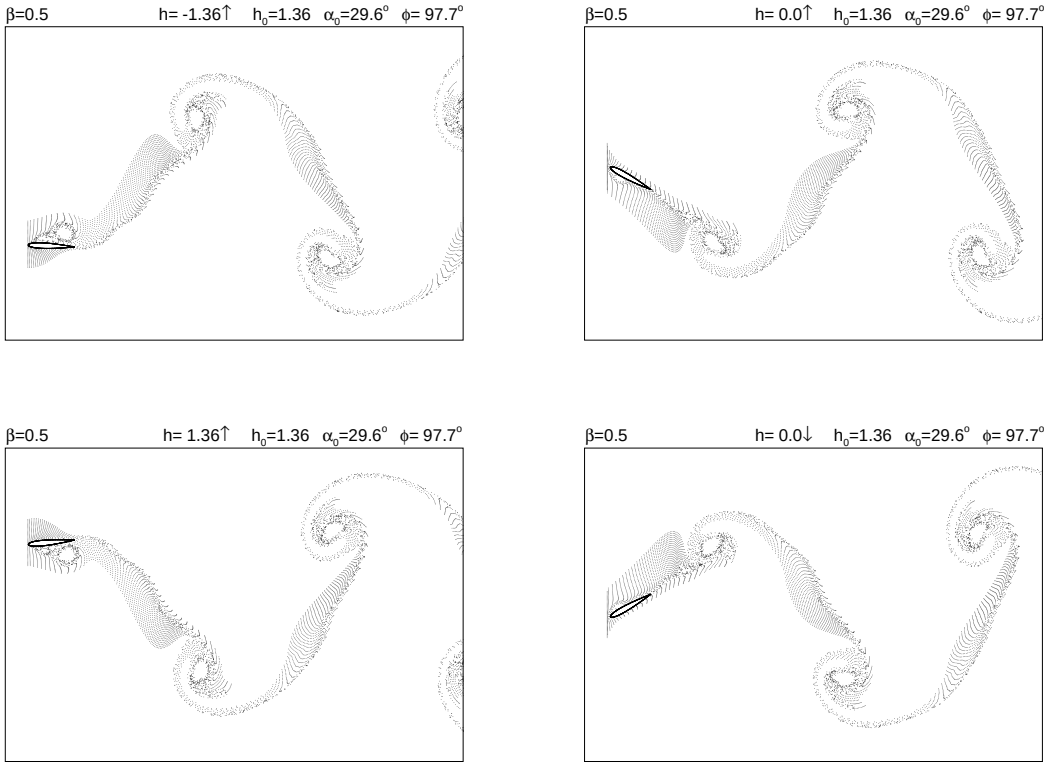


Figure 5.50: Instantaneous particle traces along a period of optimized flapping motion, Case 7

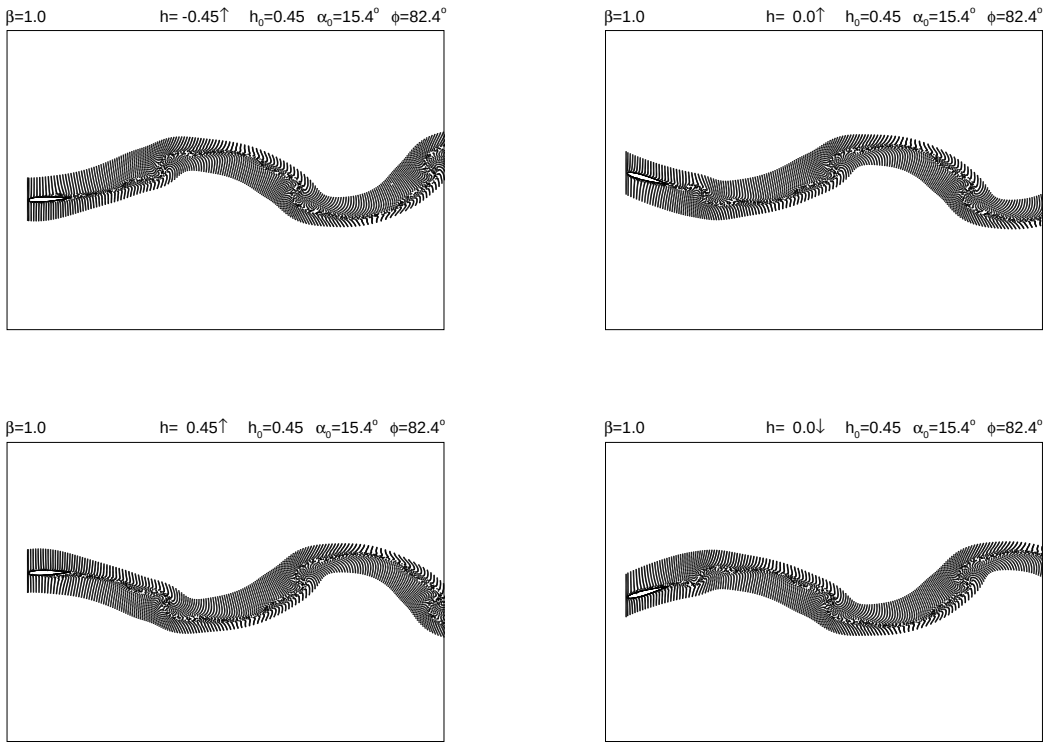
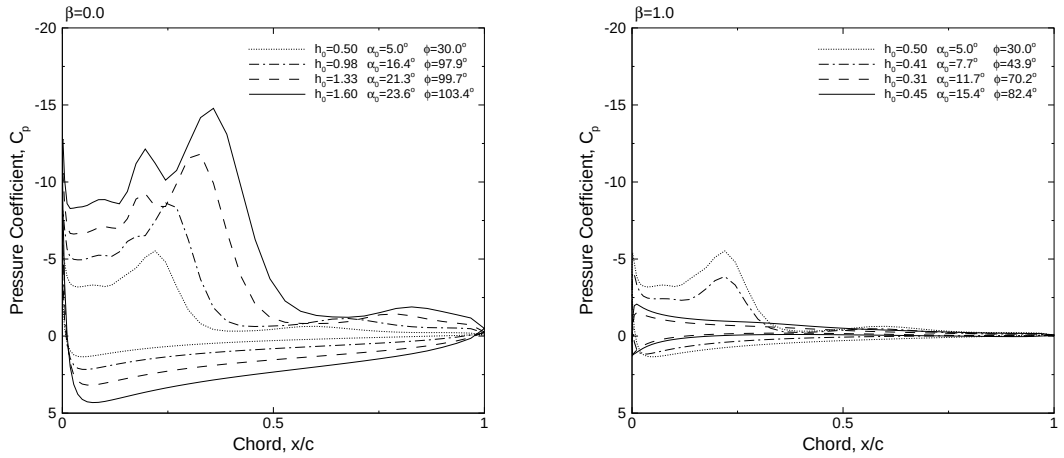


Figure 5.51: Instantaneous particle traces along a period of optimized flapping motion, Case 8



(a) $\beta = 0$, Case 6

(b) $\beta = 1$, Case 8

Figure 5.52: Instantaneous pressure distribution at maximum thrust positions, Cases 6 and 8

the mean amplitude location as a stronger leading edge vortex develops. A similar leading edge vortex formation along the optimization steps is observed in Case 7. However, the vortex development is not as strong as in Case 6. Whereas, in Case 8, the leading edge vortex formation is prevented along the optimization steps, which incidently maximizes the propulsive efficiency. The unsteady flow becomes more streamlined with the motion of the airfoil. Figure 5.49 shows the optimized flowfield for maximum thrust in Case 6. The flowfield is observed to be highly vortical with strong leading edge vortices during the upstroke and the downstroke. The flowfield is periodic, and anti-symmetric along the upstroke and the downstroke. The optimized flowfields in Cases 7 and 8 are shown in Figures 5.50 and 5.51, respectively. As seen, as β increases, the flow becomes more streamlined.

In Figure 5.52, the instantaneous pressure distributions over the airfoil chord at the instant of maximum thrust along optimization steps are given. As seen from the figure, when $\beta = 0$, the suction region near the leading edge of the airfoil gets larger along the optimization steps. Whereas, as expected, when $\beta = 1$, the suction region gets smaller along the optimization steps.

Next, the optimization space is searched for other possible local maxima of the objective function for Cases 6 and 8. It is implemented by initiating the optimization process from various initial conditions, as given in Table 5.4. All the results of the optimization cases are

Table 5.5: Optimization results

Case	h_0	α_0 (deg)	ϕ (deg)	C_t	$\eta(\%)$
1	0.93	0.0	0.0	**0.12**	6.1
2	0.92	0.0	0.0	0.12	6.3
3	0.45	0.0	0.0	**0.07**	**16.5**
4	0.32	0.0	0.0	0.04	21.9
5	0.28	0.0	0.0	0.03	**22.7**
6	1.60	23.5	103.4	**1.41**	28.3
7	1.36	29.6	97.8	**1.08**	**44.1**
8	0.45	15.4	82.4	0.08	**58.5**
9	1.73	23.8	100.7	**1.44**	25.4
10	1.52	26.9	87.2	**1.27**	33.4
11	1.55	28.6	94.9	**1.45**	35.9
12	0.57	21.0	86.7	0.13	**63.8**
13	0.60	22.8	86.1	0.13	**64.8**
14	0.83	35.5	86.5	0.18	**67.5**
15	1.15	18.5	71.0	**0.61**	28.0
16	1.54	20.0	94.9	**0.95**	23.2
17	2.12	36.6	102.9	**2.64**	34.2

given in Table 5.5. The initial conditions and the optimized states at the end of the optimization processes are shown in Figure 5.53 and 5.54 for $\beta = 0$ and $\beta = 1$, respectively. Figure 5.53 reveals that all the optimization cases for $\beta = 0$ converge about the same value of the objective function, which is the thrust coefficient, and of the optimization variables. It suggests that the global maximum of the objective function may have been found. On the other hand, the optimization processes for $\beta = 1$, provides different optimum states for h_0 and α_0. But the ϕ values are about the same in all the cases. It appears that a high flapping efficiency may be achieved for a range of h_0 and α_0 values, provided that α_0 increases as h_0 does.

The optimum flapping motion for Cases $6-8$ and 14 is shown in Figure 5.55. It is clearly observed that the plunge amplitude plays a significant role in thrust generation. Higher pitch amplitudes improve the propulsive efficiency. In addition, the phase shift between the plunge and pitch motions, which is about 90 deg. for all the cases, reduces the effective angle of attack at the mid-plunge location, where the plunge velocity is maximum.

The effective angles of attack the airfoil sees along a flapping period are given in Figure 5.56 for Cases 6, 7 and 14. 0 deg. in the flapping period corresponds to the $h = -h_0$ position of the airfoil. In agreement with the previous observations, for higher thrust production, as in Cases 6 and 7, a flapping airfoil stays at large effective angles of attack longer during the flapping

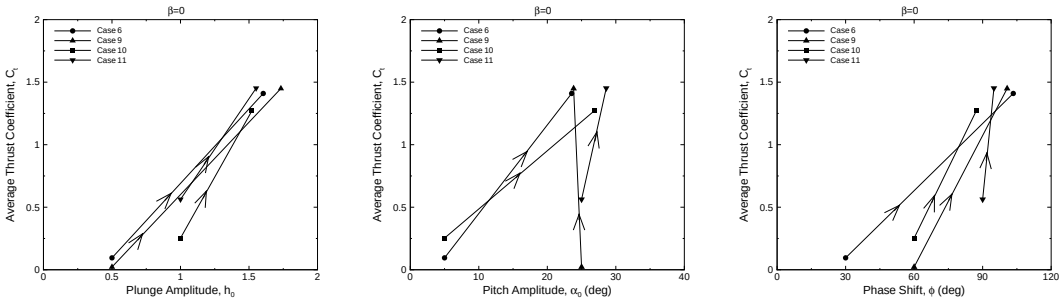


Figure 5.53: Maximization of the average thrust coefficient, C_t ($\beta = 0$) with various starting conditions, Cases 6, $9 - 11$

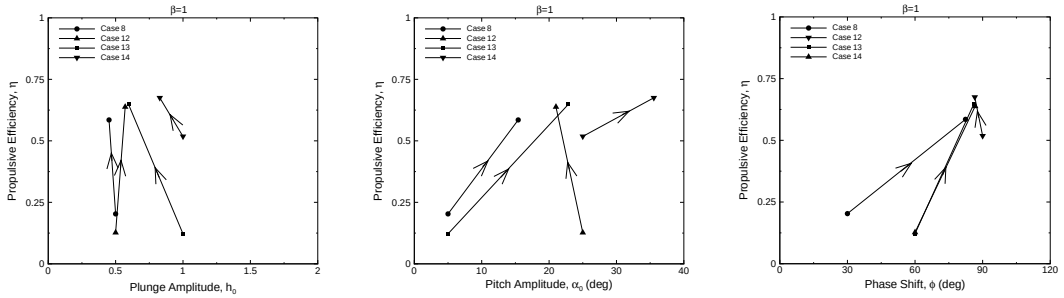
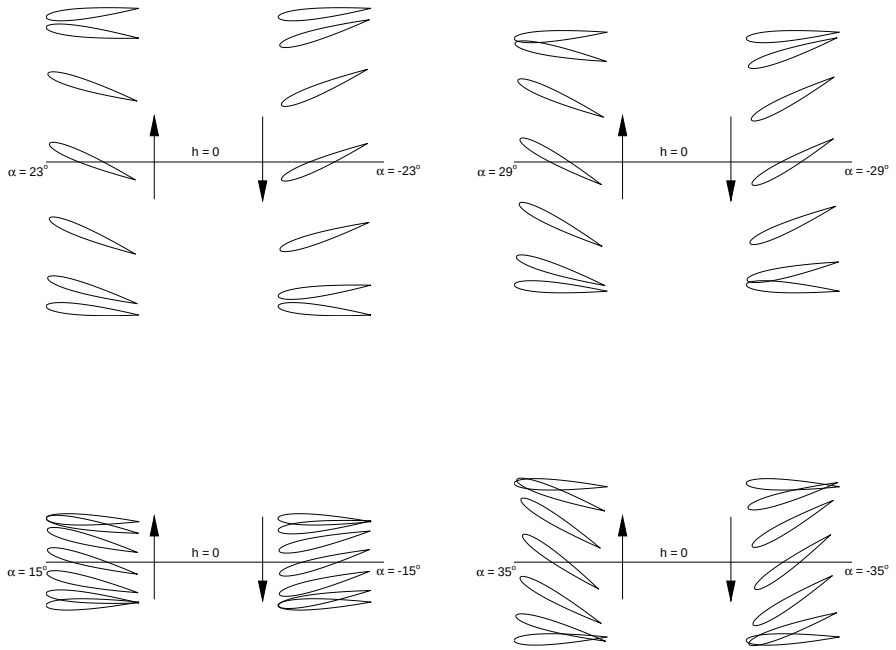


Figure 5.54: Maximization of the propulsive efficiency, η ($\beta = 1$) with various starting conditions, Cases 8, $12 - 14$



(c) Case 8 (d) Case 14

Figure 5.55: Optimized flapping motions

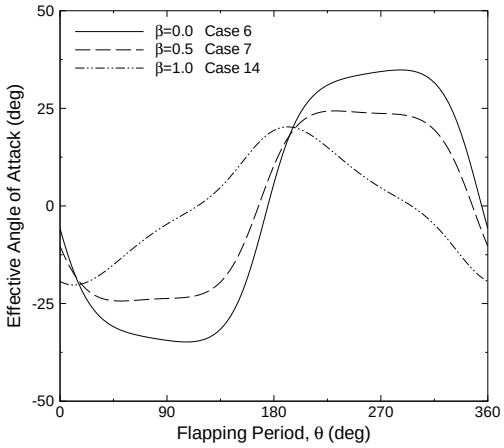


Figure 5.56: Effective angle of attack

period. For an efficient flapping (Case 14) , the effective angle of attack at the mid-plunge location ($\theta = 90, 270$ deg.) is set about 0 deg. Whereas, in Cases 6 and 7, the maximum effective angle of attack occurs around mid-plunge locations.

Cases 15 and 16 are the optimization runs carried out at the freestream Mach number of $M = 0.2$. Reynolds number is $Re = 2 \cdot 10^4$ in Case 15, and is $Re = 1 \cdot 10^4$ in Case 16. The optimization steps for both Cases are shown in Figures 5.58 and 5.59. In order to make an easy comparison with the results of Cases 15 and 16, Figure 5.45 of Case 6 is repeated on the same page as Figure 5.57. As seen from Figure 5.58, in Case 15, the maximum average thrust value is significantly smaller than $M = 0.1$ case (Case 6), and it is reached at lower h_0, α_0 and ϕ values. However, the propulsive efficiency values in both cases are almost the same. In Case 16, where the Mach number is still $M = 0.2$ but the Reynolds number is decreased to $Re = 1 \cdot 10^4$, an augmentation of thrust at the expense of propulsive efficiency is observed. When compared to Case 15, the maximum thrust is reached at higher h_0, α_0 and ϕ values, which are very close to those of Case 6. The solution with the turbulent assumption is given in Figure 5.60. About a 90% thrust enhancement and about a 20% increase in propulsive efficiency is observed when compared to Case 6. This is attributed to the delay of the flow separation due to the turbulence.

In this optimization study, the optimization of the thrust and the propulsive efficiency together is achieved with a weighted and normalized objective function. The thrust generation of a flapping airfoil is maximized at large plunge amplitudes with large leading edge vortices

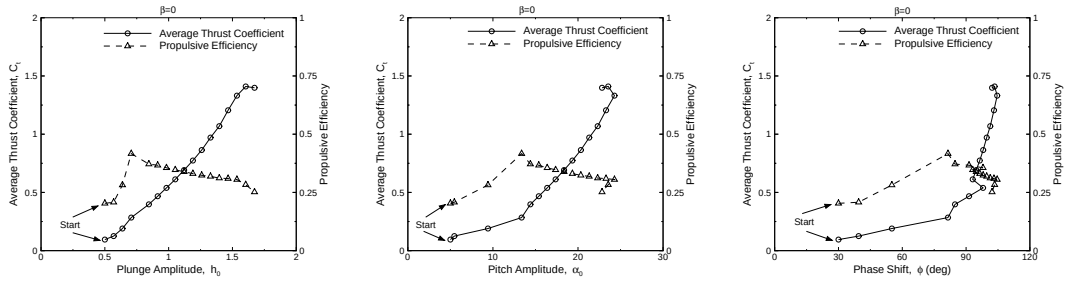


Figure 5.57: Maximization of thrust coefficient ($M = 0.1, Re = 1 \cdot 10^4$), Case 6

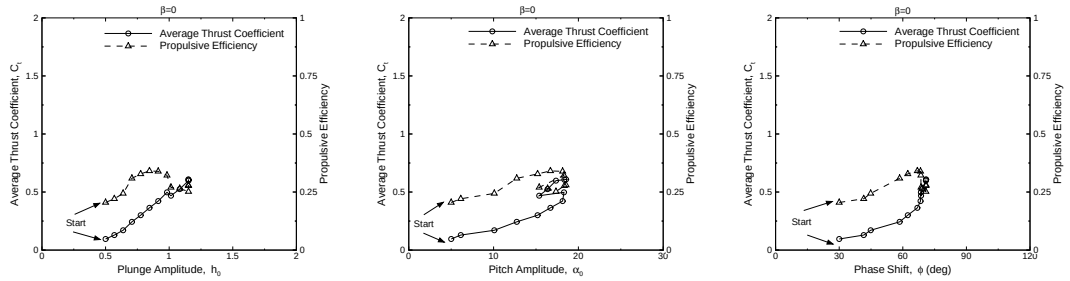


Figure 5.58: Maximization of thrust coefficient ($M = 0.2, Re = 2 \cdot 10^4$), Case 15

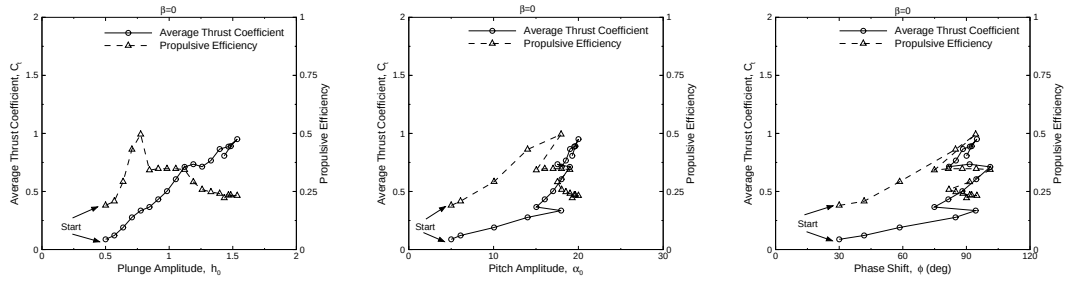


Figure 5.59: Maximization of thrust coefficient ($M = 0.2, Re = 1 \cdot 10^4$), Case 16

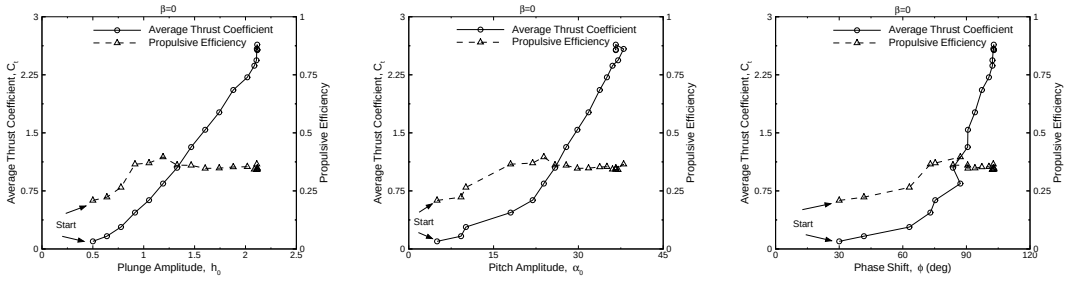


Figure 5.60: Maximization of thrust coefficient ($M = 0.1, Re = 1 \cdot 10^4$, Turbulent), Case 17

forming and shedding into the wake. The airfoil stays at a large effective angle of attack during most of the flapping period. The propulsive efficiency of the flapping airfoils may be increased by reducing the plunge amplitude and the effective angle of attack, and consequently by preventing the formation of the leading edge vortices.

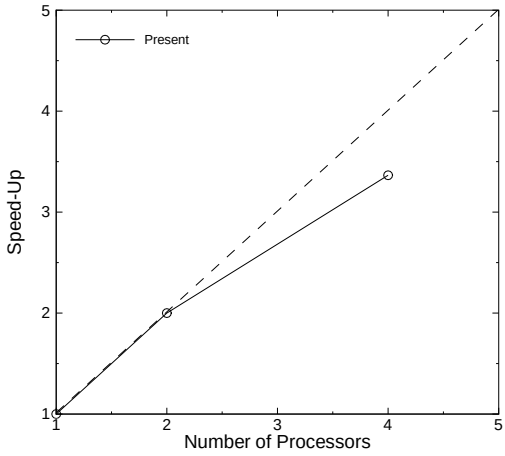


Figure 5.61: Parallel speed-up

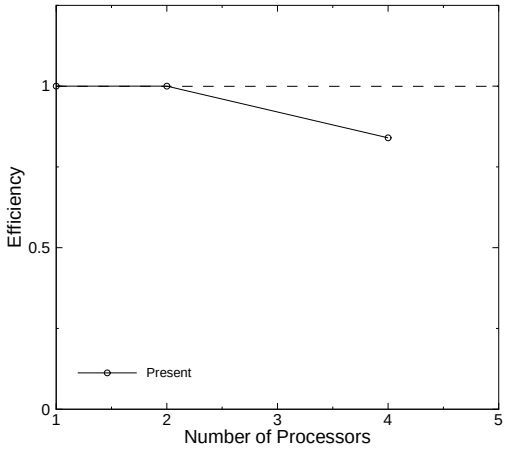


Figure 5.62: Parallel efficiency

5.8 Parallel Efficiency

The aim of parallel processing is to reduce the computational time spent. However, another important point is that one should also benefit from the computer capabilities as efficiently as possible. That is, in parallel programming, both speed-up and efficiency should be taken into consideration. The definition of the *speed-up* is given in Eqn. 4.1. The parallel efficiency is defined as the ratio of the speed-up to the number of processors used.

As an example to show the speed-up and the efficiency in parallel processing, a comparison between the ideal and real cases is given in Figure 5.61 and 5.62 for a laminar unsteady flow computation around airfoils in biplane configuration. In this example, the airfoil and back-

ground grids are of 141×31 and 135×262 size, respectively. A zoomed part of this overset grid system is shown in Figure 5.11. As explained in detail in Chapter 4, there are four subgrids in which the flow computations are to be performed separately, which are two 141×31-size C-type airfoil grids and two 135×132-size Cartesian grids decomposed from the original background grid. During the computation procedure, these four subgrids are distributed over two dual-processor computers.

Run durations shown in the figures are normalized by the computation time needed when using a single processor. Since the computers employed have dual processors, when two processors are used, the speed-up and the efficiency values coincide with the ideal ones. However, when four processors are used, parallel efficiency degrades due to the unbalanced load distribution on the airfoil and the partitioned background grids. Another reason for the decrease in the efficiency is the time lost due to the communication between the processors.

CHAPTER 6

CONCLUDING REMARKS

Unsteady viscous flowfields over flapping airfoils are computed on overset grids using a Navier-Stokes solver and the flapping motion parameters are optimized to maximize the thrust and the propulsive efficiency of flapping airfoils. Computations are performed in parallel using `PVM` library routines in a computer cluster. For the optimization processes, a gradient based algorithm, steepest ascent method implemented again in a parallel algorithm is employed. A single flapping airfoil and dual airfoils flapping in a biplane configuration are considered. The flapping motion is imposed as a combination of pitch and plunge motions. The overall conclusions are made by analyzing computed flowfields in terms of variation of aerodynamic loads in time and instantaneous distributions of the flow variables. The vortical flowfields and the wake profiles formed by the leading and trailing edge vortices and their downstream convection are visualized with unsteady particle traces.

First, the effects of compressibility and the turbulence on the thrust generation of a single flapping airfoil are investigated. Then, parametric studies for dual flapping airfoils in a biplane configuration are performed. Finally, optimization studies of flapping motion parameters are carried out.

In the preliminary studies, it is noted that the compressibility plays a significant role in thrust generation. It appears that at the low Mach number flow cases, high flow velocities at the leading edge, due to the high flapping frequency and the plunge amplitude, can be

sustained without a massive flow separation, producing very large suction pressures. Whereas, at higher Mach numbers, the rapid flow separation being shock induced near the leading edge appears to be responsible for the significantly reduced suction, and the low thrust obtained at the relatively smaller flapping frequency and amplitudes. It is concluded that the thrust generation is enhanced in a turbulent flow due to the delayed flow separation when compared to a laminar flow where an earlier flow separation is present.

The parametric studies of biplane configuration indicate that the thrust production is significantly enhanced in comparison to a single flapping airfoil as observed in the recent experiments. The computed predictions compare well with the available experimental data.

The gradient based numerical optimization algorithm in parallel is successfully applied to maximize the thrust generation and the efficiency of the thrust generation of flapping airfoils. The optimization of the thrust generation and the propulsive efficiency together is achieved with a weighted and normalized objective function. Optimization variables are the flapping motion parameters; the plunging and pitching amplitudes, the flapping frequency, and the phase shift between the pitch and plunge motions. The parallel implementation of the optimization algorithm is quite robust and the optimum solutions agree well with the parametric studies. Thrust generation of a flapping airfoil is found to be strongly dependent on the phase angle between the plunge and pitch motions. The maximum thrust is produced at large plunge amplitudes with large leading edge vortices forming and shedding into the wake. The airfoil stays at a large effective angle of attack during most of the flapping period. Propulsive efficiency of the flapping airfoils may be increased by reducing the plunge amplitude and the effective angle of attack, and consequently by preventing the formation of leading edge vortices to keep the unsteady flow streamlined.

In the near future, the flapping motion and the airfoil profile with a limited number of parameters will be included among the optimization variables.

Further research is also in progress to implement the present optimization method to the thrust generation of flapping airfoils in a biplane configuration. In this case, the optimization problem will have a constraint due to the restrictions on the pitch and plunge amplitudes of

the airfoils. In addition, a new overset grid topology, which will decrease the limitations in the biplane configuration, needs to be developed.

For a faster convergence in the optimization process, the optimization algorithm and the parallel efficiency of the computations may further be improved.

REFERENCES

[1] Shyy, W., Berg, M. and Ljungqvist, D., *Flapping and flexible wings for biological and micro air vehicles*, Progress in Aerospace Sciences 35, 1999, pp. 455-505.

[2] Kamakoti, R., Berg, M., Ljungqvist, D. and Shyy, W. *A Computaitonal Study for Biological Flapping Wing Flight*, Trans. Of the Astronautical Society of the Republic of China, Vol. 32, No. 4.,2000, pp. 265-279.

[3] Vick A., Stillion J., Frelinger, D.R., Kvitky, J.S., Lambeth, B.S., Marquis, J., Waxman, M.C., *Aerospace Operations in Urban Environments*, Appendix B, RAND, 2000, pp. 210.

[4] Committee on Materials, Structures, and Aeronautics for Advanced Uninhabited Air Vehicles, *Uninhabited Air Vehicles: Enabling Science for Military Systems*, National Academy Press, Washington, D.C., 2000.

[5] http://www.darpa.mil/tto/programs/mav.html

[6] Jones, K.D., Duggan, S.J. and Platzer, M.F., *Flapping-Wing Propulsion for a Micro Air Vehicle*, AIAA Paper No. 2001-0126, 39*th* AIAA Aerospace Sciences Meeting, Reno, Nevada, Jan., 2001.

[7] Jones, K.D., Castro, B.M., Mahmoud, O. and Platzer, M.F., *A Numerical and Experimental Investigation of Flapping-Wing Propulsion*, AIAA Paper No. 2002-0866, 40*th* AIAA Aerospace Sciences Meeting, Reno, Nevada, Jan., 2002.

[8] Alexander, R.Mc., *The U, J and L of bird flight*, Nature, 1997; 390:13

[9] Knoller, R., *Die Gesetze des Luftwiderstandes*, Flug- und Motortechnik (Wien), Vol. 3, No. 21, 1909, pp. 1-7.

[10] Betz, A., *Ein Beitrag zur Erklärung des Segelfluges*, Zeitschrift für Flugtechnik und Motorluftschiffahrt, Vol. 3, Jan. 1912, pp. 269-272.

[11] Katzmayr, R., *Effect of Periodic Changes of Angle of Attack on Behavior of Airfoils*, NACA Report No.147, Oct. 1922. (translated from Zeitschrift für Flugtechnik und Motorluftschiffahrt, Mar. 31, pp. 80-82, and Apr. 13, 1922,pp. 95-101).

[12] Von Karman, T. and Burgers, J.M., *General Aerodynamic Theory - Perfect Fluids*, Division E, Vol.II, "Aerodynamic Theory", Ed. Durand, W.F., 1943, p. 308.

[13] Theoderson, T., *General Theory of Aerodynamic Instability and the Mechanism of Flutter*, NACA Report No. 496, 1935.

[14] Garrick, I.E., *Propulsion of a Flapping and Oscillating Airfoil*, NACA Report No. 567, 1936.

[15] Silverstein, A. and Joyner, U.T., *Experimental Verification of the Theory of Oscillating Airfoils*, NACA Report No. 673, 1939.

[16] Schmidt, W., *Der Wellpropeller, ein neuer Antrieb fuer Wasser-, Land-, und Luftfahrzeuge*, Zflugwiss, Vol. 13, 1965, pp. 472-479.

[17] DeLaurier, J. D. and Harris, J. M., *Experimental Study of Oscillating-Wing Propulsion*, Journal of Aircraft, Vol. 19, No. 5, May, 1982, pp. 368-373.

[18] Lai, J.C.S. and Platzer, M.F., *The Jet Characteristics of a Plunging Airfoil*, 36*th* AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan. 1998.

[19] Jones, K.D. and Platzer, M.F., *Numerical Computation of Flapping-Wing Propulsion and Power Extraction*, AIAA Paper No. 97-0826, 1997.

[20] Jones, K.D., Dohring, C.M. and Platzer, M.F., *Wake Structures Behind Plunging Airfoils: A Comparison of Numerical and Experimental Results*, AIAA Paper No. 96-0078, Jan., 1996.

[21] Jones, K.D., Dohring, C.M. and Platzer, M.F., *An Experimental and Computational Investigation Of the Knoller-Betz Effect*, AIAA Journal Vol. 36, No. 7, 1998, pp. 1240-1246.

[22] Anderson, J.M., Streitlien, K., Barrett, D.S. and Triantafyllou, M.S., *Oscillating Foils of High Propulsive Efficiency*, Journal of Fluid Mechanics, Vol. 360, 1998, pp. 41-72.

[23] Platzer, M.F. and Jones, K.D., *The Unsteady Aerodynamics of Flapping-Foil Propellers*, 9*th* International Symposium on Unsteady Aerodynamics, Aeroacoustics and Aeroelasticity of Turbomachines, Ecole Centrale de Lyon, Lyon, France, September 4-8, 2000.

[24] Jones, K.D., Castro, B.M., Mahmoud, O., Pollard S.J., Platzer, M.F., Neef, M.F., Gonet K. and Hummel, D., *A Collaborative Numerical and Experimental Investigation of Flapping-Wing Propulsion*, AIAA Paper No. 2002-0706, 40*th* AIAA Aerospace Sciences Meeting, Reno, Nevada, Jan., 2002.

[25] Jones, K.D. and Platzer, M.F., *Experimental Investigation of the Aerodynamic Characteristics of Flapping-Wing Micro Air Vehicles*, AIAA Paper No. 2003-0418, 2003.

[26] Birnbaum, W., *Der Schlagflügelpropeller und die kleinen Schwingungen elastisch befestigter Tragflügel*, Zeitschrift für Flugtechnik und Motorluftschiffahrt, Vol. 15, 1924, pp. 128-134.

[27] Philips, P.J., East R.A. and Pratt N.H., *An unsteady lifting line theory of flapping wings with application to the forward flight of birds*, J. Fluid Mech. Fluids, 112:97-125, 1981.

[28] Vest, M.S. and Katz, J., *Unsteady aerodynamic model of flapping wings*, AIAA Journal Vol. 34, No. 7, 1996, pp. 1435-1440.

[29] Leroy, A. and Devinant, P.H., *A General Approach For Computing Unsteady* 3*D Thin Lifting And/Or Propulsive Systems Derives From a Complete Theory*, Int. J. Numer. Meth. Fluids, 29:7595, 1999.

[30] Sunada, S. and Ellington, C.P., *A new method for explaining the generation of aerodynamic forces in flapping flight*, Math. Meth. Appl. Sci., 24:13771386, 2001.

[31] Isogai, K., Shinmoto Y. and Watanabe, Y., *Effects of Dynamic Stall on Propulsive Efficiency and Thrust of a Flapping Airfoil*, AIAA Journal, Vol. 37, No. 10, 2000, pp. 1145-1151.

[32] Isogai, K. and Shinmoto Y., *Study on Aerodynamic Mechanism of Hovering Insects*, AIAA Paper No. 2001-2470, 2001.

[33] Togashi, F., Ito, Y., Murayama, M., Nakahashi, K. and Kato T., *Flow Simulation of Flapping Wings of an Insect Using Overset Unstructured Grid*, AIAA Paper No. 2001-2619, 2001.

[34] Szmelter, J. and Żbikowski R., *A study of flow arising from insect wing flapping motion*, Int. J. Numer. Meth. Fluids, 40:497-505, 2002.

[35] Tuncer, I.H. and Platzer, M.F., *Thrust Generation due to Airfoil Flapping*, AIAA Journal, Vol. 34, No. 2, 1995, pp. 324-331.

[36] Tuncer, I.H., *A 2-D Unsteady Navier-Stokes Solution Method with Moving Overset Grids*, AIAA Journal, Vol. 35, No. 3, March 1997, pp. 471-476.

[37] Tuncer, I.H., Lai, J., Ortiz, M.A. and Platzer, M.F., *Unsteady Aerodynamics of Stationary/Flapping Airfoil Combination in Tandem*, AIAA Paper 97-0659, 1997.

[38] Tuncer, I.H., Walz, R. and Platzer, M.F., *A Computational Study on the Dynamic Stall of a Flapping Airfoil*, AIAA Paper No. 98-2519, 1998.

[39] Tuncer, I.H. and Platzer, M.F., *Computational Study Of Flapping Airfoil Aerodynamics*, AIAA Journal of Aircraft, Vol. 35, No. 4, 2000, pp. 554-560.

[40] Osher, S. and Chakravarty, S.R., *A New Class of High Accuracy TVD Schemes for Hyperbolic Conservation Laws*, AIAA Paper, No. 85-0363, 1985.

[41] Chakravarty, S.R. and Osher, S., *Numerical Experiments with the Osher Upwind Scheme for the Euler Equations*, AIAA Journal, Vol. 21, No. 11, 1983, pp. 1241-1248.

[42] Baldwin, B.S. and Lomax, H., *Thin Layer Approximation and Algebraic Model for Seperated Turbulent Flow*, AIAA Paper, No. 78-257, 1978.

[43] Spalart, P.R. and Allmaras, S.R., *A One-Equation Turbulence Model for Aerodynamic Flows*, AIAA Paper, No. 92-0439, 1992.

[44] Rai, M.M. and Chakravarty, S.R., *An Implicit Form of the Osher Upwind Scheme*, AIAA Journal, Vol. 24, No. 5, 1986, pp. 735-743.

[45] Steger, J.L. and Warming, R.F., *Flux Vector Splitting of the Inviscid Gas Dynamic Equations with Applications to Finite-Difference Methods*, Journal of Computational Physics, Vol. 40, 1981, pp. 263-293.

[46] Tuncer, I.H., *Parallel Computation of Multi-Passage Cascade Flows with Overset Grids*, Parallel CFD Workshop, Istanbul, June 16-18, 1997.

[47] Steinbrenner, J.P., Chawner, J.R. and Fouts, C.L., *Multiple Block Grid Generation in the Interactive Environment*, AIAA Paper No. 90-1602, 1990.

[48] Thompson, J.F., *Composite Grid Generation Code for General 3-D Regions - the EAGLE Code*, AIAA Journal Vol. 26, No. 3, pp. 271-272, 1988.

[49] Thompson, J.F. and Weatherill, N.P., *Aspects of Numerical Grid Generation: Current Science abd Art*, AIAA Paper No. 93-3539-CP, 1993.

[50] Barth, J.T., *Aspects of Unstructured Grids and Finite-Volume Solvers for Euler and Navier-Stokes Equations*, AGARD Report No. 787, 1992.

[51] Benek, J.A., Buning, P.G. and Steger, J.L., *A 3-D Chimera Grid Embedding Technique*, AIAA Paper No. 85-1523, 1985.

[52] Giles, M.B., *Nonreflecting Boundary Conditions for Euler Equation Calculations*, AIAA Journal, Vol. 28, No. 12, Dec 1990, pp. 2050-2058.

[53] Kuruvila, G., Ta'asan S. and Salas, M.D, *Airfoil Optimization by the One-Shot Method*, AGARD Report No. 803, 1994.

[54] Greig, D.M., *Optimisation*, Longman Mathematical Texts, 1980.

[55] Hwang, K. and Xu, Z., *Scalable Parallel Computing, Technology, Architecture, Programming*, Published by WCB-Mc Graw Hill, USA.

[56] Geist, A., Beguelin, A., Dongarra, J., Manchek, R. and Sunderam, V., *PVM 3 USER's Guide and Reference Manual*, Oak Ridge National Laboratory, Tennessee, 1994

[57] Açıkgöz, N., *Application of Turbulence Models to Unsteady Cascade Flows in a Parallel Environment*, Master Thesis, METU, January 2001.