PARALLEL, NAVIER-STOKES COMPUTATION OF THE FLOWFIELD OF A HOVERING HELICOPTER ROTOR

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

BY

MURAT GEÇGEL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

THE DEPARTMENT OF AEROSPACE ENGINEERING

DECEMBER 2003

Approval of the Graduate School of Natural and Applied Sciences.

Prof. Dr. Canan Özgen
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. Yusuf Özyörük
Supervisor

Examining Committee Members

Prof. Dr. Haluk Aksel _____

Prof. Dr. Nafiz Alemdaroğlu _____

Assoc. Prof. Dr. Sinan Eyi _____

Assoc. Prof. Dr. Yusuf Özyörük _____

Assoc. Prof. Dr. İsmail H. Tuncer _____

ABSTRACT

PARALLEL, NAVIER – STOKES COMPUTATION OF THE FLOWFIELD OF A HOVERING HELICOPTER ROTOR BLADE

Geçgel, Murat

M.S., Department of Aerospace Engineering

Supervisor: Assoc. Prof. Dr. Yusuf Özyörük

December 2003, 97 pages

The aim of this study is to extend a parallel Fortran90 code to compute three – dimensional laminar and turbulent flowfields over rotary wing configurations. The code employs finite volume discretization and the compact, four step Runge – Kutta type time integration technique to solve unsteady, thin – layer Navier – Stokes equations. Zero – order Baldwin – Lomax turbulence model is utilized to model the turbulence for the computation of turbulent flowfields. A fine, viscous, H type structured grid is employed in the computations. To reduce the computational time and memory requirements parallel processing with distributed memory is used. The data communication among the processors is executed by using the MPI (Message Passing Interface) communication libraries. Laminar and turbulent solutions around a two bladed UH – 1 helicopter rotor and turbulent solution around a flat plate is

obtained. For the rotary wing configurations, nonlifting and lifting rotor cases are handled seperately for subsonic and transonic blade tip speeds. The results are, generally, in good agreement with the experimental data.

Keywords: Rotary wing, thin – layer Navier – Stokes equations, finite volume method, structured grid, parallel processing, MPI, blade – vortex interactions, flat plate.

ÖZ

HAVADA ASILI DURAN HELİKOPTER PALİ ETRAFINDAKİ AKIM ALANININ PARALEL, NAVIER-STOKES HESAPLANMASI

Geçgel, Murat

Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü

Tez Yöneticisi: Doç. Dr. Yusuf Özyörük

Aralık 2003, 97 sayfa

Bu çalışmanın amacı, döner kanat konfigurasyonları etrafındaki üç boyutlu laminar ve türbulanslı akış alanlarını hesaplayabilmek için paralel bilgisayarlar üzerinde çalışabilecek bir Fortran 90 programı geliştirmek için mevcut bir Euler kodunu modifiye etmektir. Program, zamana bağlı, ince – tabakalı Navier – Stokes denklemlerini çözmek için sonlu hacim ayrıştırma metodunu ve dört aşamalı Runge – Kutta tipi zaman integrasyon tekniğini uygulamaktadır. Türbulanslı akış alanlarının hesaplanması için sıfır – derece Baldwin – Lomax türbulans modeli kullanılmaktadır. Hesaplamalarda viskoz çözümler için üretilen H tipi yapısal olan çözüm ağı kullanılmaktadır. Kullanılan hesaplama zamanı ve hafıza gereksinimlerini azaltmak için paralel işlem tekniği uygulanmaktadır. İşlemciler arasındaki veri iletişimi, MPI (

Message Passing Interface) iletişim kütüphaneleri kullanılarak gerçekleştirilmektedir. İki palli UH – 1 helikopter rotoru etrafındaki laminer ve türbulanslı çözümler ve düz tabaka etrafındaki türbülanslı çözüm elde edilmektedir. Döner kanat konfigurasyonları için, kaldırma kuvveti üreten ve üretmeyen rotor durumları ses altı ve ses altı ses üstü geçiş hızları için ayrı ayrı ele alınmaktadır. Elde edilen sonuçlar, deneysel sonuçlarla oldukça iyi uyum göstermektedir.

Anahtar Kelimeler : Döner kanat, ince – tabaklı Navier – Stokes denklemleri, sonlu hacim metodu, yapısal olan çözüm ağı, paralel işlem, pal – girdap etkileşimleri, düz tabaka.

To my dear parents

ACKNOWLEDGMENTS

I would like to express sincere appreciation to my thesis supervisor Assoc. Prof. Dr. Yusuf Özyörük for his valuable guadiance and directions throughout this study.

I gratefully thank Assoc. Prof. Dr. İ. Hakkı Tuncer for his supports, suggestions and comments.

I also want to thank the Department of Aerospace Engineering for creating a family like atmosphere .

I am very thankfull also to my family, for their understanding, extraordinary support and assistance.

Finally, I offer sincere thanks to my research assistant friends; Mustafa Kaya and ömer Onur for their valuable support, friendship and understanding.

TABLE OF CONTENTS

LIST OF FIGURES

Figures

LIST OF SYMBOLS

c	local speed of sound	Q	source of q
C_L	lift coefficient	$\vec{r}$	position vector
C_p	pressure coefficient	R	gas constant
D	artificial dissipation term	$\vec{R}$	vector of residuals
e	internal energy per unit mass	R_∞, R_e	fixed and extrapolated Riemann invariants
E	total energy per unit mass	Re_∞	freestream Reynolds Number
E_r	total roenergy		
$\vec{f}, \vec{g}, \vec{h}$	inviscid flux components along ξ, η and ζ directions	$\vec{S}$	area vector
		T	static temperature
$\vec{f}_v, \vec{g}_v, \vec{h}_v$	viscous flux components along ξ, η and ζ directions	$\vec{T}$	rotational sources in inertial frame of reference
$\vec{F}$	flux vector	$\vec{T}_r$	rotational sources in rotating frame of reference
$\tilde{\vec{F}}$	matrix containing the flux terms	u, v, w	cartesian components of fluid velocity
k	coefficient of thermal conduc-divity	u_r, v_r, w_r	cartesian components of relative velocity
MPI	Message Passing Interface Communication Standart	$\vec{V}$	fluid velocity vector
		$\vec{V}_r$	velocity vector relative to a rotating frame of reference
M_{tip}	tip Mach Number	x, y, z	cartesian coordinates
M_∞	freestream Mach Number	α_o	collective pitch angle
P	fluid static pressure	Δt	local time step
P_r	Prandtl Number	γ	ratio of specific heats
q	conservative quantity	λ	second coefficient of viscosity
$\vec{q}$	flowfield vector		

$\alpha_1, \alpha_2, \alpha_3, \alpha_4$	Runge-Kutta coefficients
$\lambda_\xi, \lambda_\eta, \lambda_\zeta$	maximum eigenvalues of the flux Jacobian matrices
μ	dynamic viscosity
ω	angular velocity
Ω	control volume
ρ	fluid density
$\sigma_{xx}, \sigma_{yy}, \sigma_{zz}$	normal stress components
$\tau_{xy}, \tau_{xz}, \tau_{yz}$	shear stress components
ξ, η, ζ	curvilinear coordinates
ξ_x, ξ_y, ξ_z	x, y, z components of the area vector of a constant ξ surface
η_x, η_y, η_z	x, y, z components of the area vector of a constant η surface
$\zeta_x, \zeta_y, \zeta_z$	x, y, z components of the area vector of a constant ζ surface
α	dissipation exponent

CHAPTER 1

INTRODUCTION

Accurate computation of helicopter rotor flows in both hover and forward flight is a particularly challenging task since the conditions in which helicopter rotor blades operate is very complex [1]. The effects that are encountered within rotor flows include unsteady vortex interaction with following blades (BVI), a range of compressible flows from incompressible at the root to highly compressible at the rotor tip, shock boundary layer interactions, strong tip vortices dominating the flowfield and constantly changing aerodynamic environment and loads. The goal of this work is to modify an existing Euler code [26] to develop a computer code having the capability of obtaining viscous, 3-D flow solutions of rotary wing configurations. Computational fluid dynamics is resorted for this aim. It is a powerful tool to achieve this purpose since CFD is the most appropriate way of incorporating more complete mathematical models to obtain favourable predictions of complex flows dominated by non – linear effects.

A helicopter is a flying machine with rotating wings (i.e.,rotors) that provide lift, propulsion, and control forces that enable the aircraft to hover relative to the ground without forward speed or fly forward or even backward. The thrust on the rotor(s) is generated by the aerodynamic forces created on the spinning blades. The

helicopter must be able to fly forward, climb, cruise at speed, and then descend and come back into a hover for landing. This demanding flight capability comes at a price including mechanical and aerodynamic complexity and high power requirements than for a fixed wing aircraft of the same gross weight. All these factors influence the design, acquisition and operational costs of the helicopter. Although it is considered by some to be a basic and somehow cumbersome looking aircraft, the modern helicopter is indeed a machine of considerable engineering sophistication and refinement and plays a unique role in modern aviation provided by no aircraft. Besides generating all of the vertical lift, the rotor is also the primary source of control and propulsion for the helicopter, whereas these functions are seperated on a fixed-wing aircraft [2]. This property of helicopter rotor blades distinguishes it from fixed wing aircraft and therefore aerodynamic analysis and design of a helicopter requires considerable effort.

Increasing demands for low-noise rotors and smaller rotor-airframe spacings have fueled the need to understand the formation and evolution of the near-wake of a helicopter rotor. Blade Vortex Interaction (BVI) noise is one of the biggest concerns in community acceptance of helicopters as a commercial transport and is an issue for detection avoidance in military purposes [3]. BVI usually occurs at very early stages of the tip-vortex. Hence, knowledge of the vortex structure in the near wake is of great importance in formulating prediction models. The wake from the main rotor, especially the tip-vortex, is the primary component in many aerodynamic interactions. Smaller rotor-airframe spacings imply that the tip-vortex interacts with the airframe without losing its strength much and therefore the airframe is subject to severe vortex loads and vortex - induced separation. The ability to modify the rotor-wake in order to reduce interaction effects also depends on knowledge of the tip-

vortex formation at the blade tip. The understanding of the near-wake is thus essential to ease the difficulties in prediction of rotorcraft aerodynamics.

An improved understanding of the aerodynamics of helicopter rotors is one key to advanced rotor designs with better performance, lower vibratory loads, and less noise. The loads and overall performance of the helicopter rotor are highly dependent upon the vortical wake structure below the rotor, and are primarily influenced by the strengths and locations of the blade tip vortices.

The motion of a fluid is usually defined as *laminar* or *turbulent*. A laminar flow is an ordered, predictable and layered flow, as opposed to the chaotic, swirly and fluctuating turbulent flow. In a laminar flow the velocity gradients and the shear stresses are smaller; consequently the drag force over the surface of a vehicle is much lower than in a turbulent flow [5]. One of the major challenges in aircraft design is in fact to obtain a laminar flow over the wings to reduce the friction in order to save fuel. On the other hand, a turbulent flow provides an excellent mixing in the flow because of the chaotic motion of the fluid particles, and it is, therefore, required in chemical reactors or combustion engines. In real applications, as the velocity of the fluid or the physical dimension bounding the flow increases, a laminar motion cannot be sustained; the perturbations inevitably present within the flow are amplified and the flow evolves into a turbulent state. This phenomenon is called *transition* [5]. Within the range of Reynolds numbers at which helicopter rotor blades operate, boundary layer laminar to turbulent transition will occur and this will have many significant effects on rotor aerodynamics [6]. This is of particular importance in the calculation of the unsteady dynamic stall of the retreating blade in forward flight. In order to accurately simulate the detailed features of this flow phenomenon it is critical that the flow at the leading edge of the blade be correctly

represented. In this study, however, transition is not investigated. Fully laminar and turbulent solutions are obtained for rotary wing configurations by solving numerically the so called thin - layer Navier - Stokes equations.

For laminar flows, solutions to the Navier-Stokes equations are considered to be as accurate as numerical computations can be. But unfortunately, most problems where viscous effects are important are classified as turbulent flows. The turbulent energy of any shear flow consists of production, convection, diffusion, and dissipation [7]. For a zero pressure gradient flow, dissipation and production are the main contributors to the turbulent energy balance except near the outer edge. In the case of backflow, negligible turbulent energy production occurs [7]. Instead, diffusion and dissipation are the main terms in the energy balance. Near the wall, regions of low kinetic energy will exist, while away from the wall shear stresses are large. Normal stress contributions, which are negligible in zero pressure gradient flows, become significant in the momentum and turbulence energy equations. This production of both the normal and shear stresses supply turbulence energy in the outer region through turbulent diffusion [8]. To model the turbulence and to compute the turbulent viscosity, there are many turbulence models available. In this study Baldwin – Lomax turbulence model is utilized. The Baldwin and Lomax turbulence model [9] is an algebraic eddy viscosity, zero equation model. The model is based on the mixing length hypothesis, and is very similar to the Cebeci-Smith model [9]. The model has been widely used and tested, and remains a very popular turbulence model. Several advantages of the model are its simplicity in coding and the fact that it does not require the boundary layer edge thickness or edge velocity as an input parameter. The calculation of these parameters is not always straightforward and can lead to errors in the turbulence model if not computed correctly. The Baldwin and

Lomax model works best in wall bounded flows with favorable pressure gradients. As the flow physics become more complicated, as well as the geometry of the model being tested, the performance of this turbulence model greatly decreases.

Computational fluid dynamics can be utilized as the most effective tool for prediction of rotorcraft aerodynamics and noise [10]. It is a powerfull, cost-effective tool for the study of the aerodynamic characteristics of a flow. Through careful selection of the grid geometry and resolution, as well as the computational technique used, the sophistication of the analysis can be tailored to meet the needs of the researcher , while conforming to meet the imposed time and cost constraints. In order to conduct a CFD analysis, three main tasks must be completed: grid generation, or pre-processing; the actual computational processing of the analysis; and visualization of the computational results or post-processing. Before CFD can be used to calculate the aerodynamic charecteristics of a flow, the physical space through which the flow passes must be divided into a number of discrete points, forming a grid. The grid must be then transformed from the physical domain to computational domain which is a form that the flow solver program understands and uses to perform the numerical calculations.

In general, any type of analytical method that can be applied over a discrete set of points can be used as the basis of a CFD analysis. In rotor analysis, the methods that can be used include basic momentum theory, blade element theory or blade element momentum theory, which are used to predict rotor performance and vortex lattice methods , which are used to investigate vortex wake effects [10]. However, since they provide the highest level of sophistication available, and take into account effects such as compressibility, rotationality and interaction between components , which are neglected in other methods, unsteady, three-dimensional ,compressible

Navier-Stokes equations are more convenient to be solved for predicting sophisticated flow domains. The system of Navier - Stokes equations constitutes the most complete description of viscous non - heat conducting flows and hence is the highest level of approximation for viscous fluids [11].

Many current algorithms in CFD are based on the numerical solution of conservation laws. The main motivation is the ability to threat flow discontinuities automatically. The main difference between the Euler and Navier-Stokes equations is the inclusion of viscosity. Solutions found using the Euler equations are valid for inviscid flow, while solutions derived from the Navier-Stokes equations include viscous effects. The Navier - Stokes equations are more costly both in time and money since the inclusion of viscosity introduces additional complexity into the flow solver code. Besides, Navier – Stokes computations requires extremely fine meshes to resolve the thin layer regions. Instead of a complete solution, the gradients parallel to the body surface are neglected. This leads to a set of viscous equations, so called thin - layer Navier - Stokes equations [12]. Particularly, in order to obtain viscous solutions of rotating flowfields, solving the thin - layer Navier - Stokes equations causes a valuable decrease in both computational time and CPU load.

In order to solve the Euler or Navier-Stokes equations numerically, either finite difference or finite volume formulation can be used. In finite difference formulation, the values of the flow properties at each grid point are found by using an algebraic equation, specific to that grid point, that approximates the governing partial differential equations. The algebraic finite difference equations calculate the flow properties at a given point based on the flow properties at adjacent points, so this formulation requires the use of structured grid in order to determine the relationships between grid points. In the finite volume formulation, which is applied in this work

also, the governing equations written in integral form so the restriction on the type of grid that can be used is no longer valid. In finite volume formulation, flow properties are computed at cell centers and assumed uniform through the cell volume. The finite difference formulation is more convenient because it uses structured grids, but the finite volume formulation is more useful since it can use either structured or unstructured grids. On the other hand, the effect of using one of these methods of approach on the numerical results is generally small in most cases.

Thin – layer Navier – Stokes equations [11,12] have been numerically solved for the flowfield of the rotating helicopter blade and for flowfield of a flat plate. Solutions of three - dimensional Navier-Stokes equations require considerable amount of CPU time and memory. The power of conventional serial or vector computers is not adequate to allow computations of three-dimensional Euler or Navier-Stokes equations especially for the domains having high number of grid points. In order to meet this high cpu memory and time demand , parallel processing, in which computational job is distributed over finite number of precessors can be resorted. In this process, the computational domain is decomposed into sub domains and computational work is performed simultaneously by each processor which communicates among themselves using ghost cells which are the first and last two cells of each domain in each direction of computational space. Data transfer between the processors is performed by using Message Passing Interface (MPI) communication standart.

A parallel TLNS solver is developed by modifying a pre – existing Euler code. It can predict viscous flowfield solutions around rotary wings as well as it can predict viscous solutions for fixed wings. Euler solutions for rotary wing configurations was obtained by E.Alpman [26]. Comparions between viscous and Euler results are

available in chapter 5. This is an ongoing study. Later, the code can be modified to utilize for aeroelastic investigations. The developed code is run on a cluster of dual Pentium-III processors (700 MHz and 512 MB Ram on each) housed in the Department of Aerospace Engineering of the Middle East Technical University.

CHAPTER 2

GOVERNING EQUATIONS AND BOUNDARY CONDITIONS

2.1 Navier – Stokes Equations

The set of Navier – Stokes Equations is the most general description of the fluid flow in thermodynamic equilibrium [13]. It is basically the collection of conservation of mass, conservation of momentum and conservation of energy equations written for a Newtonian fluid. Differential form can be written as :

$$\frac{\partial \vec{Q}}{\partial t} + \frac{\partial\left(\vec{E} - \vec{E}_v\right)}{\partial x} + \frac{\partial\left(\vec{F} - \vec{F}_v\right)}{\partial y} + \frac{\partial\left(\vec{G} - \vec{G}_v\right)}{\partial z} = 0 \qquad (2.1)$$

The integral form, which is also the basis of all finite volume algorithms, can be written as :

$$\frac{\partial}{\partial t}\int_{\Omega} \vec{q}\, d\Omega + \oint_S \vec{F} \cdot d\vec{S} = 0 \qquad (2.2)$$

where $\vec{q}$ is the vector containing the conservative variables.

$$\vec{q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix} \tag{2.3}$$

and $\vec{F}$ is the matrix containing the flux terms. The columns are as follows:

$$\vec{F}_1 = \begin{bmatrix} \rho u \\ \rho u^2 + p - \sigma_{xx} \\ \rho uv - \tau_{xy} \\ \rho uw - \tau_{xz} \\ (E+p)u - u\sigma_{xx} - v\tau_{xy} - w\tau_{xz} - k\dfrac{\partial T}{\partial x} \end{bmatrix} \tag{2.4}$$

$$\vec{F}_2 = \begin{bmatrix} \rho v \\ \rho uv - \tau_{xy} \\ \rho v^2 + p - \sigma_{yy} \\ \rho vw - \tau_{yz} \\ (E+p)v - u\tau_{xy} - v\sigma_{yy} - w\tau_{yz} - k\dfrac{\partial T}{\partial y} \end{bmatrix} \tag{2.5}$$

$$\vec{F}_3 = \begin{bmatrix} \rho w \\ \rho uw - \tau_{xz} \\ \rho vw - \tau_{yz} \\ \rho w^2 + p - \sigma_{zz} \\ (E+p)w - u\tau_{xz} - v\tau_{yz} - w\sigma_{zz} - k\dfrac{\partial T}{\partial z} \end{bmatrix} \tag{2.6}$$

where

$$\sigma_{xx} = 2\mu\frac{\partial u}{\partial x} - \frac{2}{3}\mu\nabla\cdot\vec{V} \tag{2.7}$$

$$\sigma_{yy} = 2\mu\frac{\partial v}{\partial y} - \frac{2}{3}\mu\nabla\cdot\vec{V} \tag{2.8}$$

$$\sigma_{zz} = 2\mu\frac{\partial w}{\partial z} - \frac{2}{3}\mu\nabla\cdot\vec{V} \tag{2.9}$$

$$\tau_{xy} = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \tag{2.10}$$

$$\tau_{xz} = \mu\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \tag{2.11}$$

$$\tau_{yz} = \mu\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \tag{2.12}$$

Note that,

$$\vec{F}_1 = \left(\vec{E} - \vec{E}_v\right) \tag{2.13}$$

$$\vec{F}_2 = \left(\vec{F} - \vec{F}_v\right) \tag{2.14}$$

$$\vec{F}_2 = \left(\vec{G} - \vec{G}_v\right) \tag{2.15}$$

There are six unknowns ρ, u, v, w, p, E and T, but five equations. So the number of equations is not adequate to close the system. To close the system of equations, equation of state is used:

$$p = (\gamma - 1)\left[E - \frac{\rho}{2}\left(u^2 + v^2 + w^2\right)\right] \tag{2.16}$$

$$T = \frac{\gamma - 1}{R}\left[\frac{E}{\rho} - \frac{1}{2}\left(u^2 + v^2 + w^2\right)\right] \tag{2.17}$$

where R is the gas constant and γ is the ratio of specific heats. $R = 287\, m^2/s^2 K$ and $\gamma = 1.4$ for air.

A relation also can be established between dynamic viscosity and temperature by using Sutherland's formula:[14]

$$\mu = 1.458 \times 10^{-6}\, kg \Big/ \left(msK^{\frac{1}{2}}\right) \frac{T^{\frac{3}{2}}}{T + 110.4K} \tag{2.18}$$

k is the thermal conductivity coefficent and can be related to the dynamic viscosity by using Prandtl number Pr, which describes the ratio of momentum and thermal diffusivities:[15]

$$\mathrm{Pr} = \frac{\mu C_p}{k} \tag{2.19}$$

where C_P is the specific heat coefficient under constant pressure. After these definitions we can define fluid conductivity as :

$$k = \frac{\mu C_p}{\mathrm{Pr}} \tag{2.20}$$

For the applications in which a rotationary motion exists, such as turbomachinery problems or the propellers, a rotating (blade fixed) frame of reference has to be introduced properly to the conservation equations [16]. By using

a properly defined rotational frame together with the absolute (inertial) reference of frame and establishing a correct pattern for transformations between relative and absolute variables, one can obtain a system of equations with rotational effects to be solved in order to capture proper solutions of flow domains around rotating bodies.

Let ω is the angular velocity of a system rotating steadily, V_R is the velocity relative to the rotating system and V is the velocity relative to the absolute reference system.

The relation between the absolute and the relative velocities can be established as:

$$\vec{V} = \vec{V}_R + \vec{\omega} \times \vec{r} \tag{2.21}$$

Here $\vec{r}$ is the distance vector from the center of rotation to cell centers.The term $\vec{\omega} \times \vec{r}$ is simply grid velocity or entrainment velocity. Two additional terms comes with momentum equation. First one is coriolis force per unit volume and can be defined as :

$$\vec{F}_{cor} = -2\rho\left(\vec{\omega} \times \vec{V}_r\right) \tag{2.22}$$

And the second one is the centrifugal force per unit volume :

$$\vec{F}_{cent} = -\rho\vec{\omega} \times \left(\vec{\omega} \times \vec{r}\right) \tag{2.23}$$

After the relations are defined properly between absolute and relative velocities the system of Navier - Stokes equations in relative frame of referance can be written as :

$$\frac{d}{dt}\int_{\Omega}\vec{q}_r\,d\Omega+\oint_{S}(\vec{F}_r-F_v)\cdot d\vec{S}+\int_{\Omega}\vec{T}_r\,d\Omega=0 \qquad (2.24)$$

The system of equations defined above includes relative velocities and can be solved for relative flow variables. If one decides to solve the sytem of equations for the absolute flow variables, the equations have to be rewritten for absolute velocities. The system of Navier - Stokes equaitons in absolute frame of referance can be written as :

$$\frac{\partial}{\partial t}\int_{\Omega}\vec{q}\,d\Omega+\oint_{S}\left(\vec{F}-\vec{F}_v\right)\cdot d\vec{S}+\int_{\Omega}\vec{T}d\Omega=0 \qquad (2.25)$$

The obtained set of equations can be assigned to acquire steady – state, viscous flow solutions around hovering helicopter rotor blades.

2.2 Thin Layer Navier – Stokes Equations

The normal gradient of the stress terms are much larger than the streamwise and spanwise gradients for the cases where the flow separation is moderate [12]. Computer capability is an important constraint for the selection of full set of Navier – Stokes equations. When Navier – Stokes equations are selected, additional grid points in the streamwise and spanwise directions must be included. To resolve the viscous gradients in these directions, a large number of grid points is required. Therefore, solution of Navier – Stokes equations require more memory and storage capacity. Instead of solving the complete set, one can prefer solving a subset which can be obtained by neglecting the viscosity and heat conduction terms in streamwise and spanwise directions. This subset is called Thin Layer Navier – Stokes equations.

Thin layer Navier – Stokes equations can be given in differential form as :

$$\frac{\partial \vec{Q}}{\partial t} + \frac{\partial \left(\vec{E}\right)}{\partial x} + \frac{\partial \left(\vec{F} - \vec{F}_v\right)}{\partial y} + \frac{\partial \left(\vec{G}\right)}{\partial z} = 0 \qquad (2.26)$$

2.3 Turbulence Model

To model the turbulence in the flow field, the laminar viscosity is replaced by

$$\mu = \mu_l + \mu_{tur} \qquad (2.27)$$

where the the turbulent viscosity μ_{tur} is computed using the Baldwin-Lomax turbulence model.

It is an algebraic (zero order) model and is used for wide range of cascade flow applications. It is a turbulence model with which turbulent quantities are calculated in almost all the cascade flow solutions since it is easy to apply and does not require the solutions of any partial differential equation.

The model is based on local flow properties and do not require the calculation of some boundary layer properties such as boundary layer thickness, velocity thickness and edge velocity, which are sometimes difficult especially for the separated flows and the flows with shock waves [9]. The eddy viscosity is given for the inner and outer viscous layers as follows :

Inner layer :

$$\mu_{tur_i} = l_{mix}{}^2 \left|\Gamma\right| \qquad (2.28)$$

$$l_{mix} = Ky\left[1 - e^{-y^+ / A^+_0}\right] \qquad (2.29)$$

where the mixing length is the length scale of the turbulence.

Outer layer :

$$\mu_{tur_o} = \alpha C_{cp} F_{wake} F_{kleb}\left(y; y_{\max} / C_{kleb}\right) \tag{2.30}$$

$$F_{kleb} = \min\left[y_{\max} F_{\max}; C_{wk} y_{\max} {U_{diff}}^2 / F_{\max}\right] \tag{2.31}$$

$$F = \left[1 + 5.5\left(C_{Kleb} \frac{y}{y_{\max}}\right)^6\right]^{-1} \tag{2.32}$$

$$F_{\max} = \frac{1}{K}\left[\max\left(l_{mix}|\Gamma|\right)\right] \tag{2.33}$$

where $y_{\max}$ is the value of y at where $l_{mix}|\Gamma|$ reaches its maximum value and the coefficients are :

K = 0.40 α = 0.0168 ${A_0}^+$ = 26

C_{cp} = 1.6 C_{kleb} = 0.3 C_{wk} = 1

The function F_{kleb} is Klebanoff's intermittency function and Γ is the magnitude of the vorticity vector,i.e.,

$$\Gamma = \sqrt{\left(\frac{\partial w}{\partial y} - \frac{\partial v}{\partial z}\right)^2 + \left(\frac{\partial u}{\partial z} - \frac{\partial w}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right)^2} \tag{2.34}$$

U_{diff} is the maximum value of U for boundary layers. For more general flows it is defined as ,

$$U_{diff} = (\sqrt{u^2 + v^2})_{\max} - \left(\sqrt{u^2 + v^2}\right)_{y=y\max} \tag{2.35}$$

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

2.4 Boundary Conditions

The numerical solution of the governing eguations requires the appropriate application of the boundary conditions. In this thesis different types of boundary conditions are applied such as wall boundary conditions, farfield boundary conditions, and the periodic boundary conditions.

Wall boundary conditions are applied at the airfoil surfaces. Computation of the fluxes are performed by using absolute velocities. But boundary conditions are applied at the wall surface using relative velocities. Therefore, absolute velocities are transformed to relative velocities. That is performed as;

$$u_{rel} = u_{abs} + \omega \times z(i, j, k)$$

$$v_{rel} = v_{abs}$$

$$w_{rel} = w_{abs} - \omega \times x(i, j, k)$$

For viscous flow calculations no slip boundary conditions are applied for the velocity as;

$$\vec{V}_{rel}\Big|_{surface} = 0$$

The pressure is obtained by setting the normal pressure gradient to zero. That is;

$$\frac{\partial P}{\partial n}\Big|_{surface} = 0$$

The density is extrapolated from the interior. Total energy is obtained from the equation of state.

One – dimensional Riemann invariants [17] are utilized for the farfield boundary conditions. For a subsonic farfield, the fixed and the extrapolated Riemann invariants are defined as [17]:

$$R_{\infty} = V_{n\infty} - \frac{2c_{\infty}}{\gamma - 1} \tag{2.36}$$

$$R_{e} = V_{ne} - \frac{2c_{e}}{\gamma - 1} \tag{2.37}$$

where ∞ and e indicates freestream and the values extrapolated from the interior cells, respectively, V_n and c are the velocity component normal to the boundary and the local speed of sound, respectively.

Actual normal velocity and the actual speed of sound at the farfield boundary can be obtained by using these invariants as :

$$V_n = \frac{1}{2}\left(R_e + R_\infty\right) \tag{2.38}$$

$$c = \frac{1}{4}(\gamma - 1)\left(R_e - R_\infty\right) \tag{2.39}$$

At the outflow boundary, tangential velocity components and entropy are extrapolated from the interior cells while at an inflow boundary they are assigned as having farfield values.

For hovering helicopter rotor problems, both the geometry and the flow domain are asymmetric. Solving the flow around a single blade is a way of saving the computational load and time. To achieve this, periodic boundary conditions are applied at the vertical inflow and outflow boundaries in order to perform the calculations for only one blade sector. These conditions are used to transmit data from vertical inflow plane to vertical outflow plane and vice versa [18].

CHAPTER 3

DISCRETIZATION OF THE GOVERNING EQUATIONS

Selection of spatial discretization and time integration techniques for a CFD work is quite important duty since the performance of the code depends on these approaches. In this study, a cell centered finite volume scheme is used for spatial discretization and four-step Runge-Kutta type time integration technique is used for temporal discretization. In order to avoid odd-even decoupling occured in central difference schemes, and to suppress numerical oscillations which are mainly encountered strongly in the vicinity of the shock waves and stagnation points, artificial dissipation terms are added explicitly to the system of equations.

3.1 Finite Volume Method

The conservation laws of fluid motion may be expressed mathematically in either differential or integral form. Finite volume methods are based on the discretization of the integral form of the conservation equations directly in the physical space [19]. When a numerical scheme is applied to the differential form of equations, the domain of solution is divided into discrete points, upon which the finite difference equations are solved. On the other hand, when the integral form of the equations is utilized, the

domain of solution is divided into small volumes (or areas for a two-dimensional case) whose vertices are the grid points.

Since the majority of physical domains are irregular in shape, a coordinate transformation from a physical space to a computational space is performed where the computational domain is rectangular. However, even with the coordinate transformation is available, domains which are highly irregular would create serious difficulties in accuracy and convergence of the solution. The reason is that the metrics and Jacobian of transformation and the corresponding gradients which are used in the governing equations may include numerical discontinuities if the grid system is not relatively smooth. In general the finite difference methods posses inherited weakness for highly complicated domains while finite volume schemes do not encounter such waekness [10]. That is because the independent variables are integrated directly on the physical domain and therefore, grid smoothness is no longer an important issue.

The computational grid divides the flowfield into computational cells where the grid points are the cell vertices. The finite volume method is based on the discretization of the general form of conservation law with the surface integral replaced by the sum of discrete integrals over the faces of the computational cell. The discretized form of equation can be written as :

$$\frac{\partial}{\partial t}\left(q_{i,j,k}\Delta V_{i,j,k}\right)+\sum_{sides}\left(\vec{F}\cdot\vec{S}\right)=0 \qquad (3.1)$$

Here, $q_{i,j,k}$ refers to the value of q at the cell center (i,j,k), $\Delta V_{i,j,k}$ refers to the volume of the computational cell and the term with the summation mark refers to

total flux through all external faces. It is generally considered that $q_{i,j,k}$ is the average value of q in the (i, j, k) ordered cell. But in order to calculate a surface flux, it is convenient to think of $q_{i,j,k}$ as the value of q at some average point in the cell. A charecteristic of the finite volume method is that the precise location of this average point is not required during the calculation [19]. Only in the output of the solution the location of this point is desired. Some investigators have suggested the cell centroid for this point. This is strictly valid if and only if all the components of q vary linearly througout the cell. Since the distribution of q is not known, the centroid has no particular advantage over the cell center defined as the vectorial average of the cell vertices. The latter point is easier to calculate, and is therefore preferable. One also has to define how to estimate the volume and cell surface areas, and how to approximate the fluxes at the faces. On the other hand, the sum of the cell volumes should be equal to the volume of vhole domain. Another important constraint has to be satisfied is the closed – cell condition which can be defined as follows :

$$\sum_{sides} \vec{S} = 0 \qquad (3.2)$$

For a three-dimensional problem, the computational cells are hexahedrons as illustrated in Figure 3.1, where ξ, η and ζ denote the curvilinear coordinates. For a hexahedron, the cell surface areas can be approximated as the cross products of the diagonal vectors. Here, one has to take into account the direction of the surface normal properly. In this study, the surface area vectors are computed such that the surface normal of a constant ξ-face always points positive ξ-direction.

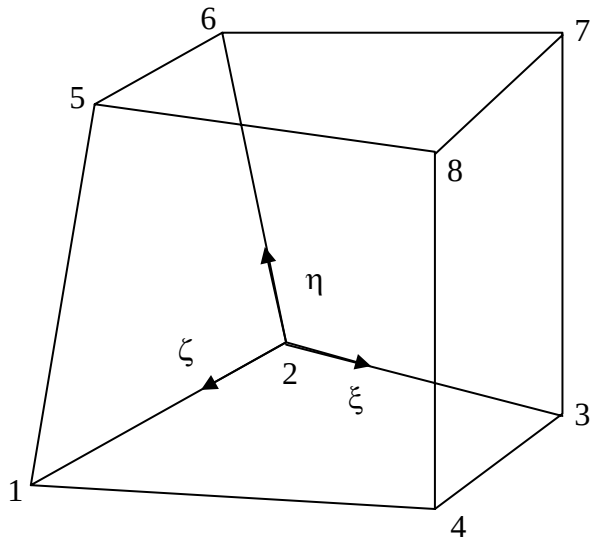

Figure 3.1:Typical hexahedral cell

Here, one has to take into account the direction of the surface normal properly. In this study, the surface area vectors are computed such that the surface normal of a constant ξ-face always points positive ξ-direction. Then, the surface area vector of face 1562, for example, can be computed as [19]:

$$\vec{S}_{1562} = \frac{1}{2}\left(\vec{r}_{61} \times \vec{r}_{25}\right) \tag{3.3}$$

The cell volume can be computed with the following equation [19]:

$$\Delta V_{12345678} = \frac{1}{6}\left[\left(\vec{S}_{1485} + \vec{S}_{1234} + \vec{S}_{1562}\right) \cdot \left(\vec{r}_7 - \vec{r}_1\right)\right] \tag{3.4}$$

where i, j and k denote the indices in the ξ, η and ζ surfaces respectively. The integer subscripts indicate the cell centers and the fructional subscripts indicate cell faces. Equation (3.1) can be rewritten as :

$$\frac{\partial}{\partial t}\left(q_{i,j,k}\Delta V_{i,j,k}\right)+\vec{F}_{i+\frac{1}{2},j,k}\cdot\vec{S}_{i+\frac{1}{2},j,k}-\vec{F}_{i-\frac{1}{2},j,k}\cdot\vec{S}_{i-\frac{1}{2},j,k}+\vec{F}_{i,j+\frac{1}{2},k}\cdot\vec{S}_{i,j+\frac{1}{2}}-\vec{F}_{i,j-\frac{1}{2},k}\cdot\vec{S}_{i,j-\frac{1}{2},k}+\vec{F}_{i,j,k+\frac{1}{2}}\cdot\vec{S}_{i,j,k+\frac{1}{2}}-\vec{F}_{i,j,k-\frac{1}{2}}\cdot\vec{S}_{i,j,k-\frac{1}{2}}=0 \qquad (3.5)$$

3.2 Artificial Dissipation

In order to avoid odd-even point decoupling experienced by the application of central schemes, aritificial dissipation can be added explicitly to the discretized equations [21]. It provides damping of the high frequency modes, which can originate due to odd-even decoupling. On the other hand, artificial dissipation terms eliminate numerical oscillations in the neighbourhood of shock waves.

Artificial dissipation terms are added to the numerical scheme by modifying the flux terms. Modified form of the conservation equation can be rewritten as :

$$\frac{\partial}{\partial t}\left(q_{i,j,k}\cdot\Delta V_{i,j,k}\right)+\vec{F}_{i+\frac{1}{2},j,k}\cdot\vec{S}_{i+\frac{1}{2},j,k}-\vec{D}_{i+\frac{1}{2},j,k}-\vec{F}_{i-\frac{1}{2},j,k}\cdot\vec{S}_{i-\frac{1}{2},j,k}+\vec{D}_{i-\frac{1}{2},j,k}+\vec{F}_{i,j+\frac{1}{2},k}\cdot\vec{S}_{i,j+\frac{1}{2},k}-\vec{D}_{i,j+\frac{1}{2},k}-\vec{F}_{i,j-\frac{1}{2},k}\cdot\vec{S}_{i,j-\frac{1}{2},k}+\vec{D}_{i,j-\frac{1}{2},k}$$

$$+\vec{F}_{i,j,k+\frac{1}{2}}\cdot\vec{S}_{i,j,k+\frac{1}{2}}-\vec{D}_{i,j,k+\frac{1}{2}}$$
$$-\vec{F}_{i,j,k-\frac{1}{2}}\cdot\vec{S}_{i,j,k-\frac{1}{2}}+\vec{D}_{i,j,k-\frac{1}{2}}=0 \qquad (3.6)$$

where $D_{i\mp\frac{1}{2},j,k}$, $D_{i,j\mp\frac{1}{2},k}$ and $D_{i,j,k\mp\frac{1}{2}}$ denotes the dissipative terms in the i, j and k directions, respectively. A typical dissipative term in the i- direction can be written as [21]:

$$D_{i+\frac{1}{2},j,k}=\lambda_{i+\frac{1}{2},j,k}\left[\varepsilon^{(2)}{}_{i+\frac{1}{2},j,k}\left(q_{i+1,j,k}-q_{i,j,k}\right)-\right.$$
$$\left.\varepsilon^{(4)}{}_{i+\frac{1}{2},j,k}\left(q_{i+2,j,k}-3q_{i+1,j,k}+3q_{i,j,k}-q_{i-1,j,k}\right)\right] \qquad (3.7)$$

where the coefficients $\varepsilon^{(2)}$ and $\varepsilon^{(4)}$ control the amount of dissipation added and are adapted to the flow.

Coefficients can be defined as [22]:

$$\varepsilon^{(2)}{}_{i+\frac{1}{2},j,k}=K^{(2)}\max\left(\upsilon_{i+1,j,k},\upsilon_{i,j,k}\right) \qquad (3.8)$$
$$\varepsilon^{(4)}{}_{i+\frac{1}{2},j,k}=\max\left[0,\left(K^{(4)}-\varepsilon^{(2)}{}_{i+\frac{1}{2},j,k}\right)\right] \qquad (3.9)$$

with

$$\upsilon_{i,j,k}=\frac{\left|p_{i+1,j,k}-2p_{i,j,k}+p_{i-1,j,k}\right|}{p_{i+1,j,k}+2p_{i,j,k}+p_{i-1,j,k}} \qquad (3.10)$$

The constants, $K^{(2)}$ and $K^{(4)}$ should be selected by the user. During this selection, one has to consider the fact that, the artificial dissipation terms must be

large enough for a satisfactory convergence rate and yet sufficiently small that the accuracy is not compromised[22]: For both Laminar and turbulent runs, these constants are selected as :

$$K^{(2)} = 0.5 \qquad K^{(4)} = 0.5$$

The term $\lambda_{i+\frac{1}{2},j,k}$ is a function of the eigenvalues associated with the flux Jacobian matrix, which is obtained by linearization of the system of conservation equations. In the dissipation model of Jameson [22], this term is defined as :

$$\lambda_{i+\frac{1}{2},j,k} = \left|\lambda_\xi + \lambda_\eta + \lambda_\zeta\right|_{i+\frac{1}{2},j,k} \tag{3.11}$$

Where $\lambda_\xi, \lambda_\eta$ and λ_ζ represent maximum eigenvalues of the flux Jacobian matrices in ξ, η and ζ directions, respectively. Note that the above formulation represents an isotropic scaling in the three coordinate directions. It has been shown that the model produces accurate results when the aspect ratio of the cells are close to one. But when meshes with high aspect ratio cells are encountered, as in the boundary layers, isotropic scaling fails to give accurate results because the high frequency modes are different in different coordinate directions. An anisotropic scaling model, where the linear sum of the eigenvalues is replaced by the individual eigenvalue in a given direction, can be used to overcome this problem [22]:

$$\lambda_{i+\frac{1}{2},j,k} = \frac{1}{2}\left[\left(\lambda_\xi\right)_{i+1,j,k} + \left(\lambda_\xi\right)_{i,j,k}\right] \tag{3.12}$$

$$\lambda_{i,j+\frac{1}{2},k} = \frac{1}{2}\left[\left(\lambda_\eta\right)_{i,j+1,k} + \left(\lambda_\eta\right)_{i,j,k}\right] \tag{3.13}$$

$$\lambda_{i,j,k+1} = \frac{1}{2}\left[\left(\lambda_\zeta\right)_{i,j,k+1} + \left(\lambda_\zeta\right)_{i,j,k}\right] \tag{3.14}$$

Martinelli [23] obtained good convergence rates and accurate solutions by reformulating the dissipation model as a function of the aspect ratio and individual eigenvalues. In this model the modified scaling factors can be defined as [21,22]:

$$\lambda_{i+\frac{1}{2},j,k} = \frac{1}{2}\left[\left(\bar{\lambda}_\xi\right)_{i+1,j,k} + \left(\bar{\lambda}_\xi\right)_{i,j,k}\right] \tag{3.15}$$

$$\lambda_{i,j+\frac{1}{2},k} = \frac{1}{2}\left[\left(\bar{\lambda}_\eta\right)_{i,j+1,k} + \left(\bar{\lambda}_\eta\right)_{i,j,k}\right] \tag{3.16}$$

$$\lambda_{i,j,k+1} = \frac{1}{2}\left[\left(\bar{\lambda}_\zeta\right)_{i,j,k+1} + \left(\bar{\lambda}_\zeta\right)_{i,j,k}\right] \tag{3.17}$$

where

$$\left(\bar{\lambda}_\xi\right)_{i,j,k} = \left(\lambda_\xi\right)_{i,j,k}\left[1 + \left(\frac{\lambda_\eta}{\lambda_\xi}\right)^\alpha + \left(\frac{\lambda_\zeta}{\lambda_\xi}\right)^\alpha\right] \tag{3.18}$$

$$\left(\bar{\lambda}_\eta\right)_{i,j,k} = \left(\lambda_\eta\right)_{i,j,k}\left[1 + \left(\frac{\lambda_\xi}{\lambda_\eta}\right)^\alpha + \left(\frac{\lambda_\zeta}{\lambda_\eta}\right)^\alpha\right] \tag{3.19}$$

$$\left(\bar{\lambda}_\zeta\right)_{i,j,k} = \left(\lambda_\zeta\right)_{i,j,k}\left[1 + \left(\frac{\lambda_\xi}{\lambda_\zeta}\right)^\alpha + \left(\frac{\lambda_\eta}{\lambda_\zeta}\right)^\alpha\right]$$

The selection of the exponent α requires some expertise based on numerical experimentation [21]. α was taken as 0.01 in the calculations.

3.3 Time Integration Technique

The applied discretization method leads to a set of coupled ordinary differential equations which can be written in the form [13]:

$$\frac{d}{dt}\left(\vec{q}\Delta V\right) = \vec{R}(\vec{q}) \tag{3.20}$$

where $\vec{q}$ is the vector containing the conservative flow variables and $\vec{R}$ is the vector of residuals, consisting of flux differences together with the dissipative terms :

$$\vec{R}(\vec{q}) = \sum_{sides}\left(\vec{F}\cdot\vec{S} + \vec{D}\right) \tag{3.21}$$

Two different families of methods can be used to discretize the equation (3.20), explicit methods and implicit methods. In explicit methods, the solution, $\vec{q}$, at the new time level depends only upon the previous time solutions. Thus a minimum number of arithmetic operations is required at each time step. However, this advantage is counterbalanced by the fact that the stability and convergence conditions impose severe restrictions on the maximum admissible time step. While this might not be a limitation for transient problems, it leads to the necessity of large number of iteration steps in order to reach steady state solutions for time-independent problems [16]. Implicit methods, where the solution $\vec{q}$ at a grid point may depend also upon the solutions at the current time level, are one possible remedy to the slow

convergence of explicit methods. However, most of the implicit methods require inversion of a block pentadiagonal matrix for two-dimensional problems or a block septadiagonal matrix for three-dimensional problems. These matrices have no efficient direct inversion and instead approximately factorised into simpler block tridiagonal matrices. Unfortunately, the approximate factorisation limits the step size and prevents fast convergence compared to full implicit methods [24]. It is also difficult to parallelize an implicit method due to the requirement for inverting a linear system of equations to obtain the solution.

In this study, an explicit, one step multi-stage method is prefferred and the compact four stage Runge - Kutta type time integration technique is applied. Runge - Kutta schemes provide a family of time integration techniques, which are of high order of accuracy, explicit but non-linear, and limited to two time levels. The main idea is to evaluate the right-hand side of the differential system (3.20) at several values of $\vec{q}$ in the time interval between $n\Delta t$ and $(n+1)\Delta t$ then to combine them in order to obtain an approximation of $\vec{q}^{\,n+1}$.

The general form of the four stage Runge - Kutta method is [24]:

$$\vec{q}^{(0)} = \vec{q}^{\,n} \tag{3.22}$$

$$\vec{q}^{(1)} = \vec{q}^{\,n} + \frac{\Delta t}{\Delta V}\alpha_1 \vec{R}\left(\vec{q}^{(0)}\right) \tag{3.23}$$

$$\vec{q}^{(2)} = \vec{q}^{\,n} + \frac{\Delta t}{\Delta V}\alpha_2 \vec{R}\left(\vec{q}^{(1)}\right) \tag{3.24}$$

$$\vec{q}^{(3)} = \vec{q}^{\,n} + \frac{\Delta t}{\Delta V}\alpha_3 \vec{R}\left(\vec{q}^{(2)}\right) \tag{3.25}$$

$$\vec{q}^{(4)} = \vec{q}^{\,n} + \frac{\Delta t}{\Delta V}\alpha_4 \vec{R}\left(\vec{q}^{(3)}\right) \tag{3.26}$$

and

$$\vec{q}^{\,n+1} = \vec{q}^{\,(4)} \tag{3.27}$$

where

$$\alpha_1 = \frac{1}{4},\ \alpha_2 = \frac{1}{3},\ \alpha_3 = \frac{1}{2} \quad \text{and} \quad \alpha_4 = 1$$

The scheme is fourth-order accurate in time and is stable for $CFL \le 2\sqrt{2}$ in smooth regions (linear problems) [24].

The numerical operations required for evaluating fourth differencing needed for dissipation are quite large in comparing with those required for the evaluation of the flux terms. To reduce the computational effort, it is suggested to freeze the dissipative part at its value at the first stage. But with this modification, the stability bound reduces to $CFL_{\max} \approx 2.6$ [24].

In order to achieve fast convergence rates it is recommended to choose the coefficients as [24]:

$$\alpha_1 = 0.25, \alpha_2 = 0.5 \quad \text{and} \ \alpha_3 = 0.55$$

However, this modification reduces the stability bound and maximum allowable CFL number becomes $CFL_{\max} \approx 2.0$ [24].

3.4 Discretization of Thin Layer Navier-Stokes Equations

The most general discretized form of the conservation law for a scalar quantity is given by equation (3.1). An equivalent form of this equation can be written for a system of conservation equations as:

$$\frac{\partial}{\partial t}\left(\vec{q}_{i,j,k}\Delta V_{i,j,k}\right) + \sum_{sides}\left(\vec{F}\cdot\vec{S}\right) = 0 \tag{3.28}$$

Where $\vec{q}$ is the column vector containing the conservative variables stored at the cell centers and $\vec{F}$ is the matrix containing the flux components. The above formulation can be further expanded on a structured grid as:

$$\frac{\partial}{\partial t}\left(\vec{q}_{i,j,k}\Delta V_{i,j,k}\right)+\vec{F}_{i+\frac{1}{2},j,k}\cdot\vec{S}_{i+\frac{1}{2},j,k}-\vec{F}_{i-\frac{1}{2},j,k}\cdot\vec{S}_{i-\frac{1}{2},j,k}+\vec{F}_{i,j+\frac{1}{2},k}\cdot\vec{S}_{i,j+\frac{1}{2},k}-\vec{F}_{i,j-\frac{1}{2},k}\cdot\vec{S}_{i,j-\frac{1}{2},k}+\vec{F}_{i,j,k+\frac{1}{2}}\cdot\vec{S}_{i,j,k+\frac{1}{2}}-\vec{F}_{i,j,k-\frac{1}{2}}\cdot\vec{S}_{i,j,k-\frac{1}{2}}=0 \qquad (3.29)$$

here, $\vec{S}_{i\pm\frac{1}{2},j,k}$, $\vec{S}_{i,j\pm\frac{1}{2},k}$ and $\vec{S}_{i,j,k\pm\frac{1}{2}}$ are the cell face area vectors that correspond to the constant ξ, η and ζ surfaces, respectively. One can write this equation in a different manner as:

$$\frac{\partial}{\partial t}\left(\vec{q}_{i,j,k}\Delta V_{i,j,k}\right)+\left(\vec{f}-\vec{f}_v\right)_{i+\frac{1}{2},j,k}-\left(\vec{f}-\vec{f}_v\right)_{i-\frac{1}{2},j,k}+\left(\vec{g}-\vec{g}_v\right)_{i,j+\frac{1}{2},k}-\left(\vec{g}-\vec{g}_v\right)_{i,j-\frac{1}{2},k}+\left(\vec{h}-\vec{h}_v\right)_{i,j,k+\frac{1}{2}}-\left(\vec{h}-\vec{h}_v\right)_{i,j,k-\frac{1}{2}}=0 \qquad (3.30)$$

where $\vec{f}$, $\vec{g}$ and $\vec{h}$ are the inviscid flux components while $\vec{f}_v$, $\vec{g}_v$ and $\vec{h}_v$ are the viscous flux components through the cell faces in the ξ,η and ζ directions, respectively.

It is possible to obtain the most general description of the flow of a newtonian fluid in thermodynamic equilibrium by the solution of the full Navier-Stokes

equations [16]. Furthermore, there is no need to add artificial dissipation terms explicitly to the equations in order to capture the shock waves since the system has physical dissipation terms, if it is properly discretized. But this way of solution requires very fine mesh spacing especially near the solid boundaries, not only in the normal direction but also in the stream wise and span wise directions. Since the number of grid points will increase with a finer mesh, this type of solution increases the computational cost severely [22]. Instead of solving the complete set of Navier - Stokes equations, one can prefer to solve a subset of whole system, which is called thin-layer Navier - Stokes equations. This subset can be obtained by neglecting the viscosity and heat conduction terms in the streamwise and spanwise directions and gives accurate results for high Reynolds number flows where the viscous effects are confined into a thin shear layer. With this approximation the system can be written as:

$$\frac{\partial}{\partial t}\left(\vec{q}_{i,j,k}\Delta V_{i,j,k}\right)+\vec{f}_{i+\frac{1}{2},j,k}-\vec{f}_{i-\frac{1}{2},j,k}$$
$$+\left(\vec{g}-\vec{g}_v\right)_{i,j+\frac{1}{2},k}-\left(\vec{g}-\vec{g}_v\right)_{i,j-\frac{1}{2},k}$$
$$+\vec{h}_{i,j,k+\frac{1}{2}}-\vec{h}_{i,j,k-\frac{1}{2}}=0 \qquad (3.31)$$

The following step of the formulation is to define the flux components $\vec{f}$, $\vec{g}$, $\vec{g}_v$ and $\vec{h}$. Firstly, let ξ be the stream wise direction, η be the normal direction, and ζ be the span wise direction. Afterwards, Let ξ_x, ξ_y, ξ_z denotes the x, y, z components of the constant ξ surface area vector; η_x, η_y, η_z be the x, y, z components of the constant η surface area vector; and $\zeta_x, \zeta_y, \zeta_z$ be the x, y, z

components of the constant ζ surface area vector, respectively. Now, the flux components can be defined as [14]:

$$\vec{f} = \begin{bmatrix} \rho U \\ \rho u U + p\xi_x \\ \rho v U + p\xi_y \\ \rho w U + p\xi_z \\ (E+p)U \end{bmatrix} \tag{3.32}$$

$$\vec{g} = \begin{bmatrix} \rho V \\ \rho u V + p\eta_x \\ \rho v V + p\eta_y \\ \rho w V + p\eta_z \\ (E+p)V \end{bmatrix} \tag{3.33}$$

$$\vec{h} = \begin{bmatrix} \rho W \\ \rho u W + p\zeta_x \\ \rho v W + p\zeta_y \\ \rho w W + p\zeta_z \\ (E+p)W \end{bmatrix} \tag{3.34}$$

$$\vec{g}_v = \frac{1}{\Delta V} \begin{bmatrix} 0 \\ \mu m_1 \dfrac{\partial u}{\partial \eta} + \dfrac{\mu}{3} m_2 \eta_x \\ \mu m_1 \dfrac{\partial v}{\partial \eta} + \dfrac{\mu}{3} m_2 \eta_y \\ \mu m_1 \dfrac{\partial w}{\partial \eta} + \dfrac{\mu}{3} m_2 \eta_z \\ m_1 \left[\dfrac{\mu}{2} \dfrac{\partial}{\partial \eta} \left(u^2 + v^2 + w^2 \right) + k \dfrac{\partial T}{\partial \eta} + \dfrac{\mu}{3} V m_2 \right] \end{bmatrix} \tag{3.35}$$

where

$$U = u\xi_x + v\xi_y + w\xi_z \tag{3.36}$$

$$V = u\eta_x + v\eta_y + w\eta_z \tag{3.37}$$

$$W = u\zeta_x + v\zeta_y + w\zeta_z \tag{3.38}$$

and

$$m_1 = \eta_x{}^2 + \eta_y{}^2 + \eta_z{}^2 \tag{3.39}$$

$$m_2 = \eta_x \frac{\partial u}{\partial \eta} + \eta_y \frac{\partial v}{\partial \eta} + \eta_z \frac{\partial w}{\partial \eta} \tag{3.40}$$

The eigenvalues appear in the equations are given by the relations:

$$\lambda_\xi = |U| + c\sqrt{\xi_x{}^2 + \xi_y{}^2 + \xi_z{}^2} \tag{3.41}$$

$$\lambda_\eta = |V| + c\sqrt{\eta_x{}^2 + \eta_y{}^2 + \eta_z{}^2} \tag{3.42}$$

$$\lambda_\zeta = |W| + c\sqrt{\zeta_x{}^2 + \zeta_y{}^2 + \zeta_z{}^2} \tag{3.43}$$

where c is the local speed of the sound, and for a perfect gas it is given by:

$$c = \sqrt{\frac{\gamma p}{\rho}} \tag{3.44}$$

The time integration of the thin-layer Navier - Stokes equations requires an appropriate time step Δt, limited by the stability bound of the scheme. The local time step is computed through a stability analysis for the inviscid and viscous flux operators seperately[23].

From the analysis for the inviscid part, the time step can be written as :

$$(\Delta t)_{inv} = \frac{1}{\lambda_\xi + \lambda_\eta + \lambda_\zeta} \tag{3.45}$$

and from the stability analysis of the viscous part, the time step is determined as :

$$(\Delta t)_{vis} = \frac{\Delta V \rho}{\mu(\eta_\xi + \eta_\eta + \eta_\zeta)} \tag{3.46}$$

The local time step that will be used during the computations can be obtained by taking the minimum of these values and multiplying it by the CFL number:

$$\Delta t = CFL \times \min[(\Delta t)_{inv}, (\Delta t)_{vis}] \tag{3.47}$$

An additional condition called entropy condition has, therefore, to be added in order to exclude these non-physical solutions. Entropy condition is satisfied by adding appropriate amount of artificial dissipation terms to the discretized equations.

The metric terms used in the above formulations are:

$$\xi_x = J(y_\eta z_\zeta - y_\zeta z_\eta) \tag{3.48}$$

$$\xi_y = J(x_\zeta z_\eta - x_\eta z_\zeta) \tag{3.49}$$

$$\xi_z = J(x_\eta y_\zeta - x_\zeta y_\eta) \tag{3.50}$$

$$\eta_x = J(y_\zeta z_\xi - y_\xi z_\zeta) \tag{3.51}$$

$$\eta_y = J(x_\xi z_\zeta - x_\zeta z_\xi) \tag{3.52}$$

$$\eta_z = J(x_\zeta y_\xi - x_\xi y_\zeta) \tag{3.53}$$

$$\zeta_x = J(y_\xi z_\eta - y_\eta z_\xi) \tag{3.54}$$

$$\zeta_y = J(x_\eta z_\xi - x_\xi z_\eta) \tag{3.55}$$

$$\zeta_z = J(x_\xi y_\eta - x_\eta y_\xi) \tag{3.56}$$

CHAPTER 4

MESH TOPOLOGY AND PARALLELIZATION

4.1 Mesh Topology

Accuracy of the results are highly dependent on the structure of the mesh in computational fluid dynamics analysis. The quality of the mesh, especially on the solid boundaries and in the regions which will capture boundary layer and tip vortex formation, is particularly important. According to the meshes used in inviscid flow solutions, viscous meshes have to provide higher number of grid points in these regions. Besides, orthogonality has to be enforced on the solid body surfaces.

The fine mesh used in the computations of rotary wing configurations was constructed using the CFD – GEOM software [25]. The computational mesh structure is H – H type with 121 × 96 × 65 points where surface mesh has 41 grid points in streamwise direction and 41 grid points in spanwise direction. The blades have rectangular planforms with round tip caps and made up of NACA 0012 airfoil sections. Since both the geometry and the flowfield are asymmetric, computations were performed for a single blade by applying periodic boundary conditions at the inflow and outflow boundaries of a single blade sector.

Fig. (4.1) shows the outer boundaries of the computational mesh. Sectional views of the computational mesh are displayed in Fig. (4.2) and Fig. (4.3). In Fig. (4.4) and

Fig. (4.5) , surface mesh of the UH – 1 rotor blade is displayed. In Fig. (4.6), surface mesh of the tip region is displayed.

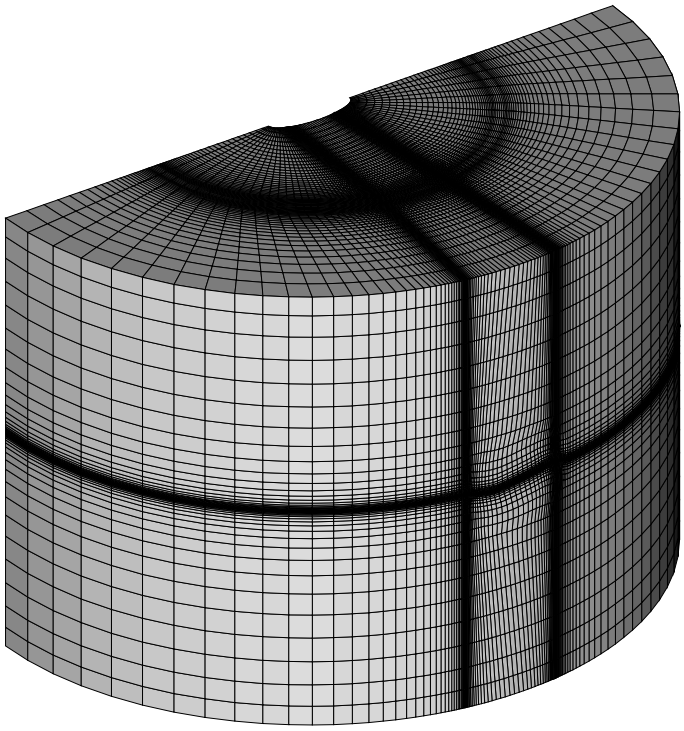

Figure 4.1: Outer boundaries of the computational mesh.

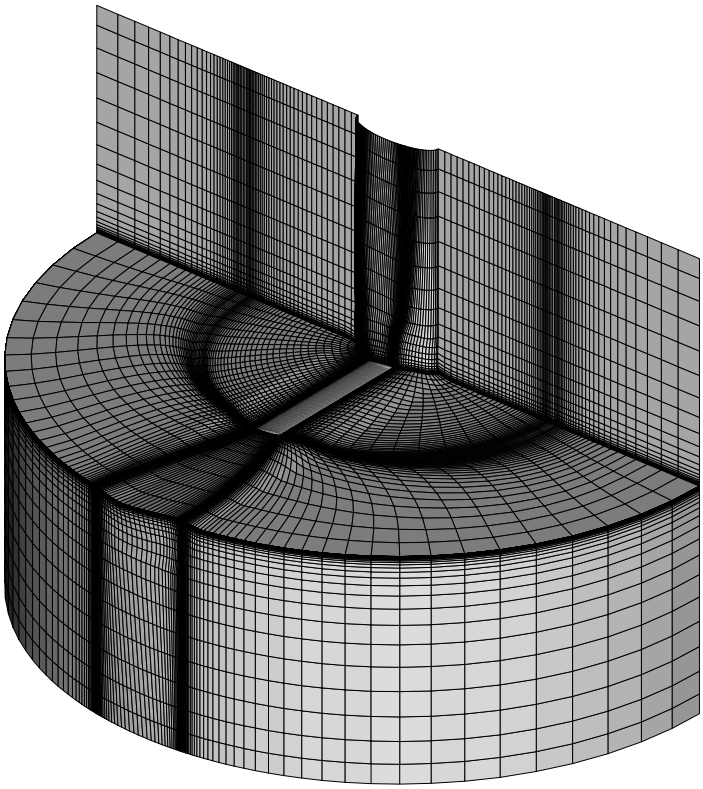

Figure 4.2: Sectional view of the computational mesh.

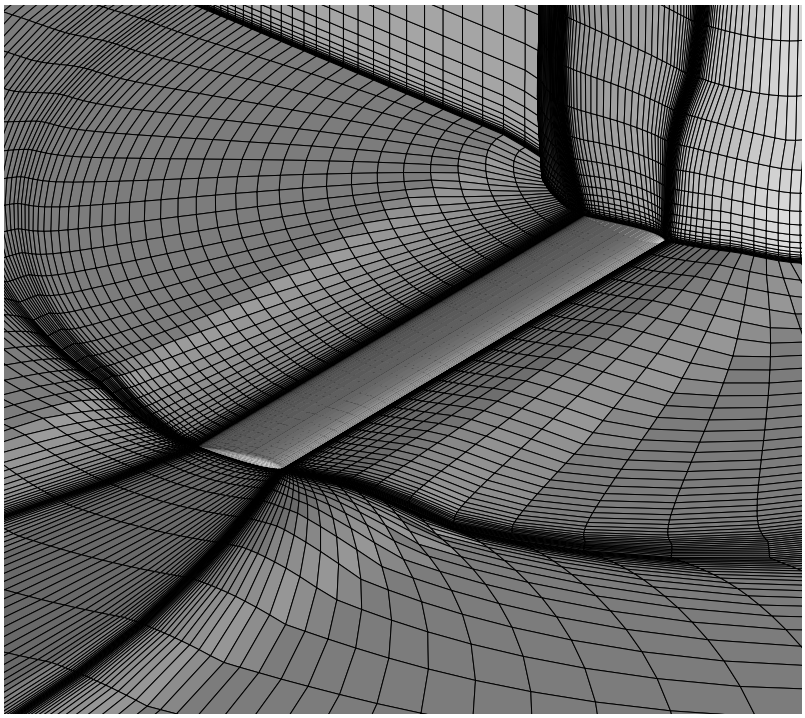

Figure 4.3: Sectional view of the computational mesh.

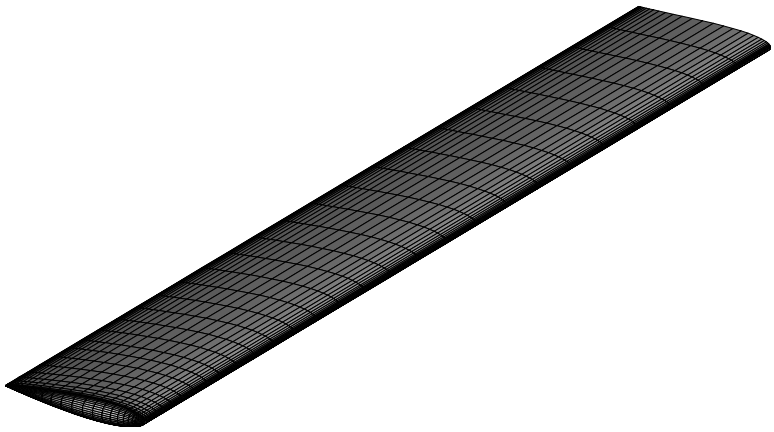

Figure 4.4: Surface mesh of UH – 1 Helicopter blade.

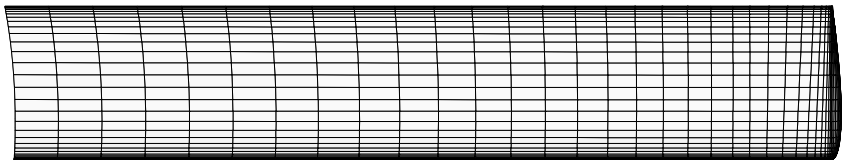

Figure 4.5: Surface mesh of UH – 1 Helicopter blade.

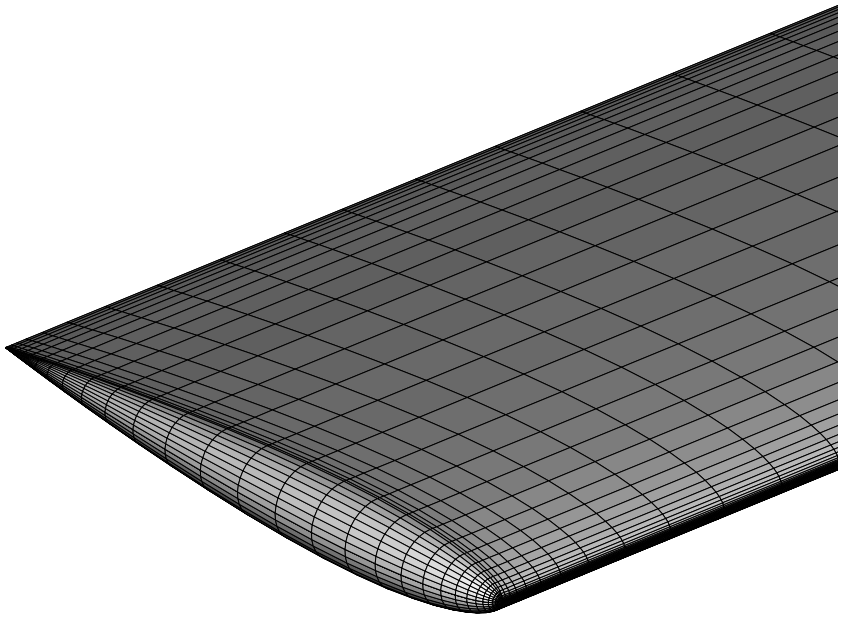

Figure 4.6: Surface mesh of UH – 1 Helicopter blade tip.

4.2 Domain Decomposition Technique

In this study, the governing equations are aimed to be solved numerically using a cell - centered finite volume scheme and a four step Runge-Kutta time-integration technique. A parallel fortran90 code has been developed by modifying an existing code. The code consists of a main program and 33 subroutines. The pre-existing code could handle different grid topologies and such as H –H , C –H , C – C and C - O and be utilized to obtain inviscid solutions around rotary geometries and viscous solutions around fixed wings. The first modification was made by applying non – slip conditions at solid wall boundaries and adding viscous terms to obtain laminar results. Later on, a turbulence subroutine was added to include turbulent viscosity to obtain turbulent results. Baldwin – Lomax turbulence model was utilized to model turbulence. The result was a parallel code which obtains laminar and turbulent solutions for rotary wing configurations. Since the flow around hovering helicopter blades is periodic in the span-wise direction, it was enough to solve the flow around

just one blade. On symmetry surfaces, periodic boundary conditions were applied to provide flow continuity.

Three-dimensional CFD applications demand considerable amount of computational memory and time. The capacity of serial and vector computers are not adequate for this kind of massive applications. In order to overcome this significant problem, one can take the advantage of a technique called parallel processing, in which computational domain is decomposed in to smaller sub – domains in i, j and k directions, and the job is distributed over more than one processors to be performed simultaneously. Data transfer is the most essential progress in parallel processing. It is necessary for the continuity of the flow variables across the boundaries of sub domains. Data transfer across the boundaries prevents them to act as sources by which the solution is effected drastically in a manner.

The domain decomposition is achieved by dividing the flow domain into computational sub – domains in desired directions in accordance with the processors used. A domain decomposition algorithm is used for this purpose. The decomposition is performed to provide a balanced work load on each processor with N_i sub-domains in the i-direction, N_j sub-domains in the j-direction and N_k sub-domains in the k-direction giving a total number of sub-domains :

$$N_{total} = N_i \times N_j \times N_k \tag{4.1}$$

To reduce processor load used for communication at sub-domain boundaries, ratio of the number of surface cells to the number of volume cells is kept minimum. The computational and the job for each sub-domain assigned to the respective process according to the following scheme :

```
do taskid  = 0,  N – 1
   task_k  = taskid  / ( Ni * Nj ) + 1
   task_j  = ( taskid – ( task_k – 1) * Ni * Nj ) / Ni + 1
   task_i  = taskid – ( task_j – 1) * Ni – (task_k – 1 ) * Ni * Nj + 1
```

Here, taskid indicates the number of a process which is responsible for the $(task_i, task_j, task_k)$ sub-domain work.. Heterogeneouty of the processors employed is an important constraint. Since the capacity of the slowest processor determines the overall performance of the code, one has to take care of the available memory and speed while distributing the processes over the computers. In this study, computational mesh is divided in to eight sub-domains; 2 in the i-direction, 2 in the j-direction and 2 in the k-direction.

4.3 Data Transfer Process

The essential element of the parallel processing is data communication among the processes. The continuity of the flow variables across the inner sub-domains can be provided by following a proper data transfer communication pattern. The finite volume method requires one auxiliary cell to transmit the fluxes through the cell faces. But the artificial dissipation terms requires two adjacent cells. In order to establish a favourable data transformation between the sub-domains, two ghost cells are used. Data are then communicated between two neighbouring processes across the inner sub-domain boundaries. A schematic of the data transfer process for a 8 – process decomposition can be seen in Fig. (4.7).

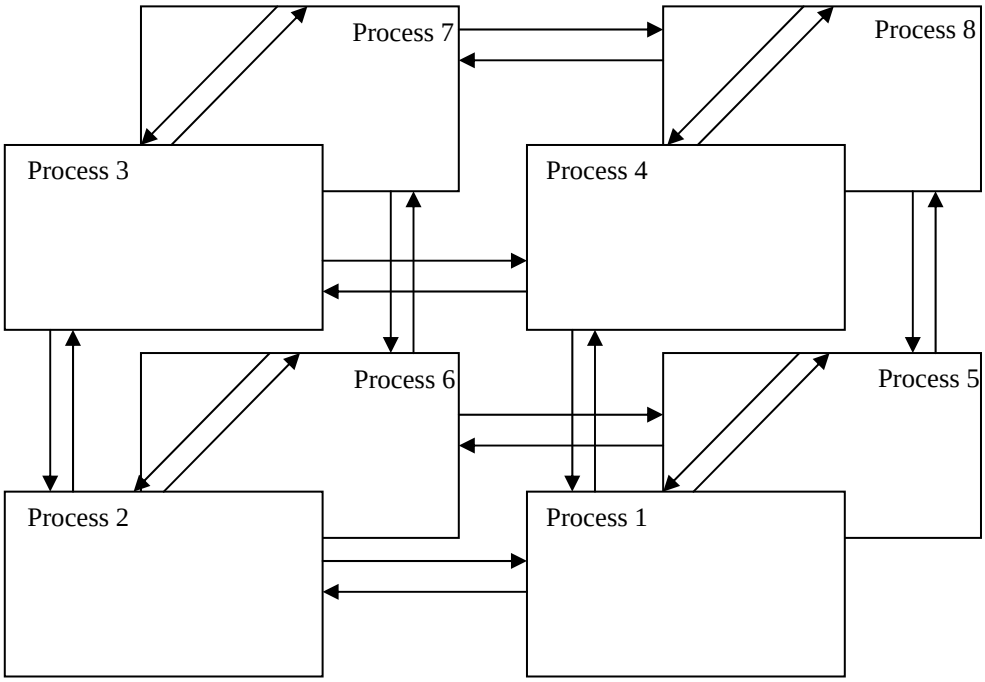

Figure 4.7: Schematic of data transfer between processes.

Data transfer is performed using library calls from the MPI (Message Passing Interface) communication software. Specifically blocking send and receive calls are used for this purpose. The expression *blocking* states that the send call blocks until the send buffer can be reclaimed, and similarly the receive function blocks until the receive buffer actually contains the contents of the message. MPI also provides *non-blocking* send and receive operations that allow possible overlap of message transmittals with one – another. In this thesis blocking send and receive operations are preferred in order to ensure the continuity of flow variables across the inner boundaries and in order to avoid the wrong update of boundary conditions.

One of the processes is chosen as master process. It is responsible for carrying out additional work compared to the others. The master reads input information and broadcasts it to each process, performs the domain decomposition. Each process reads assigned part of the computational grid and performs computations for each

iteration step. Information about the residual history, CPU time and restart parameters are send from each process to the master and it prints all this information. Outputs such as solution files or restart files are written by each processes. The details of some MPI commands can be found in appendix.

CHAPTER 5

RESULTS AND DISCUSSION

In this chapter, results of turbulent runs for flat plate and 3 –D results of laminar and turbulent runs for the UH – 1 blade are presented. The code has been run on the parallel cluster of computers housed in the Aerospace Engineering department at the Middle East Technical University. Each computer has dual 700 MHz processors with 512 MB Ram. Reasonable agreement with experimental data has been achieved. The computed results, convergence histories and comparisons with experimental data and inviscid results are available in the following sections of the chapter.

5.1 Flat Plate

One of the basic test cases for turbulent flow is the unsteady turbulent flow over a flat plate. For this flow, no pressure gradient exists and so a constant boundary layer edge velocity occurs. The investigation of this flow also gives rise to similar solutions. If we look in a bit more detail at the structure of a turbulent boundary layer , we see that there are more or less three distinct regions within the boundary layer [10]:

1. *Outer Layer* : relatively sensitive to the external flow properties.
2. *Inner Layer* : turbulent mixing is the dominant influence.
3. *Laminar Sublayer* : closest to the surface, in which, because the no-slip

condition forces the fluctuating velocity components to zero at the wall, the turbulent stresses are negligible compared to $\partial u/\partial y$.

The code was run to obtain the turbulent flowfield solution around a flat plate. Obtained results for different streamwise stations are compared with the results of the analytical solution.

In laminar sublayer, friction velocity can be given as :[10]

$$u^+ = y^+$$

This is valid for a region from $y^+ \approx 1$ to $y^+ \approx 30$. The overlap region spans $30 \approx y^+ \approx 300$ and has the following equation:[10]

$$u^+ = \frac{1}{0.41}\ln\left(y^+\right) + 5$$

where the relations

$$u^+ = \frac{u}{u_\tau}$$

$$y^+ = \frac{y u_\tau}{\upsilon}$$

$$u_\tau = \sqrt{\tau_w/\rho}$$

$$\tau_w = \mu \frac{\partial u}{\partial y}\Bigg)_{y=0}$$

are used to obtain computational results. Fig. (5.1) displays the comparison of the results with the analitical solution at different streamwise locations.

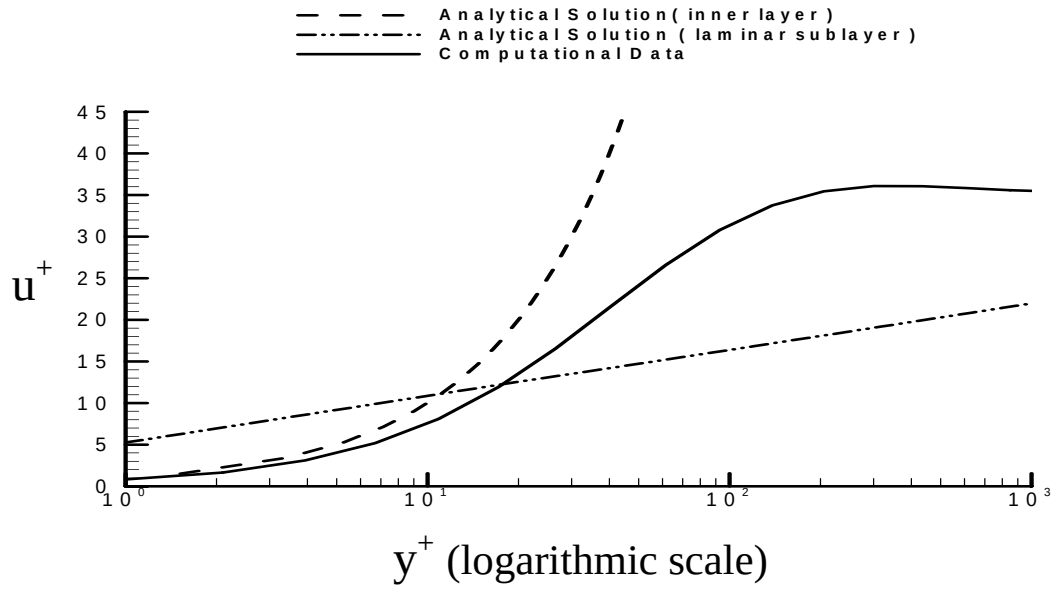

Figure 5.1: Wall law comparison of the numerical solution with the analytical solution for a turbulent flat plate.

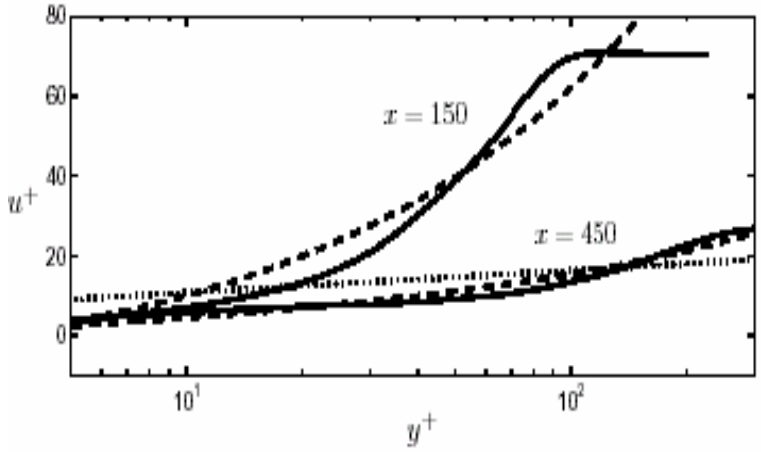

Figure 5.2: Wall law comparison of the numerical solution with the analytical solution for a flat plate obtained by a DNS; solid lines - DNS, dashed lines - analytic solution, dotted line - Analytic solution ($u^+ = \frac{1}{0.41}\ln(y^+) + 5.1$) [29].

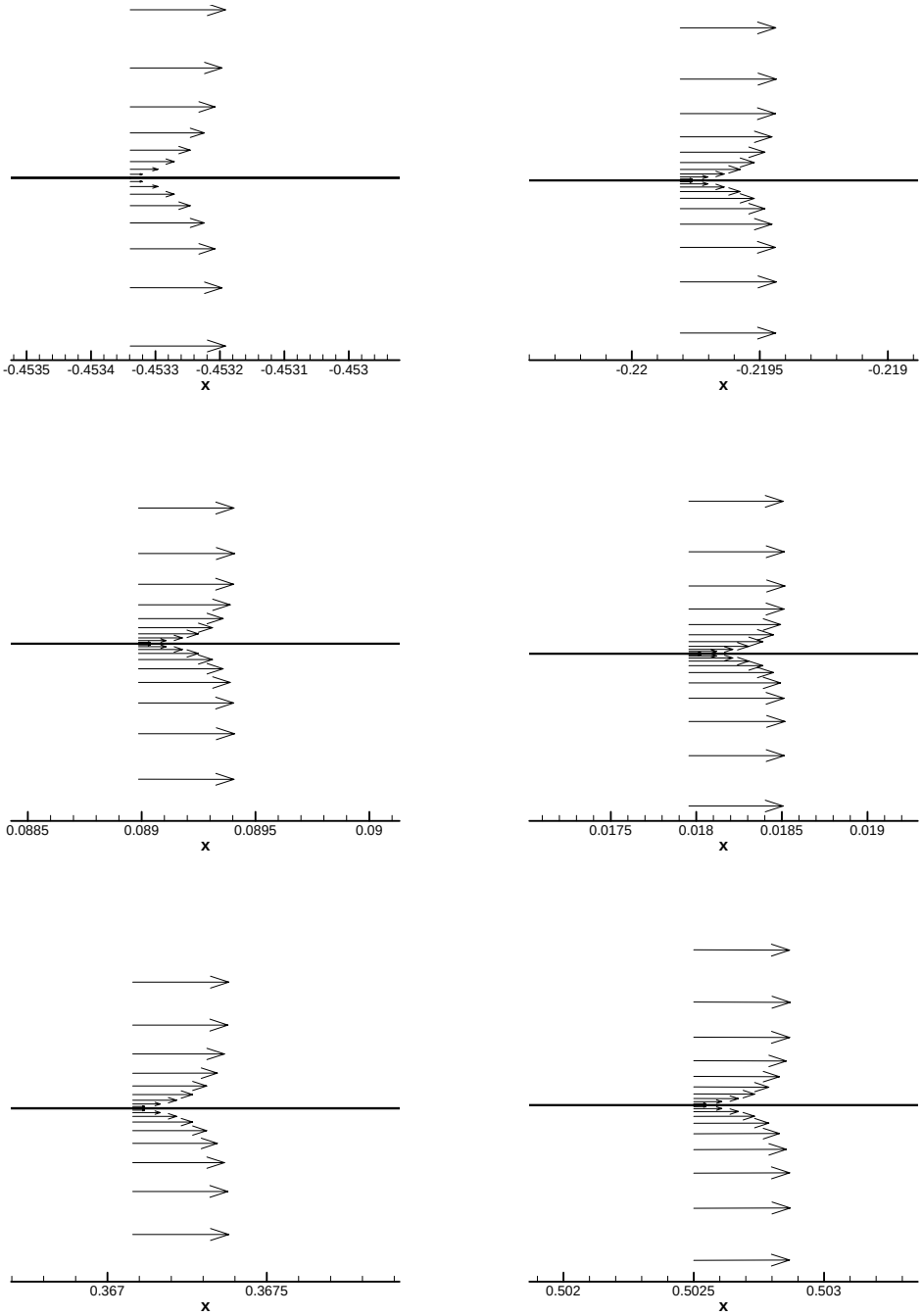

Fig: 5.3: Boundary Layer velocity profiles on flat plate, laminar case, $M_0 = 0.3$, $\mathrm{Re} = 3.96 \times 10^6$

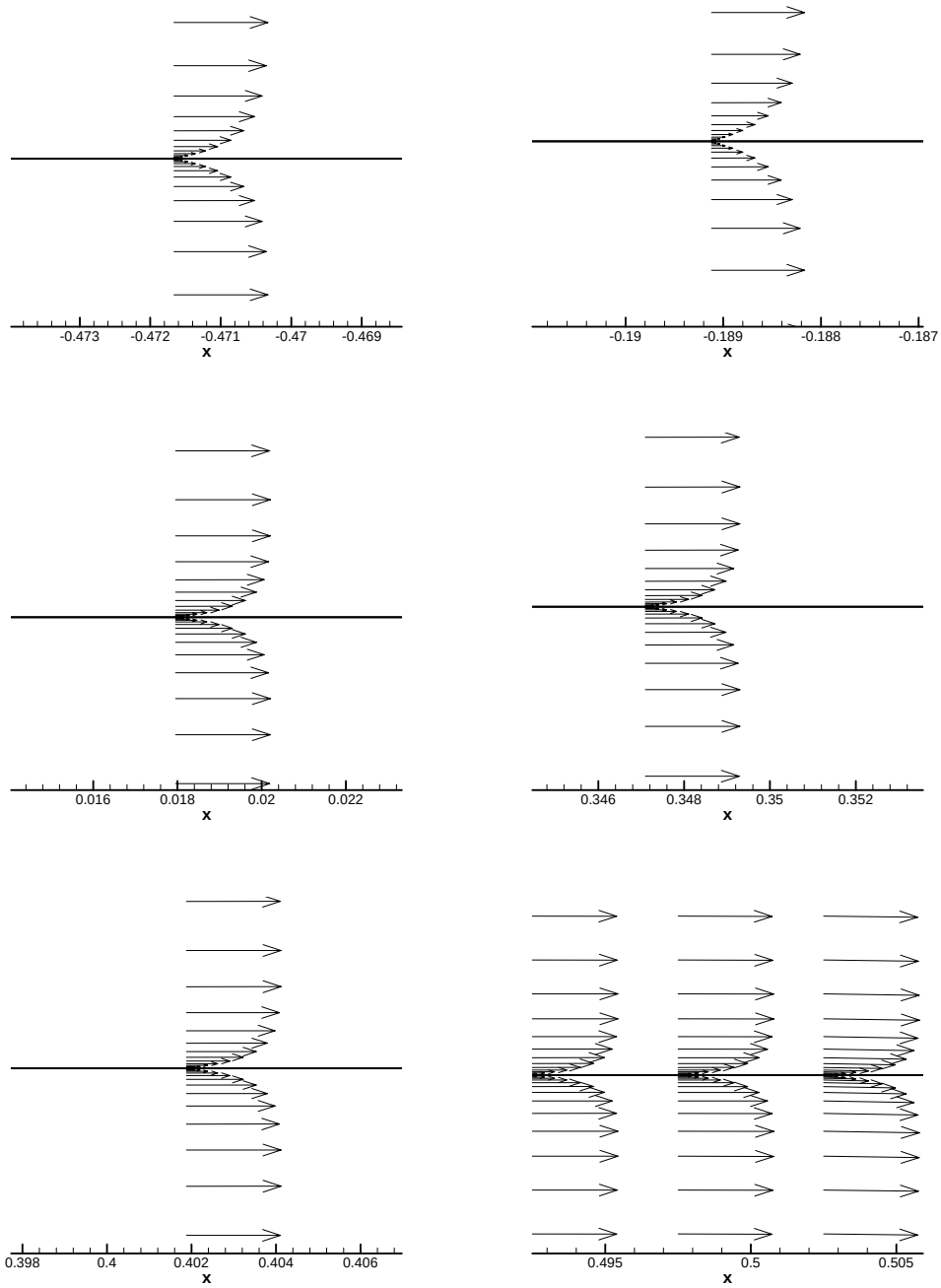

Fig: 5.4: Boundary Layer velocity profiles on flat plate, turbulent case, $M_0 = 0.3$, $\mathrm{Re} = 3.96 \times 10^6$

5.2 Hovering UH – 1 rotor

This section of the thesis involves the viscous rotary wing solutions around a two – bladed UH – 1 helicopter main rotor. Since both the geometry and the flowfield are asymmetric, computations were performed for a single blade by applying periodic boundary conditions at the inflow and outflow boundaries. Solutions were obtained both laminar and turbulent cases.

In many of the studies, wake modelling techniques were used to model the vortical wake. In this thesis, no wake modelling technique was followed and the vortical wake and its influences were captured as part of of the overall flowfield solution. For turbulent cases, algebraic eddy viscosity, zero equation Baldwin – Lomax turbulance model was employed to model the turbulance.

Runs were executed for different flow regimes such as non – lifting hovering rotor at subsonic tip speed, lifting hovering rotor at subsonic tip speed, and lifting hovering rotor at transonic tip speed for both laminar and turbulent cases. For all solutions, freestream Mach number was taken as zero (hover). Eight processors were employed for the solutions. Laminar and turbulent results are compared with experimental data[18] and Euler results[27].

5.2.1 $\alpha_0 = 0^o$, $M_{tip} = 0.52$ Case

In non – lifting cases, solutions were obtained by setting tip Mach number to 0.52 and the collective pitch angle to 0°. Since a non – lifting blade does not create a tip vortex, BVI is not investigated for these cases. The residual histories collected in one of the eight of the solutions are displayed in Fig. (5.5). Good convergence rates were obtained for both laminar and turbulent cases. $K^{/2)} = 0.5$, $K^{(4)} = 0.5$, $CFL = 1$ and $\alpha = 0.01$. Actually, a significant difference between the convergence rates of laminar and turbulent cases was expected because of the included calculations for the turbulance and additional turbulent viscosity. But,

however, convergence histories of laminar and turbulent runs are quite similar to each other.

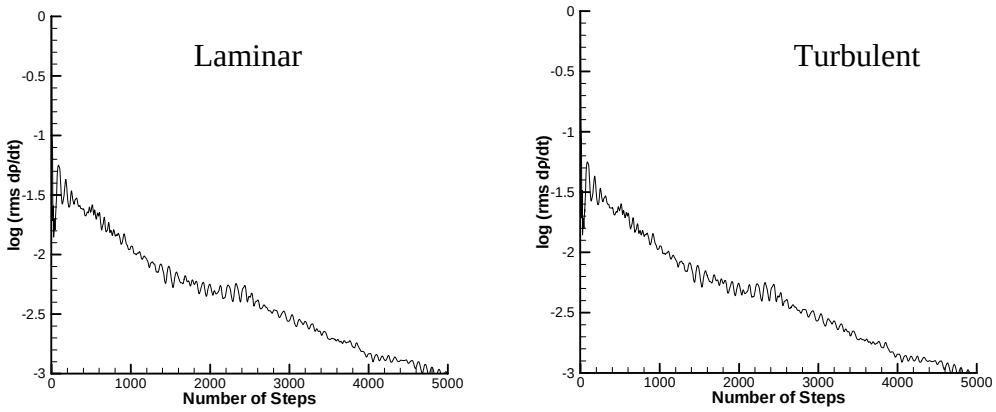

Figure 5.5: Residual histories for laminar and turbulent flow solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$, for an outer block of the computational domain.

.

An important aerodynamic force during low – speed subsonic flight is the shear force caused by viscous airflow over the surfaces of the body. This shear force is referred to as the skin – friction force or skin – friction drag and depends strongly on the Reynolds number, surface roughness and pressure gradients.[6] In addition to the pressure forces that act everywhere perpendicular to (normal to) a body in moving air, viscous forces are also present. These viscous forces modify the lift and create an important part of drag. Within the boundary layer, where the velocity is changing from zero to a constant value, there are relative velocities between the layers and an internal friction between them. This internal friction extends to the surface of the body. The cumulative effect of all these friction forces is to produce drag on the body. This drag is referred to as skin –friction drag.[5]

Figures (5.6) through (5.8) show the non – dimensional surface pressure contours, while Fig. (5.9) and Fig. (5.10) show the skin – friction coefficient contours for the non – lifting laminar and turbulent cases.

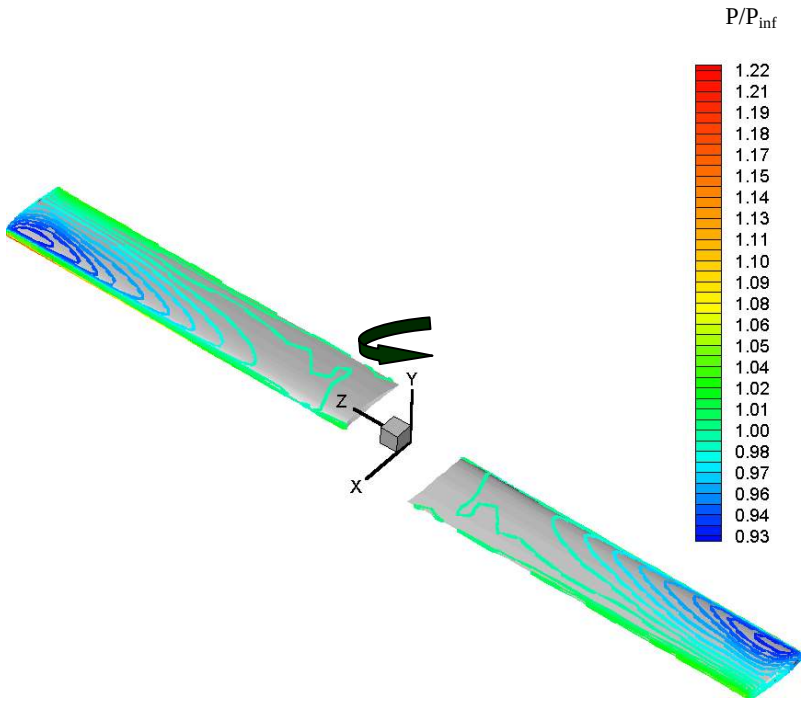

Figure 5.6: Surface pressure contours for laminar solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$.

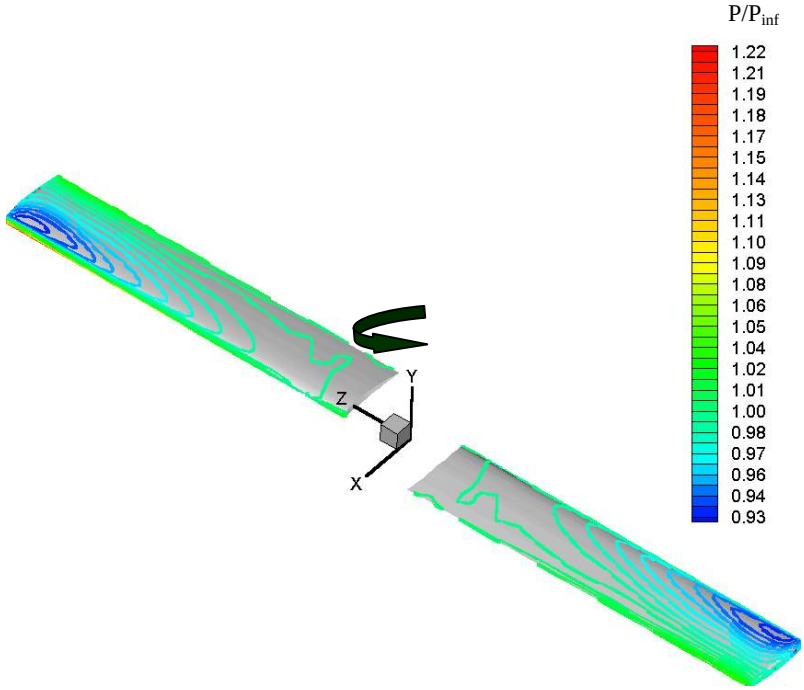

Figure 5.7: Surface pressure contours for turbulent solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$.

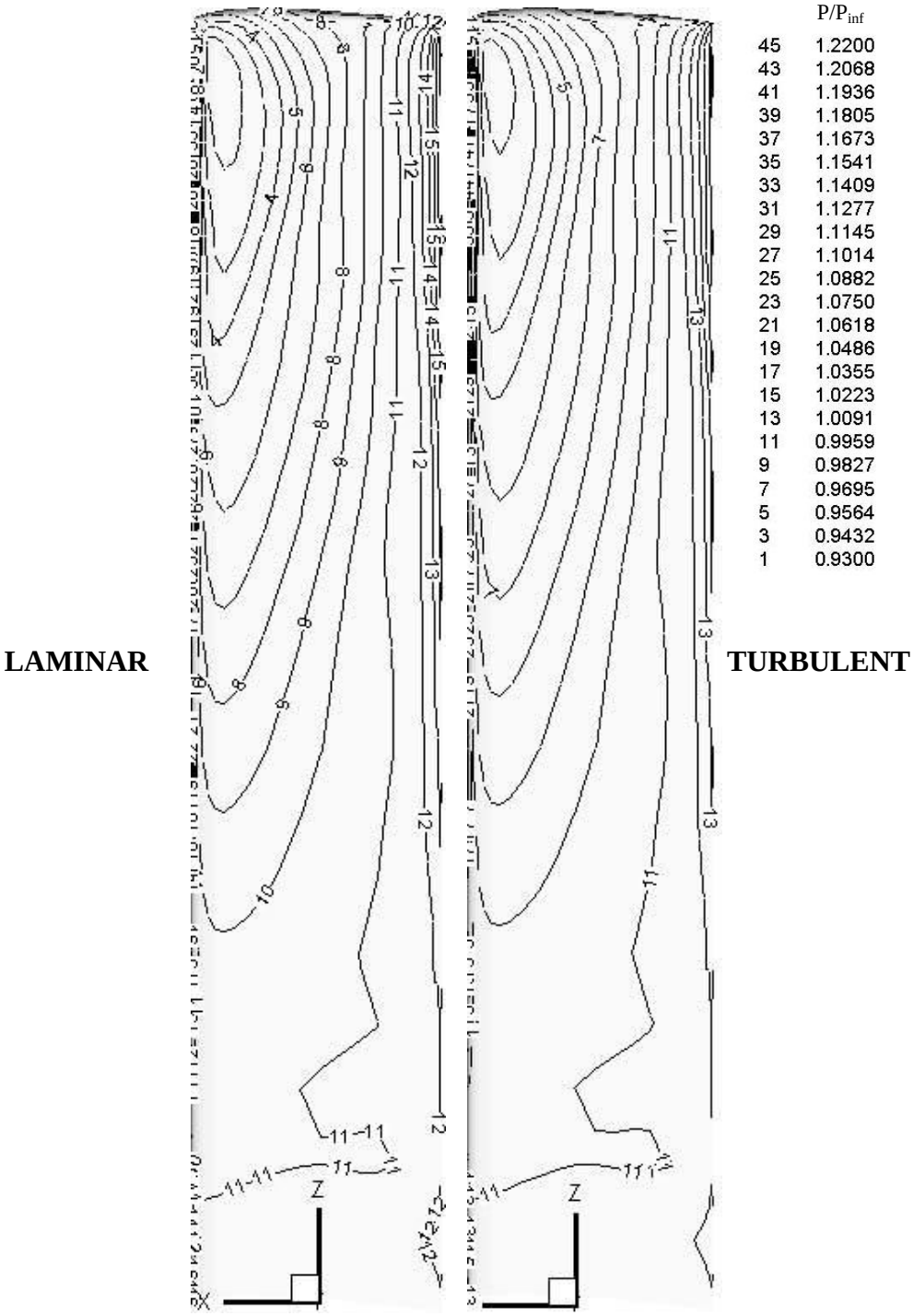

Figure 5.8: Non – dimensional surface pressure contours for laminar and turbulent solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$.

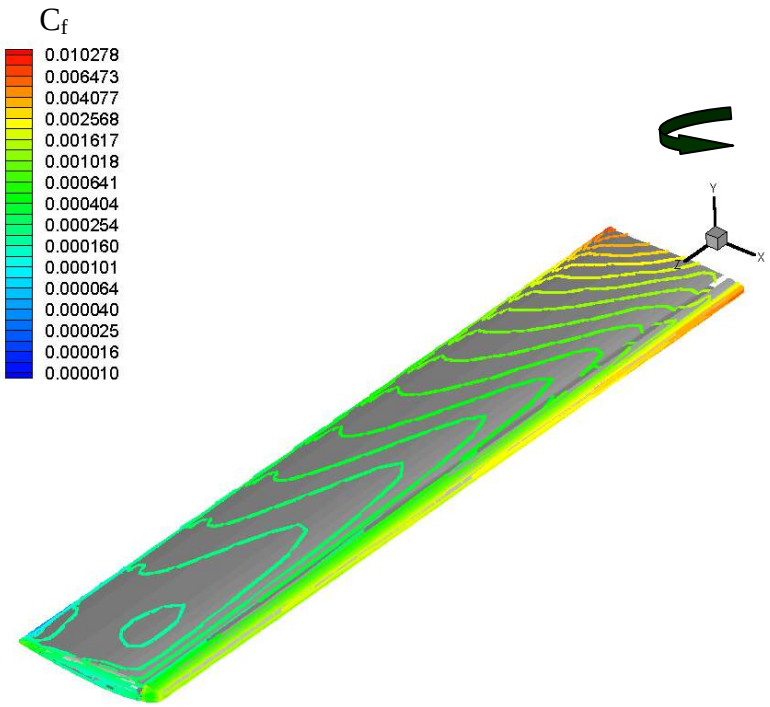

Figure 5.9: Surface C_f contours around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$, laminar case.

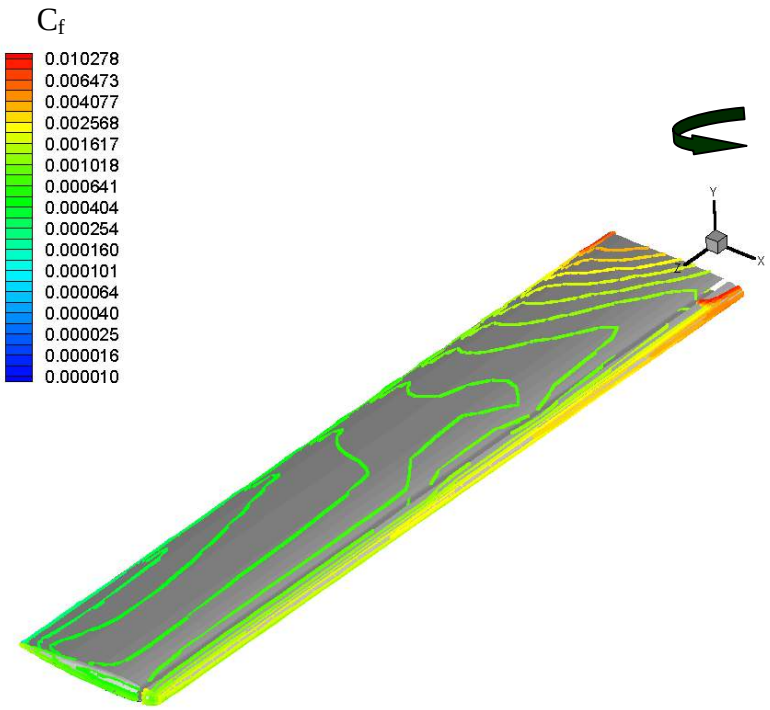

Figure 5.10: Surface C_f contours around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$, turbulent case.

However, pressure contours for laminar and turbulent cases are quite similar. Fig. (5.8) indicates a smaller accelerated pocket near tip for turbulent case. On the other hand, as it can be observed on Fig. (5.9) and Fig. (5.10), there is a obvious difference between the skin friction coefficient values of the laminar and turbulent cases. The effect of the additional turbulent viscosity can be seen easily. Turbulent C_f values are quite higher than those of laminar case, as expected. Besides, while C_f exhibits a smooth decrease in streamwise direction for turbulent case, there occurs a minimum C_f pocket for laminar case which can be seen on Fig. (5.9) and Fig. (5.10).

In Fig. (5.11) and fig. (5.12), non – dimensional pressure contours at different spanwise locations for laminar and turbulent cases , are displayed. As one goes to the hub, the effect of decreasing angular velocity can be seen as a diminish in the pressure contours. Fig. (5.13) presents the 2–D non-dimensional pressure contours at different spanwise stations for nonlifting laminar and turbulent cases. There is not an important difference that can be observed easily in the 2-D pressure contours. The difference between the pressure values of laminar and turbulent cases is observed mostly on the surface pressure values. Fig. (5.14) displays the C_p distributions at various stations and comparisons with Euler solution and experimental data. It is clear that the results for laminar and turbulent cases are in fairly good agreement with experimental data, proving that the added viscous terms and the viscous mesh worked very well. The computational results are quite similar to the experimental values. Fig. (5.15) displays C_f distributions at different spanwise locations for laminar and turbulent cases. As expected, the skin friction coefficient is highest near leading edge. In Fig. (5.16) and Fig. (5.17), velocity profiles on UH-1 blade for nonlifting laminar and nonlifting turbulent cases are given. Normally, since the Re number is quite high a flow separation in the laminar case would be expected but this is not observed here because of 3-D effects caused by the rotation of the blade.

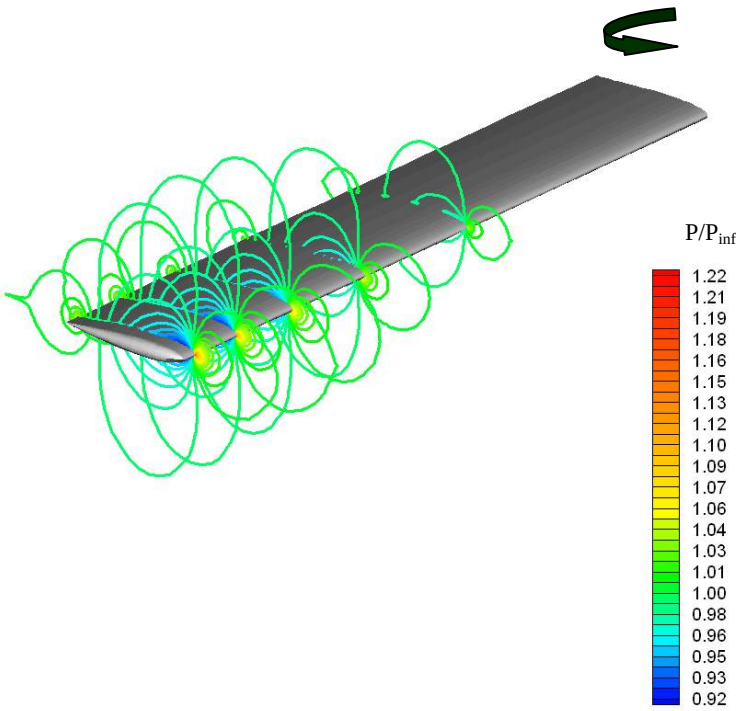

Figure 5.11: Pressure contours at different spanwise locations for laminar flow solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$, $z/R = 0.5,0.68,0.8,0.89,0.96$.

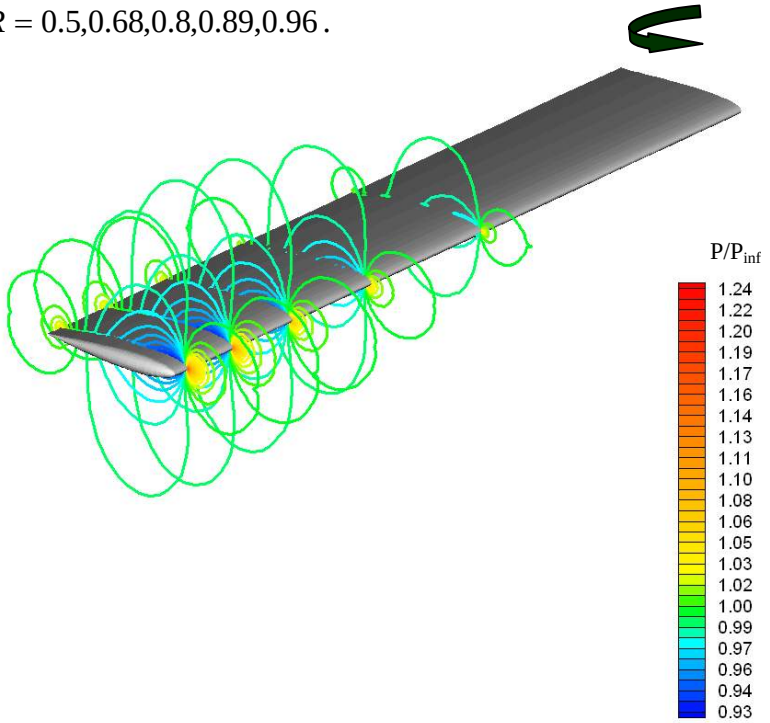

Figure 5.12: Pressure contours at different spanwise locations for turbulent flow solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$, $z/R = 0.5,0.68,0.8,0.89,0.96$.

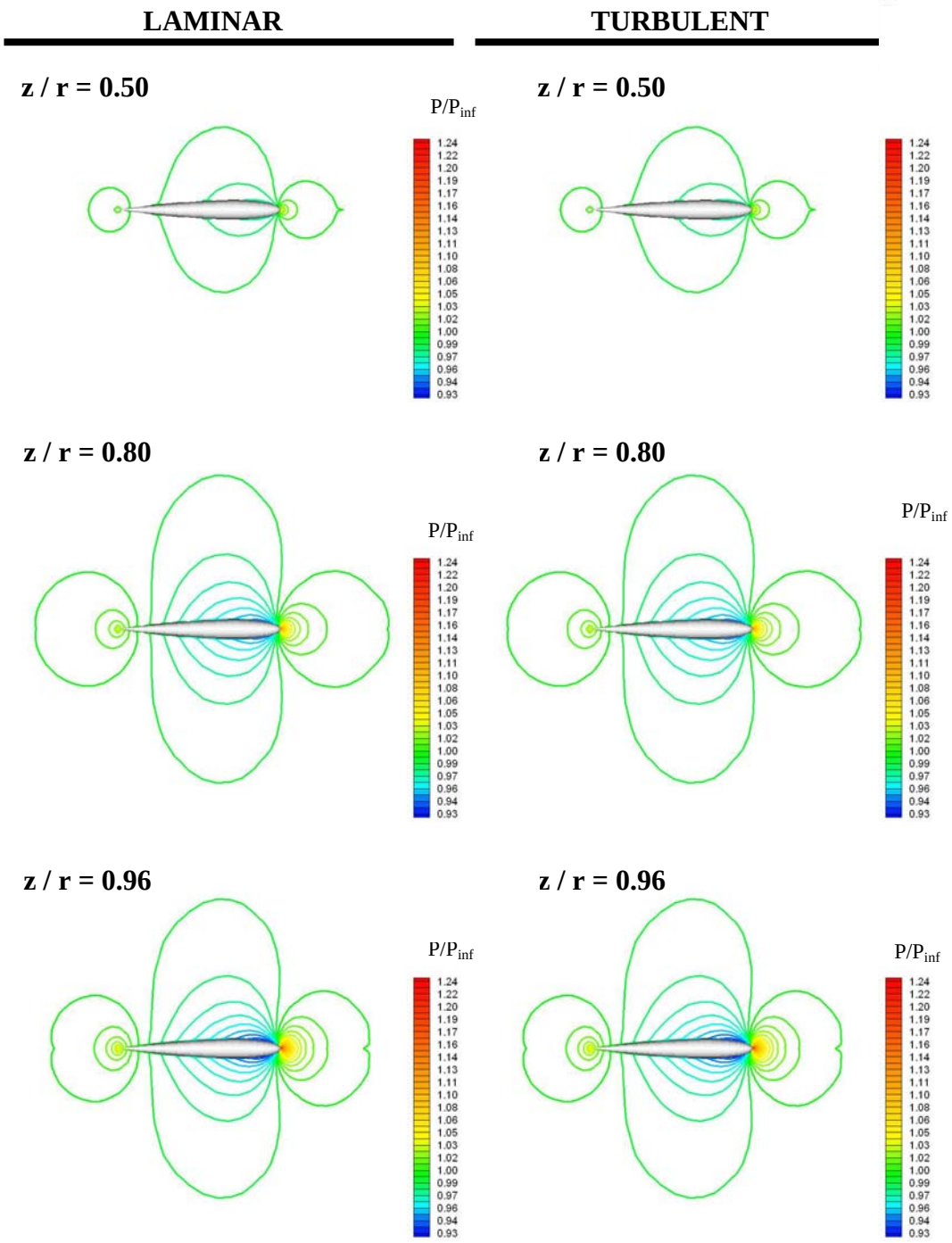

Figure 5.13: Pressure contours at different spanwise locations for laminar and turbulent flow solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$, $z/R = 0.5, 0.8, 0.96$.

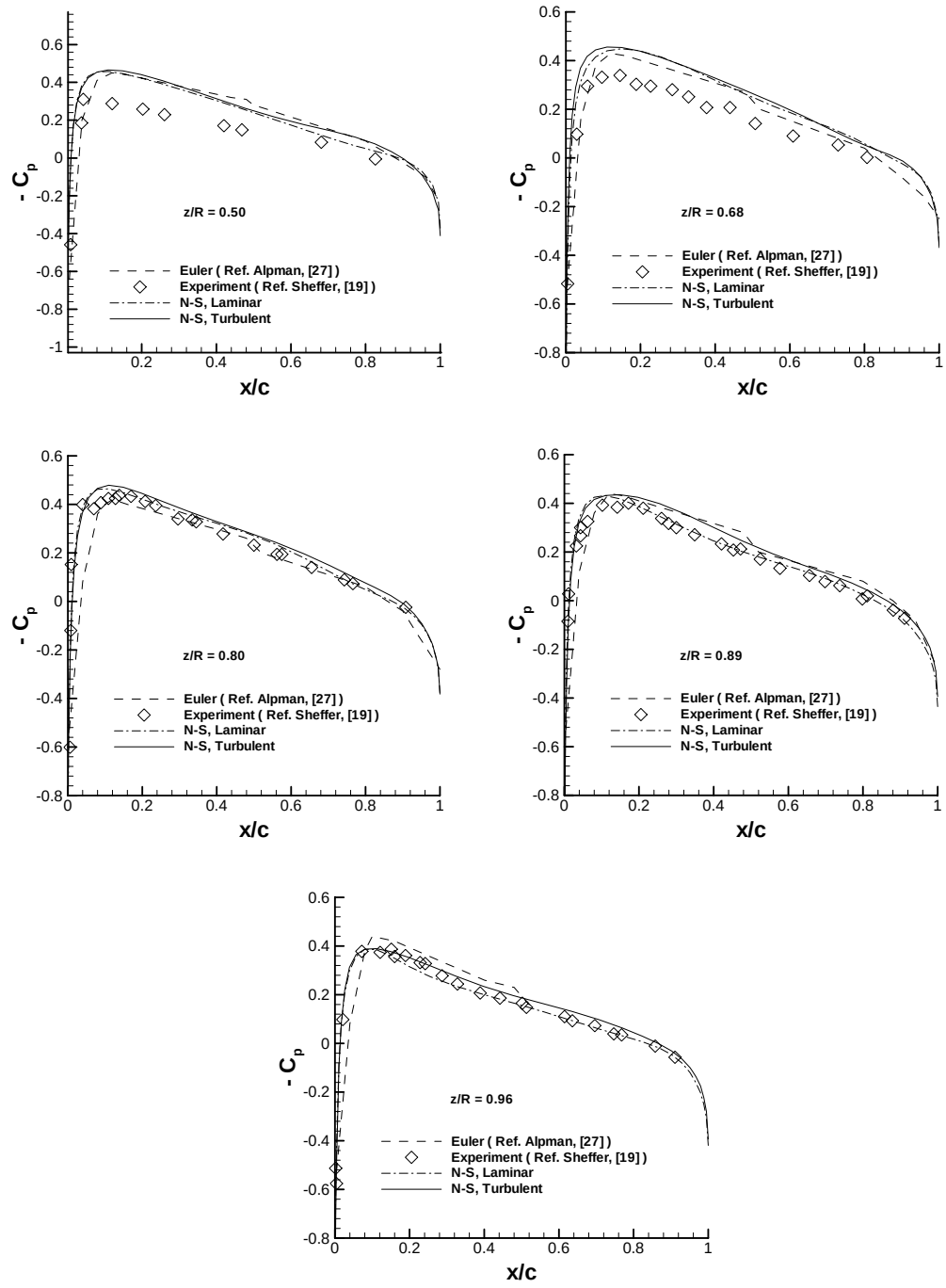

Figure 5.14: C_p distributions of various spanwise locations around UH – 1 rotor blade , $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$.

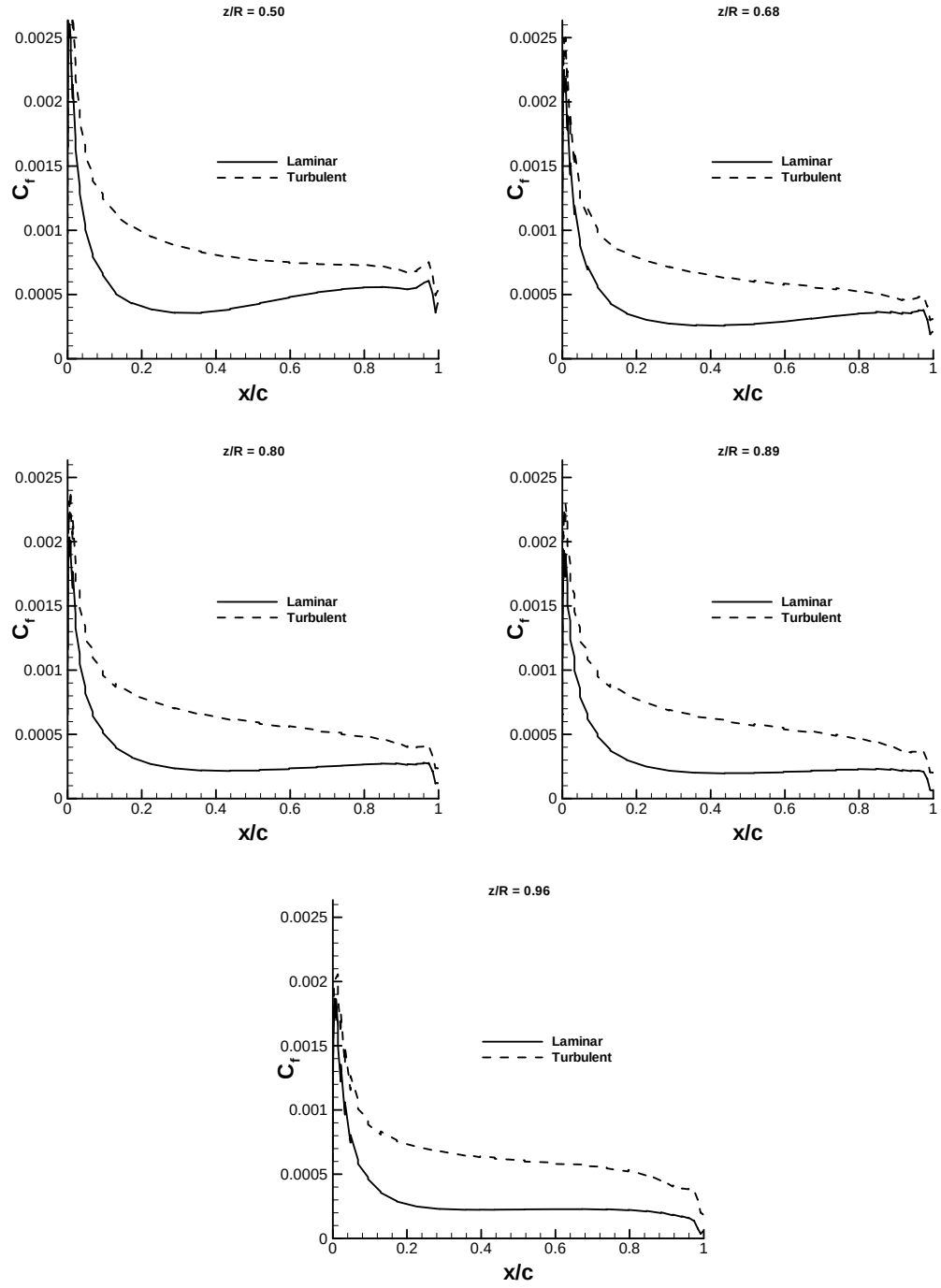

Figure 5.15: C_f distributions of various spanwise locations around UH – 1 rotor blade , $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 0^\circ$, turbulent case.

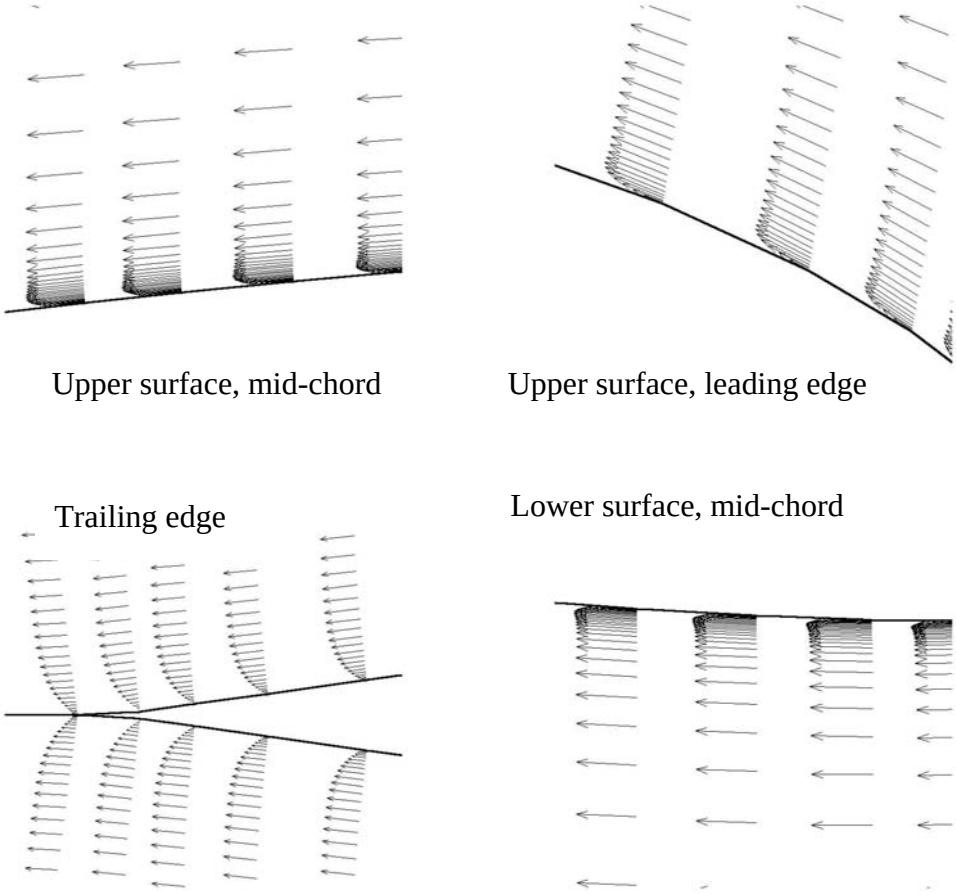

Fig: 5.16: Boundary Layer velocity profiles over UH-1 rotor blade , laminar case, $M_{tip} = 0.52$, $\alpha = 0^{\circ}$, $z/R = 0.96$, $\mathrm{Re} = 3.96 \times 10^{6}$

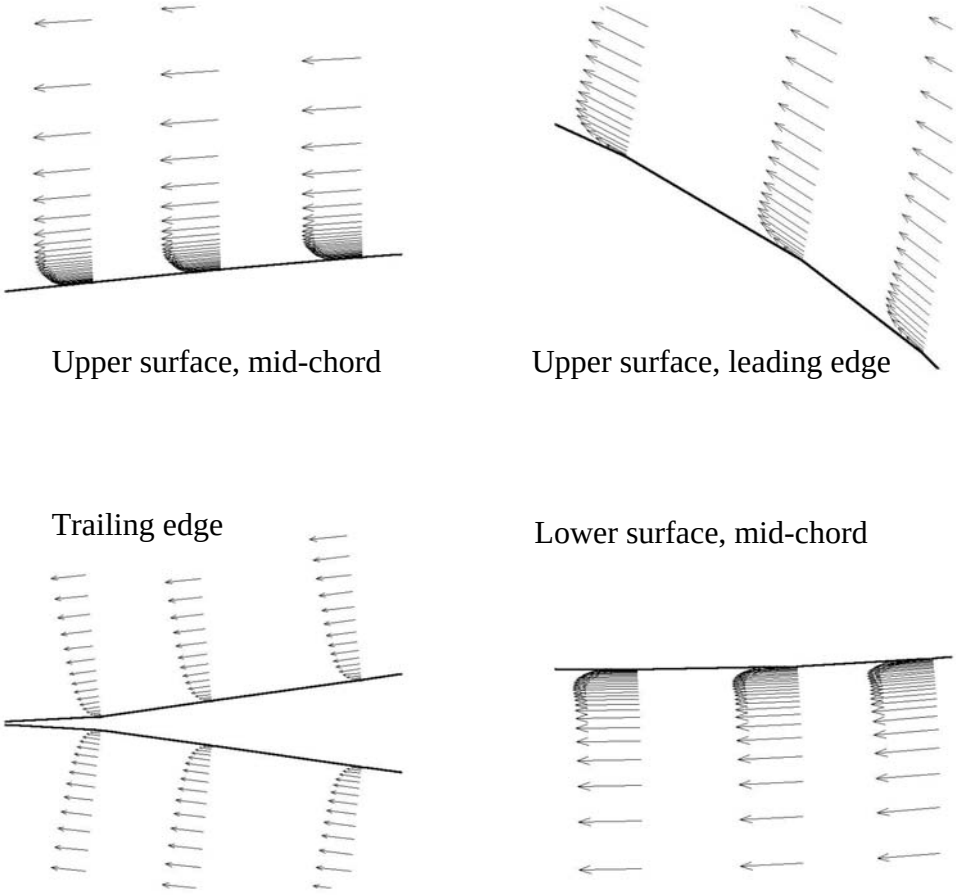

Fig: 5.17: Boundary Layer velocity profiles over UH-1 rotor blade , turbulent case, $M_{tip} = 0.52$, $\alpha = 0^{\circ}$, $z/R = 0.96$, $\mathrm{Re} = 3.96 \times 10^{6}$

5.2.2 $\alpha_0 = 8^o$, $M_{tip} = 0.439$ Case

Lifting rotor problem was solved for four different cases; for a laminar subsonic tip speed, a turbulent subsonic tip speed, a laminar transonic tip speed and a turbulent transonic tip speed. In the subsonic tip speed cases, tip Mach number was set to 0.439, whereas for the transonic tip speed cases it was taken as 0.877. For all cases , collective pitch angle was assigned as 8° to the blades. The blades were tilted to the desired angle by making local deformations in the computational mesh. Wake modelling approach was not followed and the vortical wake captured as part of the overall flowfield solution. $K^{/2)} = 0.5$, $K^{(4)} = 0.5$, $CFL = 1$ and $\alpha = 0.01$.

The convergence histories of a lifting rotor problem at $M_{tip} = 0.439$ are displayed at Fig. (5.18). The convergence histories displayed here were stored just for a single block which incidently happened to be away from the blade wall. Inspection of the Fortran code revealed that this was caused by a communication error in collecting the residuals from all the blocks. Nevertheless good convergence rates were obtained for the laminar and turbulent flowfields in this block, which may be considered as an indication of a globally converged solution.

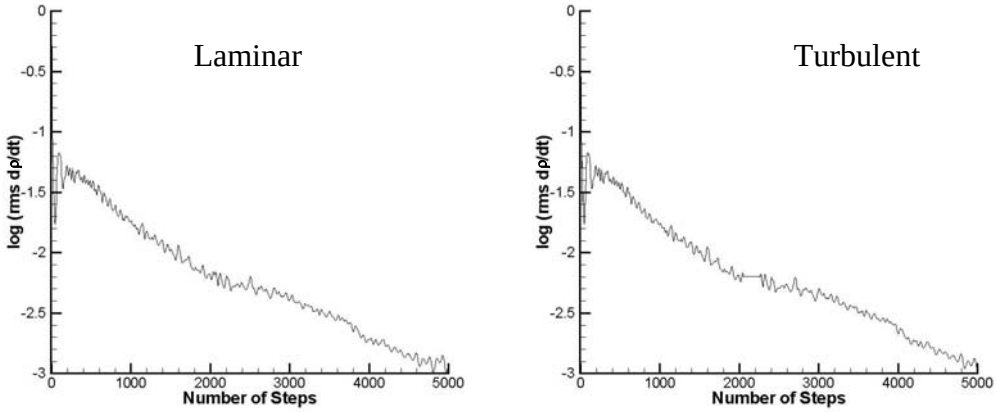

Figure 5.18: Residual histories for laminar and turbulent flow solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$, for an outer block of the computational domain.

Fig. (5.19), Fig. (5.20), and Fig. (5.21) show the non – dimensional surface pressure contours for laminar and turbulent cases. From these figures, the effect of radially decreasing speed and tilted rotor can be easily seen. Also, pressure distributions for both cases are very similar. It is obvious from Fig. (5.21) that the turbulent case resulted in slightly narrower low pressure region near the leading edge / tip junction. In Fig. (5.22) and Fig. (5.23) surface C_f contours for laminar and turbulent cases are presented. The effect of the turbulent viscosity can not be seen on the C_p values, but can be observed on C_f values mostly. In Fig. (5.24) and Fig. (5.25), pressure contours at different spanwise locations are diplayed for tip Mach speed 0.439. Fig. (5.26) presents the 2 – D non-dimensional pressure contours at different spanwise stations for lifting laminar and turbulent cases. No significant differences are observed between laminar and turbulent C_p distributions in volume, but surface distributions exhibit some differences, especially near the tip as will be seen next. Fig. (5.27) displays C_p distributions at various spanwise locations. It can easily be observed that laminar and turbulent results are in good agreement with experimental

data. Fig. (5.28) displays C_f distributions at different spanwise locations for turbulent case. For the lifting, sunsonic case, C_f values of turbulent case are higher than those of laminar case, as expected. In Fig. (5.29) and Fig. (5.30) boundary layer velocity profiles are presented for laminar and turbulent cases respectively. Boundary layer was captured by using 23 grid points in normal direction. Normally, a flow separation would be expected particularly in the laminar case but if the velocity profiles and surface C_f contours are inspected, separation is not observed for both laminar and turbulent cases. This can be thought as result of the 3-D effects arising from the rotation of the rotor blade.

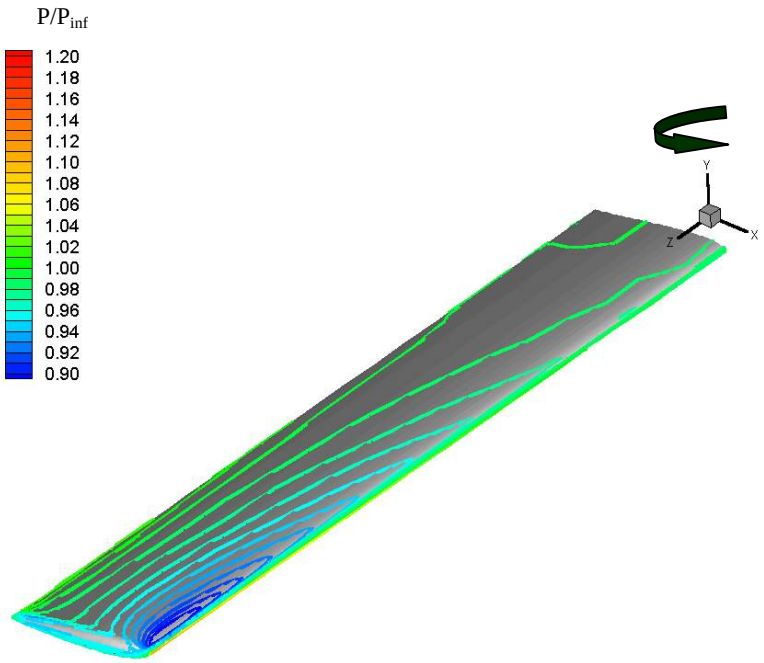

Figure 5.19: Surface pressure contours for laminar solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$.

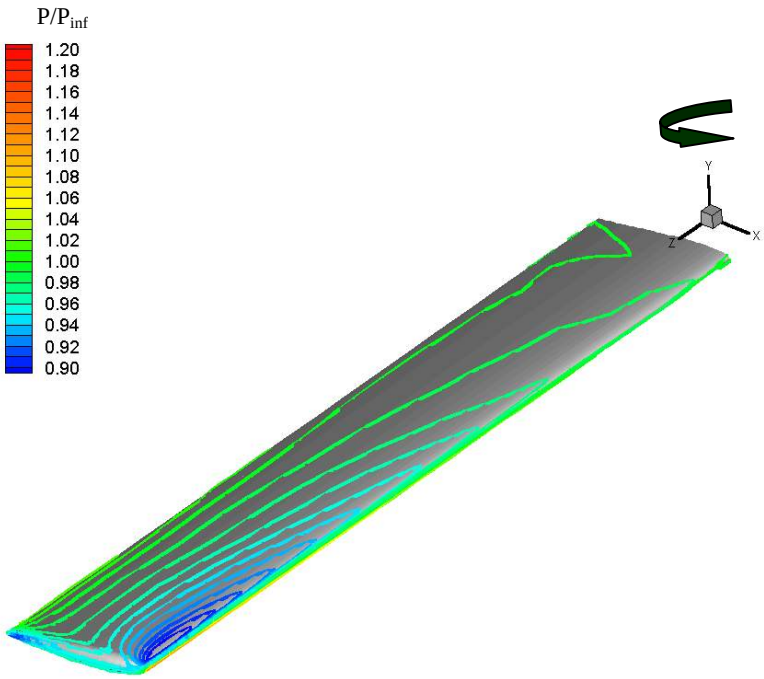

Figure 5.20: Surface pressure contours for turbulent solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$.

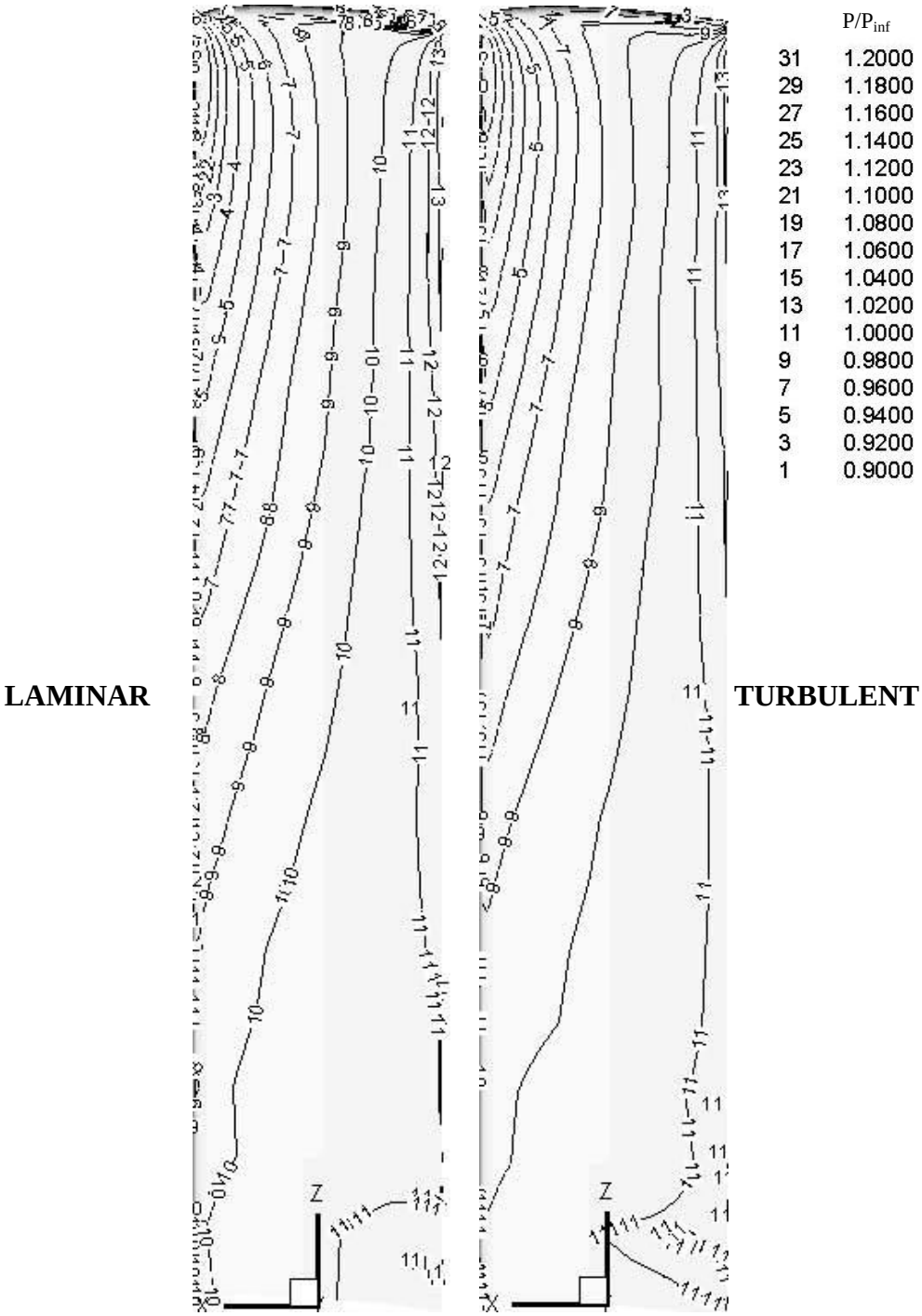

Figure 5.21: Non – dimensional surface pressure contours for laminar and turbulent solutions around UH – 1 rotor blades, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$.

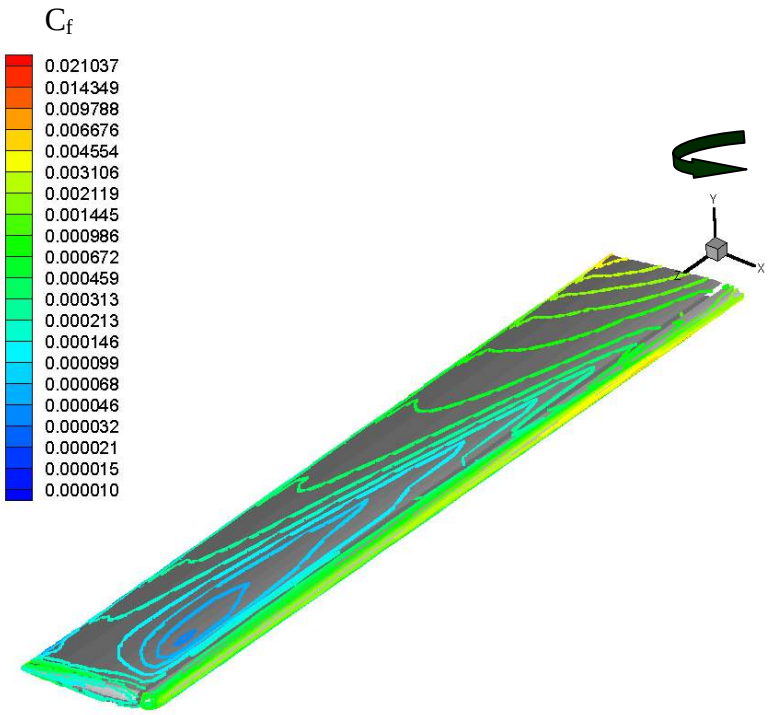

Figure 5.22: Surface C_f contours around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$, laminar case.

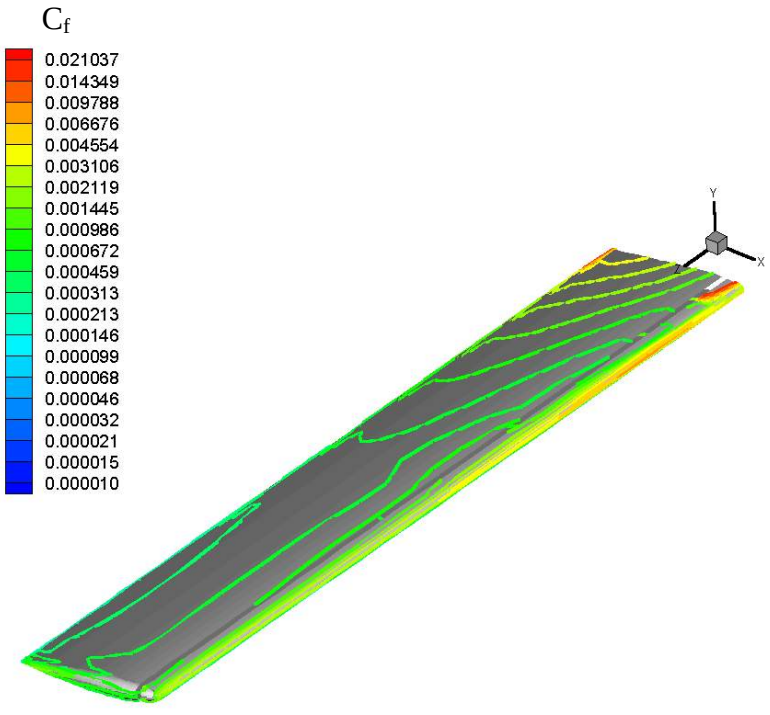

Figure 5.23: Surface C_f contours around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$, turbulent case.

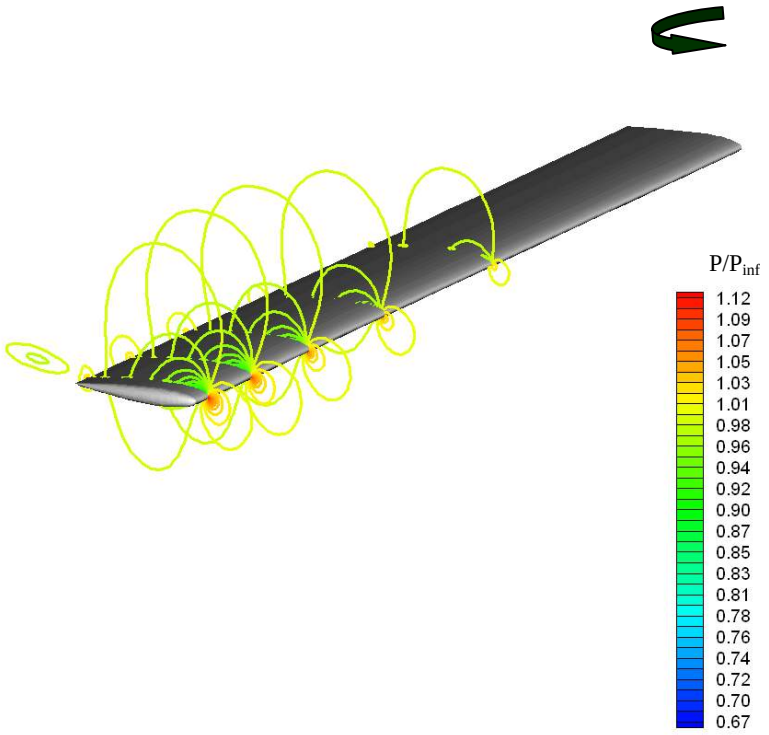

Figure 5.24: Pressure contours at different spanwise locations for laminar flow solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$, $z/R = 0.5,0.68,0.8,0.89,0.96$.

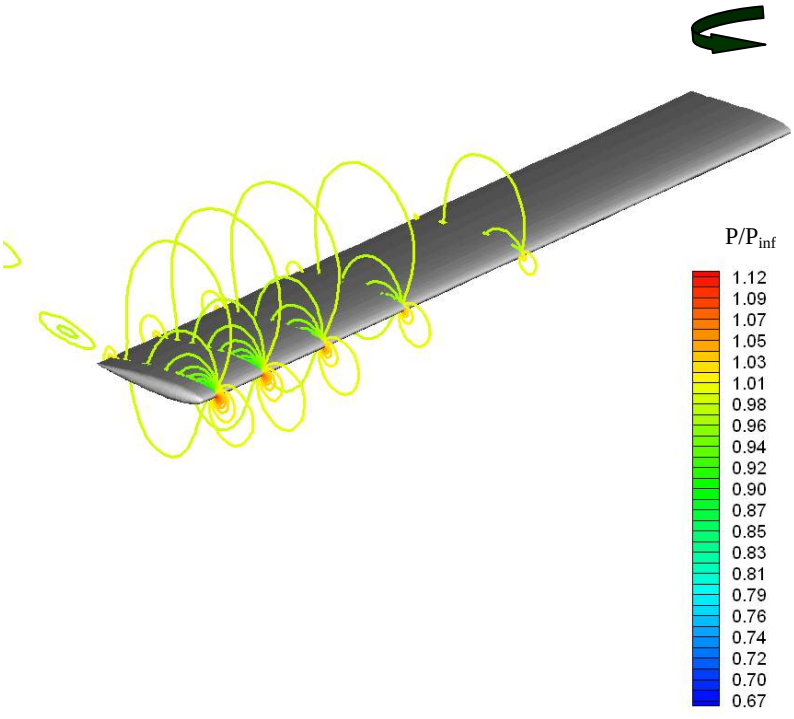

Figure 5.25: Pressure contours at different spanwise locations for turbulent flow solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$, $z/R = 0.5,0.68,0.8,0.89,0.96$.

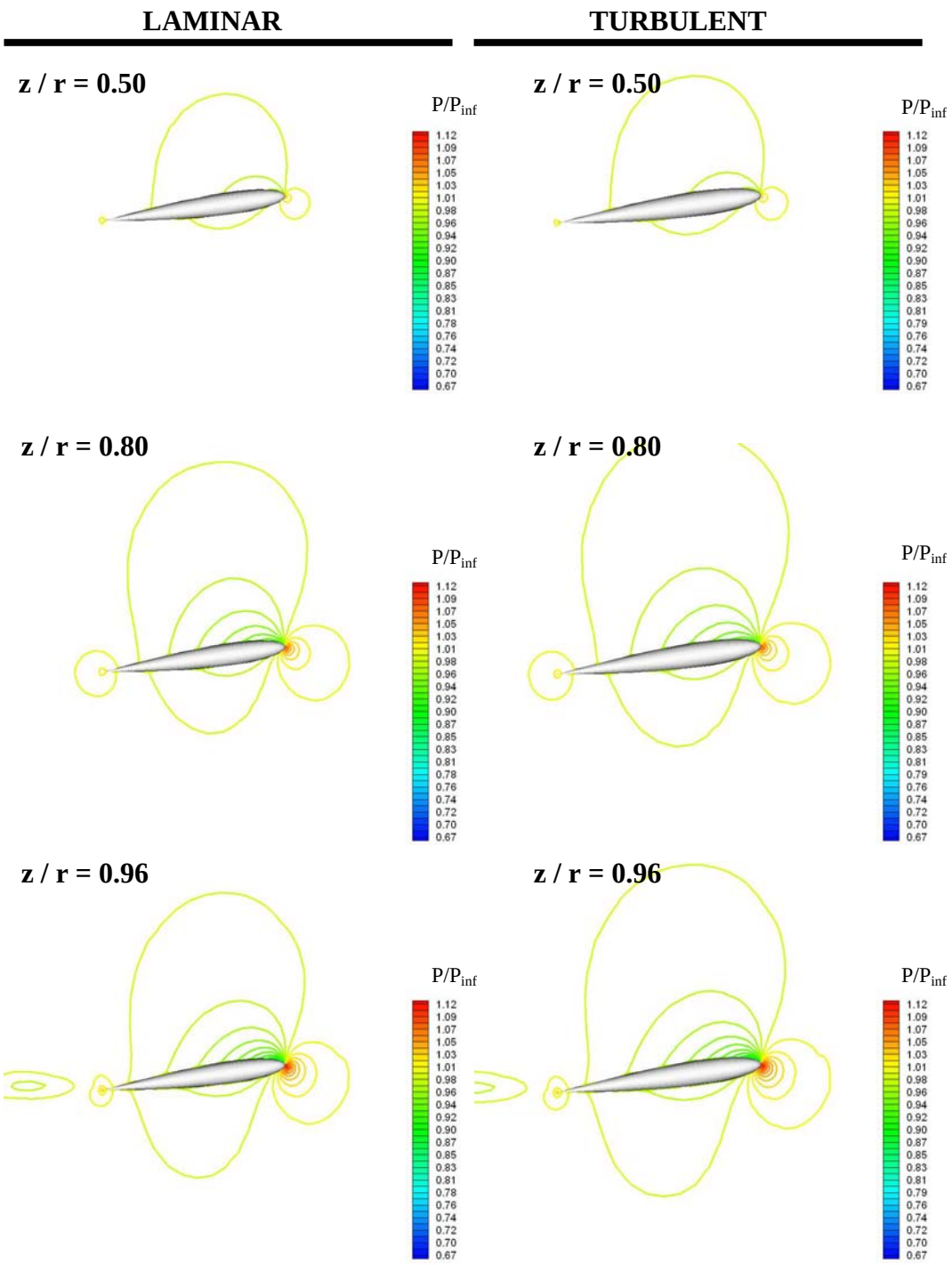

Figure 5.26: Pressure contours at different spanwise locations for laminar and turbulent flow solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 8^\circ$, $z/R = 0.5, 0.8, 0.96$.

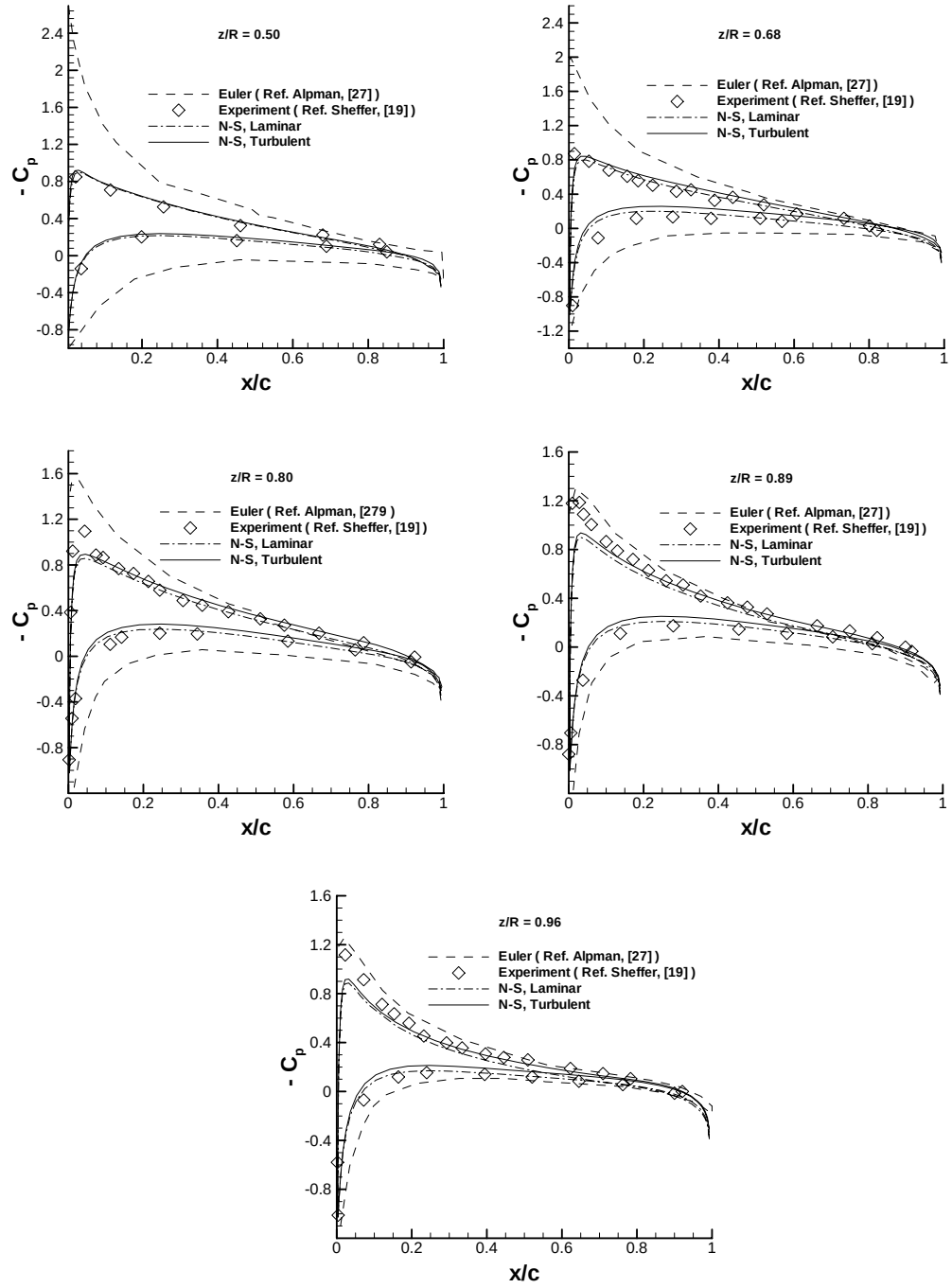

Figure 5.27: C_p distributions of various spanwise locations around UH – 1 rotor blade , $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$.

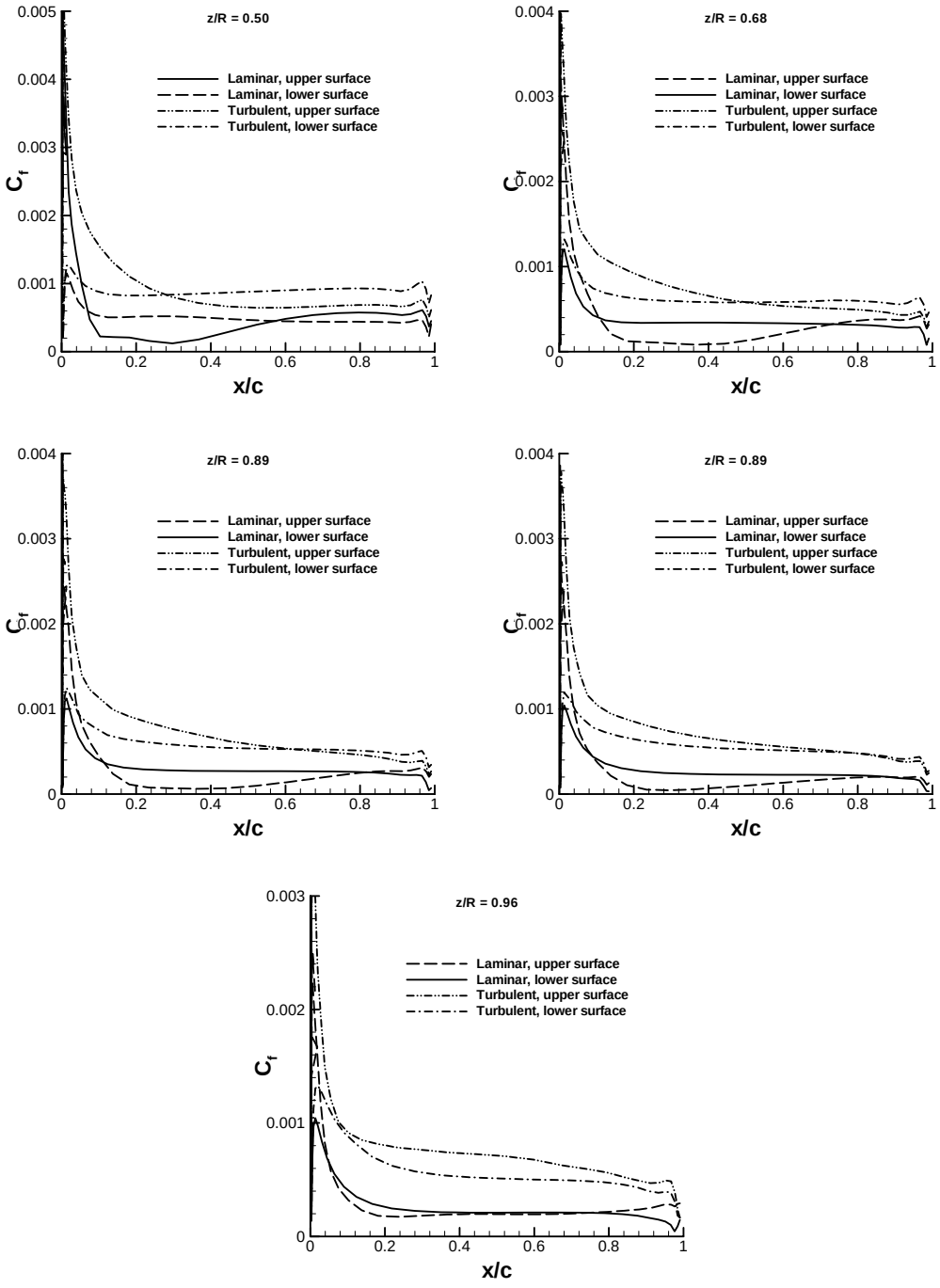

Figure 5.28: C_f distributions of various spanwise locations around UH – 1 rotor blade , $M_\infty = 0$, $M_{tip} = 0.439$, $\alpha_0 = 8^\circ$.

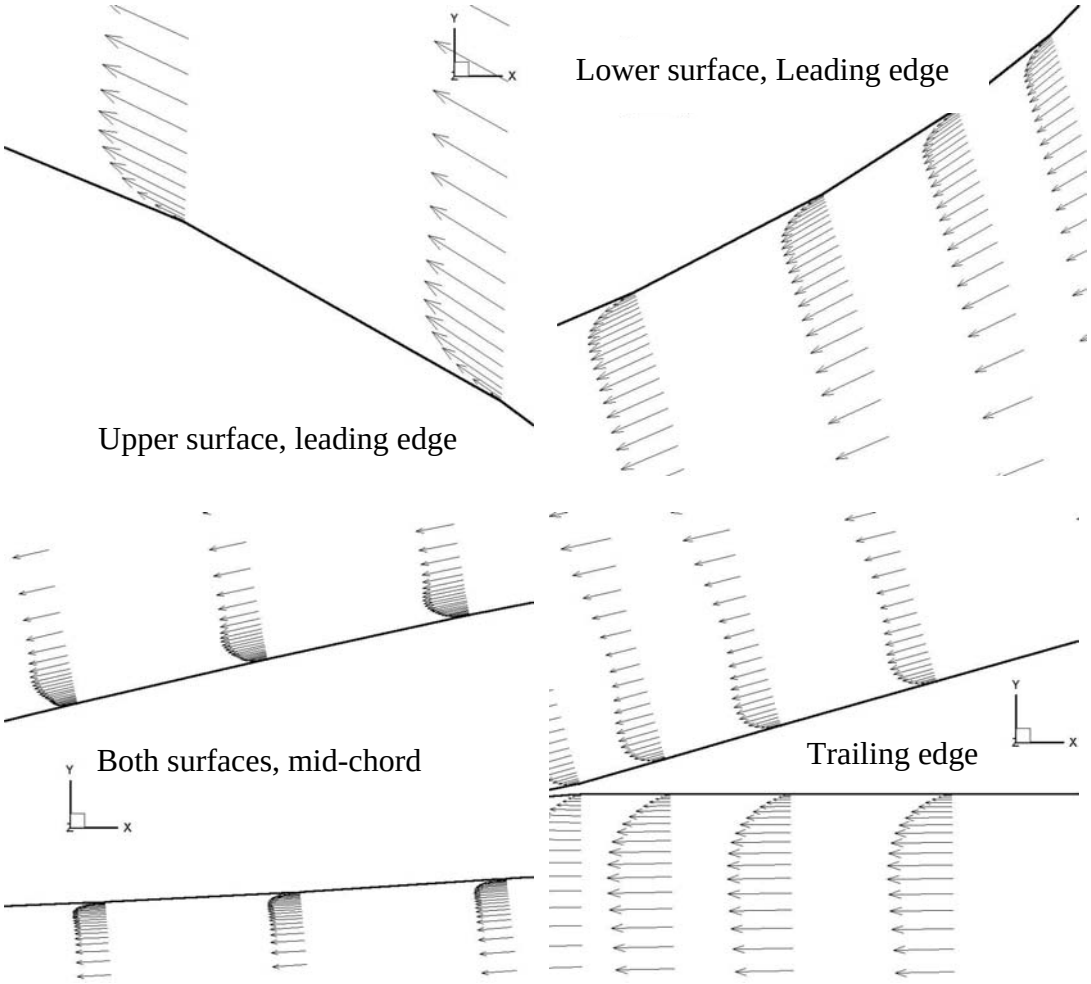

Fig: 5.29: Boundary Layer velocity profiles over UH –1 rotor blade , laminar case, $M_{tip} = 0.439$, $\alpha = 8^{\circ}$, $z/R = 0.50$, $\mathrm{Re} = 3.96 \times 10^{6}$

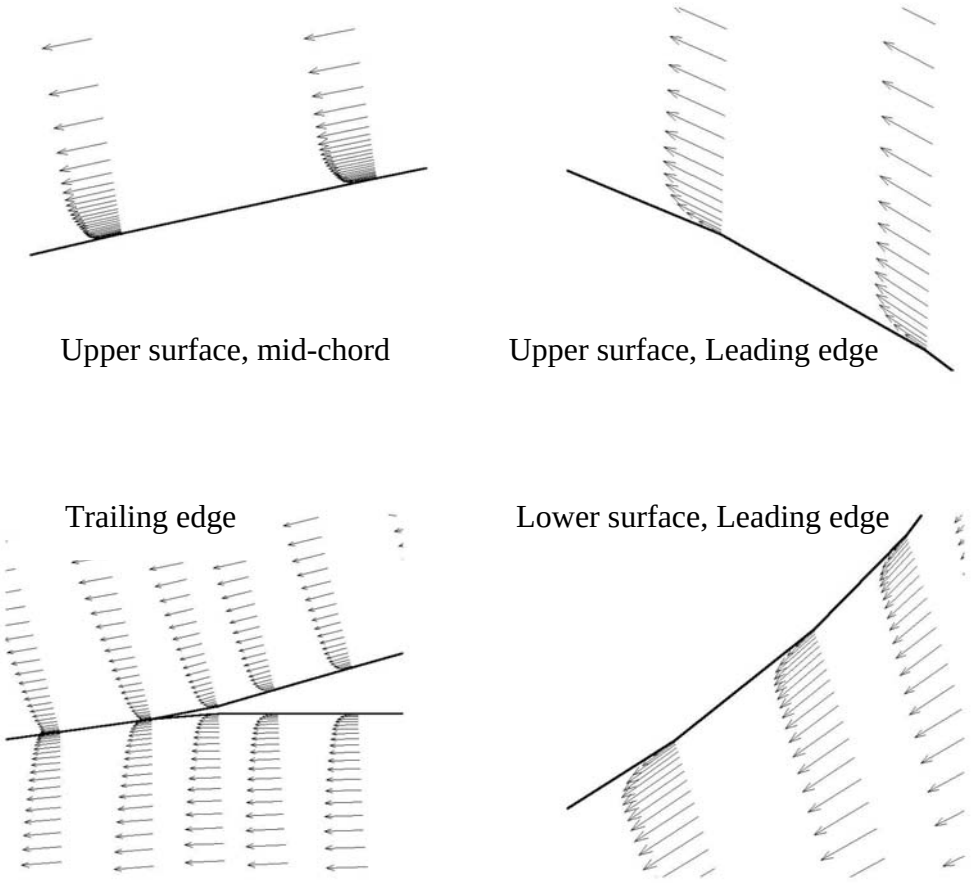

Fig: 5.30: Boundary Layer velocity profiles over UH –1 rotor blade , turbulent case, $M_{tip} = 0.439$, $\alpha = 8^{\circ}$, $z/R = 0.96$, $\mathrm{Re} = 3.96 \times 10^{6}$

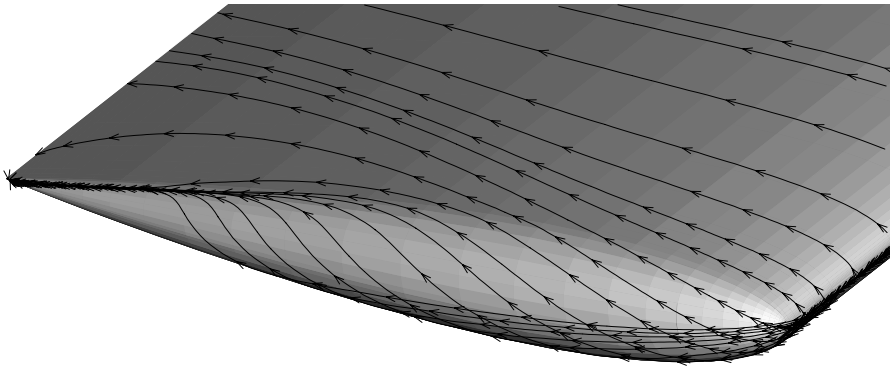

Fig: 5.31: Streamlines at the tip region of the UH-1 blade, turbulent case, $M_{tip} = 0.439$, $\alpha = 8^{\circ}$, $\mathrm{Re} = 3.96 \times 10^{6}$

In Fig. (5.31), streamlines at the tip region of the UH-1 blade and the evolution of the tip vortex are displayed. The major cause for this formation is the pressure unbalance between the lower and upper surface. The fluid particles at the high pressure region (lower surface for lifting case) move to the region where the pressure is lower. So a fluid flow occurs from the lower surface to the upper surface at the tip. This continuous flow causes the tip vortex formation (vortex roll up).

5.2.3 $\alpha_0 = 8^o$, $M_{tip} = 0.877$ Case

Fig. (5.32) displays the residual histories for transonic tip speed. Good convergence rates were obtained for transonic tip speed. Unlike the other cases, the difference between the convergence rate of laminar case and that of turbulent case can be observed obviously. If they are compared, carefully, it can be seen that convergence rate of laminar case is higher than that of turbulent case as expected. Fig. (5.33), Fig. (5.34) and Fig. (5.35) display non – dimensional surface pressure contours for transonic tip speed. Although there is not any big difference in the volume pressure contours, a significant difference in the surface pressure contours can be seen. In the tip region where the fluid velocity reaches its maximum values and the pressure reaches its minimum value, acceleration of the flow exhibits different characteristics for laminar and turbulent cases. In Fig. (5.35), it can be observed that, additional turbulent viscosity results in a smooth acceleration mechanism for the turbulent case at the tip region of the blade.

Fig. (5.36) and Fig. (5.37) show surface C_f contours for transonic tip speed. C_f values of turbulent case are higher than those of laminar case and they are relatively higher at leading edge as expected..In Fig. (5.38) and Fig. (5.39), non – dimensional pressure contours at various spanwise locations are displayed for laminar and turbulent cases. Fig. (5.40) presents the 2 – D non-dimensional pressure contours at different spanwise stations for lifting laminar and turbulent cases. As it was mentioned before, there is not a significant difference between the volume C_p contours of laminar and turbulent cases.

Fig. (5.41) shows the C_p distributions and comparisons with experimental data and Euler results. Generally, a very good agreement with the experimental data

is observed particularly in the hub region where the flow velocity is small relative to the velocity in the tip region. The shock position is almost identical for all solutions. Better results can be obtained in the tip region with a finer grid resolution. Surface C_p results of N-S solutions are closer to the experimental data than those of Euler solutions. The effect of the transonic shock wave is observed on C_p values in the transonic tip case. Fig. (5.42) displays C_f distributions at different spanwise locations for laminar and turbulent cases. As expected, C_f exhibits higher values at the leading edge. Besides, turbulent C_f values are higher than those of laminar case both on upper and surface surfaces.

In Fig. (5.43) and Fig. (5.44) boundary layer velocity profiles on UH-1 helicopter blade are presented for laminar and turbulent cases. In Fig. (5.43), an adverse flow can be observed on the upper survace. In Fig. (5.45) and Fig. (5.46), velocity fields at a streamwise station are given. In these figures the velocity vectors represent the V and W components of the total velocity. At the cross-section, which is near to the trailing edge, flow roll up and tip vortex formation can be seen. In Fig. (5.47), a closer view of the tip vortex formation can be observed. Fig. (5.48) and Fig. (5.49) displays velocity fields after the blade and before the blade, respectively.

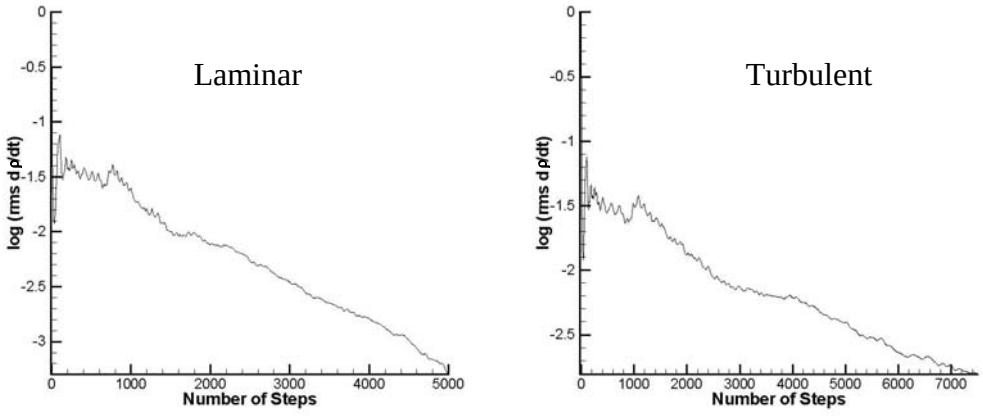

Figure 5.32: Residual histories for laminar and turbulent flow solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$.

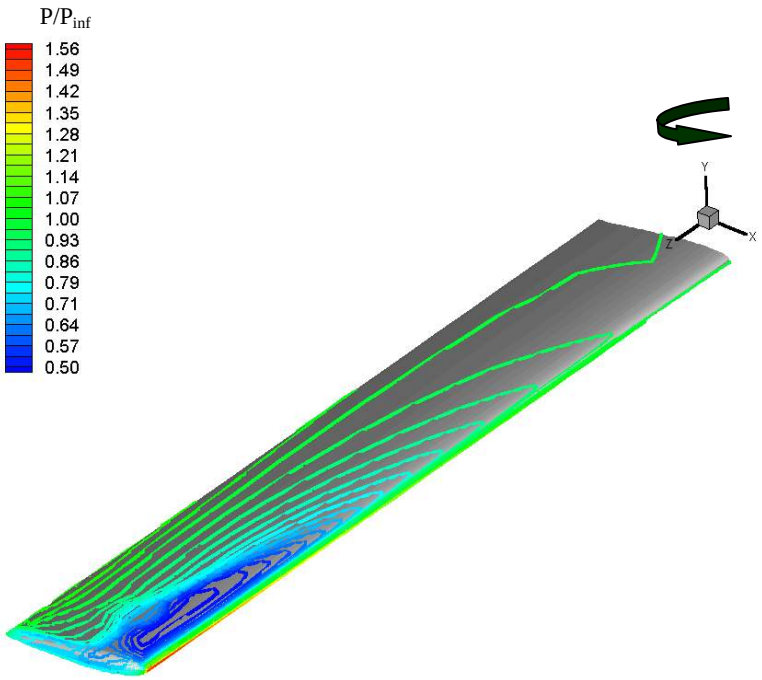

Figure 5.33: Surface pressure contours for laminar solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$.

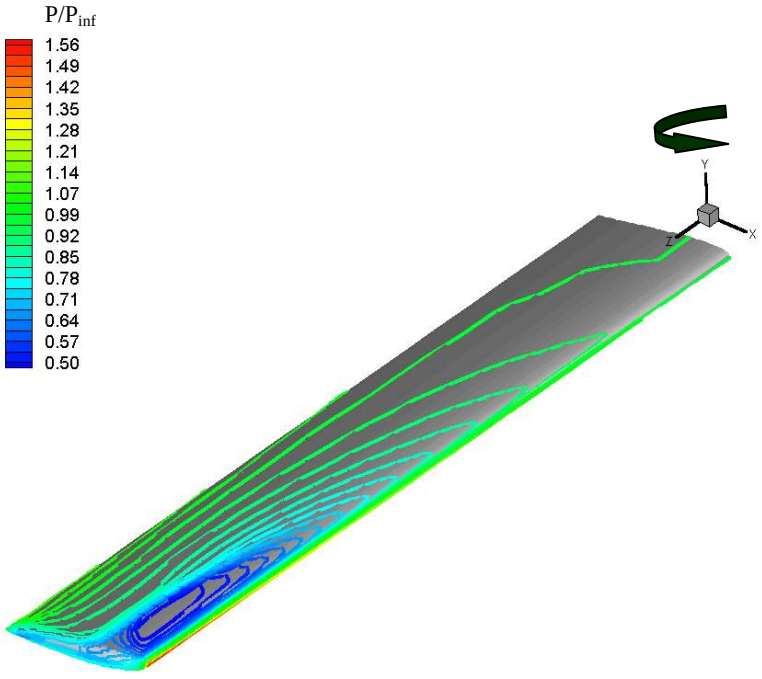

Figure 5.34: Surface pressure contours for turbulent solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$.

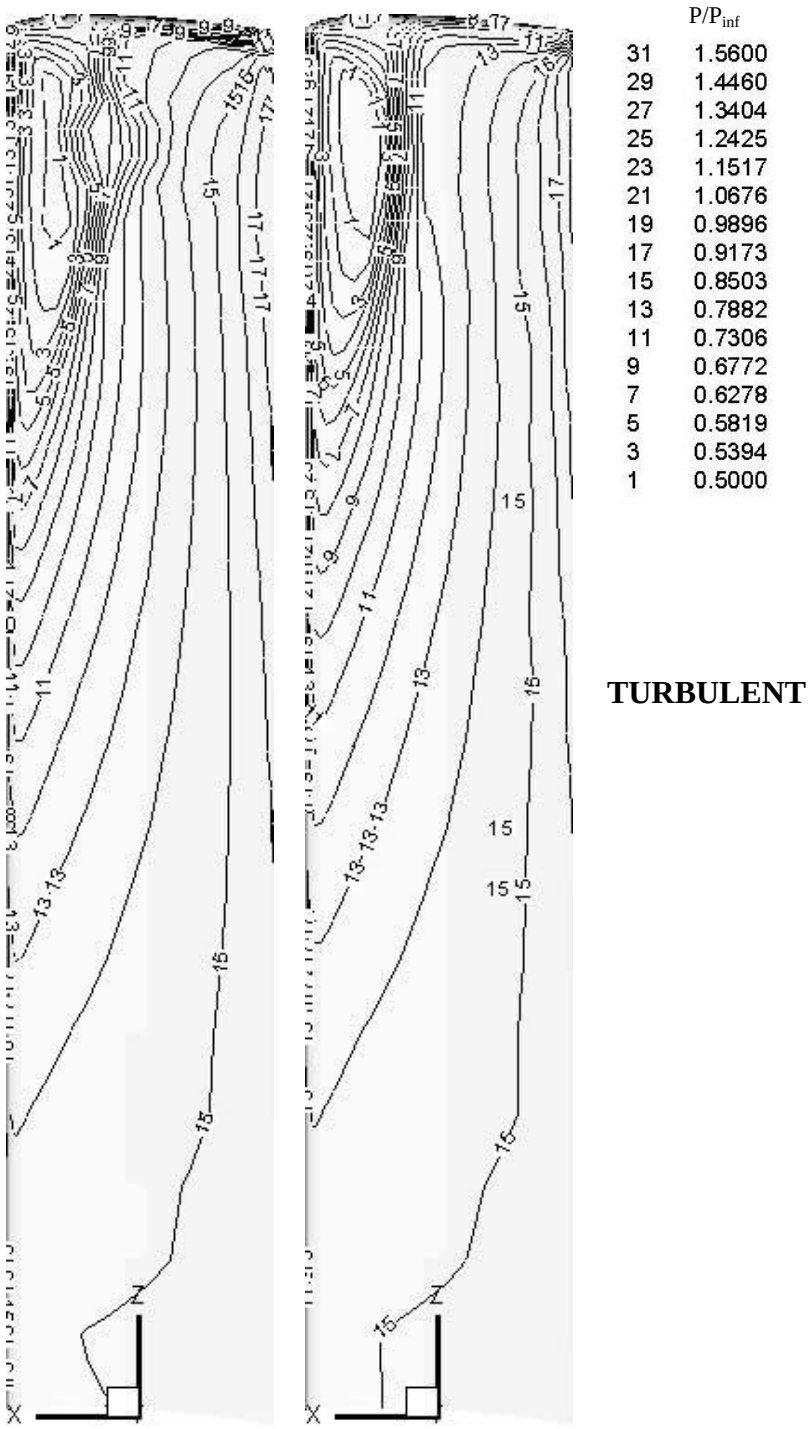

Figure 5.35: Non - dimensional surface presure contours for laminar and turbulent solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$.

C_f

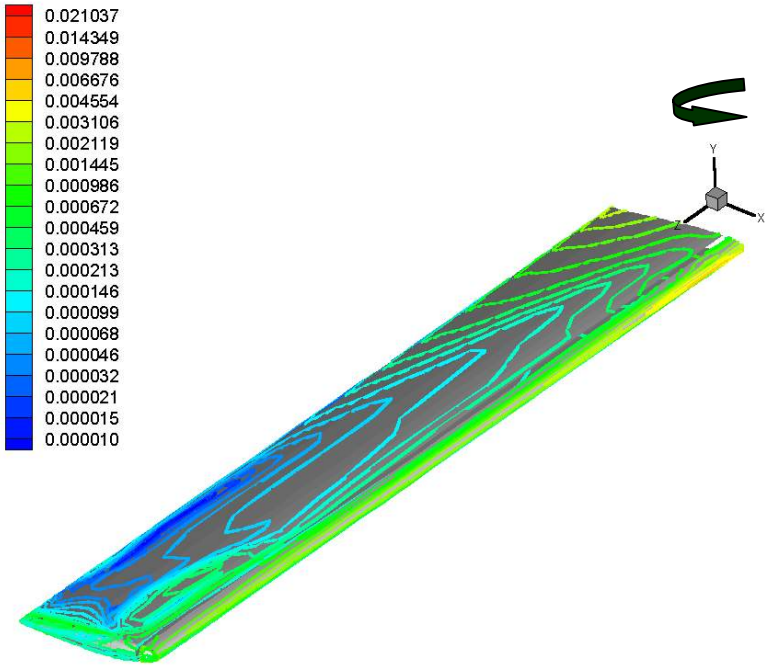

Figure 5.36: Surface C_f contours around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$, laminar case.

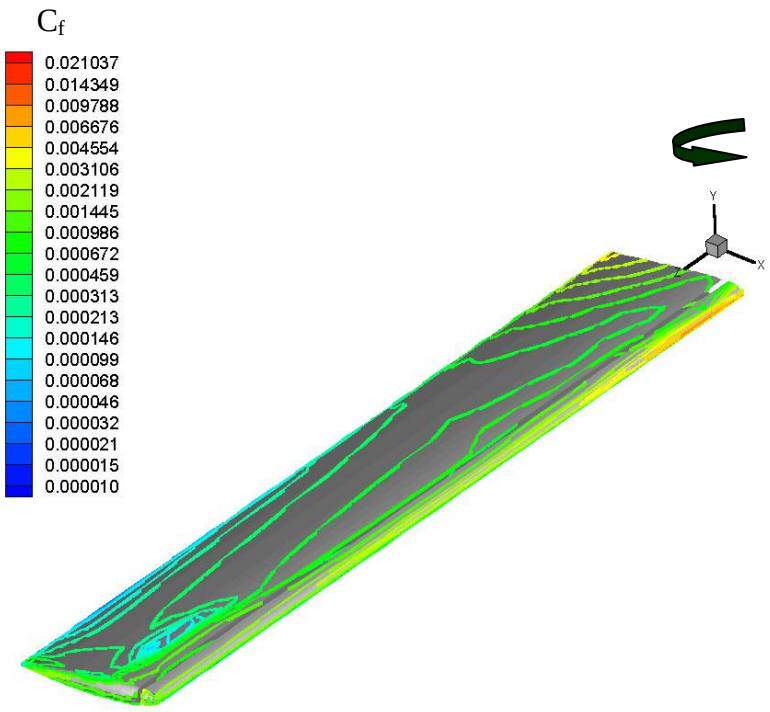

Figure 5.37: Surface C_f contours around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$, turbulent case.

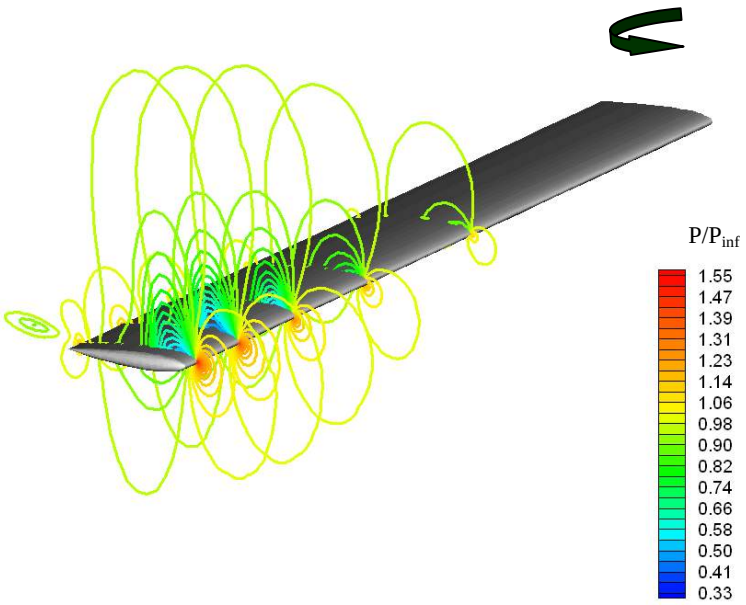

Figure 5.38: Pressure contours at different spanwise locations for laminar flow solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$, $z/R = 0.5,0.68,0.8,0.89,0.96$.

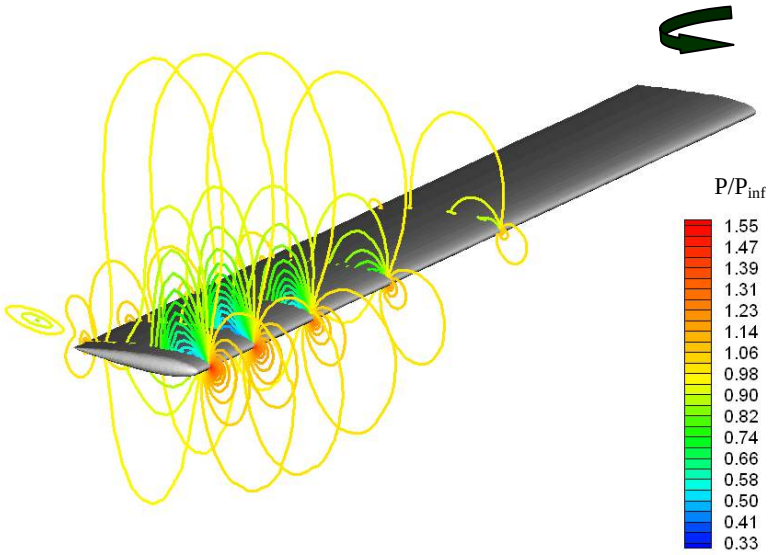

Figure 5.39: Pressure contours at different spanwise locations for turbulent flow solution around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$, $z/R = 0.5,0.68,0.8,0.89,0.96$.

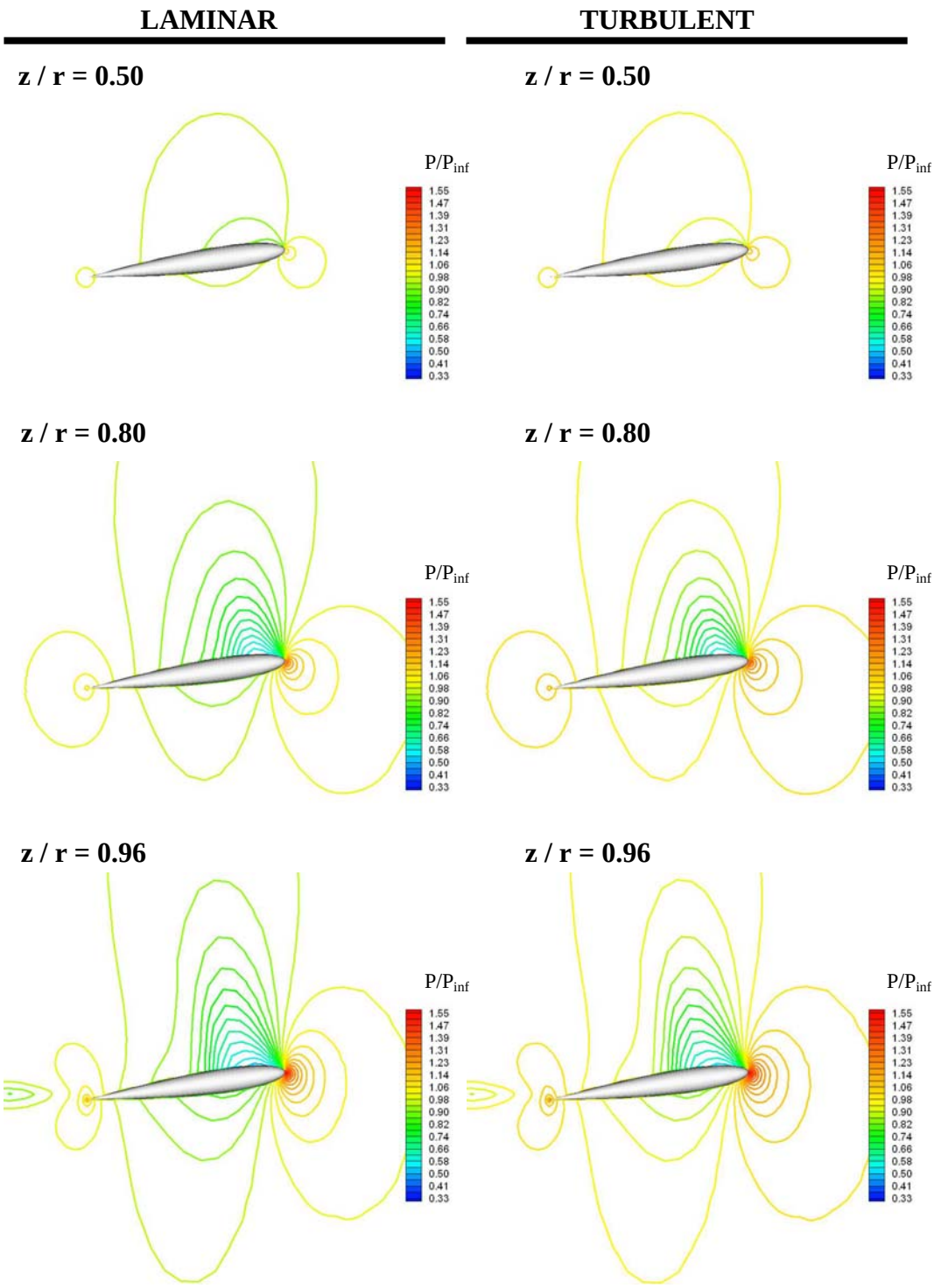

Figure 5.40: Pressure contours at different spanwise locations for laminar and turbulent flow solutions around UH – 1 rotor blade, $M_\infty = 0$, $M_{tip} = 0.52$, $\alpha_0 = 8^\circ$, $z/R = 0.5,0.8,0.96$.

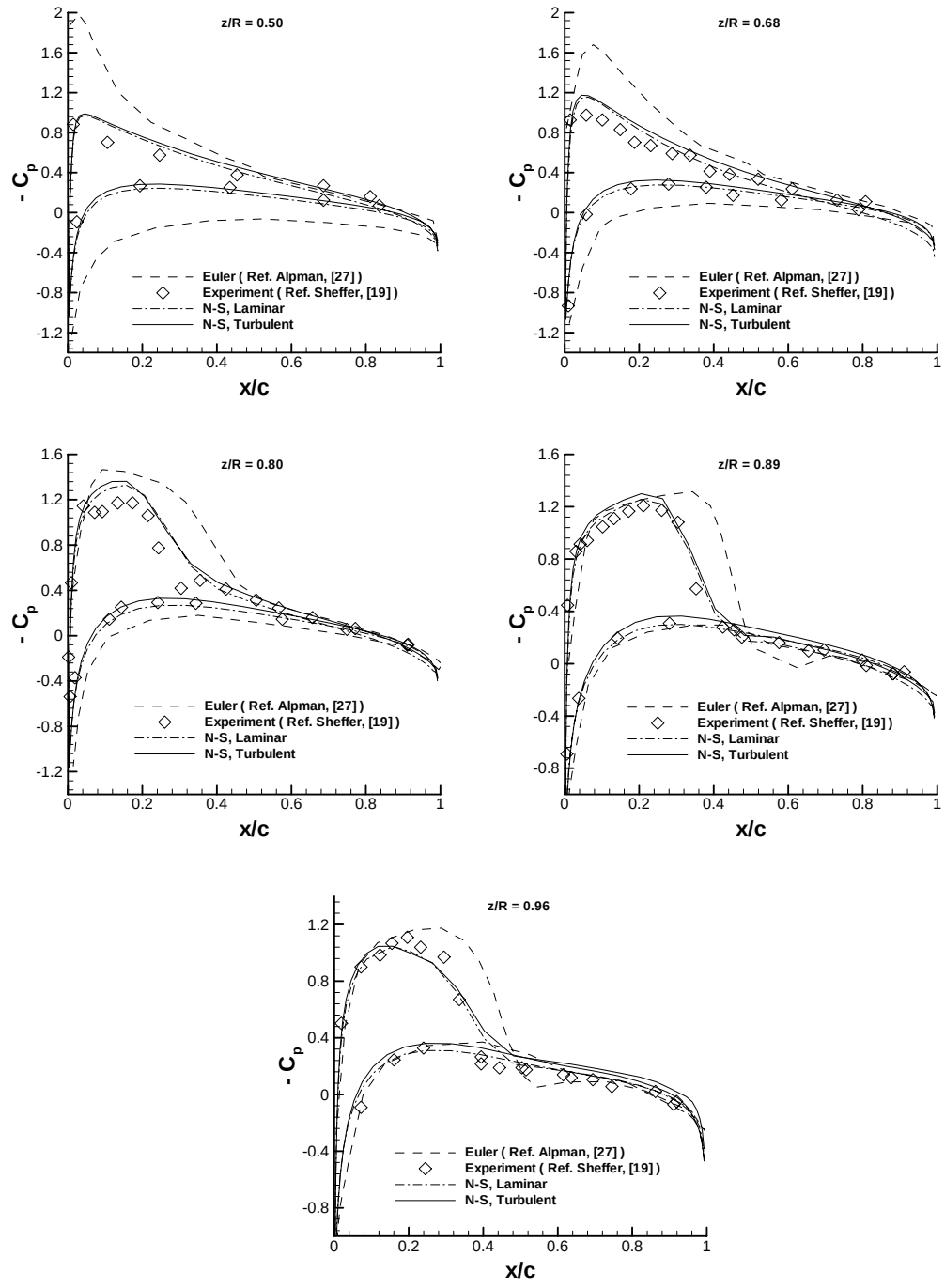

Figure 5.41: C_p distributions of various spanwise locations around UH – 1 rotor blade , $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$.

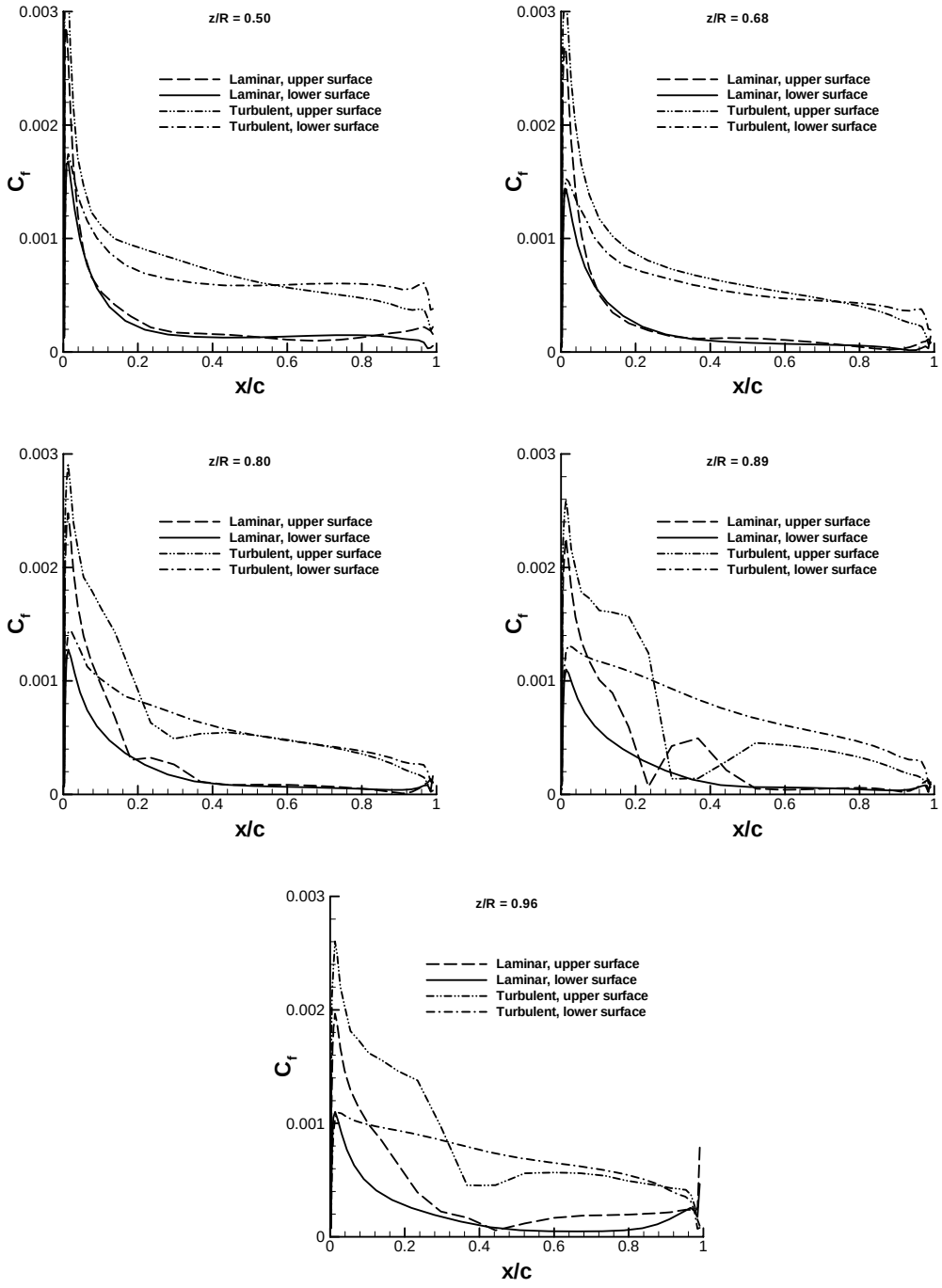

Figure 5.42 : C_f distributions of various spanwise locations around UH – 1 rotor blade , $M_\infty = 0$, $M_{tip} = 0.877$, $\alpha_0 = 8^\circ$.

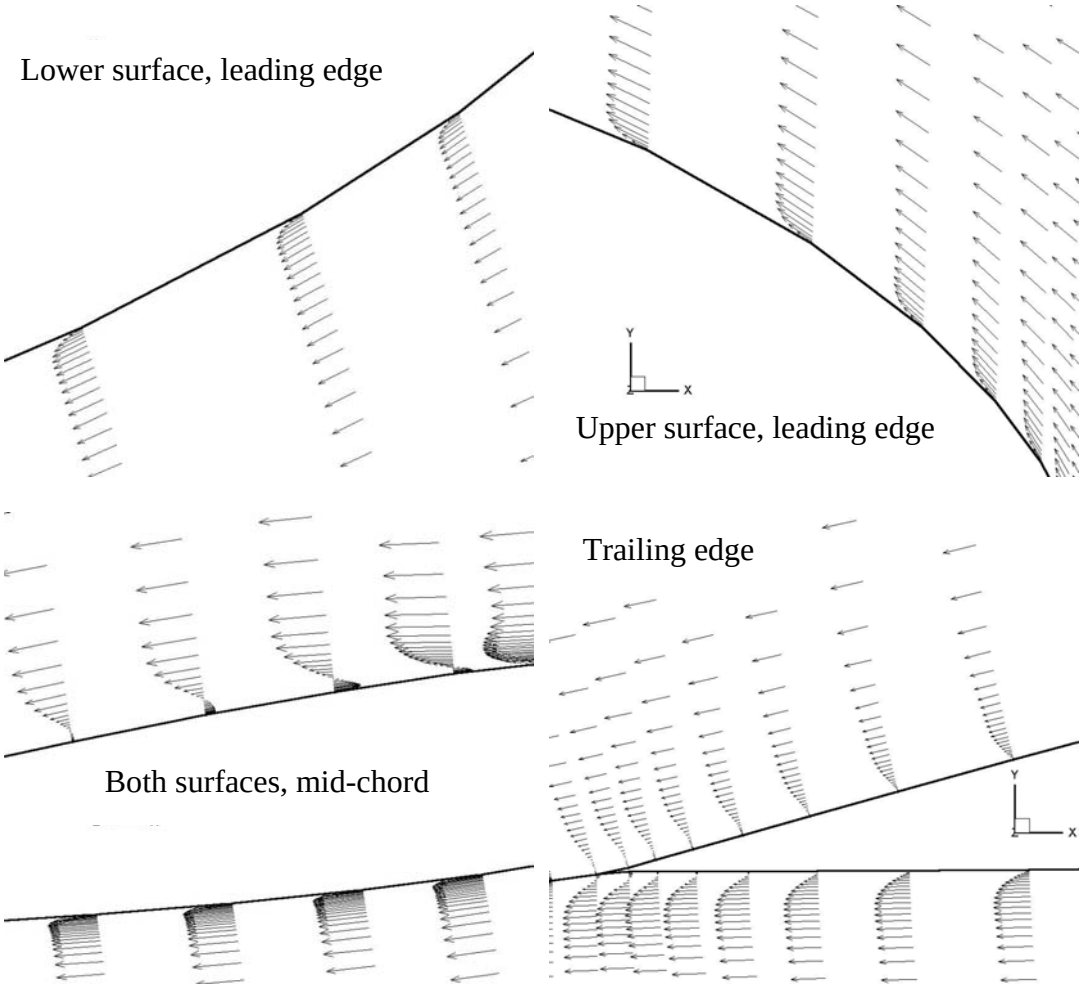

Fig: 5.43: Boundary Layer velocity profiles over UH-1 rotor blade , laminar case, $M_{tip} = 0.877$, $\alpha = 8^{\circ}$, $z/R = 0.96$, $\mathrm{Re} = 3.96 \times 10^{6}$

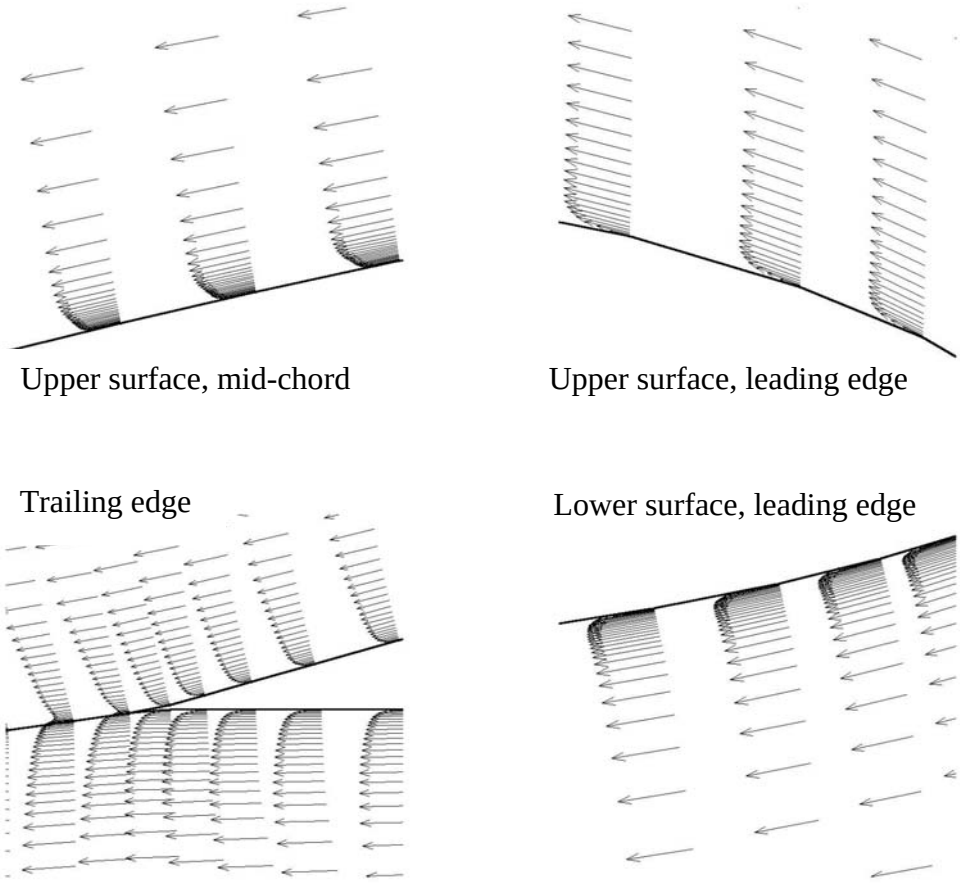

Fig: 5.44: Boundary Layer velocity profiles over UH-1 rotor blade , turbulent case, $M_{tip} = 0.877$, $\alpha = 8^{\circ}$, $z/R = 0.96$, $\mathrm{Re} = 3.96 \times 10^{6}$

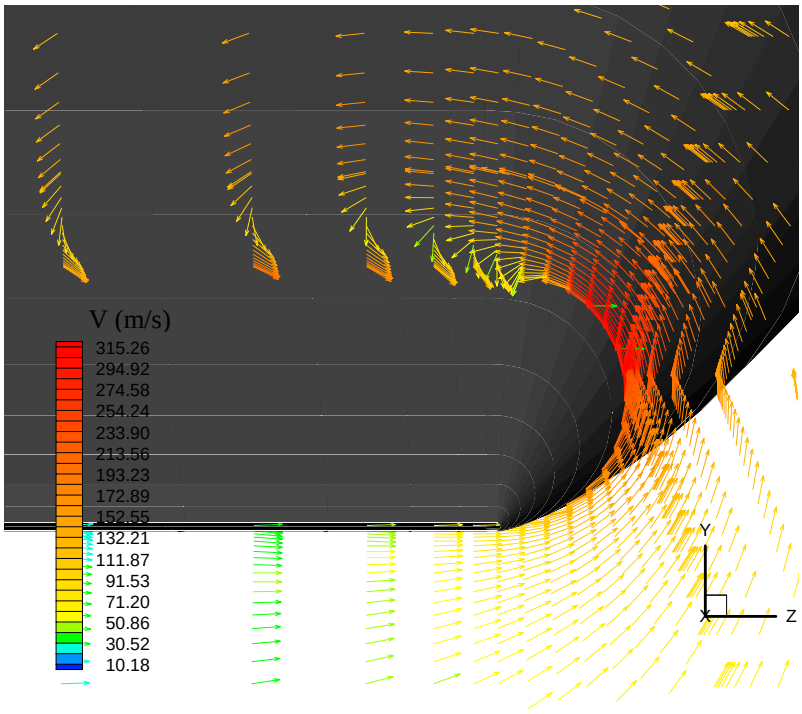

Figure 5.45: Vortex formation at the tip of the blade. Vectors are colored by velocity magnitude, $M_{tip} = 0.877$, $\alpha = 8^{\circ}$, $\mathrm{Re} = 3.96 \times 10^{6}$

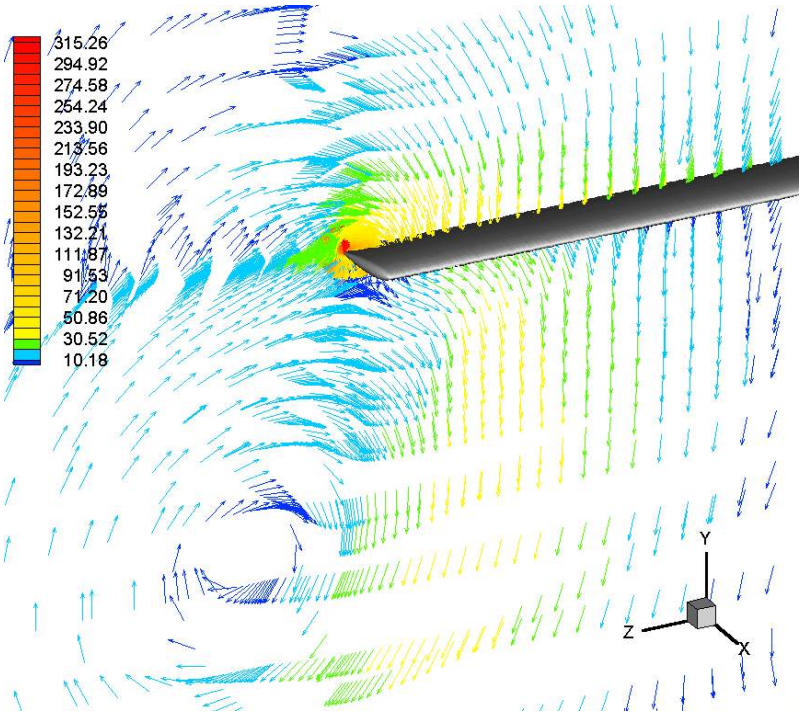

Figure 5.46: Vortex formed by a retreating blade which has been subject to rotor downwash –perspective view. $M_{tip} = 0.877$, $\alpha = 8^{\circ}$, $\mathrm{Re} = 3.96 \times 10^{6}$

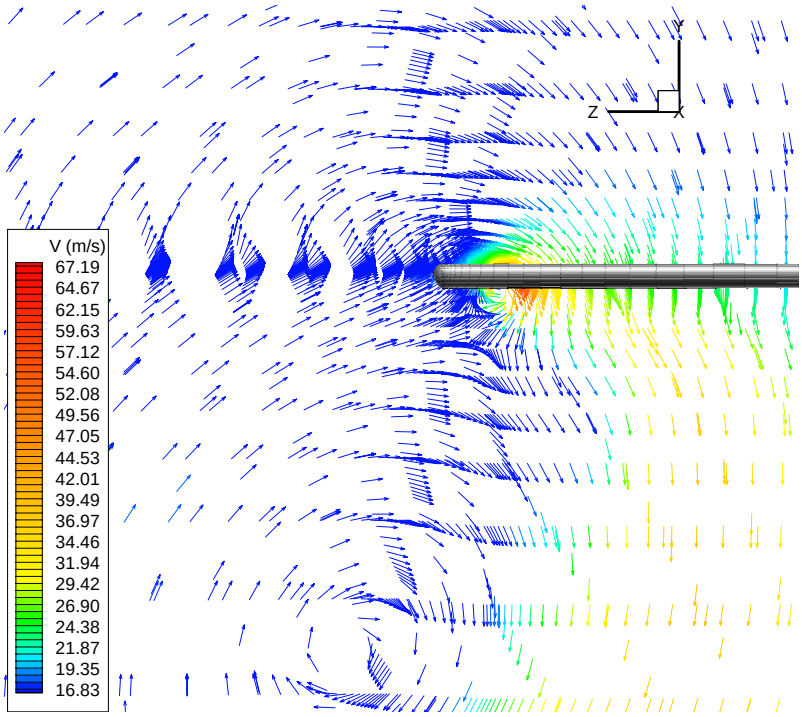

Figure 5.47: Vortex formed by a retreating blade which has been subject to rotor downwash –in YZ plane. $M_{tip} = 0.877$, $\alpha = 8^{\circ}$, $\mathrm{Re} = 3.96 \times 10^{6}$

CHAPTER 6

CONCLUSION

This study contains numerical simulations of flowfields of a flat plate and rotary wing configurations. The solutions have been obtained using a parallel Fortran 90 code which solves the thin – layer Navier – Stokes equaitons. A cell centered finite volume method for spatial discretization and a four step Runge – Kutta type time stepping scheme has been used. In order to decrease the required computational time and memory, parallel processing approach was utilized. Baldwin – Lomax turbulence model has been used to model turbulence.

The developed code has been run for various problems on a cluster of dual Pentium – III processors (700 MHz and 512 MB Ram on each) housed in the Aerpspace Engineering Department of the Middle East Technical University. The computed results have been, in general, in good agreement with experimental data.

The most crucial element in the evaluation of three – dimensional flowfields around lifting surfaces is an accurate prediction of the vortical wake. This is especially very important for rotary wing problems, where, unlike in fixed wing aircraft case, the tip vortices remain close to the rotor and produce a complicated induced velocity field that has a primary influence on the blade loads and the rotor performance. The strength and the proximity of the tip vortices result in powerful interactions between the individual vortex filaments and also between the blades

[27]. For nonlifting cases, formation of the tip vortex was investigated. For subsonic tip number case, streamlines at the tip region are displayed. Rolling up of the flow and vortex formation can be easily observed on the Fig. (5.31). For transonic tip number case, velocity fields at different streamwise stations are given. Velocity field is an important tool since it exhibits the general behaviour of the flow. On the displayed graphs for transonic tip speed, formation of the tip vortex and location of the helical wake can be seen.

Convergence histories for all cases are displayed. The convergence histories displayed here were stored just for a single block which incidently happened to be away from the blade wall. Inspection of the Fortran code revealed that this was caused by a communication error in collecting the residuals from all the blocks. Nevertheless good convergence rates were obtained for the laminar and turbulent flowfields in this block, which may be considered as an indication of a globally converged solution.

Modified Jameson type artificial dissipation terms were utilized for the solutions due to highly stretched grid points in the vicinity of the solid boundaries to capture the thin shear layer. Laminar and turbulent solutions have been obtained for hovering UH – 1 blade for lifting and non – lifting cases and compared with experimental data. Surface pressure contours and C_p contours at different spanwise locations are compared and displayed. Most of the N-S results are in good agreement with experimental data. Smal differences can be observed at the tip region because of the grid resolution at that region. It is possible that better results can be obtained in tip region with a finer grid resolution. However, N-S results are better than Euler results and are very similar to the experimental data. It proves that, new viscous fine mesh, additional viscous terms and turbulence subroutine work well. 3-D and 2-D

non-dimensionalized pressure contours are also displayed. In general, there is not a significant difference between the pressure contours of laminar and turbulent cases. The difference mostly can be observed on the surface pressure contours especially at the tip region.

An important parameter for viscous solution, skin friction coefficient over solid body surface has been computed. Surface C_f contours and C_f distributions at different spanwise locations are displayed. Generally, turbulent C_f values are higher than those of laminar case as expected. This result shows the effect of additional turbulent viscosity which is added to tye viscosity for turbulent solutions. Asit can be seen on the C_f distributions at different spanwise locations, C_f values are relatively high at the leading edge.

Boundary layer velocity profiles are displayed for all cases. Normally, a flow separation would be expected particularly in laminar cases but it is not observed because of the 3-D effects coming from the rotation of the blade. The only thing that can be critical is a small adverse flow region on the upper surface of the blade in transonic tip speed laminar case. It can be said that, as the angular velocity increases 3-D effects lose their effectivity and flow separation can be observed in laminar cases in transonic cases.

Results are generally satisfactory. But better results can be obtained with a higher memory and CPU capacity which enables the usage of finer mesh with a higher number of grid points in all directions. The modified code also can be utilizied for the aeroelastic analysis in advance.

REFERENCES

[1] Li, H.,Burggraf, O., R., Conlisk, A., T., ”Formation of a Rotor Tip Vortex “, Journal of Aircraft, 39, no. 5, pp. 739-749, 2002.

[2] Leishman, J., G., “Principles of Helicopter Aerodynamics”, University of Maryland, Cambridge University press, January 2003.

[3] Mahalingam, R., Komerath, NM., “Measurements of the Near - Wake of a Helicopter Rotor in Forward Flight”, AIAA Paper 98-0692, 36th AIAA Aerospace Sciences Meeting, January 1998, Reno, NV.

[4] Stack, J., “Experimental Investigation of Rotor Vortex Wakes in Descent”, AIAA Region VI Graduate Student Paper Competition,, January 2002.

[5] Brandt, L., “Numerical Studies of Bypass Transition in the Blasius Boundary Layer”,Technical Reports from Royal Institute of Technology, Department of Mechanics,SE-100 44 Stockholm, Sweden, April 2003.

[6] Hill, J., L.,Qin, N., “Development of a Transition Modeling Capability for Helicopter Rotor CFD”, Centre for Computational Aerodynamics, Cranfield College of Aeronautics, Cranfield University, 2000.

[7] Neel, R., E., “Advances in Computational Fluid Dynamics; Turbulent Seperated Flows and Transonic Potential Flows”, Ph.D. dissertation,Blackburg, Virginia, 1997.

[8] White, E., B.,”Transient Growth of Stationary Disturbances in a Flat Plate Boundary Layer” Department of Mechanical and Aerospace Engineering, Case Western University, Cleveland, November 2002.

[9] Cebeci, T., Cousteix, J., “Modeling and Computation of Boundary Layer Flows ; Laminar, Turbulent and Transitional Boundary Layers in Incompressible Flows”, Longbeach, California; Horizons pub;Berlin;Newyork;Springer, May 2001

[10] Bridges, D., “Introduction to Computational Fluid Dynamics with Rotorcraft Applications”, November 2000

[11] Hoffmann, K., A., Chiang, S., T., “ Computational Fluid Dynamics For Engineers”, Vol.1, Wichita, KS Engineering Education systems, 1993

[12] Hoffmann, K., A., Chiang, S., T., “ Computational Fluid Dynamics For Engineers”, Vol.2, Wichita, KS Engineering Education systems, 1993

[13] Jameson, A.,"Success and Challanges in Computational Aerpdynamics",AIAA paper 87 – 1184, AIAA 8th CFD Conference, Honolulu, Hawaii, June 1987.

[14] Anderson, D., A., Tannehill, J., C., Pletcher, R., H., "Computational Fluid Mechanics and Heat Transfer", Hemisphere Publishing Corporation,1984.

[15] Incropera, F., P., DeWitt , D., P.," Fundementals of Heat and Mass Transfer",John Wiley & Sons,1996.

[16] Hirsch, C., "Numerical Computation of Internal and External Flows,Vol.1:Fundementals of Numerical Discretization", John Wiley & Sons, 1988.

[17] Agarwal, R., K., Deese, J., E., "Euler Calculation of a Flowfield of a Helicopter Rotor in Hover", Journal of Aircraft, 24(4), pp. 231 – 238, April 1987.

[18] Sheffer, S., G., Alanso, J., J., Martinelli, L., Jameson, A.,"Time Accurate Simulation of Helicopter Rotor Flows Including Aeroelastic Effects", AIAA paper, 97 – 0339, 35th Aerospace Sciences and Exhibit, Reno, NV, January 1997.

[19] Hirsch, C., "Numerical Computation of Internal and External Flows,Vol.2", John Wiley & Sons, 1988.

[20] Vinokur, M., "An Analysis of Finite – Difference and Finite – Volume Formulations of Conservation Laws", Journal of Computational Physics 81, pp.1-52 (1989).

[21] Vatsa, V., N., Wedan, B., W., "Development of a Multigrid Code for 3 – D Navier – Stokes Equations and Its Applications to a Grid Refinement Study",Computers & Fluids, 18(4), pp,391 – 403 , May 1990.

[22] Swanson, R.,C., Turkel, E.,"Artificial Dissipation and Central Difference Schemes For the Euler and Navier – Stokes Equations", AIAA paper, 87-1107, AIAA 8th CFD Conference, Honolulu, Hawaii, June 1987.

[23] Martinelli, L.,"Calculation of Viscous Flows with Multigrid Methods", Ph.D. dissertation, MAE Department, Princeton University,1987.

[24] Ni, R.,"Multigrid Convergence Acceleration Techniques for Explicit Euler Solvers and Applications to Navier – Stokes Calculations, VKI Lecture Notes, VKI, April 2002.

[25] CFD Research Corporation, "CFD – GEOM, Interactive Geometric Modelling and Grid Genaration Software", Cummings Research Park, Version 5, October 1998.

[26] Alpman, E., “Numerical Simulation of Rotary Wing Flowfields on Parallel Computers”,M.S. Thesis, METU,Ankara, July 2001.

[27] G., Bagai,G., A.Leishman, “Experimental Study of Rotor wake/Body Interactions in Hover”, Journal of the American Helicopter society, Vol. 37, (4), October 1992

[28] Snir, M., Otto, S., W., Huss-Ledermann, S., Walker, D., W., Dongarra, J.,”MPI, The Complete Reference”, MIT press, 1996

[29] Martin, S.,”Studies of Turbulent Boundary Layer Flow Through Direct Simulation”,Technical Reports From Royal Institute of Technology, Department of Mechanics, SE-100 44 Stockholm, Sweden

APPENDIX A

MPI COMMANDS

This appendix contains the MPI commands used throughout the thesis for parallel pro cessing. These commands are specifically blocking send an receive, gather, broadcast, barrier, and abort commands. The details of these commands are available in the following sections.[29]

A.1 Blocking Send and Receive Commands

The standart mode blocking send operation is performed bye the MPI_SEND function.This function is defined as :

MPI_SEND (buf , count , datatype , dest , tag , comm)

where

buf is the addres of the send buffer

count is the number of entries to send

datatype is the datatype of each entry

dest is the rank of destination

tag is the message tag

comm is the communicator

Here all the semantics described above are input variables.

The standart blocking receive operation is performed by the MPI_RECV function, which is defined as :

MPI_RECV (buf , count , datatype , source , tag , comm , status)

where

buf is the addres of the receive buffer

count is the maximum number of entries to receive

datatype is the datatype of each entry

dest is the rank of source

tag is the message tag

comm is the communicator

status is the return status

Here the semantics buf and status are output variables while others are input variables.

A.2 Gather Command

Gathering operation is performed when the master process collects some data from the other processes. The g ather operation is performed by the MPI_GATHER function which is defined as :

MPI_GATHER (sendbuf , sendcount , sendtype , recvbuf , recvcount , recvtype , root , comm)

where

sendbuf is the starting address of the send buffer

sendcount is the number of elements in the send buffer

sendtype is datatype of the send buffer elements

recvbuf is the address of the receive buffer

recvcount is the number of elements for any single receive

recvtype is the datatype of receive buffer elements

root is the rank of receiving process

comm is the communicator

Here semantic recvbuf is output variable and the others are input variables.

A.3 Broadcast Command

Broadcast operation is performed when the master process distributes some data to the other processes. The broadcast operation is performed by the MPI_BCAST function which is defined as:

MPI_BCAST (buffer , count , datatype , root , comm)

where

buffer is the starting address of the buffer

count is the number of entries in the buffer

datatype is the datatype of the buffer

root is the rank of broadcast root

comm is the communicator

Here all semantics are input variables.

A.4 Barrier Command

MPI_BARRIER function blocks the caller until all processes have call it. It is defined as :

MPI_BARRIER (comm)

where

comm is the communicator

A.5 Abort Command

MPI_ABORT function makes the best attempt to abort all processes. It is defined as :

MPI_ABORT (comm , errorcode)

where

comm is communicator

errorcode is the error code to return to invoking environment