

# SEGMENTED DOMAIN DECOMPOSITION MULTIGRID FOR 3-D TURBOMACHINERY FLOWS

A dissertation submitted to the  
Division of Graduate Studies and Research  
of the University of Cincinnati

in partial fulfillment of the  
requirements for the degree of

DOCTOR OF PHILOSOPHY  
in the Department of Aerospace Engineering and Engineering Mechanics  
1999

by  
Mark L. Celestina

B.S.A.E, Parks College of St. Louis University, 1981  
M.S., University of Cincinnati, 1983

Committee Chair: Professor S. G. Rubin

## ABSTRACT

In the numerical simulation of flow through rotating turbomachinery, resolution of the various length scales is imperative. Mesh stretching is currently used to cluster points to resolve these length scales and reduce the required computer resources to available levels. It will be shown that large mesh stretching and high aspect ratio cells can cause instability. In fact, the convergence rate on high aspect ratio and stretched meshes is slowed as the inverse function of the aspect ratio and/or stretching parameter for exponential stretching. The alternative to the problems associated with high aspect ratio and stretched meshes is to deal only with uniform meshes.

With this in mind, the intent of this work is to develop a Segmented Domain Decomposition Multigrid (SDDMG) strategy for three-dimensional viscous problems as they apply to turbomachinery flows. The computational domain is first divided into a coarse mesh comprised of uniformly spaced cells. This is called the global parent mesh. To resolve smaller length scales such as the viscous layer near a surface, this region of the parent mesh is subdivided into a finer mesh of half the spacing of the parent mesh. This is repeated until adequate resolution of the smallest relevant length scale is obtained. Multigrid is used to communicate information between the different grid levels and solve the coarse grid equations with the accuracy of the fine grid. As regions of the mesh are refined, the solutions on the finer mesh levels converge to a grid independent solution.

To test the procedure, a number of simulations will be presented. The first case to be presented using the above procedure is channel flow. The results will be compared to a solution obtained with a conventional stretched mesh and a fine mesh solution. Flow through a compressor cascade and turbine cascade are also presented with experimental data to show some of the benefits of the method. Finally, results for a high-speed rotor typical of a modern day core compressor rotor are presented and compared to experimental data.

## ACKNOWLEDGMENTS

I would first like to thank my co-workers, Mr. Richard Mulac, Dr. Aamir Shabbir and Dr. Wai-Ming To for their very helpful insight and advice during the course of this work. Their encouragement and support helped keep me focused on completing this work.

I would also like to thank Prof. Eli Turkel, who provided useful insight into the multigrid method. Dr. Kumar Srinivasan provided me with background on how he implemented a domain decomposition procedure by discussing his strategy with me. Dr. Eric McFarland provided me with solutions from his panel method code and two boundary layer codes, BLAYER and STAN5, which were used for comparisons in the ARL Compressor Cascade section. I would also like to thank Dr. Ken Suder for his help in interpreting the comparison of my results to his experimental data for the Rotor 37 test case.

The review of this dissertation was performed primarily by my committee chair, Prof. Stan Rubin, and Dr. John Adamczyk. I would like to thank them for the time and effort they spent in helping me shape it to its final form. I would also like to thank Dr. Tony Strazisar for the perspective insights he provided through his comments. Many thanks also to my committee members, Prof. Prem Khosla and Prof. Paul Orkwis, for taking the time to read this work.

To my advisor, Prof. Stan Rubin, it has been a long journey. You have been a constant source of support through all of my years as a graduate student.

Dr. John Adamczyk deserves special thanks and I am deeply indebted to him for supporting this work from the beginning. His constant encouragement helped me bring this work to a successful conclusion.

My pride and joy are my wonderful children, Richie, Joanna, and Jonathan, who were born during the course of this work. It was a comfort knowing that you were at home waiting for me at the end of each of the many long days of working on this dissertation. I hope the sacrifices I made to complete this work will someday be a source of strength to you on your journeys through life.

My love and admiration go to my wife Joanne. Without your encouragement, patience, love, and support, I would not have considered pursuing this degree. The hardship of living without you for one year at the beginning of this journey to your struggle of caring for our children without me as I wrote this dissertation have deepened my admiration and love for you. Without you, I could not have completed this.

Finally, to my parents, <sup>†</sup>Miroslav and Mary, who sacrificed so much for their seven children. This dissertation is as much a gift from all of us as it is from me.

This dissertation is dedicated to my father  
who passed away before I was able to complete this.

# TABLE OF CONTENTS

	<u>Page</u>
Abstract . . . . .	ii
Acknowledgments . . . . .	iii
Table of Contents . . . . .	1
List of Tables . . . . .	3
List of Figures . . . . .	4
Chapter 1. Introduction . . . . .	9
1.1 Background . . . . .	9
1.2 Motivation . . . . .	11
1.3 Objective . . . . .	14
Chapter 2. Governing Equations and Discretization . . . . .	18
2.1 Governing Equations with Approximations . . . . .	18
2.2 Nondimensionalization . . . . .	21
2.3 Turbulence Model . . . . .	22
2.3 Discretization . . . . .	25
2.4 Boundary Conditions . . . . .	26
2.5 Solution Procedure . . . . .	27
2.6 Grid Generation . . . . .	31
Chapter 3. Procedure . . . . .	33
3.1 Multigrid For A Single Domain . . . . .	33
3.2 Multigrid For Sub-domains . . . . .	38
3.3 Domain Decomposition Strategy . . . . .	39
3.3.1 Multiblock Structure . . . . .	40
3.3.2 Multigrid Strategy . . . . .	41
3.3.3 Grid Generation Strategy . . . . .	42
3.3.4 Boundary Conditions . . . . .	43
3.3.5 Interface Conditions . . . . .	44
3.3.6 Implementation . . . . .	46

	<u>Page</u>
Chapter 4. Results . . . . .	57
4.1 Channel Flow . . . . .	57
4.2 ARL Compressor Cascade . . . . .	62
4.3 VKI Turbine Cascade . . . . .	66
4.4 Rotor 37 . . . . .	70
Chapter 5. Summary, Conclusions and Future Work . . . . .	114
5.1 Summary and Conclusions . . . . .	114
5.2 Future Work . . . . .	117
Bibliography . . . . .	119
Appendix A . . . . .	123
A.1 Model Equation . . . . .	123
A.2 Exponential Stretching . . . . .	126
A.3 Time Integration Scheme . . . . .	131
A.4 Numerical Results . . . . .	133
A.5 Conclusion . . . . .	135

## LIST OF TABLES

		<u>Page</u>
1.1	Blade Gridding Practice for APNASA . . . . .	10
4.1	SDDMG Mesh Refinement Schedule for ARL Compressor . . . . .	63
4.2	SDDMG Mesh Refinement Schedule for VKI Turbine. . . . .	67
4.3	SDDMG Mesh Refinement Schedule for Rotor 37 . . . . .	71
A.1	Comparison of Amplification Factor . . . . .	133
A.2	Comparison of Amplification Factor . . . . .	135



## LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1.1	Mesh of a transonic rotor at midspan. . . . .	16
1.2	The effect of AR on the convergence rate of the model problem in Appendix A. . . . .	17
1.3	The effect of stretching on the convergence rate of the model problem in Appendix A. . . . .	17
2.1	Schematic of a typical hexahedral cell. . . . .	32
3.1	Multigrid process for a single block domain using two grid levels. . . . .	48
3.2	Schematic of interior and image cells at a boundary. . . . .	49
3.3	Global parent mesh with two child meshes in computational coordinates. . . . .	50
3.4	Multigrid process for segmented domain blocks. . . . .	51
3.5	Resolution of blade surface definition with mesh refinement. . . . .	52
3.6	Alignment of child mesh interface boundaries with parent mesh grid lines. . . . .	53
3.7	Schematic of parent and child cells at an interface. . . . .	54
3.8	Schematic of a parent mesh / child mesh interface surface. . . . .	55
3.9	Schematic of SDDMG procedure. . . . .	56
4.1	Schematic of channel flow geometry. . . . .	77
4.2	Sectional view of the SDDMG mesh at midspan with 4 mesh levels for the channel geometry. . . . .	78
4.3	Sectional view of the stretched mesh at midspan for the channel geometry. Mesh size is 101 x 11 x 51. . . . .	79
4.4	Comparison of $u^+$ vs. $y^+$ for SDDMG and stretched mesh with law of the wall at 90% channel length and midspan. . . . .	80

<u>Figure</u>	<u>Page</u>
4.5	(a) Convergence history of maximum error for two mesh levels of SDDMG system and stretched mesh for channel flow. . . . . 81
4.5	(b) Convergence history of percent massflow difference for SDDMG global parent mesh and stretched mesh for channel flow. . 81
4.6	Comparison of $u^+$ vs. $y^+$ for each SDDMG mesh refinement for channel flow. . . . . 82
4.7	Schematic of the ARL Compressor Cascade. . . . . 83
4.8	Sectional view of the ARL compressor SDDMG mesh at midspan with 7 mesh levels. . . . . 83
4.9	(a) SDDMG mesh near the leading edge region of the ARL compressor at midspan. . . . . 84
4.9	(b) SDDMG mesh near the trailing edge region of the ARL compressor at midspan. . . . . 84
4.10	Comparison of Blade Surface $C_p$ vs. percent chord at midspan for experimental data and panel method. . . . . 85
4.11	Comparison of Blade Surface $C_p$ vs. percent chord at midspan for panel method and SDDMG. . . . . 85
4.12	Percent span vs. axisymmetric axial velocity at inlet and outlet. . 86
4.13	Percent span vs. axisymmetric flow angle, $\alpha$ , at outlet. . . . . 86
4.14	Contours of velocity magnitude at midspan. . . . . 87
4.15	Comparison of axial velocity, $v_z$ vs. pitch at midspan for SDDMG and experimental data. . . . . 88
4.16	Velocity vectors near the suction surface trailing edge at midspan for SDDMG. . . . . 89
4.17	Convergence of Blade Surface $C_p$ for all the SDDMG mesh levels vs. normalized distance from the leading edge at midspan. . . . . 90
4.18	Schematic of VKI Turbine Cascade. . . . . 91
4.19	Perspective view of the VKI turbine cascade geometry with 6 mesh levels outlined. . . . . 92

<u>Figure</u>		<u>Page</u>
4.20	Sectional view of the VKI turbine SDDMG mesh at midspan with 6 mesh levels. . . . .	92
4.21	(a) Two passages of the SDDMG mesh near the leading edge of the VKI turbine at midspan. . . . .	93
4.21	(b) Two passages of the SDDMG mesh near the trailing edge of the VKI turbine at midspan. . . . .	93
4.22	Axisymmetric axial momentum vs. fraction of span at the inlet and exit. . . . .	94
4.23	Mach Number contours at midspan. VKI turbine cascade. Min=0.00, Max=0.95, Inc=0.05. . . . .	95
4.24	Flow vectors near the trailing edge at midspan for the VKI turbine cascade. . . . .	96
4.25	Loss coefficient at 143% chord at midspan for the VKI turbine cascade. . . . .	97
4.26	$y^+$ versus % chord on suction surface at midspan for the VKI turbine cascade. . . . .	97
4.27	$y^+$ versus % chord on suction surface at midspan for the VKI turbine cascade. . . . .	98
4.28	$y^+$ versus % chord on suction surface at midspan for the VKI turbine cascade. . . . .	98
4.29	$v_z$ versus $\theta$ on suction surface at midspan for the VKI turbine cascade. . . . .	99
4.30	Comparison of Loss coefficient at 143% chord at midspan for the VKI turbine cascade. . . . .	99
4.31	Rotor 37. . . . .	100
4.32	Rotor 37 flowpath indicating locations of Stations 1, 3 & 4. . . . .	101
4.33	Perspective view of the grid outline for a passage of Rotor 37. . . . .	102
4.34	Sectional view of the Rotor 37 SDDMG mesh at midspan. . . . .	103

<u>Figure</u>		<u>Page</u>
4.35	(a) Two passages of the SDDMG mesh near the leading edge of the rotor at midspan. . . . .	104
4.35	(b) Two passages of the SDDMG mesh near the trailing edge of the rotor at midspan. . . . .	104
4.36	(a) Mass flow ratio vs. Pressure Ratio at design speed for Rotor 37. . . . .	105
4.36	(b) Mass flow ratio vs. Efficiency at design speed for Rotor 37. . . . .	105
4.37	(a) Mass-averaged pressure ratio vs. percent span at Station 4 for Rotor 37. . . . .	106
4.37	(b) Mass-averaged temperature ratio vs. percent span at Station 4 for Rotor 37. . . . .	106
4.37	(c) Efficiency vs. percent span at Station 4 for Rotor 37. . . . .	107
4.37	(d) Absolute flow angle, $\alpha$ , vs. percent span at Station 4 for Rotor 37. . . . .	107
4.38	Percent pitch vs. relative Mach number at midspan at Station 3 for Rotor 37. . . . .	108
4.39	Solution convergence of static pressure on mesh levels at leading edge for Rotor 37 at midspan. . . . .	108
4.40	(a) Convergence History of Maximum Error of SDDMG and single block stretched mesh for Rotor 37. . . . .	109
4.40	(b) Convergence History of Percent Massflow Difference of SDDMG and single block stretched mesh for Rotor 37. . . . .	109
4.41	Comparison of blade loading of SDDMG and single block stretched mesh at midspan for Rotor 37. . . . .	110
4.42	(a) Contours of Relative Mach Number from the SDDMG solution at midspan for Rotor 37. . . . .	111

<u>Figure</u>		<u>Page</u>
4.42	(b) Contours of Relative Mach Number from the single block stretched mesh solution at midspan for Rotor 37. . . . .	112
4.43	Comparison of wake profiles of SDDMG and single block stretched mesh at Station 3 at midspan for Rotor 37. . . . .	113
A.1	Gerschgorin disk for $N\delta \sim O(\epsilon)$ in the imaginary plane. . . . .	137
A.2	Gerschgorin disk for $N\delta \sim O(1)$ in the imaginary plane. . . . .	138
A.3	Gerschgorin disk for $N\delta \sim O(N^P)$ in the imaginary plane. . . . .	139
A.4	Variation of AR on the model problem, CFL = 2.8, $\nu = \frac{1}{32}$ , $N = 100$ . . . . .	140
A.5	Plot of $N$ vs. $\delta$ for a maximum AR of 250, 500, and 1000 for the exponential stretching function. . . . .	140
A.6	Variation of stretching on the model problem, CFL = 2.8, $\nu = \frac{1}{32}$ , $N = 100$ . . . . .	141
A.7	Variation of stretching on the model problem, CFL = 2.8, $\nu = \frac{1}{32}$ , $N = 40$ . . . . .	141

# CHAPTER 1

## INTRODUCTION

### 1.1 Background

The three-dimensional flow through a turbomachinery blade row is complex and very difficult to predict due to the wide range of length scales inherent in the problem. Adequate resolution of these length scales is imperative to obtain accurate solutions of the flow-field. A number of researchers [1-6] have developed codes that predict the time-average flow-field through a single blade row of a turbomachine. Almost all turbomachines are composed of a series of blade rows each with a different blade count. Adamczyk [7] derived a set of equations known as the average-passage system for use in simulating multistage turbomachinery flows. Adamczyk, *et al.* [8] then developed a code solving this equation set for multistage machines. Part of the procedure for solving the flow within a multistage machine requires the solution of each blade row. Thus, any improvement to each blade row solution should improve the multistage solution. The methods used by the developers of the above codes range from finite difference discretization to finite volume schemes. The solution strategies vary from implicit schemes based on the Beam-Warming algorithm to explicit schemes based on the Runge-Kutta algorithm with multigrid. Industry users of turbomachinery codes seem to favor a particular type of code, namely, the explicit, cell-centered, finite volume scheme. This type of code is the underlying solution algorithm of the current work.

One of the early developers of a three-dimensional turbomachinery viscous code was Denton [1] who approximated the viscous forces near the solid walls by body forces on an inviscid-type mesh. The codes developed by Hah [2], Dawes [4], Chima [5], and Adamczyk [8] for turbomachinery applications solved the full Navier-Stokes equations with meshes that were characterized by large stretchings and large cell aspect ratio. This was done to minimize the computer memory required to run them and obtain solutions in a reasonable amount of time.

As computer memory and speed increased, and the need to resolve more flow detail arose, researchers found that more grid cells were needed particularly to resolve the blade boundary layers. The number of mesh points increased but the grid stretching remained unchanged. Researchers were also striving to more accurately grid the known geometry details. One example of this is rotor tip clearance [5,42]. Accounting for the clearance between the rotor tip and the shroud is necessary to obtain accurate predictions of blade row performance and predict the location on the speedline where the rotor will stall. This was made self-evident by Adamczyk, *et al.* [9]. Another example of more accurate geometry detail is the leading edge and trailing edge geometry. Resolution of the leading edge geometry of a transonic rotor is important in obtaining the proper shock stand-off distance and hence, an accurate estimate of shock entropy generation. Resolution of the trailing edge geometry of transonic turbines is important in obtaining the correct shock structure and an assessment of loss mechanisms [43]. It also became evident as these codes were moved from supercomputers to desktop workstations that turbomachinery designers needed codes which could resolve the critical details of their designs and could be executed in a timely fashion so as to impact the design process.

Table 1.1 gives a historical perspective on how the gridding practice has changed in the past 12 years for one code that is accepted and widely used by U.S. engine manufacturers, APNASA. The table gives the typical number of grid points describing the blade geometry with time. The stretching range is approximate. Typically, the stretchings differ for the three directions.

Date	Mesh Type	Chord-wise Points	Span-wise Points	Pitch-wise Points	Stretching Range
1986	Inviscid	21	22	16	1.1-1.3
1989	Viscous	41	25	41	1.2-1.4
1991	Viscous	31	31	31	1.2-1.4
1993	Viscous	41	31	31	1.2-1.4
1995	Viscous	41	41	41	1.2-1.4
1997	Viscous	51	51	51	1.2-1.3

Table 1.1 Blade Gridding Practice for APNASA.

The move towards higher resolution has increased the number of grid points that describe a geometry by as much as 200% in each direction. Part of this increase can be attributed to the implementation of better turbulence models which require resolution of the near surface flow-field. Part can be attributed to more complete modeling of the geometry, e.g., fillets, clearances, and bleed ports.

Stretching the grid can create problems in obtaining converged solutions. This is particularly true if large stretchings are used. Another issue which impacts convergence of finite volume discretization formulations is associated with large aspect ratio. The aspect ratio is defined as the ratio of the largest surface area to the smallest surface area within a cell. The code developers, from experience, develop rules on the maximum amount of stretching that can be used in their grids and the users of the code must follow them in order to obtain converged solutions. This, in some applications, is not sufficient nor does it guarantee adequate resolution of flow details.

## 1.2 Motivation

The motivation for this work came from the use of grids which are less than optimum to resolve flow features which are relevant to accurately predicting the aerodynamic performance of turbomachinery blading and the impact these grids have on convergence of Runge-Kutta based solution algorithms. To illustrate this point, consider the grid shown in Figure 1.1. Such simple sheared meshes facilitate ease of numerical implementation of multistage turbomachinery flow models.

In order to resolve the leading and trailing edges, grid points are packed in the axial direction. Grid points must also be packed in the tangential direction to resolve blade boundary layers. A close-up of the mesh near the leading edge region is shown in the lower right-hand box. At mid-pitch, shown in the upper left-hand box, the stretching in the pitch-wise direction is relaxed but the stretching in the axial direction is maintained. The aspect ratio of the mesh nears 300. The effect of aspect ratios like this in regions where the flow is primarily inviscid has a



detrimental impact on the convergence rate.

The effects of mesh stretching and aspect ratio on the convergence of the two-dimensional advection equation is presented in Appendix A. The results of the analysis show that mesh stretching and high aspect ratio cell sizes have an adverse effect on the stability of the Runge-Kutta iteration scheme to solve the discretized equations. The effect of large cell aspect ratio and no stretching on convergence for this model problem is shown in Figure 1.2. As the aspect ratio is increased, the convergence rate is dramatically reduced which increases the number of iterations to attain the same residual level. A key parameter, which came out of this study, is the stretching ratio times the number of cells in the stretched direction. This term plays a role similar to the aspect ratio for uniform meshes. The effect of stretching on convergence rate for the model problem is shown in Figure 1.3. The residual history with stretching shows a decrease in convergence rate, exhibits an oscillatory behavior and is subject to round-off error. Both the aspect ratio and stretching parameter would impact any iterative scheme chosen.

Another motivation for this work is the difficulty of refining a grid in order to capture all the flow details of interest. Standard grid refinement procedure is to re-grid the geometry, interpolate the solution onto the new grid and begin the solution algorithm again. This procedure could be considered a manual version of adaptive gridding. The interpolated solution contains errors which must be eliminated by the solution algorithm. The convergence of this process is slow.

The simplest alternative to using stretched meshes illustrated in Figure 1.1 is to use a mesh with cells that are uniform. The size of the cells would be such as to resolve small length scales such as a wall boundary layer. The computer memory and speed requirements needed to carry out a simulation based on such a gridding strategy would be enormous and not feasible even with today's processor technology. An alternative approach is to use a domain decomposition method. To resolve small length scales using a domain decomposition method, a segment of the domain containing the smaller length scale is refined with a finer mesh of

half the spacing of the original mesh. To resolve even smaller length scales, a segment of the domain is further refined. A large amount of effort is currently being expended by researchers to develop this area of CFD. Some of the more well-known approaches include Adaptive Mesh Refinement (AMR) developed by Berger [19-21] in which grid patches are used to resolve flow features. Another approach is the CHIMERA method originally developed by Steger [22-23]. This approach uses overset grids which are independently generated and body fitted for simulating flow about complex configurations.

Subdividing the domain can be repeated until adequate resolution of the smallest relevant length scale is obtained. Multigrid, as suggested by Brandt [10], can be incorporated into this refinement strategy and can be used to communicate information between the different mesh levels. This resulting methodology is called Segmented Domain Decomposition Multigrid Strategy (SDDMG).

The first implementation of an SDDMG procedure was by Srinivasan and Rubin [11-13,45,46]. An adaptive approach was used to achieve optimal grid resolution for a wide range of internal flow problems using the Reduced Navier-Stokes equations (RNS) equations. This work showed that the fine grid sub-domains improve the accuracy of a solution even outside regions of refinement. This work also developed a terminology regarding the coarse and fine mesh level refinements which is adopted here.

### 1.3 Objective

The objective of this work is to develop a segmented domain decomposition multigrid strategy within the framework of a three-dimensional finite volume, cell-centered scheme which solves the Reynolds-Averaged Navier-Stokes equations for turbomachinery flows. A key objective will be the capability to efficiently generate grid independent solutions. The strategy is to use uniform meshes in the blade-to-blade direction and a stretched mesh in the span-wise direction. This is done to simplify the complexities of applying the SDDMG strategy to this class of problems.

The main elements of the current work are: a multiblock structure; a multigrid strategy; a grid generation procedure; boundary conditions; and interface conditions. Although each of these elements have previously been developed, their integration into a segmented domain decomposition multigrid (SDDMG) strategy for turbomachinery flows represents a unique contribution.

The multiblock structure is used to pass information from one grid level to the next. The multiblock formulation must have the capability to pass information from a coarse grid to a fine grid and detect whether the fine grid is a sub-domain or a global refinement of the coarse grid. It must also have the capability to pass information from a fine grid to a coarse grid or to the region of the coarse grid that the fine grid encompasses.

Multigrid is key to this multiblock formulation as it plays two roles. The first is to accelerate the convergence of the fine grid solution. The second role is to allow regions of the flow described by the coarse grid equations to be solved to the same accuracy as the regions of the flow described by the fine grid.

The grid generation procedure must be able to generate refined grids anywhere in the domain subject to certain specified constraints. The typical blade geometry found in turbomachinery is a carefully designed stack of airfoils with well-defined leading and trailing edges. Careful attention must be given to gridding these regions. In addition to the surface geometry, the sub-domain grid boundaries must

be coincident with its coarser parent mesh.

The boundary conditions at the inlet, exit and solid surfaces must be consistent with the finest grid level and thus must be developed within the framework of multigrid.

The SDDMG solution requires that the interface between a fine sub-domain grid and a coarse grid must be conservative. A conservative interface implies that the sum of the flux of mass, momenta and energy formulated from the fine grid must be identical to the result formulated from the underlying coarse grid.

The next two chapters discuss the above elements in more detail. The SDDMG method is applied to four cases. The first case is flow through a channel and shows comparisons to similar solutions with a single block stretched mesh. The next two cases are a compressor and turbine cascade for which measurements of their aerodynamic performance exists. The SDDMG solution will be compared to these measurements where applicable and further show convergence to the underlying fluid dynamic field equations as the grid is refined. The final case is a three-dimensional rotating blade row for which there is extensive experimental data. The fifth chapter presents a summary of this work with conclusions based on the results along with a discussion of future work. Appendix A presents details of the model problem used to investigate the effects of mesh stretching and aspect ratio on convergence.

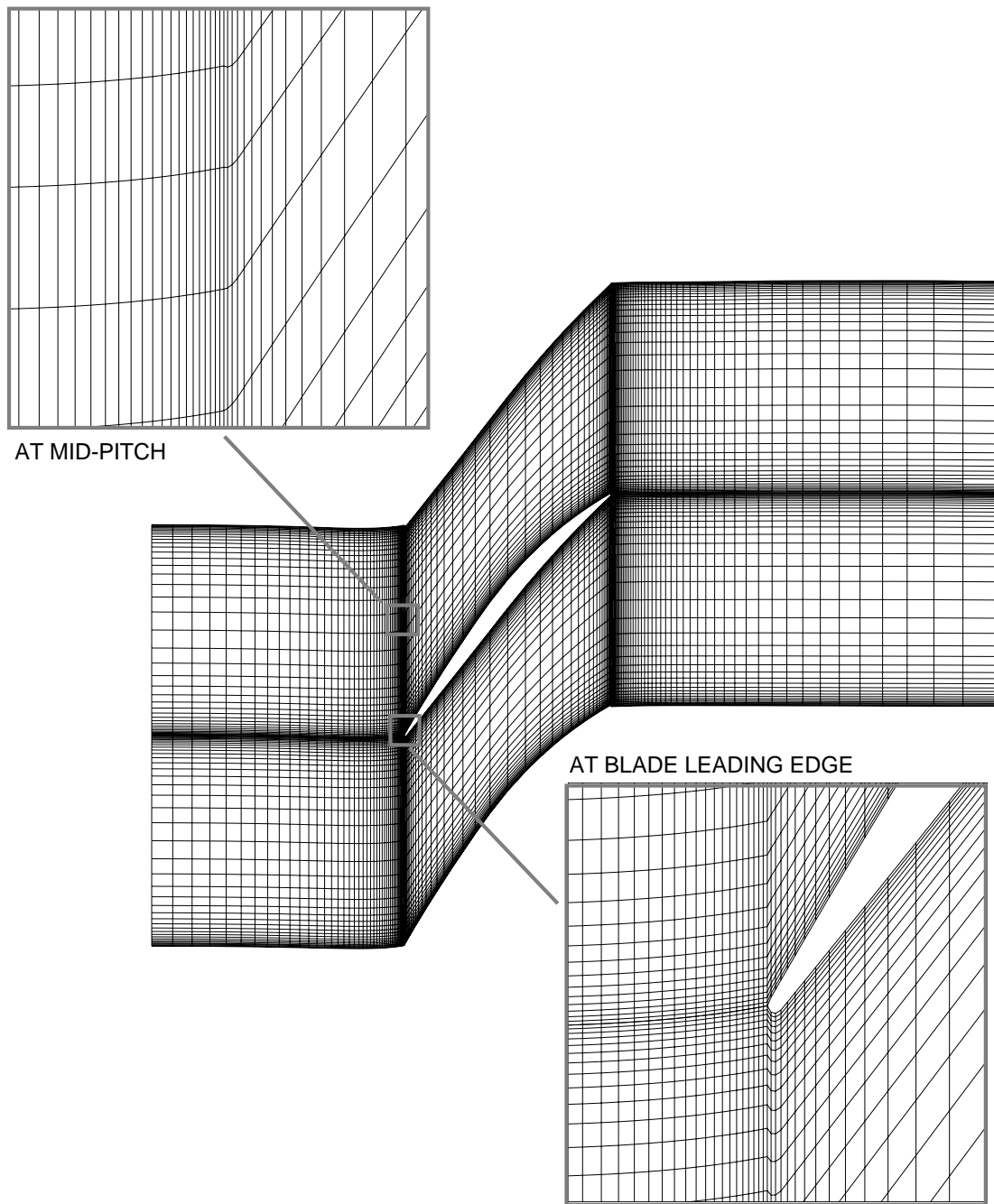


Figure 1.1. Mesh of a transonic rotor at midspan.

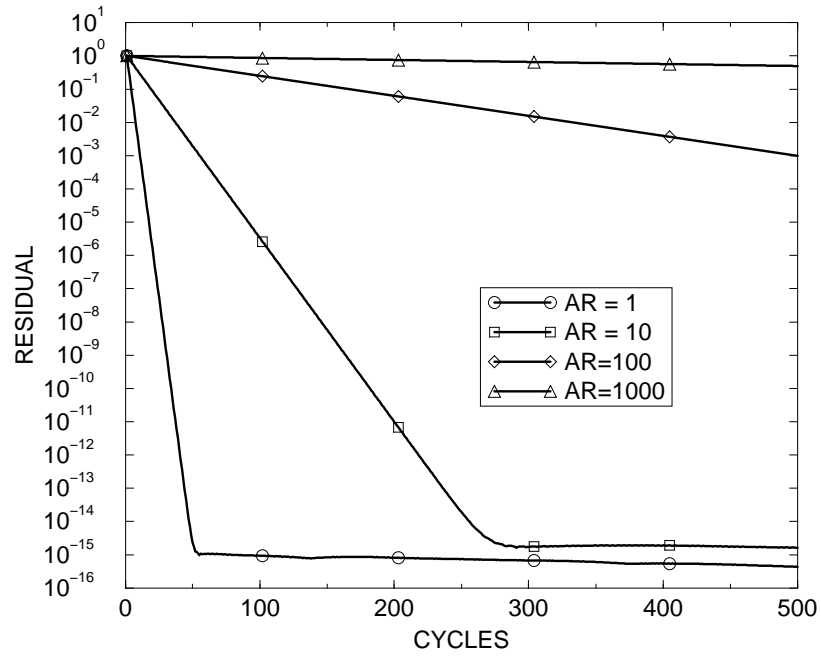


Figure 1.2. The effect of AR on the convergence rate of the model problem in Appendix A.

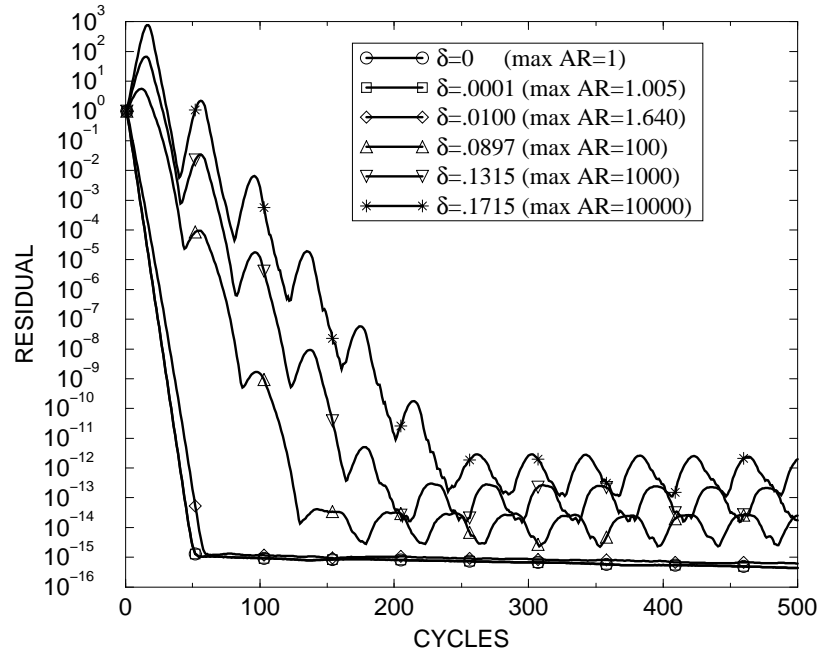


Figure 1.3. The effect of stretching on the convergence rate of the model problem in Appendix A.

## CHAPTER 2

### GOVERNING EQUATIONS AND DISCRETIZATION

#### 2.1 Governing Equations with Approximations

The Navier-Stokes equations govern the motion of a fluid and are written here in integral form for a cylindrical  $(z, r, \theta)$  coordinate system

$$\frac{\partial}{\partial t} \int_{dV} \mathbf{q} dV + \mathbf{L}(\mathbf{q}) + \int_{dV} \mathbf{S} dV = \mathbf{L}_v(\mathbf{q}) \quad (2.1)$$

The vector  $\mathbf{q}$  contains the flow variables density  $(\rho)$ , axial  $(\rho v_z)$ , radial  $(\rho v_r)$ , and angular  $(\rho v_\theta)$  momenta, and total internal energy  $(\rho e_0)$ . The term  $\mathbf{L}(\mathbf{q})$  balances the mass, momenta and energy through a control volume,  $\int \mathbf{S} dV$  is a source term due to the cylindrical coordinate system, and  $\mathbf{L}_v(\mathbf{q})$  contains the viscous and heat transfer terms. The vector  $\mathbf{q}$  and terms  $\mathbf{L}$  and  $\mathbf{L}_v$  are defined as

$$\mathbf{q} = [\rho, \rho v_z, \rho v_r, \rho v_\theta, \rho e_0]^T$$

$$\mathbf{L} = \int_{dA} \mathbf{F} dA_z + \mathbf{G} dA_r + \mathbf{H} dA_\theta \quad (2.2)$$

and

$$\mathbf{L}_v = \int_{dA} \mathbf{F}_v dA_z + \mathbf{G}_v dA_r + \mathbf{H}_v dA_\theta \quad (2.3)$$

where

$$\begin{aligned} \mathbf{F} &= [\rho v_z, \rho v_z^2 + p, \rho v_r v_z, r \rho v_z v_\theta, \rho H v_z]^T \\ \mathbf{G} &= [\rho v_r, \rho v_r v_z, \rho v_r^2 + p, r \rho v_r v_\theta, \rho H v_r]^T \\ \mathbf{H} &= [\rho v_\theta, \rho v_\theta v_z, \rho v_\theta v_r, r(\rho v_\theta^2 + p), \rho H v_\theta]^T \\ \mathbf{F}_v &= [0, \tau_{zz}, \tau_{zr}, \tau_{z\theta}, v_z \tau_{zz} + v_r \tau_{zr} + v_\theta \tau_{z\theta} + q_z^h]^T \\ \mathbf{G}_v &= [0, \tau_{rz}, \tau_{rr}, \tau_{r\theta}, v_z \tau_{rz} + v_r \tau_{rr} + v_\theta \tau_{r\theta} + q_r^h]^T \\ \mathbf{H}_v &= [0, \tau_{\theta z}, \tau_{\theta r}, \tau_{\theta\theta}, v_z \tau_{\theta z} + v_r \tau_{\theta r} + v_\theta \tau_{\theta\theta} + q_\theta^h]^T \end{aligned}$$

$$\mathbf{S} = [0, 0, \frac{\rho v_\theta^2 + p}{r} - \frac{\tau_{\theta\theta}}{r}, 0, 0]^T$$

In the above equations,  $v_z$ ,  $v_r$ , and  $v_\theta$  represent the axial, radial and tangential velocity components, respectively.  $p$  represents the pressure,  $T$  represents the temperature, and  $H$  is the enthalpy.  $\tau_{ij}$  is the stress tensor and  $q_i^h$  is the heat flux vector. The differential  $dV$  is the volume of the control cell and  $dA_z$ ,  $dA_r$ ,  $dA_\theta$  are the differential areas of its sides.

Assuming the fluid is Newtonian, the shear stresses are linearly dependent on the fluid strain rates and the components of the stress tensor can be written as

$$\begin{aligned}\tau_{zz} &= 2\mu \frac{\partial v_z}{\partial z} + \lambda \nabla \cdot \mathbf{V} \\ \tau_{zr} &= \tau_{rz} = \mu \left( \frac{\partial v_r}{\partial z} + \frac{\partial v_z}{\partial r} \right) \\ \tau_{z\theta} &= \tau_{\theta z} = \mu \left( \frac{1}{r} \frac{\partial v_z}{\partial \theta} + \frac{\partial v_\theta}{\partial z} \right) \\ \tau_{rr} &= 2\mu \frac{\partial v_r}{\partial r} + \lambda \nabla \cdot \mathbf{V} \\ \tau_{r\theta} &= \tau_{\theta r} = \mu \left( \frac{1}{r} \frac{\partial v_r}{\partial \theta} + \frac{\partial v_\theta}{\partial r} - \frac{v_\theta}{r} \right) \\ \tau_{\theta\theta} &= 2\mu \left( \frac{1}{r} \frac{\partial v_\theta}{\partial \theta} + \frac{v_r}{r} \right) + \lambda \nabla \cdot \mathbf{V}\end{aligned}\tag{2.4}$$

where  $\mu$  is the coefficient of molecular viscosity and  $\lambda$  is the coefficient of bulk viscosity. The divergence of velocity for a cylindrical coordinate system is

$$\nabla \cdot \mathbf{V} = \frac{\partial v_z}{\partial z} + \frac{\partial v_r}{\partial r} + \frac{1}{r} \frac{\partial v_\theta}{\partial \theta} + \frac{v_r}{r}\tag{2.5}$$

Stokes hypothesis assumes  $\lambda = -\frac{2}{3}\mu$ . Sutherland's law

$$\frac{\mu}{\mu_{Ref}} = \left( \frac{T}{T_{Ref}} \right)^{\frac{3}{2}} \left[ \frac{T_{Ref} + 198.6}{T + 198.6} \right]\tag{2.6}$$

is used to determine the coefficient of molecular viscosity.  $\mu_{Ref}$  is the molecular viscosity evaluated at  $T_{Ref}$ . Also assuming the fluid obeys Fourier's Law, the heat flux components are

$$\begin{aligned}q_z^h &= -k_l \frac{\partial T}{\partial z} \\ q_r^h &= -k_l \frac{\partial T}{\partial r} \\ q_\theta^h &= -k_l \frac{1}{r} \frac{\partial T}{\partial \theta}\end{aligned}\tag{2.7}$$



where  $k_l$  is the coefficient of thermal conductivity.

For a perfect gas, the equation of state,

$$p = \rho RT \quad (2.8)$$

relates the pressure to the density and temperature.  $R$  is the gas constant.

The specific heats,  $C_p$  and  $C_v$  are assumed constant and are defined as  $C_p = \frac{\gamma R}{\gamma-1}$  and  $C_v = \frac{R}{\gamma-1}$ . With these assumptions, the total internal energy is related to the pressure

$$e_0 = C_v \left( \frac{p}{\rho} \right) + \frac{1}{2} (v_z^2 + v_r^2 + v_\theta^2) \quad (2.9)$$

and the total enthalpy,  $H$ , is related to  $p$  and  $e_0$  by

$$H = e_0 + R \left( \frac{p}{\rho} \right) \quad (2.10)$$

For rotating flows, the absolute (fixed) reference frame is transformed to the relative (rotating) frame by the transformation

$$\theta_{\text{absolute}} = \theta_{\text{relative}} + \Omega t \quad (2.11)$$

where  $\Omega$  is the rotational wheel speed and  $t$  is time. Introducing equation (2.11) into equations (2.2) yields

$$L = \int_{dA} \mathbf{F} dA_z + \mathbf{G} dA_r + (\mathbf{H} - r\Omega\mathbf{q}) dA_\theta \quad (2.12)$$

The direction of rotation is oriented with increasing  $\theta$ .

## 2.2 Nondimensionalization

The governing equations can be nondimensionalized by defining a new set of variables in terms of reference values. Let

$$\begin{aligned}\bar{z}, \bar{r} &= \frac{z, r}{L_{Ref}}, & \bar{p} &= \frac{p}{P_{Ref}}, & \bar{T} &= \frac{T}{T_{Ref}}, \\ \bar{v}_z, \bar{v}_r, \bar{v}_\theta &= \frac{v_z, v_r, v_\theta}{\sqrt{RT_{Ref}}}, & \bar{\rho} &= \frac{\rho}{\frac{P_{Ref}}{RT_{Ref}}}\end{aligned}$$

The over-barred quantities are nondimensional. By replacing the dimensional quantities in the governing equations with the nondimensional variable and reference quantity and simplifying, a nondimensional set of equations results. The effect of this nondimensionalization is that the viscous operator  $L_v$  is now multiplied by the inverse of the reference Reynolds number.

$$Re_{Ref} = \frac{P_{Ref} L_{Ref}}{\sqrt{RT_{Ref}} \mu_{Ref}}$$

The heat flux vector in the energy equation can be recast in terms of a Prandtl number

$$Pr = \frac{\mu C_p}{k_l}$$

From here on, the overbar indicating a nondimensional quantity will be dropped. The viscous operator resulting from the above nondimensionalization is

$$L_v = \frac{1}{Re_{Ref}} \int_{dA} \mathbf{F}_v dA_z + \mathbf{G}_v dA_r + \mathbf{H}_v dA_\theta \quad (2.14)$$

The heat flux components can be written as

$$\begin{aligned}q_z^h &= - \frac{\mu C_p}{Pr} \frac{\partial T}{\partial z} \\ q_r^h &= - \frac{\mu C_p}{Pr} \frac{\partial T}{\partial r} \\ q_\theta^h &= - \frac{\mu C_p}{Pr} \frac{1}{r} \frac{\partial T}{\partial \theta}\end{aligned} \quad (2.15)$$

## 2.3 Turbulence Model

The flow simulations that will be presented in this work are at a Reynolds number for which the flow is considered to be fully turbulent. The focus of this work is not on the random turbulent fluctuations but on the mean flow. To solve for the mean flow quantities, the governing equations are Reynolds averaged. The resulting equations have apparent (or Reynolds) stresses that are related to local flow conditions as well as physical properties. The isotropic eddy viscosity models relate these stresses linearly to the local strain rate. This is similar to laminar flow in which a coefficient of molecular viscosity is defined. The stress tensor for the mean flow becomes

$$\begin{aligned}
\tau_{zz} &= 2(\mu + \mu_t) \frac{\partial v_z}{\partial z} + \lambda \nabla \cdot \mathbf{V} - \frac{2}{3} \rho \mathbf{k} \\
\tau_{zr} &= \tau_{rz} = (\mu + \mu_t) \left( \frac{\partial v_r}{\partial z} + \frac{\partial v_z}{\partial r} \right) \\
\tau_{z\theta} &= \tau_{\theta z} = (\mu + \mu_t) \left( \frac{1}{r} \frac{\partial v_z}{\partial \theta} + \frac{\partial v_\theta}{\partial z} \right) \\
\tau_{rr} &= 2(\mu + \mu_t) \frac{\partial v_r}{\partial r} + \lambda \nabla \cdot \mathbf{V} - \frac{2}{3} \rho \mathbf{k} \\
\tau_{r\theta} &= \tau_{\theta r} = (\mu + \mu_t) \left( \frac{1}{r} \frac{\partial v_r}{\partial \theta} + \frac{\partial v_\theta}{\partial r} - \frac{v_\theta}{r} \right) \\
\tau_{\theta\theta} &= 2(\mu + \mu_t) \left( \frac{1}{r} \frac{\partial v_\theta}{\partial \theta} + \frac{v_r}{r} \right) + \lambda \nabla \cdot \mathbf{V} - \frac{2}{3} \rho \mathbf{k}
\end{aligned} \tag{2.17}$$

Note that the turbulent (or eddy) viscosity is not constant and must be obtained from a turbulence model.

Reynolds analogy relates the turbulent heat flux to the temperature gradient in the same way as laminar flow. Defining a turbulent Prandtl number

$$Pr_t = \frac{\mu_t C_p}{k_t}$$

and using Reynolds analogy, the heat flux components for turbulent flows can be

written as

$$\begin{aligned}
q_z^h &= - \left( \frac{\mu C_p}{Pr} + \frac{\mu_t C_p}{Pr_t} \right) \frac{\partial T}{\partial z} \\
q_r^h &= - \left( \frac{\mu C_p}{Pr} + \frac{\mu_t C_p}{Pr_t} \right) \frac{\partial T}{\partial r} \\
q_\theta^h &= - \left( \frac{\mu C_p}{Pr} + \frac{\mu_t C_p}{Pr_t} \right) \frac{1}{r} \frac{\partial T}{\partial \theta}
\end{aligned} \tag{2.18}$$

The present work uses the two-equation standard  $k - \epsilon$  model of Launder and Spalding [14] to estimate the eddy viscosity. This is a high Reynolds number model which requires the use of wall functions at solid surfaces and is valid from the outer region of a turbulent boundary layer to the beginning of the log-layer. The eddy viscosity is computed from the following equation,

$$\mu_t = C_\mu \rho \frac{k^2}{\epsilon} \tag{2.19}$$

The term  $k$  is the turbulence kinetic energy and  $\epsilon$  is the dissipation rate of turbulence kinetic energy.  $C_\mu$  is a constant and  $\rho$  is the density. The integrated equations for  $k$  and  $\epsilon$  for a control volume are

$$\frac{\partial}{\partial t} \int_{dV} \mathbf{q}_{k-\epsilon} dV + \mathbf{L}_{k-\epsilon}(\mathbf{q}_{k-\epsilon}) + \int_{dV} \mathbf{S}_{k-\epsilon} dV = \mathbf{L}_{vk-\epsilon}(\mathbf{q}_{k-\epsilon}) \tag{2.20}$$

The vector  $\mathbf{q}_{k-\epsilon}$  contains the variables  $\rho k$  and  $\rho \epsilon$ . The term  $\mathbf{L}_{k-\epsilon}(\mathbf{q}_{k-\epsilon})$  is the convective term,  $\int \mathbf{S}_{k-\epsilon} dV$  is a source term and  $\mathbf{L}_v(\mathbf{q})$  is the viscous term. The vector  $\mathbf{q}_{k-\epsilon}$  and terms  $\mathbf{L}_{k-\epsilon}$  and  $\mathbf{L}_{vk-\epsilon}$  are defined as

$$\mathbf{q}_{k-\epsilon} = [\rho k, \rho \epsilon]^T$$

$$\mathbf{L} = \int_{dA} \mathbf{F}_{k-\epsilon} dA_z + \mathbf{G}_{k-\epsilon} dA_r + \mathbf{H}_{k-\epsilon} dA_\theta \tag{2.21}$$

and

$$\mathbf{L}_v = \int_{dA} \mathbf{F}_{vk-\epsilon} dA_z + \mathbf{G}_{vk-\epsilon} dA_r + \mathbf{H}_{vk-\epsilon} dA_\theta \tag{2.22}$$

where

$$\mathbf{F}_{k-\epsilon} = [\rho k v_z, \rho \epsilon v_z]^T$$

$$\mathbf{G}_{k-\epsilon} = [\rho k v_r, \rho \epsilon v_r]^T$$

$$\mathbf{H}_{k-\epsilon} = [\rho k v_\theta, \rho \epsilon v_\theta]^T$$

$$\begin{aligned}
\mathbf{F}_{vk-\epsilon} &= \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial z}, \left( \mu + \frac{\mu_t}{\sigma_\epsilon} \right) \frac{\partial \epsilon}{\partial z} \right]^T \\
\mathbf{G}_{vk-\epsilon} &= \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial r}, \left( \mu + \frac{\mu_t}{\sigma_\epsilon} \right) \frac{\partial \epsilon}{\partial r} \right]^T \\
\mathbf{H}_{vk-\epsilon} &= \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \frac{1}{r} \frac{\partial k}{\partial \theta}, \left( \mu + \frac{\mu_t}{\sigma_\epsilon} \right) \frac{1}{r} \frac{\partial \epsilon}{\partial \theta} \right]^T \\
\mathbf{S} &= \left[ (P_k - \rho\epsilon), \frac{\epsilon}{k} (C_{\epsilon 1} P_k - C_{\epsilon 2} \rho\epsilon) \right]^T
\end{aligned}$$

The term  $P_k$  is called the production of turbulence kinetic energy and is given as

$$\begin{aligned}
P_k &= \tau_{zz} \frac{\partial v_z}{\partial z} + \tau_{zr} \frac{\partial v_z}{\partial r} + \tau_{z\theta} \left[ \frac{1}{r} \frac{\partial v_z}{\partial \theta} \right] \\
&\quad + \tau_{rz} \frac{\partial v_r}{\partial z} + \tau_{rr} \frac{\partial v_r}{\partial r} + \tau_{r\theta} \left[ \frac{1}{r} \frac{\partial v_r}{\partial \theta} - \frac{v_\theta}{r} \right] \\
&\quad + \tau_{\theta z} \frac{\partial v_\theta}{\partial z} + \tau_{\theta r} \frac{\partial v_\theta}{\partial r} + \tau_{\theta\theta} \left[ \frac{1}{r} \frac{\partial v_\theta}{\partial \theta} + \frac{v_r}{r} \right]
\end{aligned} \tag{2.23}$$

where

$$\begin{aligned}
\tau_{zz} &= 2\mu_t \left( \frac{\partial v_z}{\partial z} + \frac{1}{3} (\nabla \cdot \mathbf{V}) \right) - \frac{2}{3} \rho k \\
\tau_{zr} &= \mu_t \left( \frac{\partial v_z}{\partial r} + \frac{\partial v_r}{\partial z} \right) \\
\tau_{z\theta} &= \mu_t \left( \frac{1}{r} \frac{\partial v_z}{\partial \theta} + \frac{\partial v_\theta}{\partial z} \right) \\
\tau_{rr} &= 2\mu_t \left( \frac{\partial v_r}{\partial r} + \frac{1}{3} (\nabla \cdot \mathbf{V}) \right) - \frac{2}{3} \rho k \\
\tau_{r\theta} &= \mu_t \left( \left[ \frac{1}{r} \frac{\partial v_r}{\partial \theta} - \frac{v_\theta}{r} \right] + \frac{\partial v_\theta}{\partial r} \right) \\
\tau_{\theta\theta} &= 2\mu_t \left( \left[ \frac{1}{r} \frac{\partial v_\theta}{\partial \theta} + \frac{v_r}{r} \right] + \frac{1}{3} (\nabla \cdot \mathbf{V}) \right) - \frac{2}{3} \rho k
\end{aligned}$$

The values of the model constants are found in the original Launder and Spalding paper. They are given here as:  $\sigma_k = 1.0$ ;  $\sigma_\epsilon = 1.3$ ;  $C_{\epsilon 1} = 1.44$ ;  $C_{\epsilon 2} = 1.92$ ; and  $C_\mu = 0.09$ . The boundary conditions on  $k$  and  $\epsilon$  will be presented in Section 2.5 along with the boundary conditions of the flow solver.

## 2.4 Discretization

The integral equation (2.1) is discretized using a finite volume approach. First, a computational domain is defined and subdivided into hexahedral cells. The surface integrals in  $L$  and  $L_v$  are approximated by the midpoint rule. Uncoupling the time derivative results in a system of ordinary differential equations of the form

$$\frac{d}{dt} \int_{dV} \mathbf{q} dV + \mathbf{L}(\mathbf{q}) + \int \mathbf{S} dV = \mathbf{L}_v(\mathbf{q}) \quad (2.24)$$

The integral in the inviscid term  $\mathbf{L}$  is replaced by a summation and the discrete form of  $\mathbf{L}$  becomes

$$\mathbf{L} = \sum \mathbf{F} dA_z + \mathbf{G} dA_r + (\mathbf{H} - r\Omega\mathbf{q}) dA_\theta \quad (2.25)$$

This states that the mass, momenta and energy flux through each cell face is summed over that cell. Figure 2.1 shows a schematic of a typical hexahedral cell in the computational domain. The surfaces of the cell are projected outward from the volume and have been shaded. The indices  $i$ ,  $j$ , and  $k$  are used to indicate the location of the cell in computational space. The  $i$  index indicates the  $i^{th}$  axial cell, the  $j$  index indicates the  $j^{th}$  radial cell, and the  $k$  index indicates the  $k^{th}$  tangential or pitch-wise cell. The surface areas,  $dA_z$ ,  $dA_r$ , and  $dA_\theta$  are calculated using the cross product of the diagonals of each face. The volume of the cell is determined using the formula described by Holmes and Tong [15]. Their formula sums a scalar obtained from each face to determine the volume of the cell. The scalars are obtained by taking the cross product of the two diagonals of each face; dot this with the vector from any point to the center of the face (defined as the average of the four corner points).

The integral in the viscous operator  $\mathbf{L}_v$  is replaced by a summation and the discrete form of  $\mathbf{L}_v$  becomes

$$\mathbf{L}_v = \sum \mathbf{F}_v dA_z + \mathbf{G}_v dA_r + \mathbf{H}_v dA_\theta \quad (2.26)$$

The viscous and heat transfer terms are discretized by evaluating the shear stress and heat flux at the center of each cell face and summing over that cell. Derivatives

necessary to compute the stresses and flux at each cell face are estimated using central differences of the velocity and temperature field [8].

The  $k$  and  $\epsilon$  equations are discretized in the same way as the governing equations with the exception of the advection term. This term was approximated using a first order upwind difference scheme. The details of the discretization can be found in Shabbir, *et al.* [16].

## 2.5 Boundary Conditions

Since the inlet flow to the computational domain is assumed to be subsonic, four quantities need to be specified for the mean flow. Total pressure, total temperature, and radial and tangential flow angle profiles are specified. These are obtained from experiment or by design conditions. From the interior of the flow domain, the upstream running one-dimensional Riemann invariant  $C^-$  defined as

$$C^- = v_z - \frac{2a}{\gamma - 1} \quad (2.27)$$

is computed and extrapolated to the boundary. Here  $a$  is the local speed of sound. From these five quantities, the static pressure, velocity components and temperature at the inlet flow boundary can be established. The turbulence model requires that  $k$  and  $\epsilon$  be specified at the inlet. A turbulence intensity is assumed and a length scale  $l$  which is taken as some fraction of the characteristic length scale of the geometry is specified.

The flow exiting the computational domain is assumed to be subsonic. When the exiting flow is subsonic, only one boundary condition may be specified. For this work the pressure is set. For a three-dimensional flow, the one-dimensional Riemann invariant  $C^+$  defined as

$$C^+ = v_z + \frac{2a}{\gamma - 1}, \quad (2.28)$$

entropy,  $v_\theta$ , and  $v_r$  are extrapolated from the interior to the exit boundary. Along with the specified pressure and the extrapolated variables, the flow variables exiting

the computational domain can be established. The pressure distribution can be specified either from experiment or design condition or the exit hub static pressure can be fixed and the radial equilibrium equation integrated using the tangential velocity profile extrapolated from the interior. The gradients of  $k$  and  $\epsilon$  in the turbulence model with respect to the flow direction are set to zero at the exit.

Solid surfaces in the computational domain are modeled as no-slip and are assumed to be adiabatic. The values of  $k$  and  $\epsilon$  in the turbulence model are specified in the first cell away from the wall from the following relations

$$k = \frac{u_\tau^2}{\sqrt{C_\mu}}, \quad \epsilon = \frac{u_\tau^3}{\kappa d}$$

where  $C_\mu = 0.09$ ,  $\kappa = 0.41$ , and  $u_\tau = \sqrt{\tau_w/\rho}$  is the friction velocity.  $d$  is the distance from the wall.

## 2.6 Solution Procedure

To integrate the ODE that results from the spatially discretized Reynolds-averaged Navier Stokes equations, an explicit 4-stage Runge-Kutta algorithm is employed. Letting  $q^n$  be a solution at time  $t$  and advancing the equations to time level  $n + 1$ ,

$$\begin{aligned} q^{n+a_1} &= q^n - \frac{a_1 \Delta t}{V} (-L(q^n) + L_v(q^n) + D(q^n)) + a_1 \Delta t (S(q^n)) \\ q^{n+a_2} &= q^n - \frac{a_2 \Delta t}{V} (-L(q^{n+a_1}) + L_v(q^n) + D(q^n)) + a_2 \Delta t (S(q^n)) \\ q^{n+a_3} &= q^n - \frac{a_3 \Delta t}{V} (-L(q^{n+a_2}) + L_v(q^n) + D(q^n)) + a_3 \Delta t (S(q^n)) \\ q^{n+a_4} &= q^n - \frac{a_4 \Delta t}{V} (-L(q^{n+a_3}) + L_v(q^n) + D(q^n)) + a_4 \Delta t (S(q^n)) \end{aligned} \tag{2.29}$$

where  $a_1 = \frac{1}{4}$ ,  $a_2 = \frac{1}{3}$ ,  $a_3 = \frac{1}{2}$ , and  $a_4 = 1$ .  $V$  is the volume of the cell and  $\Delta t$  is the timestep.

$\Delta t$  is the maximum allowable timestep and is defined here as a combination of the maximum timestep in each grid direction,

$$\Delta t = \text{CFL} \frac{\Delta t_\xi \Delta t_\eta \Delta t_\zeta}{\Delta t_\xi \Delta t_\eta + \Delta t_\eta \Delta t_\zeta + \Delta t_\zeta \Delta t_\xi} \tag{2.30}$$



where  $\xi$  is the axial grid direction,  $\eta$  is the radial grid direction, and  $\zeta$  is the pitch-wise grid direction.

The directional timestep is the minimum of the convective and viscous timestep. As an example, the directional timestep in the  $\xi$  direction will be given. The timesteps in the other two directions are similarly defined.

$$\Delta t_\xi = \min(\Delta t_\xi^{inv}, \Delta t_\xi^{vis})$$

The convective timestep,  $\Delta t_\xi^{inv}$ , is a function of the cell volume and maximum local wave speed:

$$\Delta t_\xi^{inv} = \frac{V}{\lambda_\xi} \quad (2.31)$$

where

$$\begin{aligned} \lambda_\xi &= q_s + a\sqrt{S_s} \\ q_s &= v_z dA_{z\xi} + v_r dA_{r\xi} + (v_\theta - r\Omega) dA_{\theta\xi} \\ S_s &= dA_{z\xi}^2 + dA_{r\xi}^2 + dA_{\theta\xi}^2 \end{aligned}$$

Here,  $a$  is the local speed of sound,  $dA_{(z,r,\theta)\xi}$  is the projected area of the  $\xi$  grid plane of the cell in the  $(z, r, \theta)$  directions.  $S_s$  is the sum of the squares of those projected areas.

The viscous timestep,  $\Delta t_\xi^{vis}$ , is based on a stability analysis of the diffusion equation which states that the cell Reynolds number cannot exceed 2. Extending this to three-dimensional cells, the viscous timestep can be written as

$$\Delta t_\xi^{vis} = \frac{2Re_{Ref}V^2}{(\nu_l + \nu_t)S_s} \quad (2.32)$$

Operator  $D$  in Eqn. 2.29 adds artificial dissipation to the scheme to suppress odd-even point decoupling of the solution within inviscid regions of the flow. It is a blend of second and fourth difference smoothing operators patterned after the model developed by Jameson *et al.* [17]. It is composed of three spatial operators

$$D(q) = (D_\xi + D_\eta + D_\zeta)(q) \quad (2.33)$$

that are evaluated separately in computational space. The dissipation in the  $\xi$  direction  $D_\xi$  is expressed as

$$D_\xi(q_{i,j,k}) = d_{i-\frac{1}{2},j,k} - d_{i+\frac{1}{2},j,k} \quad (2.34)$$

where

$$d_{i+\frac{1}{2},j,k} = \frac{V_{i+\frac{1}{2},j,k}}{\Delta t_{\xi_{i+\frac{1}{2},j,k}}} [\epsilon_{i+\frac{1}{2},j,k}^{(2)} \delta_\xi^{(1)}(q_{i,j,k}) - \epsilon_{i+\frac{1}{2},j,k}^{(4)} \delta_\xi^{(2)}(\delta_\xi^{(1)}(q_{i,j,k}))]$$

and the indices,  $i$ ,  $j$ , and  $k$  indicate the location of the cell in computational space. The index  $i + \frac{1}{2}$  is a quantity located at the back or rearward surface of the cell shown in Figure 2.1 in the axial direction. The index  $i - \frac{1}{2}$  is a quantity at the front or forward surface of the cell shown in Figure 2.1 in the axial direction. The same holds true for the half indices in the  $j$  and  $k$  directions.

The coefficients  $\epsilon^{(2)}$  and  $\epsilon^{(4)}$  are evaluated as follows:

$$\begin{aligned} \epsilon_{i+\frac{1}{2},j,k}^{(2)} &= \kappa^{(2)} \max(\nu_{i+1,j,k}, \nu_{i,j,k}) \\ \epsilon_{i+\frac{1}{2},j,k}^{(4)} &= \max(0, \kappa^{(4)} - \epsilon_{i+\frac{1}{2},j,k}^{(2)}) \end{aligned}$$

$\kappa^{(2)}$  and  $\kappa^{(4)}$  are constants set at  $\frac{1}{2}$  and  $\frac{1}{64}$ , respectively. The function  $\nu_{i,j,k}$  is a pressure sensitive switch that is used primarily to prevent oscillations near stagnation points and shocks. It is defined as

$$\nu_{i,j,k} = \left| \frac{p_{i+1,j,k} - 2p_{i,j,k} + p_{i-1,j,k}}{p_{i,j,k} \min(4, \frac{p_{i+1,j,k} + 2p_{i,j,k} + p_{i-1,j,k}}{p_{i,j,k}})} \right| \quad (2.35)$$

The symbols  $\delta_\xi^{(1)}$  and  $\delta_\xi^{(2)}$  are the central difference approximations of a first and second difference in the  $\xi$  direction. The above operations in the  $\xi$  direction is also performed in the  $\eta$  and  $\zeta$  directions.

To ensure conservation of the artificial viscosity operator,  $D_\xi$ ,  $D_\eta$ , and  $D_\zeta$  are summed over their respective grid indices and set to 0,

$$\begin{aligned} \sum_{i=2}^{i=il} D_\xi &= d_{\frac{1}{2},j,k} - d_{il+\frac{1}{2},j,k} = 0 \\ \sum_{j=2}^{j=jl} D_\eta &= d_{i,\frac{1}{2},k} - d_{i,jl+\frac{1}{2},k} = 0 \\ \sum_{k=2}^{k=kl} D_\zeta &= d_{i,j,\frac{1}{2}} - d_{i,j,kl+\frac{1}{2}} = 0 \end{aligned}$$

where  $il$ ,  $jl$ , and  $kl$  are the total number of axial, radial and pitch-wise cells, respectively. This yields the conditions that must be set on the artificial dissipation at the boundaries. At the inlet, exit and solid boundaries,

$$d_{\frac{1}{2},j,k} = d_{il+\frac{1}{2},j,k} = 0$$

$$d_{i,\frac{1}{2},k} = d_{i,jl+\frac{1}{2},k} = 0$$

$$d_{i,j,\frac{1}{2}} = d_{i,j,kl+\frac{1}{2}} = 0$$

and for periodic boundaries,

$$d_{i,j,\frac{1}{2}} = d_{i,j,kl+\frac{1}{2}}$$

To prevent the artificial dissipation from affecting the near wall solution, the distance from all solid surfaces defined in terms of wall units,  $y^+ = \frac{l\sqrt{\frac{\tau_w}{\rho}}}{\nu}$  with  $l$  being the physical distance, was stored and used to shut-off the dissipation at a predetermined value. This value was typically set to 1000. The dissipation cut-off was done in all three directions.

The maximum permissible time step for this scheme is restricted by the CFL stability limit and has been determined to be  $2\sqrt{2}$  [18]. To relax this limit and enhance the convergence rate of this scheme, two additional procedures are employed: local time-stepping and implicit residual smoothing. The technique of local time-stepping uses the largest time step calculated for each cell to advance the discrete equations in time as opposed to the minimum value of the time step associated with all the cells in the domain. Implicit residual smoothing applies the following

$$(I - \epsilon_\zeta \delta_\zeta^{(2)})(I - \epsilon_\eta \delta_\eta^{(2)})(I - \epsilon_\xi \delta_\xi^{(2)})\bar{R}_{i,j,k} = R_{i,j,k} \quad (2.36)$$

after the 2nd ( $m = 2$ ) and 4th ( $m = 4$ ) stages to the residual defined as

$$R_{i,j,k} = \frac{a_m \Delta t_{i,j,k}}{V_{i,j,k}} (-L(q_{i,j,k}^{n+a_m}) - S(q_{i,j,k}^{n+a_m}) + L_v(q_{i,j,k}^n) + D(q_{i,j,k}^n))$$

This operation is applied sequentially and is reduced to solving a set of tridiagonal equations using a Thomas algorithm. The coefficients  $\epsilon_\zeta$ ,  $\epsilon_\eta$ , and  $\epsilon_\xi$  are constants.

The  $k - \epsilon$  equations are uncoupled from the main flow solver and are solved using an ADI scheme. The turbulence viscosity  $\mu_t$  is updated every 5 iterations based on the updated values of  $k$  and  $\epsilon$ . Details can be found in Shabbir [16].

## 2.7 Grid Generation

A mesh generation code developed by Mulac [28] generated all of the meshes used in this work. The code is a general purpose turbomachinery mesh generator which generates a three-dimensional H-mesh by applying a series of cubic spline fits to the geometry data of the flowpath and blade. To generate a mesh, the code requires both the number of points desired and an exponential stretching factor in the axial, radial, and pitch-wise directions relative to the blade surface. Additionally, an exponential stretching factor is input for both the leading edge to inlet and trailing edge to outlet regions. If the exponential stretching factor is unity, the code yields a uniform distribution of points in that direction.

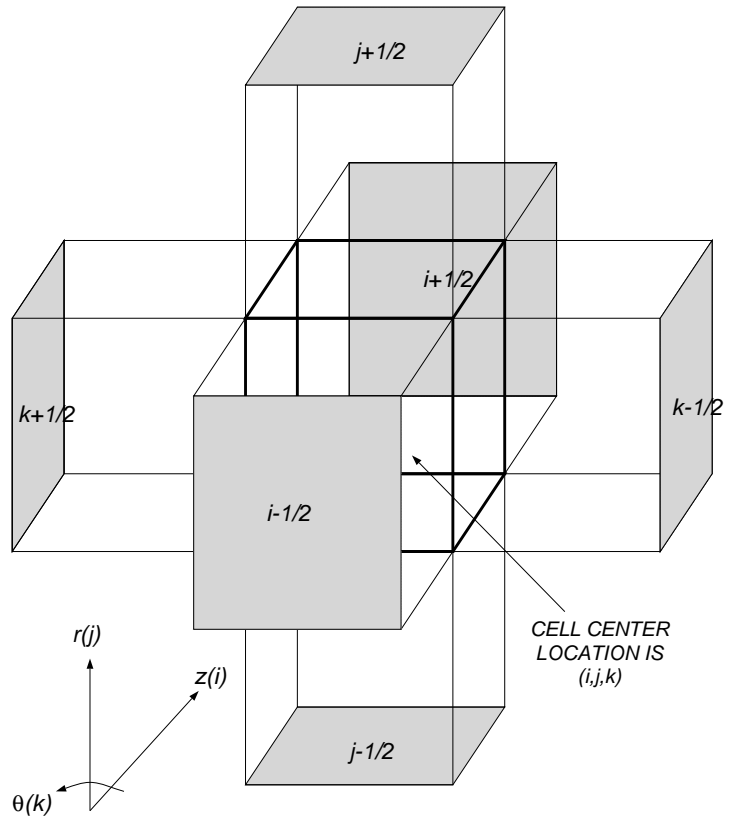


Figure 2.1. Schematic of a typical hexahedral cell.

## CHAPTER 3

### PROCEDURE

Two simple methods, local time-stepping and implicit residual smoothing, have already been presented for accelerating the convergence of the iterative scheme. Both dealt with increasing the stability limit of the underlying Runge-Kutta based iterative scheme on a given grid. A multigrid strategy is now presented to further accelerate the convergence rate.

#### 3.1 Multigrid for A Single Domain

Multigrid assumes that the iterative scheme rapidly reduces or smooths the high wave number error on a given grid. However, quite often, the iterative scheme is not effective in reducing the error associated with the low wave number error [10,26,27]. The multigrid process, therefore, works on smoothing this low wave number error with a hierarchy of coarse grids. The low wave number error is transferred onto a coarser grid where parts of the low wave number error are now high wave number error. The iterative scheme is then applied on the coarse grid and smooths this high wave number error. This is repeated on successively coarser grids. When the error is smoothed on the coarsest grid, the corrections resulting from the coarse grids are then all combined and brought back to the finest grid.

There are four steps to a multigrid process. The first step is the iterative scheme or smoother on the fine grid. For this work, the four-stage Runge Kutta integration scheme discussed earlier is used to obtain a solution on any grid level. The second step is the restriction operation which transfers information from a fine grid to a coarser grid. The third step is smoothing the coarse grid error and the fourth step is the prolongation operation which interpolates the coarse grid corrections onto the fine grid. This process is illustrated in Figure 3.1 using two grid levels. The restriction operation passes information to the coarse grid. After smoothing the coarse grid error, the prolongation operation brings the coarse grid corrections back

to the fine grid. This is referred to in the literature [26] as the V-cycle. With regard to the third step, the accuracy of the equations on the coarse grid must be consistent with the accuracy of the fine grid equations. The following section outlines the development of the coarse grid equations.

Let the problem on the finest mesh level  $N$  be represented by

$$L^N Q = S^N \quad (3.1)$$

where  $L^N$  represents a discrete spatial operator.  $Q$  is the exact discrete solution at convergence and  $S^N$  is a source term. The error is defined as  $e^N \equiv Q - q^N$  where  $q^N$  is the approximate discrete solution. The residual function is defined as

$$\begin{aligned} R^N &\equiv L^N Q - L^N q^N \\ &= S^N - L^N q^N \end{aligned}$$

At convergence,  $e^N$  approaches 0 and  $R^N$  approaches 0. On the  $N-1$  grid level, the error function is  $e^{N-1} = Q^{N-1} - I_N^{N-1} q^N$  and the residual satisfies  $R^{N-1} = \tilde{I}_N^{N-1} R^N$  where  $I_N^{N-1}$  and  $\tilde{I}_N^{N-1}$  are both restriction operators.

Two restriction operators need to be defined: one to restrict the fine grid solution  $I_N^{N-1}(q^N)$ ; and one to restrict the fine grid operator  $\tilde{I}_N^{N-1}(L^N(q^N))$ . The restriction operator for the solution is volume weighted and is defined as

$$I_N^{N-1} q^N = \frac{\sum q^N V}{\sum V} \quad (3.2)$$

where the sum is taken over the fine grid cells that define the coarse grid cell. The restriction operator for  $L$  is defined as

$$\tilde{I}_N^{N-1} L^N = \sum L^N \quad (3.3)$$

where the sum is again taken over the fine grid cells that define the coarse grid.

The residual can be expanded and written as

$$R^{N-1} \equiv L^{N-1} Q^{N-1} - L^{N-1} q^{N-1} = \tilde{I}_N^{N-1} R^N \quad (3.4)$$

Substituting equation (3.1) into equation (3.4) and rearranging, the coarse grid equation consistent with the fine grid problem is

$$L^{N-1}Q^{N-1} = \tilde{I}_N^{N-1}S^N + [L^{N-1}I_N^{N-1}q^N - \tilde{I}_N^{N-1}L^Nq^N]$$

$Q^{N-1}$  is approximated by  $q^{N-1}$  and therefore

$$L^{N-1}q^{N-1} = \tilde{I}_N^{N-1}S^N + [L^{N-1}I_N^{N-1}q^N - \tilde{I}_N^{N-1}L^Nq^N] \quad (3.5)$$

Thus, the source term,  $S^N$  and spatial operator,  $L^N$ , from grid level  $N$  are restricted to grid level  $N - 1$  along with the solution  $q^N$ .

The effect of the forcing term within the context of a multigrid strategy can be understood more clearly by examining the relationship between the pressure and the solution vector on the fine and coarse grid. The pressure,  $p$ , on the fine grid,  $N$ , is simply a function of the vector,  $\mathbf{q}$ , on the fine grid,

$$p^N = f(\mathbf{q}^N). \quad (3.6)$$

However, on the coarse grid,  $N - 1$ , the pressure is not only a function of the coarse grid vector,  $\mathbf{q}^{N-1}$  but also a function of the pressure,  $p^N$ , and the vector,  $\mathbf{q}^N$ , of the fine grid,

$$p^{N-1} = f(\mathbf{q}^{N-1}) + [I_N^{N-1}p^N - f(I_N^{N-1}\mathbf{q}^N)] \quad (3.7)$$

The terms  $I_N^{N-1}p^N$  and  $f(I_N^{N-1}\mathbf{q}^N)$  are nonlinear and do not cancel.

With a new coarse grid solution,  $q_{new}^{N-1}$ , the fine grid solution,  $q_{new}^N$ , can be improved by applying

$$q_{new}^N \leftarrow q_{old}^N + I_{N-1}^N(q_{new}^{N-1} - I_N^{N-1}q_{old}^N) \quad (3.8)$$

where  $I_{N-1}^N$  is the prolongation operator and transfers information from grid level  $N - 1$  to  $N$ . The prolongation operator is a bilinear interpolator.

To determine the boundary conditions on the coarse grid, the approach used by Jentink and Usab [27], developed for two dimensions, will be adopted and extended



to three dimensions. They began by defining a general spatial filter operator. In three dimensions, this operator has the form

$$\begin{aligned}\mathcal{L}(f) &= \bar{f}(\xi, \eta, \zeta) \\ &= \frac{1}{\lambda} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\bar{\xi}, \bar{\eta}, \bar{\zeta}) H(\bar{\xi}, \bar{\eta}, \bar{\zeta}) W(\bar{\xi} - \xi, \bar{\eta} - \eta, \bar{\zeta} - \zeta) d\bar{\xi} d\bar{\eta} d\bar{\zeta}\end{aligned}\quad (3.9)$$

where  $H$  is the Heavyside function defined as

$$H(\xi, \eta, \zeta) = \begin{cases} 1, & \text{inside the flow domain;} \\ 0, & \text{outside the flow domain.} \end{cases}\quad (3.10)$$

$W(\xi, \eta, \zeta)$  is a weighting function defined as having unit influence over the eight fine mesh cells surrounding a given mesh point and zero influence elsewhere,

$$W(\xi, \eta, \zeta) = \begin{cases} 1, & \text{if } -1 \leq \bar{\xi} - \xi \leq 1 \text{ \& } -1 \leq \bar{\eta} - \eta \leq 1 \text{ \& } -1 \leq \bar{\zeta} - \zeta \leq 1; \\ 0, & \text{if } |\bar{\xi} - \xi| > 1 \text{ or } |\bar{\eta} - \eta| > 1 \text{ or } |\bar{\zeta} - \zeta| > 1. \end{cases}\quad (3.11)$$

$\lambda$  is defined as

$$\lambda(\xi, \eta, \zeta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} H(\bar{\xi}, \bar{\eta}, \bar{\zeta}) W(\bar{\xi} - \xi, \bar{\eta} - \eta, \bar{\zeta} - \zeta) d\bar{\xi} d\bar{\eta} d\bar{\zeta}\quad (3.12)$$

For the weighting function defined in Equation (3.11), Jentink and Usab defined an “average” solution based on the fine mesh solution,  $q^N$ , and the Jacobian,  $J^{-1N}$ . The “average” or filtered solution,  $\mathcal{L}(q^N)$ , is defined as

$$\mathcal{L}(q^N) \equiv \frac{\overline{q^N J^{-1N}}}{\overline{J^{-1N}}} = \frac{\mathcal{L}(q^N J^{-1N})}{\mathcal{L}(J^{-1N})}\quad (3.13)$$

Integrating Equation (3.13) and using the mean-value theorem,  $\mathcal{L}(q^N)$ , in three-dimensions, is

$$\mathcal{L}(q^N) = \frac{\sum_{n=1}^8 q_n^N V_n^N}{\sum_{n=1}^8 V_n^N}\quad (3.14)$$

which is identical to the restriction operator,  $I_N^{N-1} q^N$ , defined in Equation (3.4). The sum is taken over the eight fine mesh cells surrounding the center of a coarse mesh cell.

Applying this filter to Eq. (3.1) Jentink and Usab showed, in two-dimensions, that Eq. (3.3) is recovered with the restriction operators defined in Eqns. (3.4) and (3.5). Thus, the coarse grid equations can be interpreted as the equations governing a spatially filtered flow state of  $Q^N$ .

The fine grid boundary conditions can be written as

$$L_b q = S_b \quad (3.15)$$

and are applied at the boundary cell face. The coarse mesh boundary conditions are also applied at the cell faces, as shown in Figure 3.2. The fine mesh cells are outlined with dashed lines while the coarse cells are outlined with solid lines. At the boundary, the function  $H$  is unity inside the domain and zero outside the domain. This means that the image cells do not contribute to the boundary condition. Applying the filter,  $\mathcal{L}$ , to the fine mesh cells adjacent to the boundary and integrating, the transferred value of  $q^N$  to the coarse mesh boundary,  $\mathcal{L}(q^N)$ , is

$$\mathcal{L}(q^N) = \frac{\sum_{n=1}^4 q_n^N V_n^N}{\sum_{n=1}^4 V_n^N} \quad (3.16)$$

where the sum is only taken over the four fine mesh cells inside the domain. Thus, one can define a coarse mesh boundary condition as

$$L_b^{N-1} q^{N-1} = L_b^{N-1}(\mathcal{L}(q^N)) \quad (3.17)$$

For a viscous flow, the forcing term at a solid wall essentially adds a slip condition as the mesh becomes too coarse to fully resolve the boundary layer.

### 3.2 Multigrid for Sub-domains

The effectiveness of multigrid is dependent on the amount of stretching used to generate a mesh. On highly stretched meshes, an iterative scheme may not be effective in reducing the high wave number error in all directions and thus multigrid is not as effective. This situation is typical of Navier Stokes solvers trying to resolve boundary layers in a reasonable amount of time and with a reasonable number of grid points. As an example, consider a mesh stretching ratio of  $\sigma$  on a fine mesh. The next coarse level mesh has a stretching ratio of  $\sigma^2$  which deteriorates the order of accuracy on the coarser grids. As a result, the coarse grid corrections do not contribute significantly to reducing the error in the fine grid solution. This point was also discussed by Srinivasan [12] in his work. Himansu and Rubin [24-25] found that, in some cases, semi-coarsening multigrid may be more effective than full coarsening for highly-stretched grids. Semi-coarsening is the application of multigrid by coarsening the mesh in one or two of the three directions. Swanson and Turkel [26] point out that the minimum time step for local time-stepping is based on the largest aspect ratio cell. The error propagation with this time step is disparate in one of the directions. This is illustrated in Figure A.4 and A.6 in the Appendix where increasing the aspect ratio decreases the rate of convergence.

The obvious solution to this problem is to increase the number of points until the stretching ratio is brought down to a reasonable level but that increases the number of points and the time and memory needed to obtain a solution. An alternative to stretching the mesh is to use uniform meshes and resolve flow details with locally refined uniform meshes where they are needed. This introduces a new level of complexity to solving the problem. However, the advantages of better accuracy, as shown by Srinivasan [12], could well outweigh the disadvantages of the added complexity.

Achi Brandt [10] describes multigrid in two complementary ways. The first is to view the coarser grids as correction grids that accelerate the convergence of an iterative scheme on the finest grid by efficiently liquidating low wave number errors.

That was the basis for the previous section. Many researchers have exploited this aspect of multigrid to increase the convergence rate of their algorithms [5-6,18,24-27]. His second view is to regard the finer grids as the correction grids, improving accuracy on coarser grids by correcting their forcing terms. This is the basis for the domain decomposition strategy to be developed here and applied to the simulation of turbomachinery flows. In the same paper, Brandt writes,

“ an efficient and very flexible general way to construct an adaptive grid is as a sequence of uniform subgrids, the same sequence used in the multi-grid process, but where the finer levels may be confined to increasingly smaller sub-domains to produce the desired local refinement. In this structure, the difference equations can be defined separately on each of the uniform subgrids, interacting with each other through the multi-grid process.”

This statement summarizes the intent of the work to be presented in the next section.

### **3.3 Domain Decomposition Strategy**

This section discusses the elements necessary to develop and implement a domain decomposition strategy for a general, three-dimensional rotating turbomachine. These elements are: a multiblock structure; a multigrid strategy; a grid generation strategy; boundary conditions; and interface conditions. Although each of these techniques have been developed and applied before, their combination into a single solution strategy, as developed herein, is unique.

Srinivasan [12] developed an SDDMG solver for the RNS (Reduced Navier-Stokes) Equations. Srinivasan refined the local regions of the computation domain by estimating the truncation error of the pressure gradient and second derivative of velocity. The marked regions of the domain were then refined and included into his solution procedure. This adaptive approach worked well for the two and

three-dimensional internal flows he addressed but is not applicable to general, three-dimensional, rotating turbomachines. The procedure employed here solves the three-dimensional, Reynolds-averaged, Navier-Stokes equations using a finite volume formulation and with an explicit Runge-Kutta solver.

In the course of developing his system, Srinivasan used a terminology to describe the mesh levels and their relationship to other mesh levels. That terminology is adopted here and is used to explain how the domain decomposition strategy is developed and implemented.

**3.3.1 Multiblock Structure.** Traditionally, a multiblock structure ties together a group of distinct block segments with common boundaries. Every multiblock grid has either an external flow boundary or an internal flow boundary through which information is passed. There are a number of reasons for implementing a multiblock structure. In a parallel processing environment, the computational work of iterating on a large single block domain can be reduced by dividing the single block into multiple blocks and iterating on each block separately. Another is to be able to grid complex configurations with multiple boundaries. This same type of structure is needed to implement the SDDMG method. The difference between the multiblock structure discussed above and the structure needed here is that the multiple block segments here are not distinct segments of the computational domain but each is treated as a sub-domain of the computational domain.

The multiblock structure for the current work begins with a single global parent mesh. This mesh is comprised of uniform cells and covers the entire computational domain. A global parent mesh is shown in Figure 3.3 in computational space.

The first level of refinement subdivides a segment or segments of the global parent mesh into sub-domain meshes. These segmented domains, relative to the global parent mesh, are called child meshes. The only rule that must be followed in refining a mesh level is that the boundary grid lines of a child mesh must be coincident with a grid line of the parent mesh. Figure 3.3 shows an example of two child meshes defined within a global parent mesh in computational space. This

example shows a refinement of both grid directions, however, the SDDMG method allows for semi-coarsening or refinement in either grid direction. The reason for this rule will be discussed in the following section. The only exception to this rule occurs on a blade surface boundary line which, as will be shown in the section on Grid Generation Strategy, requires the boundary line to lie on the blade surfaces as the mesh is refined.

Every subsequent level of mesh refinement is done in a similar manner. The child meshes on the previous mesh level are called parent meshes on their mesh level. From within these meshes, new child meshes are defined within the boundary of the parent mesh.

The information that is needed between a child mesh and a parent mesh is the location of the child mesh relative to the parent mesh and what type of coarsening strategy was used on the child mesh relative to the parent mesh. The location of the child mesh relative to the parent mesh is important as that determines what type of boundary condition is to be used on the child mesh. If the boundary grid line is coincident with an external boundary grid line then the appropriate external boundary conditions for that surface is used. If the boundary grid line is internal to the parent mesh, the appropriate interface condition is used.

**3.3.2 Multigrid Strategy.** The implementation of multigrid on sub-domain refinements is very similar to what was described in Section 3.1. The only difference is that the child meshes define the fine grid and the parent mesh defines the coarse grid. The same four steps discussed in Section 3.1 are applied to the child and parent meshes and are discussed below.

The first step of multigrid is iterating on the parent mesh on Mesh Level II in Figure 3.4 using the fine grid equations, Equation (3.1). The second step is evaluating the restriction operator over the parent mesh of Mesh Level II. The same restriction operators, Equations (3.2) and (3.3) defined in Section 3.1 are used. The resulting forcing functions are incorporated into the coarse grid equations of Mesh Level I. These forcing functions only exist over the domain of the child mesh. This

is the shaded area on Mesh Level I in Figure 3.4. The third step is smoothing the coarse grid error. This step is the primary difference between applying multigrid to a single block domain and applying multigrid to locally defined sub-domains. The parent mesh is treated as a fine mesh in regions outside of the child mesh domain. Within those regions, the forcing terms are zero and the fine grid equations are solved. The fourth step is the prolongation operation which interpolates the coarse grid corrections onto the domain of the fine grid on Mesh Level II of Figure 3.4. This operation, similar to the restriction operator, is only applied over the domain of the child mesh since the coarse grid corrections only exist over that domain. The prolongation operator, as previously defined, is a bilinear interpolator. The corrections are then used to improve the solution on the parent mesh of Mesh Level II. With the improved solution, the four steps just discussed are then repeated.

**3.3.3 Grid Generation Strategy** The domain decomposition approach presented here assumes that all meshes for a simulation are generated before the calculation is started. The primary reason for this is to preserve the integrity of the geometry, primarily the blade surface definition, across all grids levels. The first step in this process is to generate a uniform mesh with a very fine mesh spacing for the entire computational domain using the grid generation code developed by Mulac [28] and discussed in Section 2.6. To obtain the finest mesh spacing, one must consider the distance from the solid boundaries defined in terms of boundary layer wall units  $y^+$ . This must be consistent with the wall function boundary conditions of the turbulence model. Of equal importance is adequate resolution of the leading edge and trailing edge regions of the turbomachine geometry. For the present study, the only direction in which the mesh is stretched is the radial direction. Uniform grid spacing is maintained on the blade surface and across the blade passage.

The fine global mesh is used to generate the global parent mesh and all child meshes by discarding the appropriate number of mesh points from the fine global mesh. The effect of this procedure on blade surface definition as the mesh is refined is readily seen in Figure 3.5 on the leading edge region of Rotor 37. The global

parent mesh, mesh level 1, generates a wedge-like leading edge. After five levels of mesh refinement, the finest child mesh, mesh level 6, clearly defines a circle at the leading edge. Mesh levels 3 and 4 are identical because a refinement in the pitch-wise direction only and not the axial direction was performed to generate mesh level 4.

The first sub-domain boundary is set at 50% axial chord upstream and downstream of the blade surface and at 25% pitch from the blade surfaces. The first sub-domain meshes are typically refinements of both the axial and tangential direction of the global parent mesh. Subsequent finer domain boundaries are set at half that distance from the blade surfaces in both the axial and tangential directions. This is continued until the finest mesh spacing is achieved in both the axial and tangential directions.

Following the generation of all the parent and child meshes, a procedure is applied to all the sub-domain meshes to ensure that the interface grid lines of a child mesh are coincident with a grid line of the parent mesh. This is done by holding the interior tangential grid line of the parent mesh at the interface to the child mesh fixed. The interior grid points of the child mesh are redistributed uniformly from the external boundary to the interior boundary. This is illustrated in Figure 3.6. The grid in the upper box is the parent and child mesh with non-coincident interfaces and the grid in the lower box is the parent mesh and new child mesh with coincident interfaces. This procedure is important as it simplifies the specification of the interface boundary condition on the child mesh. The direction of the projected area of the child mesh at the interface is identical to the parent mesh and the value of the projected area is half that of the parent mesh.

**3.3.4 Boundary Conditions.** The implementation of external boundary conditions for the domain decomposition procedure is also modified for child and parent meshes. The fine grid boundary conditions, equation (3.13), are applied to the external boundaries of the finest child meshes and the portion of the external boundaries of the parent mesh outside the child mesh domain. The coarse grid



external boundary conditions, equation (3.16), containing the forcing term is applied to the portion of the parent mesh boundary that coincides with the child mesh external boundary.

**3.3.5 Interface Conditions.** To complete the implementation of the domain decomposition procedure, the interface condition arising from the child mesh inside the parent mesh must be derived. The interface condition for the child meshes is dependent on the parent mesh and must satisfy conservation of mass, momenta and energy.

To derive this condition, consider an interface surface between two parent cells as shown in Figure 3.7. The parent cell to the back shows the outline of the four cells of the child mesh that are a refinement of the coarse parent cell. The interface surface is between the forward parent cell and the two most forward child cells. The interface condition must conserve the flux of mass, momenta and energy across the interface area. This can be written as

$$\begin{aligned}\dot{m}_P &= \dot{m}_C \\ \vec{f}_P &= \vec{f}_C \\ i_P &= i_C\end{aligned}\tag{3.18}$$

where  $\dot{m}$  is the mass flux,  $\vec{f}$  is the momentum flux vector and  $i$  is the energy flux at the interface. The subscript  $P$  denotes the parent mesh and  $C$  denotes the child mesh. Equation (3.18) can be further expanded as

$$\begin{aligned}\dot{m} &= \rho(\mathbf{v} \cdot d\mathbf{A}) \\ \vec{f} &= \rho(\mathbf{v} \cdot d\mathbf{A})\mathbf{v} + pd\mathbf{A} \\ i &= (\mathbf{v} \cdot d\mathbf{A})(\rho e_0 + p)\end{aligned}\tag{3.19}$$

where  $d\mathbf{A}$  is the projected surface area vector of the interface. Substituting equation (3.19) into equation (3.18) yields

$$[\rho(\mathbf{v} \cdot d\mathbf{A})]_P = \sum [\rho(\mathbf{v} \cdot d\mathbf{A})]_C$$

$$\begin{aligned}
[\rho(\mathbf{v} \cdot d\mathbf{A})\mathbf{v} + pd\mathbf{A}]_P &= \sum [\rho(\mathbf{v} \cdot d\mathbf{A})\mathbf{v} + pd\mathbf{A}]_C \\
[(\mathbf{v} \cdot d\mathbf{A})(\rho e_0 + p)]_P &= \sum [(\mathbf{v} \cdot d\mathbf{A})(\rho e_0 + p)]_C
\end{aligned} \tag{3.20}$$

Where the sum is taken over all the child interface areas that make up the parent interface area. By construction, the parent interface area vector,  $d\mathbf{A}_P$ , is equal to the sum of the child interface area vectors,  $\sum d\mathbf{A}_C$ . For uniformly spaced cells, the magnitude of the child interface surface areas are exactly half of the magnitude of the parent interface surface area. A solution to Equation (3.20) is

$$\begin{aligned}
\rho_C &= \rho_P \\
[\rho\mathbf{v}]_C &= [\rho\mathbf{v}]_P \\
p_C &= p_P \\
H_{0C} &= H_{0P}
\end{aligned} \tag{3.21}$$

at the interface, where  $H_0$  is the enthalpy defined in Equation (2.10). For the interface to be conservative, the interface of a child mesh must satisfy Equation (3.21).

Figure 3.8 shows an interface surface between two parent mesh cells. The upper parent cell is within a child mesh domain and contains four child mesh cells. The lower parent mesh cell is outside but adjacent to a child mesh domain and contains the child mesh “image” cells. The open circles in Figure 3.8 represent the center of the parent mesh cells and the open squares represent the center of the child mesh cells. The open broken line squares represent the center of the child mesh “image” cells. The right-hand side of Equation (3.21) is obtained by averaging  $\rho$ ,  $\rho\mathbf{v}$ ,  $p$ , and  $H_0$  from the centers of the parent mesh cells bordering the parent interface surface which is represented by the shaded circle. To satisfy Equation (3.21),  $\rho$ ,  $\rho\mathbf{v}$ ,  $p$ , and  $H_0$  at both child interface surfaces, represented by the shaded squares, are set to the averaged value of  $\rho$ ,  $\rho\mathbf{v}$ ,  $p$ , and  $H_0$  at the parent interface surface. The value of  $\rho$ ,  $\rho\mathbf{v}$ ,  $p$ , and  $H_0$  at the parent interface surface is the average formed from the respective values at the centers of the parent mesh bordering the parent interface. The value of  $\rho$ ,  $\rho\mathbf{v}$ ,  $p$ , and  $H_0$  at the cell center of the child mesh “image” cell is

then obtained by linear extrapolation using the value at the child interface surface and the value of the center of the child mesh bordering the child interface surface. This “image” cell value with the interior cell value is then used to compute the shear stresses at the interface surface as is done in the parent mesh.

The artificial viscosity on the child mesh requires an additional “image” cell to compute the fourth difference of the flow variables at the interior cell bordering the child interface. This second “image” cell is located directly beneath the “image” cells shown in Figure 3.8. The flow variables at the center of these second “image” cells are obtained by bilinear interpolation from the centers of the parent mesh cells bordering the second “image” cells.

**3.3.6 Implementation.** The solution procedure is similar to what is termed full multigrid mode. Figure 3.9 illustrates the procedure in a schematic diagram. The calculation begins on a coarse mesh which is denoted as CM in Figure 3.9. This mesh is obtained by coarsening the global parent mesh. The global parent mesh is the finest mesh covering the entire computational domain. A preset number of iterations, indicated as  $N$  iters. at the top of the figure, are performed on the coarse mesh. The resulting solution is interpolated, indicated as the broken line in the figure, onto the global parent mesh. A preset number of iterations are performed on the global parent mesh using the V-cycle of multigrid. This procedure is outlined in Section 3.1. The next level of mesh refinement is now introduced. The solution obtained on the global parent mesh is interpolated onto the parent mesh of Mesh Level  $II$ . This parent mesh is also known as the child mesh of Mesh Level  $I$ . The iteration procedure is initiated on the parent mesh of Mesh Level  $II$ . The fine mesh equations are solved on this level followed by an evaluation of the forcing terms associated with the mesh level which are distributed onto to the child mesh domain of the global parent mesh. The next step is to iterate on the global parent mesh. This entails solving the coarse grid equations in the domain of the child mesh which contain the forcing terms and the remaining coarse grid domain void of the forcing terms. Forcing terms on the parent mesh level are evaluated and distributed to the

next coarser mesh level. On the coarsest mesh level, the equations consistent with this mesh level are solved over the entire domain. Corrections obtained from the coarsest mesh are interpolated to the global parent mesh to improve the solution on that mesh level. Corrections on the global parent mesh in the child mesh domain are used to improve the solution of the parent mesh on Mesh Level II. The interface condition on the parent mesh of Mesh Level II is now updated using the global parent mesh solution. At this point, the iteration of the equations associated with the parent mesh of Mesh Level *II* is re-initiated. A preset number of cycles are calculated in this manner. Upon reaching the preset number of cycles, the procedure checks to see if there is another mesh level within the parent mesh of Mesh Level *II*. If so, the mesh level is advanced and the child mesh becomes a parent mesh. The parent mesh solution is interpolated onto the next level parent mesh. The above procedure is then repeated involving all the previously defined mesh levels. The corrections from the various mesh levels are then used to improve the solution at each successively finer mesh level. Upon improving the solution on the new fine mesh level, the interface condition is updated based on the solution from its parent mesh and the procedure is re-initiated. This is continued until all of the predetermined mesh levels have been included in the above procedure.

If, upon examination of the final solution, it is felt that more resolution is needed, a finer mesh can be generated within the smallest child mesh(es) and the calculation restarted incorporating the new finer mesh level. To obtain the finer domain of the new child mesh, the grid generation program is used to halve the previous finest mesh spacing. The new child mesh is defined by extracting the desired sub-domain out of the newly generated fine mesh. The new child mesh level is simply incorporated into the previous solution procedure. The initial solution is estimated from the parent mesh and the new mesh is treated as part of the overall solution strategy presented above. An example of this will be shown in Section 4.3 where two additional mesh refinements are added to a converged solution.

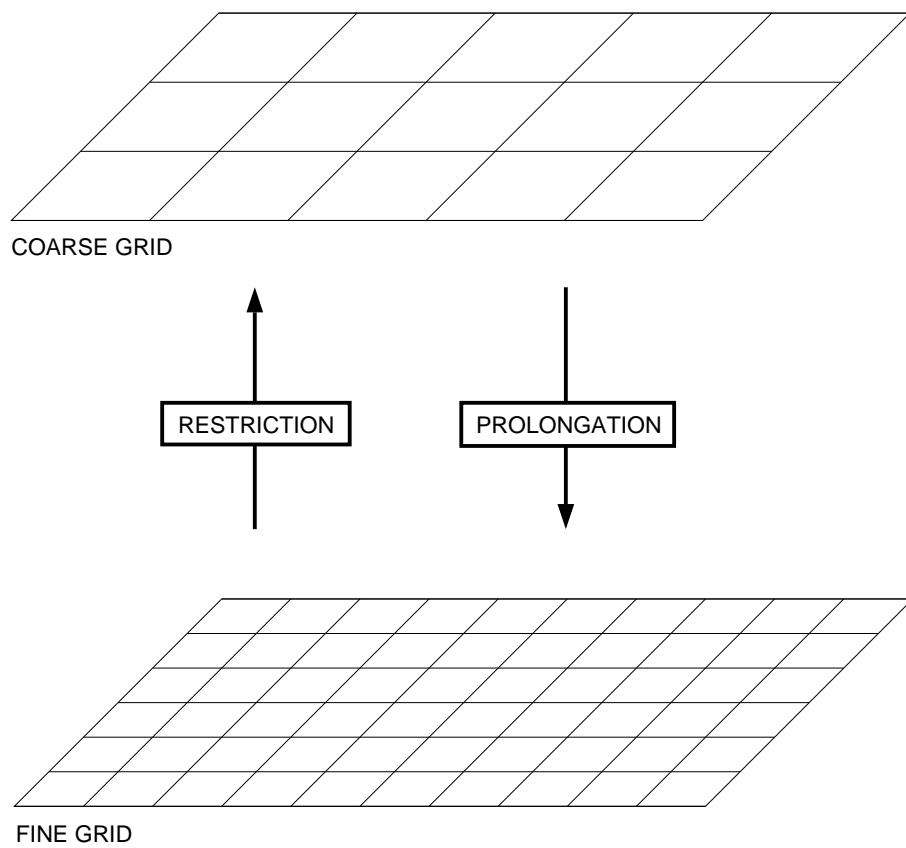


Figure 3.1. Multigrid process for a single block domain using two grid levels.

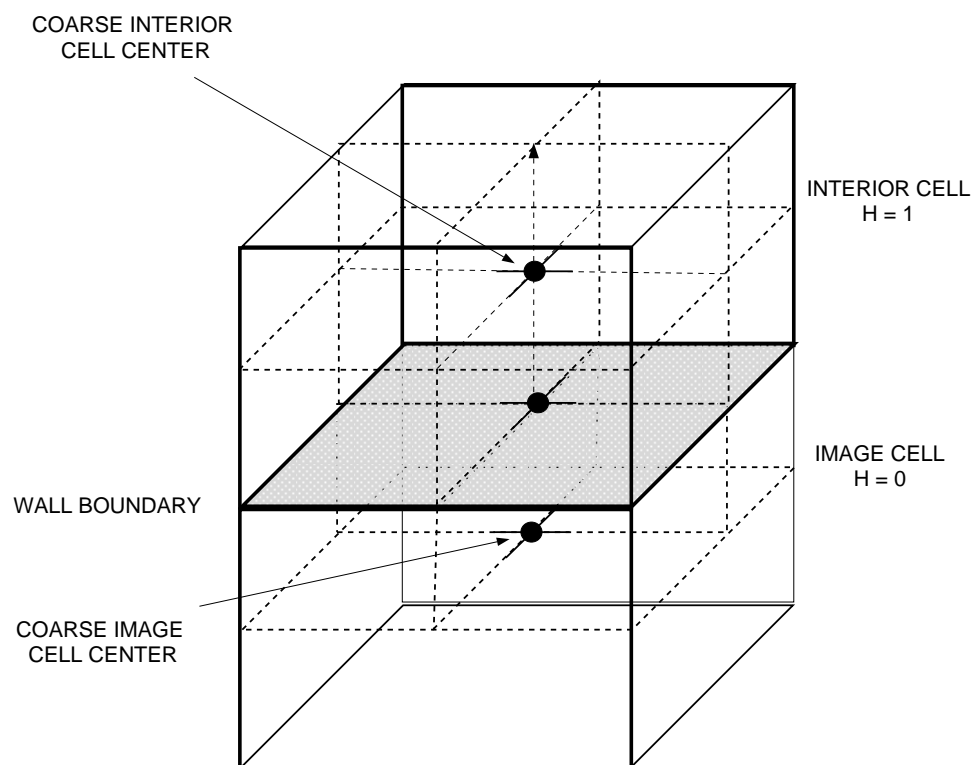


Figure 3.2. Schematic of interior and image cells at a boundary.

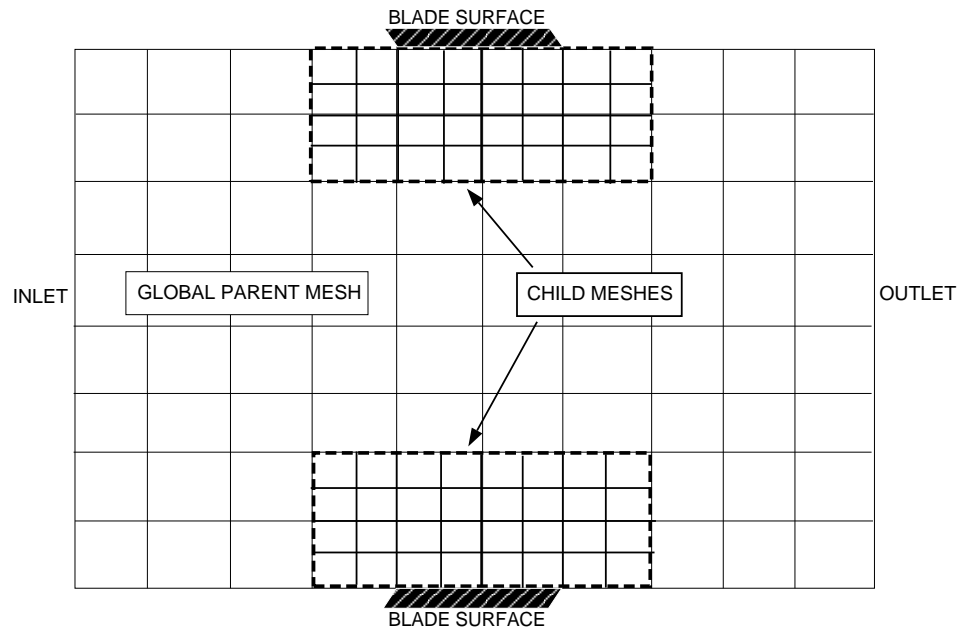


Figure 3.3. Global parent mesh with two child meshes in computational coordinates. The dashed line depicts the boundaries of the child meshes.

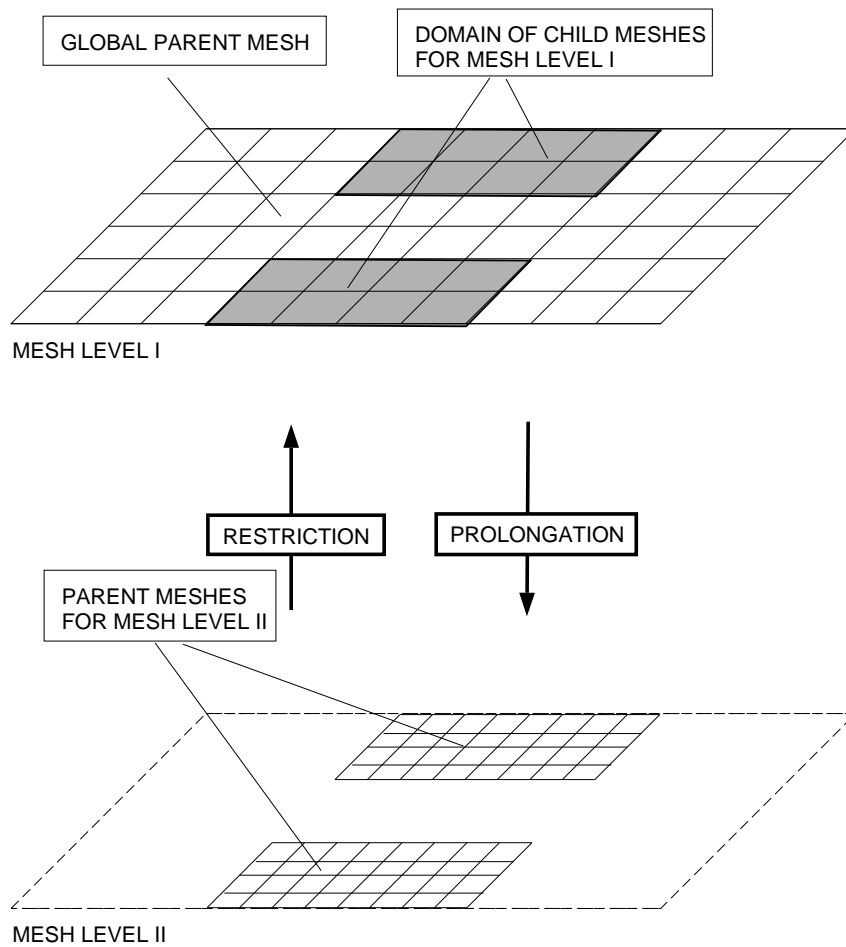


Figure 3.4. Multigrid process for segmented domain blocks.



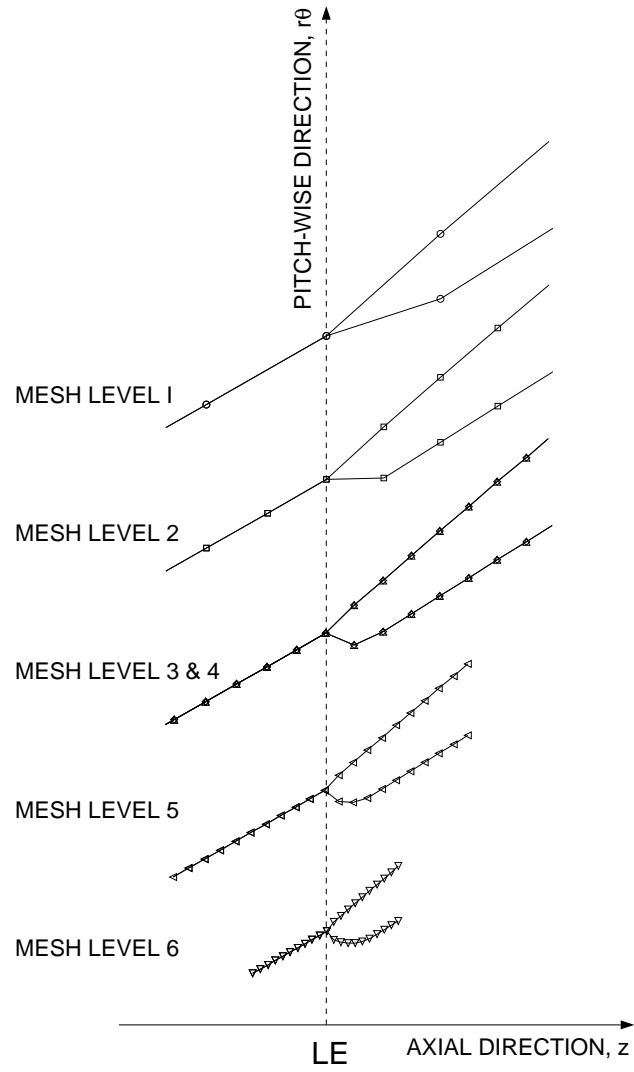
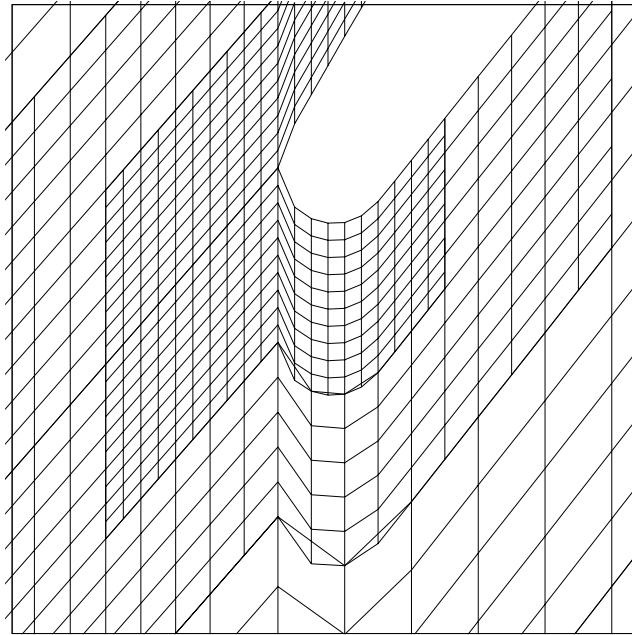
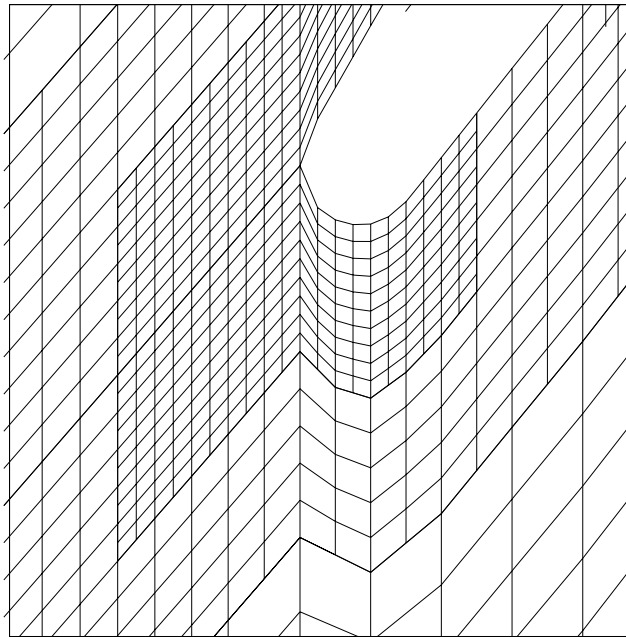


Figure 3.5. Resolution of blade surface definition with mesh refinement.



NON-COINCIDENT INTERFACES



COINCIDENT INTERFACES

Figure 3.6. Alignment of child mesh interface boundaries with parent mesh grid lines.

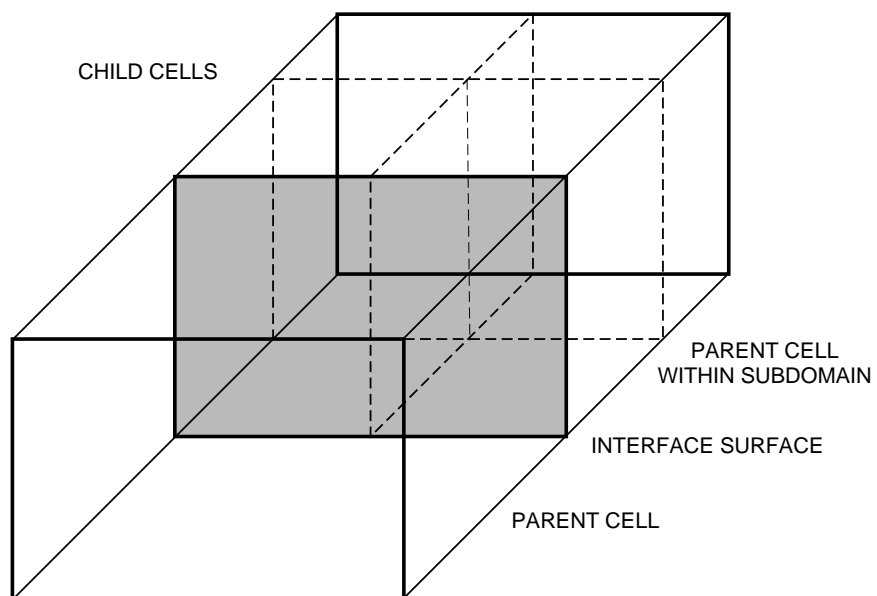


Figure 3.7. Schematic of parent and child cells at an interface.

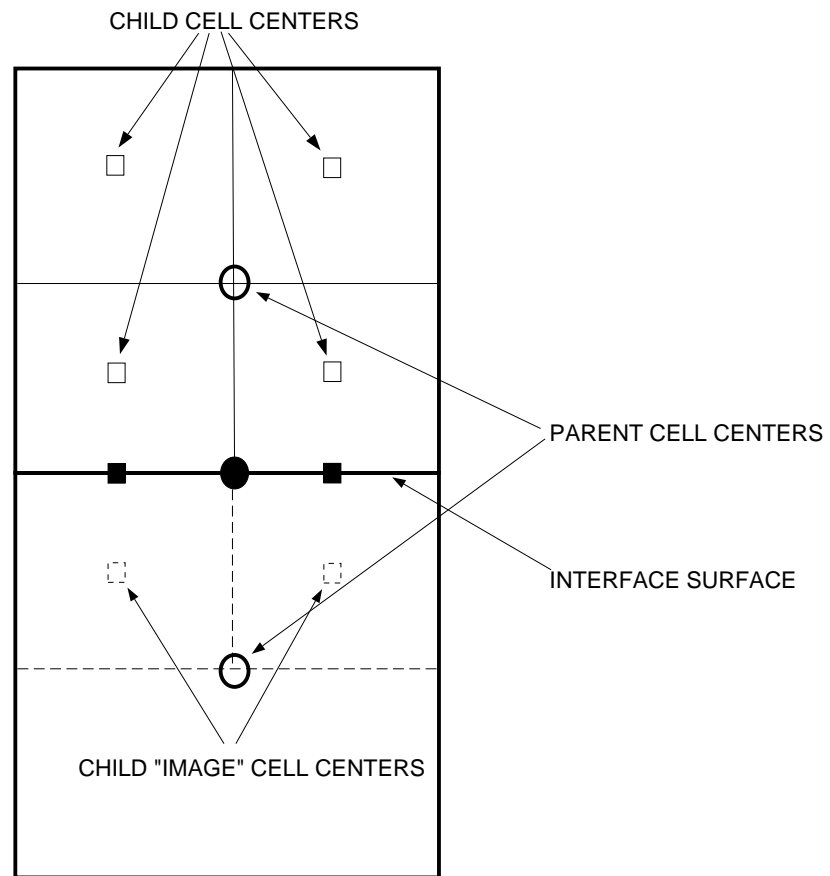


Figure 3.8. Schematic of a parent mesh / child mesh interface surface.

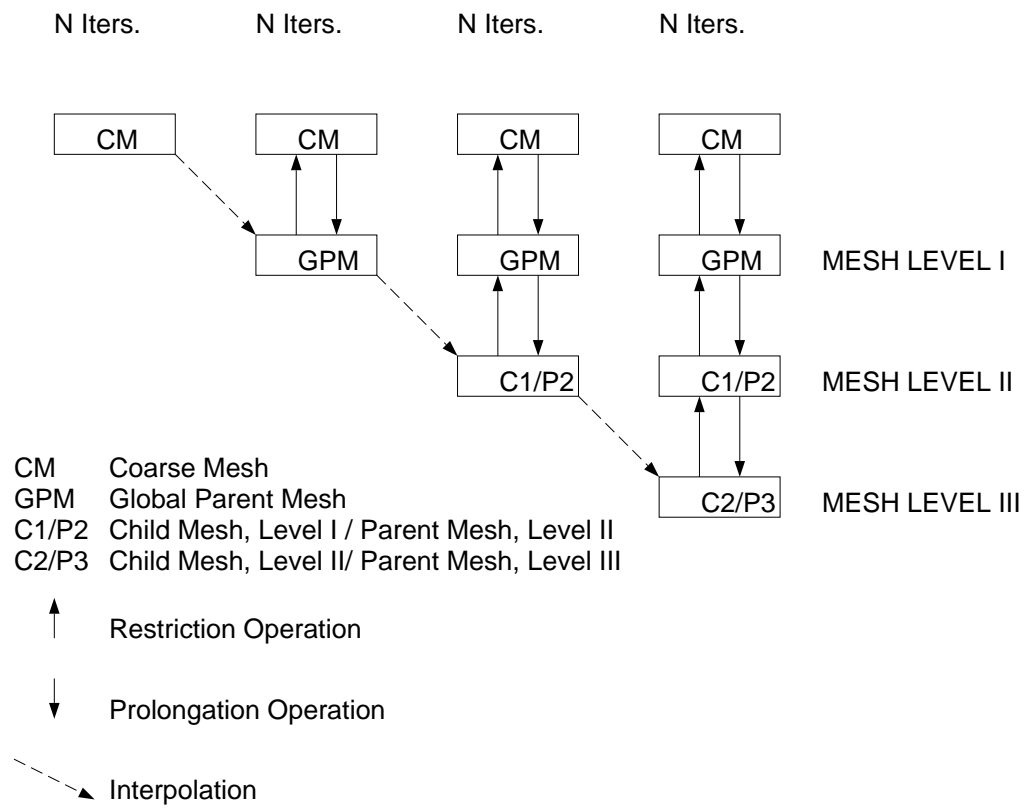


Figure 3.9. Schematic of SDDMG procedure.

## CHAPTER 4

### RESULTS

Four geometries are considered to test and validate the SDDMG method. The cases are: channel flow; a compressor cascade; a turbine cascade; and a transonic compressor rotor typical of what is found in today's modern jet engine. Three of the four cases are two-dimensional but are gridded as three-dimensional geometries. All four have experimental data available. Comparisons to this data are made where possible.

#### 4.1 Channel Flow

The channel flow geometry is shown in Figure 4.1 and is simulated primarily because it is simple in geometry and it can be used to assess various features of the SDDMG procedure. It is also used to show the advantages of the domain decomposition method over a single block method using a stretched grid. The geometry was setup according to the description of the test section given by Hussain and Reynolds [47] in their experiment. In the simulation, the gap width is the tangential or pitch-wise direction. The radial height of the channel is set to 18 times the gap width and the axial extent of the channel is set to 40 times the gap width to allow the boundary layer to fully develop. Because the coordinate system of the code is cylindrical, the channel is placed some distance away from the centerline so that the hub to tip ratio defined as the ratio of the hub radius to the tip radius is approximately unity. The hub to tip ratio, for this case, is set to 0.98. To minimize the influence of any boundary layer from the end-walls of the channel, the boundary conditions at the end-walls allowed for slip flow. The channel is simulated at a Mach number of three-tenths which is set by fixing the outlet static pressure. Uniform conditions are set at the inlet. The Reynolds number based on the channel width is 27,000 and the flow is considered fully turbulent to be consistent with the experiment.

The channel is simulated using a uniform mesh with several levels of refinement on sub-domains and also with a single block stretched mesh. Figure 4.2 shows a sectional view of the grid at midspan. 101 grid points are distributed axially from the inlet to the outlet. The gap width (pitch-wise direction) contains 4 mesh levels with 3 of the levels being sub-domain refinements. The global parent mesh spans the gap width with 21 points. All subsequent refined mesh levels contain 5 pitch-wise grid points. The total number of cells in the tangential direction is 44. The total number of cells that actually span the solution domain in the pitch-wise direction, i.e., the cells that are relevant to the final result, is 32. This number is obtained by discounting the parent cells that are contained in the child meshes. Mesh refinements for this case were only made in the tangential direction in order to compare the solution to a single block stretched mesh. All mesh levels had the same axial distribution of points.

The stretched mesh, shown in Figure 4.3, is obtained by calculating the stretching ratio required to have 51 grid points across the passage with the location of the first cell equal to the first cell of the SDDMG mesh shown in Figure 4.2. The stretching ratio in the tangential direction is 1.08614 and is defined as the ratio of cell width for two adjacent cells. This is the same definition of stretching ratio that is used in Appendix A. The solution obtained with this combination of grid points and stretching ratio was completely coupled and was therefore chosen to represent the minimum resolution required for a single block stretched mesh. Because there are two solid surfaces that encompass the gap, this stretching ratio is maintained from both solid surfaces to mid-passage. The stretched mesh contains the same axial distribution of mesh cells as the domain decomposition mesh.

The number of grid points in the radial direction is uniform for all cases and is set to 11. Slip flow is imposed at the channel end-wall boundaries.

The experimental results of Hussain and Reynolds [47] essentially validates the law of the wall and is not shown with the computations. Instead, comparisons will be made to the law of the wall. A solution was also obtained with a grid containing 160

cells in the tangential direction that was used to generate the domain decomposition grids.

To maintain an isentropic exit Mach number of 0.3, a static pressure is obtained from the isentropic tables and this value is fixed at the outlet boundary. The reference conditions are taken as Standard Day which are  $P_{Ref} = 101,325 \text{ N/m}^2$  and  $T_{Ref} = 288.15 \text{ K}$ . The inlet turbulence intensity is assumed to be 1%. The solution results for the two calculations are presented near the outlet of the channel length where the flow is considered to be fully developed. The results are shown in Figure 4.4 and are plotted in the boundary layer variables,  $u^+$  and  $y^+$ . Both the SDDMG solution and stretched mesh solution follow the law of the wall until a  $y^+$  of about 800 is achieved. The SDDMG and stretched mesh solutions show practically identical profiles. However, the SDDMG solution achieves this profile with 32 effective tangential cells as opposed to the 50 tangential cells associated with the stretched mesh. The difference of 12 tangential cells from the stretched mesh are clearly clustered outside of the log-layer where the gradients are small and the grid density is not necessary. The cells of the SDDMG solution are clustered closer to the surface where they are needed.

Figures 4.5 (a) and (b) illustrate how the SDDMG system converges relative to the single block stretched mesh. Figure 4.5(a) is a plot of the convergence rate of the two systems. The maximum error is associated with the continuity equation for both cases. The plot shows the convergence rate of two mesh levels of the SDDMG system versus the single block system of the stretched mesh. The finest SDDMG mesh level referred to in Figure 4.5(a) is mesh level 4 in Figure 4.2. The single block stretched mesh was run with one level of multigrid. Both systems began their iterations on the coarse mesh level with the same number of predetermined iterations to obtain a better initial guess for the next mesh level. Both systems were iterated upon until the residuals achieved a limit cycle which is assumed to be the lowest level of convergence of the calculation. Figure 4.5 shows that both the global parent mesh and the finest mesh level, indicated as Mesh Level 4, of the SDDMG



system achieved a lower level of convergence than the stretched mesh along with a better rate of convergence. This is indicated by the slopes of the SDDMG residual history. The spikes in the global parent mesh of the SDDMG calculation is a result of interpolating the parent mesh solution onto the child meshes at the beginning of a mesh refinement. Both calculations were run further than the 1300 iterations that is shown. The dashed line in Figure 4.5, labeled Machine Precision, is assumed to be the lowest level of error that is attainable. This value was obtained from the residual history of the mesh containing 160 tangential cells across the gap.

The global parent mesh of the SDDMG system spans the entire computational domain and thus, controls the overall mass flow of the system. Figure 4.5(b) shows the convergence rate of the percent difference of massflow, defined as

$$100\left(\frac{\dot{m}_{inlet} - \dot{m}_{outlet}}{\dot{m}_{inlet}}\right)$$

where  $\dot{m}$  is the mass flow. The mass flow is obtained by integrating the momentum in the axial direction over the cross-sectional area that comprises the inlet and outlet. The difference in massflow from the SDDMG global parent mesh converges to zero within 700 iterations whereas the single block stretched mesh has about a 1% difference. Even after 1300 iterations, the single block stretched mesh has a difference of about a half percent.

The SDDMG and stretched mesh solutions compared very well with the fine mesh solution which contains 160 cells across the gap. Neither solution had an error larger than 1% in the velocity profile.

The SDDMG and stretched mesh simulations were run on an SGI Origin 2000. The CPU time per iteration is comparable for the two methods and averages to about 3 seconds per iteration in both cases. This number accounts for iterating on all grid levels and the passing of information between grid levels with multigrid. The start-up time for the SDDMG solution is less than for the single block stretched mesh since each mesh level is started after a predetermined number of iterations on its parent mesh. Thus, the finest mesh level (Level 4) is begun only after 250 iterations have been performed on the coarser mesh levels.

The profile from each SDDMG mesh is shown in Figure 4.6 and is compared to the law of the wall. Each subsequent finer mesh level refines the profile near the wall. The effect of halving the mesh spacing to resolve the boundary layer is seen by examining the value of  $y^+$  from the first mesh cell of each level. The distance in wall units,  $y^+$ , from the center of the first mesh cell from the global parent mesh, represented by the open circles, is 180. The corresponding distance from the center of the first mesh cell from the next mesh level, represented by the open squares, is 90. The open diamond represents the second refinement and the distance from the center of the first mesh cell to the wall, in wall units, is 45. The distance, in wall units, from the center of the first mesh cell to the wall for the finest mesh level, represented by the open triangles, is 23. Thus, four levels of refinement reduced the value of  $y^+$  by a factor of eight.

As an example of the flexibility of the SDDMG system, if one required further resolution of the boundary layer to a  $y^+$  of about 5 to test, say, a new turbulence model, just two more levels of refinement would be needed to obtain that level of resolution. The total number of cells added to the system is 8 and the number of additional effective cells added to the system is four. Recalling the strategy discussed in Section 3.3, the new mesh levels would simply be incorporated into the solution procedure.

For the single block stretched mesh, the strategy is quite different. To reduce the distance of the first cell of the wall by one-fourth for the single block stretched mesh and maintaining the same stretching ratio requires increasing the number of tangential points to 81 which is an increase of almost 60%. In addition to increasing the number of tangential points, the entire solution domain must be interpolated from the current mesh of 51 tangential points to the new finer mesh. Thus, the SDDMG system is more flexible in allowing further refinements of local regions of the flow and cheaper in terms of minimizing the number of additional mesh levels and thus, grid points.

## 4.2 ARL Compressor

The ARL cascade [29] shown in Figure 4.7 is a double-circular-arc blade designed at NASA Lewis and is typical of a highly loaded compressor blade. From the test report, the chord length  $c$  is given as 228.6 mm and the solidity  $\frac{c}{s}$  is given as 2.14. The aspect ratio  $AR = \frac{h}{c}$  is 1.61. The inlet mean velocity is 33 m/s and the Reynolds number based on chord is 501,000.

The test report discussed some of the limitations of the experiment. To obtain the Reynolds number in the experiment a large chord length was required. This limited the number of cascade blades in the test section to five and limited the aspect ratio to 1.61. To obtain cascade data representative of a cascade with an infinite number of blades, the experiment was supplemented with suction both to limit end-wall boundary layer growth and maintain two-dimensionality of the flow. The suction was applied at the top and upper side channels upstream of the blades. It was reported in Ref. [29] that suction had the most influence on the periodicity of the flow. In addition to suction, positioning of tailboards downstream of the experiment was used to control the exit flow. Ref. [29] also stated that the tailboards were most useful in controlling the relative exit angles of the flow.

To simulate the above test, the  $AR$  of the geometry is increased by a factor of five to a value of 8. This is done primarily to circumvent the influence of the end-wall flow on the core flow. Suction and tailboards were not modeled in the simulation. Since this cascade is a two-dimensional geometry and the code is written in cylindrical coordinates, the hub and shroud radii are placed far from the centerline. To maintain the solidity or pitch spacing of the blade, the number of blades is chosen which then fixes the radius. To maintain a two-dimensional geometry, the hub to tip ratio  $\frac{R_h}{R_t}$  must approach unity and the radius ratio,  $R^* = \frac{R_t - R_h}{R_t}$  must approach zero. The number of blades, for this case, is set to 2000 which sets the diameter and reference length of the geometry. Based on the set number of blades, the hub to tip ratio  $\frac{R_h}{R_t} = 0.947$  and  $R^* = 0.053$ .

The test report presented data for three incidence angles. The simulation will only consider the data presented for an incidence angle of  $-1.5^\circ$ .

The inlet flow conditions were measured 15.7% chord upstream of the leading edge, however, the incidence angle was determined by the relative position of the blades to the incoming flow and was not measured. Outlet profiles were measured at either 31.9% or 52.6% chord downstream of the trailing edge.

The ARL compressor cascade is simulated as a three-dimensional geometry. Uniform meshes are used on planes of constant radius and a non-uniform distribution of points is used for the radial direction. The grid on a constant radial plane at midspan is shown in Figure 4.8. A total of seven grid levels are used in this simulation with six sub-domain mesh levels. Table 4.1 gives a summary of the mesh sizes:

Mesh Level	Axial Points	Radial Points	Tangential Points	Number of Meshes
1	61	41	21	1
2	81	41	11	2
3	121	41	11	2
4	201	41	11	2
5	361	41	11	2
6	41	41	11	4
7	41	41	11	4

Table 4.1 SDDMG Mesh Refinement Schedule for ARL Compressor.

The global parent mesh is Mesh Level 1 and contains 61 axial points with 21 of these points distributed uniformly on the blade. The inlet and outlet are set one chord upstream and downstream of the blade. The number of radial grid points is set to 41 and is distributed with a stretching ratio of 1.2 to resolve the hub and casing boundary layer. This distribution is maintained for all sub-domain meshes. 21 points were distributed evenly in the tangential direction in the global parent mesh. The first set of sub-domain refinements is located between 50% chord upstream and downstream of the blade geometry. This division axially is maintained for the next three mesh levels. Mesh levels 6 and 7 are used to refine the leading

edge and trailing edge circles of the blade. This is illustrated in Figures 4.9 (a) and (b). Two mesh passages are shown near the leading edge and trailing edge regions of the blade at midspan.

The interface boundary in the tangential direction for the first mesh level was set at 25% pitch from the blade surface. The interface boundaries for the remaining mesh levels were set midway between the surface and the location of the interface boundary of the parent mesh.

The simulation at the given incidence angle of  $-1.5^\circ$  did not compare well with the blade loading from the experiment. However, the incidence angle was never measured. A panel method code developed by McFarland [31,32] was then used to determine an incidence angle that more closely matched the blade loading determined from the experimental measurements. This incidence angle was determined to be  $0^\circ$ . Figure 4.10 compares the measured blade loading versus percent of chord with the panel method result at an incidence angle of  $0^\circ$ . The blade loading is given in terms of a pressure coefficient,  $C_p$  which is defined as

$$C_p = \frac{p_s - p_{sINLET}}{\frac{1}{2}(\rho q^2)_{INLET}}.$$

The simulation was rerun at this incidence angle of  $0^\circ$  and the resulting blade loading is shown in Figure 4.11. The blade loading derived from the simulation is compared to the panel method and experimental data. The loading on the suction surface of the simulation indicates the possibility of separation near the trailing edge.

To ensure that the results of the three-dimensional simulation are two-dimensional, it is necessary that the velocities and angle of the flow be substantially the same in span-wise planes. Figure 4.12 shows the axisymmetric axial velocity at the inlet and outlet station. The axisymmetric average of the axial velocity is obtained by integrating the axial velocity in the tangential direction and dividing by the tangential distance. The velocity at the inlet is fairly constant except for the imposed boundary layer at the end-walls. At the exit, the axial velocity is constant

from 15% to 85% span and can be considered two-dimensional in this region. The flow becomes more three-dimensional at the exit due to the significant growth of the hub and shroud boundary layer. Figure 4.13 shows the axisymmetric exit flow angle,  $\alpha$ , versus fraction of span. The flow angle is defined as:

$$\alpha = \tan^{-1} \frac{\bar{v}_\theta}{\bar{v}_z}$$

where  $\bar{v}_\theta$  and  $\bar{v}_z$  are the axisymmetric average values of  $v_\theta$  and  $v_z$ , respectively. Here also, the extent of the hub and shroud boundary layer reduces the region where the flow can be considered two-dimensional by 20% from the end-walls. As mentioned earlier, end-wall suction and exit tailboards were used in the experiment to eliminate or reduce any three-dimensional flow variations and provide a periodic flow. This was not done in the simulation. All remaining comparisons will be at the midspan plane.

Figure 4.14 shows contours of velocity magnitude at the midspan plane. The wake from the trailing edge is seen decaying with distance. In addition, the velocity contours are continuous as they pass through the various mesh levels. In the experiment, the wake was surveyed at 32.9% chord downstream of the trailing edge. Figure 4.15 shows a comparison of the axial velocity from the simulation compared to the experiment at 32.9% chord downstream of the trailing edge. The minimum axial velocity from the data was shifted in the pitch-wise direction to be aligned with the minimum axial velocity from the simulation. The wake characteristics of the simulation, i.e., velocity deficit and wake width, compare very well with the data.

To examine the effect of viscosity by an alternate method, two boundary layer codes, BLAYER [33] and STAN5 [34], were run using the panel code as the inviscid outer solution to predict the extent of separation, if any, on the suction surface of the blade. BLAYER predicted the possibility of separation at 96.9% chord and STAN5 predicted the possibility of separation at 86.5% chord. The present simulation showed separation at 97.6% chord.

Figure 4.16 shows velocity vectors on the suction surface near the trailing edge. The zero velocity line is seen going through two mesh levels with the separation region extending across three mesh levels.

The solution obtained with the SDDMG system converges to a mesh independent solution in the neighborhood of the leading edge as the mesh is refined. To show this, the blade surface pressure coefficient,  $C_p$ , for all the mesh levels is plotted near the leading edge of the blade. Figure 4.17 shows how the solution on different mesh levels is converging as the mesh levels are refined. The leading edge is at zero. The pressure surface is on the left of the leading edge and the suction surface is to the right of the leading edge.

### 5.3 VKI Turbine Cascade

The VKI cascade [30] shown in Figure 4.18 is a highly loaded transonic turbine nozzle guide vane mounted in a linear cascade arrangement. The measurements were performed in the Von Karman Institute short duration Isentropic Light Piston Compression Tube facility. The chord length is given as 67.647 mm and the solidity  $\frac{c}{s}$  is given as 1.176. The blade stagger angle is  $55^\circ$  from the axial direction. The throat to chord ratio  $\frac{h}{c}$  is 0.2207. As in the ARL cascade case, the number of blades is set which then determines the diameter and reference length. For this case, the number of blades is set to 200 and the hub to tip ratio and radius ratio are  $\frac{R_h}{R_t} = 0.964$  and  $R^* = 0.036$ .

Freestream conditions and turbulence intensity were measured 55 mm upstream of the leading edge plane (148.7% chord) and exit conditions were measured 16 mm (43.3% chord) downstream of the cascade. The simulation was run at an exit isentropic Mach number,  $M_{2,is}$ , of 0.85 and a Reynolds number,  $Re_{2,is}$  of  $1 \times 10^6$ . The Reynolds number is based on the chord and exit velocity.

The throat height of the VKI Turbine Cascade is maintained when generating the grid for the geometry. The experiment did not remove side-wall boundary layers. This cascade is simulated with the side walls included which means the

simulation is three-dimensional. A perspective view of the VKI turbine mesh is shown in Figure 4.19. The hub grid and block outlines of the global mesh and sub-domains are shown. A view of the mesh at midspan is given in Figure 4.20. The radial direction uses a stretched grid. The number of radial grid points was set to 41 and are distributed with a stretching ratio of 1.2 to resolve the hub and casing boundary layer. This distribution is maintained for all sub-grids. The inlet and outlet boundaries were set one chord upstream and downstream of the blade. Table 4.2 gives a summary of the mesh sizes:

Mesh Level	Axial Points	Radial Points	Tangential Points	Number of Meshes
1	61	41	21	1
2	81	41	11	2
3	121	41	11	2
4	101	41	11	2
5	101	41	11	2
6	21	41	11	4
7	21	41	11	4

Table 4.2. SDDMG Mesh Refinement Schedule for VKI Turbine.

Mesh level 1 is the global parent mesh. The first two sub-domain meshes, levels 2 and 3 are axial and tangential refinements of the parent mesh. Mesh levels 4 and 5 are tangential refinements only and are employed to resolve the blade boundary layers. Mesh levels 6 and 7 are axial refinements of the blade leading and trailing edge regions. Figures 4.21(a) and 4.21(b) show two passages of the SDDMG mesh near the leading and trailing edges of the blade.

To ensure that the results of the three-dimensional simulation are two-dimensional at mid-span, the axial momentum entering and exiting the blade row should be nearly identical. Figure 4.22 is a plot of the axisymmetric axial momentum versus percent span at the inlet and outlet stations. From 10% to 80% of span, the axial momentum profiles are nearly identical and thus, the flow in that region can be considered two-dimensional. All following results will be reported from the midspan plane. Figure 4.23 is a plot of Mach Number contours at midspan. The



figure shows the inlet Mach number to be 0.15. The experimental data reports the exit Mach number to be 0.85. The exit static pressure varied in order to match the reported exit Mach number.

One of the advantages of this scheme is the ability to capture fine flow details near the blade surface. One region with interesting flow detail is the trailing edge. Figure 4.24 shows flow vectors at the trailing edge of the blade obtained from the solutions at mesh level 7. One can clearly see two counter-rotating vortices caused by the merger of the flow from the suction and pressure surface.

Further downstream of the blade, the pitch-wise variation of total pressure was measured in the experiment at 142% of chord. The distribution is presented as a loss coefficient and is defined as

$$\chi = \frac{1 - \frac{p_2}{p_{t2}} \frac{\gamma-1}{\gamma}}{1 - \frac{p_2}{p_{t1}} \frac{\gamma-1}{\gamma}} - 1$$

where  $p$  is pressure,  $p_t$  is the total pressure and the indices 1 indicate the inlet and 2 indicates the downstream reporting station at 142% chord. Figure 4.25 compares the loss coefficient for the SDDMG results versus experiment. The wake width compares very well to the experimental result, however, the wake depth is over-predicted by about 30%. Integrating the loss coefficient over one pitch, the loss coefficient at that location is 0.0500 for the simulation as compared to the experimental value of 0.0247. The over-prediction in the wake depth and excess loss may be the result of flow transition not being accounted for.

Referring back to Figure 4.23, the flow accelerates rapidly near the suction surface (lower surface in the figure) of the blade. This acceleration causes the suction surface boundary layer to be very thin. Figure 4.26 illustrates this point by plotting  $y^+$  versus percent chord on the suction surface.  $y^+$  increases rapidly from the leading edge to a maximum value of 75 at 36% chord which represents a physical distance of 0.15625% of pitch. The same cell width was used on both the suction and pressure surfaces of the blade. As was pointed out in Chapter 1, an advantage of the SDDMG method is the ability to easily insert a finer mesh level

to further refine an area. As an example of this capability, an eighth mesh level is inserted near the suction surface that tangentially refines the mesh where the  $y^+$  values exceed 40. The refinement reduces the physical distance to 0.078125% pitch. This region begins at 11% chord and ends at 50% chord. The mesh was generated and incorporated into the solution by iterating another 500 cycles. The resulting plot of  $y^+$  vs. percent chord is shown in figure 4.27. The effect of the finer mesh can be seen by the two discontinuities in the  $y^+$  curve. To further illustrate this point, another mesh level was generated to tangentially refine the mesh where the  $y^+$  values exceed 30 in mesh level 8. This refinement further reduces the physical distance to 0.039% pitch. Following the above procedure, the new finer mesh (Mesh Level 9) was incorporated into the solution by iterating another 500 cycles.

Figure 4.28 is a plot of  $y^+$  versus percent chord and summarizes the effect of mesh refinement on the suction surface boundary layer of this blade. The value of  $y^+$  drops from a maximum of about 75 to under 20, almost a factor of 4, by two local tangential refinements. The SDDMG method will yield a grid independent solution with enough refinement. This is illustrated in Figure 4.29 where the axial velocity,  $v_z$  obtained from the three mesh levels is plotted versus  $\theta$ , tangential distance from the suction surface of the blade at 30% chord. Each subsequent mesh refinement better defines the velocity profile.

The effect of the two additional mesh refinements on the wakes is shown in Figure 4.30. The additional mesh refinements improve the wake profile. The integrated value of the loss coefficient for the additional refinements is 0.0460 which is a 20% improvement in the prediction.

The above example of local grid refinement on the suction surface shows that refinement not only affects the local region but is carried through several coarser mesh levels to affect the downstream wake profile.

## 5.4 Rotor 37

The above cases test the domain decomposition method for flows which are nearly two-dimensional. To test out the 3D capabilities of this code, a transonic rotor that was designed, built and tested at NASA Lewis Research Center was chosen for simulation. See Refs. [35,36].

This geometry is called Rotor 37 and was used as a blind test case by the ASME/IGTI at its 1994 meeting to assess the predictive capabilities of then current turbomachinery CFD tools. A summary of the results of the proceedings can be found in Ref. [37]. Figure 4.31 shows the blades mounted on the rotor disk and some additional views of the rotor.

The rotor has 36 blades spinning at 17,188.7 RPM at the design condition. The aspect ratio is 1.19 and radius ratio is 0.70. The tip clearance gap was given as 0.0356 cm and is the distance between the rotor tip and the shroud at the design condition. The choking mass flow from experiment was given as 20.93 kg/sec. Overall performance data from aerodynamic probe surveys were taken at Station 1 and Station 4 which are indicated in Figure 4.32. LFA (Laser Fringe Anemometry) data was taken at Station 3 (which is slightly downstream of the rotor) which provides the wake characteristic.

At the 1994 meeting and since then, a number of papers have been written that coupled the experimental data with computational techniques to help understand the flow processes that develop in a high-speed rotor [38-40]. Some of the flow processes investigated are the effect of tip clearance, the effect of hub leakage, and wake mixing all of which affect the performance of the machine.

The mesh used to simulate Rotor 37 is shown in Figure 4.33. The block outlines can be seen along with the grid used at the hub. A total of six mesh levels are used in this calculation with five sub-mesh levels. A view of the grid at midspan is shown in Figure 4.34. The finest two levels of mesh refinement are used to resolve the leading and trailing edge circles of the rotor. Figure 4.35(a) shows two passages

of the mesh near the leading edge and Figure 4.35(b) shows the two passages of the mesh near the trailing edge of the rotor at midspan. The number of points and meshes in each level are given in Table 4.3.

Mesh Level	Axial Points	Radial Points	Tangential Points	Number of Meshes
1	81	51	21	1
2	121	51	11	2
3	201	51	11	2
4	181	51	11	2
5	21	51	11	4
6	21	51	11	4

Table 4.3. SDDMG Mesh Refinement Schedule for Rotor 37.

As was done in the previous sections, the radial direction is resolved with a stretched grid (i.e., non-uniform). The stretching ratio in the radial direction and number of radial grid points were identical to the mesh used by the author in the ASME Blind Test Case and Refs. 38 and 39. The stretching ratio,  $\sigma_r$ , was set to 1.19837 and was picked so that the last two cells in the radial direction could be modeled as tip clearance over the blades. The tip clearance is modeled as an orifice flow with no loss in mass, momenta, or energy [41]. A discharge coefficient of 0.5 is used to account for the vena contracta.

In simulating a rotor, particularly one designed for transonic flows, it is important to obtain the correct choke flow which is the maximum amount of air that can be passed through the machine for a given wheel speed. To obtain the choke flow, the simulation was run at a very low back pressure. The choke flow obtained from this simulation at 100% of design wheel speed is 20.76 kg/sec. The experimental value given is 20.93 kg/sec and the resulting error is 0.8%. Upon obtaining the choke flow, the back pressure is increased until the desired operating point on the speed line is reached. The operating point for this case is the same as was outlined in the ASME Blind Test Case which required a solution near 98% of choke flow. Figure 4.36 shows the speedline obtained from experiment. The speedline is a curve

obtained at constant speed while raising the back pressure. The pressure ratio is plotted as a function of mass flow ratio as is the efficiency. Station 4 is where the global computational grid is terminated and is about one chord downstream of the blade. The pressure ratio is defined as the mass-averaged total pressure of the flow at station 4 normalized by the total pressure at the inlet. The efficiency is the ratio of ideal work input to actual work input. The simulation shows that the pressure ratio of the experiment is achieved. The operating point that will be used in subsequent comparisons is indicated in the figure as high flow. The simulation was about a point lower in efficiency than experiment at the high flow condition. Two reasons can be attributed to this under-prediction. The first is the efficiency obtained from the the simulation is computed using all the mesh points available instead of the radial locations from which the data was obtained. Another reason for this under-prediction is flow leakage from a small gap between the stationary and rotating parts of the hub upstream of the rotor. This was uncovered after an investigation by numerical simulations and experiments [39]. Modeling of this leakage flow was not accounted for in the simulation.

The next set of figures shows axisymmetric mass-averaged profiles of the simulation at high flow versus experiment. Figures 4.37 is a plot of pressure ratio, temperature ratio, efficiency and absolute flow angle versus percent span. The total pressure deficit at 15% span is the effect of hub leakage in the profile of the experimental data which is not seen in the simulation. The over-prediction of pressure ratio and temperature ratio in the tip region is due to the coarse radial grid being used. In work to be published by VanZante [42], the effect of radial resolution on the profiles of pressure and temperature ratio is researched. The conclusions are that much finer radial resolution of the tip region results in much better comparisons to the data. Figure 4.37(c) is a plot of the efficiency at station 4 which compared very well to the data as did the absolute flow angle in Figure 4.37(d).

To conclude the comparisons to data, Figure 4.38 shows the wake in terms of relative Mach number plotted at midspan at Station 3 which is at 22% chord

downstream of the trailing edge. The simulation results were translated in the pitch-wise direction until the minimum relative Mach number of both plots were aligned with each other. The wake width agrees very well with the data as well as the level of the core flow. The depth of the wake is over-predicted by about 30%.

The strength of the SDDMG method is its ability to efficiently generate grid independent solutions. This means that as a region of the flow is refined, the solution on the succession of refined meshes converges to a grid independent solution at the leading edge. This is evident in Figure 4.39 which shows the wall static pressure on all the mesh levels in the region of the leading edge. If this refinement were continued, one would obtain a grid independent solution.

Comparison of an SDDMG solution to a single block stretched mesh solution for a three-dimensional geometry is difficult because the mesh construction is so different. For the channel flow presented in Section 4.1, the axial mesh spacing was identical for both methods and the solution resulting from tangential differences in the mesh systems was compared. For Rotor37, both the axial and tangential directions must be suitably gridded for the single block stretched mesh in order to obtain a credible solution. In order to compare the two procedures, the difference in the meshing strategy of the two procedures must be accounted for. The SDDMG method uses finer mesh levels to refine the mesh and the single block stretched mesh uses stretching to refine the mesh. In order to compare the two, the pitch-wise spacing from the finest and coarsest SDDMG mesh levels are used to fix the tangential spacing for the stretched mesh. The spacing from the finest SDDMG mesh level in the tangential direction will be the minimum spacing for the single block stretched mesh and will be chosen as the distance from the wall to the first tangential grid line. The coarsest spacing from the global parent mesh of the SDDMG system is chosen as the largest tangential cell spacing, i.e., distance between tangential grid lines, in the field for the single block stretched mesh. To generate the single block stretched mesh, the number of points and pitch-wise stretching ratio is varied until the desired cell spacings are achieved. The number of points in the pitch-wise di-

rection that achieves these spacings is 71 and the corresponding stretching ratio is 1.1089.

The mesh spacing for the axial direction presents another difficulty. The SDDMG system utilized a total of 160 uniform axial mesh cells to define the blade surface in addition to two levels of local meshes to further define the leading and trailing edge circles only. A single block stretched mesh with this level of refinement requires roughly 1,850,000 grid points of which 261 points lie along the chord. The stretching ratio for this mesh is 1.012. These numbers are obtained by fixing the largest axial cell length on the blade surface to the axial cell length from Mesh Level 5 from the SDDMG mesh presented in Table 4.3. The smallest axial cell length is set to the axial cell length from Mesh Level 7 also from Table 4.3. In lieu of this, the same number of points along the chord is set equal to that across the pitch. The mesh stretching in the axial direction is that used in the pitch-wise direction. The total number of cells for the single block stretched mesh is 574,600 with 169 cells describing the axial direction from the inlet to the outlet of which 70 cells are along the chord. The radial direction is described by 50 cells and 70 cells describe the pitch-wise direction. The radial distribution of cells is identical in both systems.

In comparison, the total number of SDDMG cells is 660,000 but with more cells distributed axially on the blade.

A solution was obtained for the single block stretched mesh near the point on the speedline referred to in the ASME Test Case Exercise as high flow. This is the point on the speedline at which the integrated mass flow of the solution is 98% of the choke mass flow. The first comparison of the two systems is their convergence history shown in Figure 4.40. The global parent mesh convergence rate represents the SDDMG system and is represented by circles. The max residual of the continuity equation is plotted versus normalized iterations in Figure 4.40(a) and the percent difference in massflow is plotted versus normalized iterations in Figure 4.40(b). The normalized iterations are obtained by accounting for the disparity in CPU times for one iteration between the SDDMG system and the single block

stretched mesh system. The CPU time for the single block stretched mesh is 33.3 secs./iteration and the CPU time for the SDDMG system is 56.3 secs./iteration. The ratio of SDDMG CPU time to the single block stretched mesh CPU time is 1.69. To compare convergence history, the iteration count for the single block stretched mesh is divided through by 1.69. Figure 4.40(a) shows that the SDDMG achieves convergence (measured by the beginning of the limit cycle) in a little over 1000 iterations. The single block stretched mesh achieves convergence (measure by the beginning of the limit cycle) in a little over 2000 iterations. Thus, the SDDMG system converges twice as fast as the single block stretched mesh. Figure 4.40(b) shows the convergence history of the percent difference in massflow. The same definition of massflow as well as percent difference in massflow used in Section 4.1 is applied here. The SDDMG system has reduced the percent difference in massflow to less than half a percent within 1000 iterations which is consistent with the convergence of the maximum error presented in Figure 4.40(a). The single block stretched mesh, however, exhibits a long wavelength oscillation that is still perceptible after 1500 iterations. The error is reduced to less than half a percent for the single block stretched mesh within 1300 iterations. An interesting observation is that the single block stretched mesh shows a much larger variation in massflow initially than is seen in the SDDMG convergence. Both systems began from the same initial guess.

Figure 4.41 compares the blade loading predicted by the SDDMG system to that predicted by the single block stretched mesh versus percent chord at midspan. An immediate observation from the figure is how well the shock is defined at about 42% chord from the SDDMG system. This is clearly due to the better axial definition of the blade in the SDDMG system. The single block stretched mesh defines the shock over three grid points which span about 4% chord. Downstream of the shock, the SDDMG system predicts more diffusion, which is evident by the higher rise in the level of static pressure from downstream of the shock to the trailing edge than the single block stretched mesh.



The passage shock and suction surface boundary layer growth at midspan are illustrated in Figures 4.42 with contours of relative Mach number for the SDDMG system and the single block stretched mesh. The solution from the SDDMG system has a thinner boundary layer, evident by the contours of relative Mach number closer to the suction surface of the blade in Figure 4.42(a) compared to the same contours of the single block stretched mesh in Figure 4.42(b). Therefore, the SDDMG system predicts less blockage than the single block stretched mesh. Blockage refers to an effective reduction in flow area and affects the work output and mass flow capacity of axial compressors [48]. Because there is less blockage in the SDDMG system, there is more diffusion which is consistent with the comparisons of blade loadings from Figure 4.41.

To complete the comparison, Figure 4.43 compares the wake profile of the SDDMG system and the single block stretched mesh to the experimental data at midspan at Station 3. Both the SDDMG solution and single block stretched mesh solution predict the wake width well. However, the SDDMG system attains the right level of relative Mach number in the freestream while the single block stretched mesh solution over-predicts the level of relative Mach number. Thus, the amount of diffusion through the compressor is well predicted by the SDDMG solution. The wake depth is over-predicted in both simulations but less for the SDDMG system. The percent error in predicting the minimum wake value is 25% for the SDDMG system and 45% for the single block stretched mesh. The SDDMG system is predicting the correct diffusion because it is predicting less blockage than the single block stretched mesh. Getting the blockage right is critical in the design of high speed compressors [48].

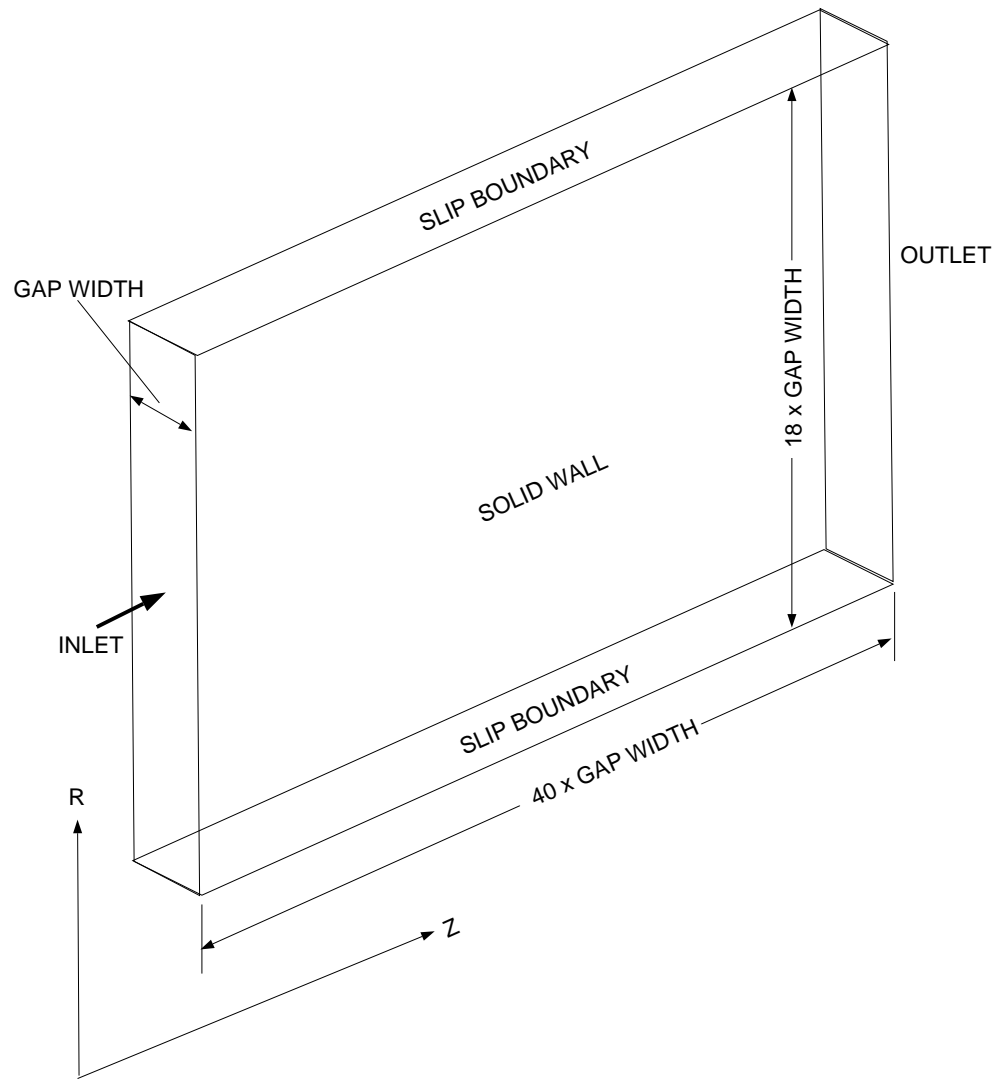


Figure 4.1. Schematic of channel flow geometry.

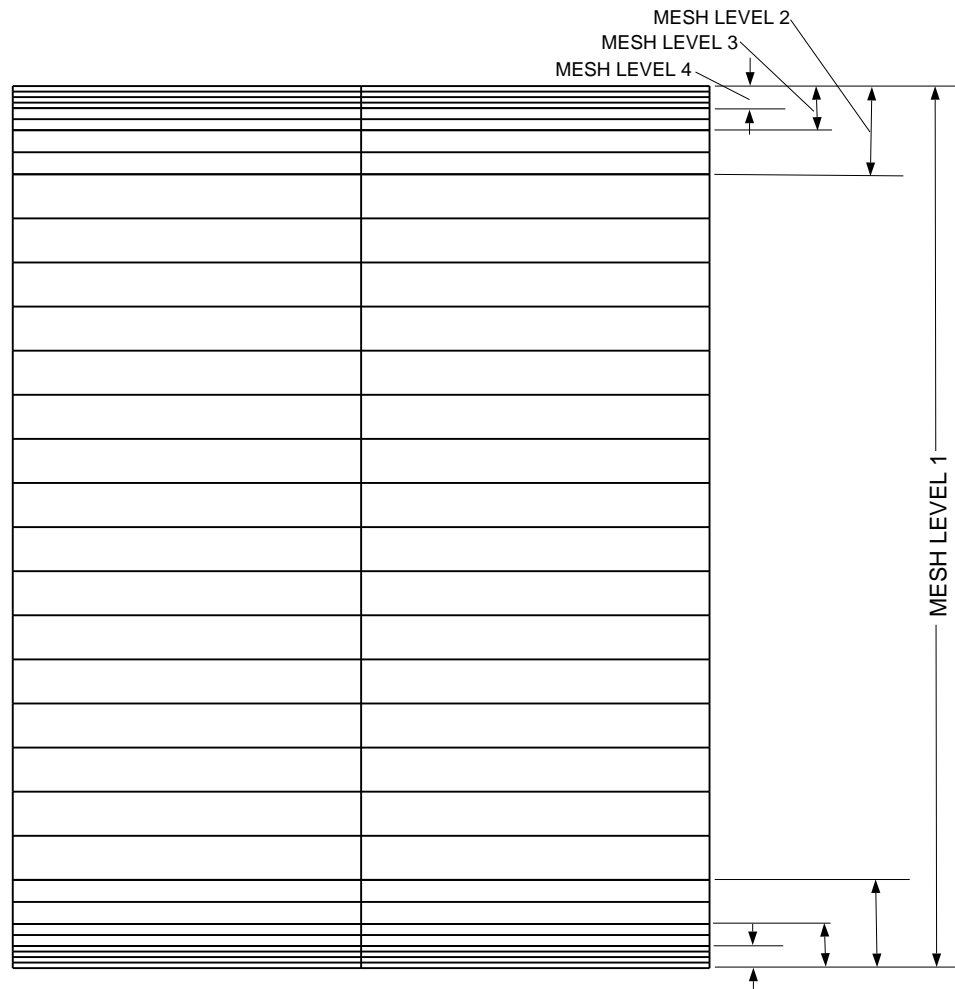


Figure 4.2. Sectional view of the SDDMG mesh at midspan with 4 mesh levels for the channel geometry.

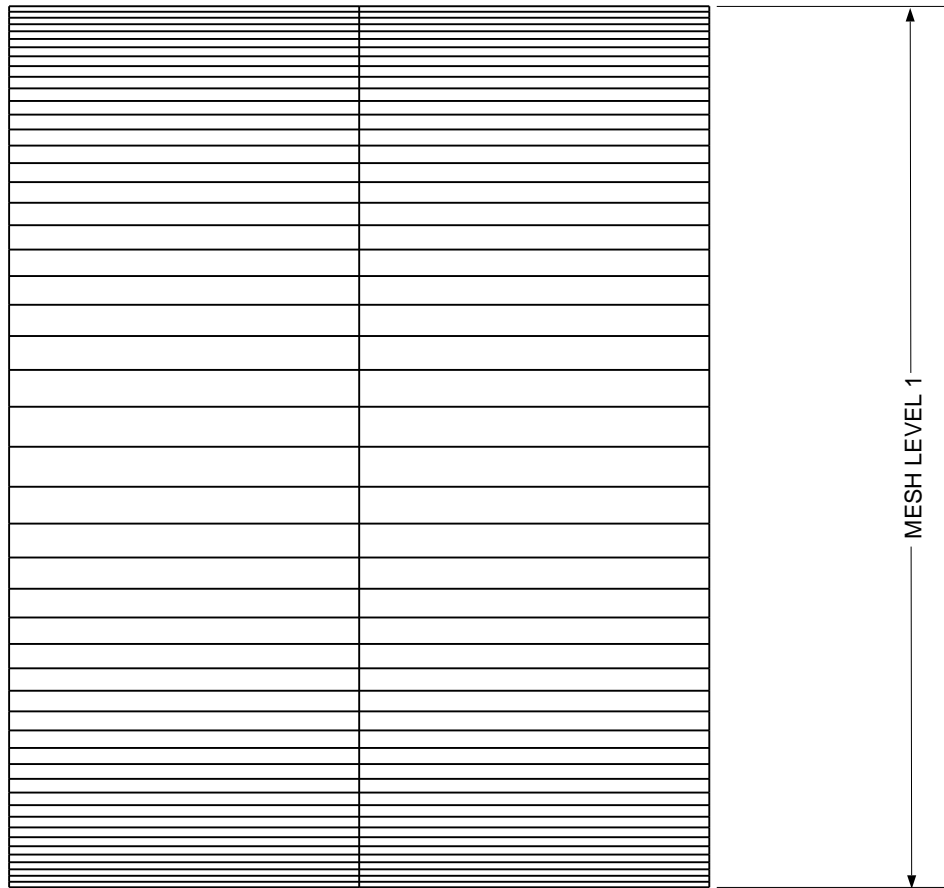


Figure 4.3. Sectional view of the stretched mesh at midspan for the channel geometry. Mesh size is 101 x 11 x 51.

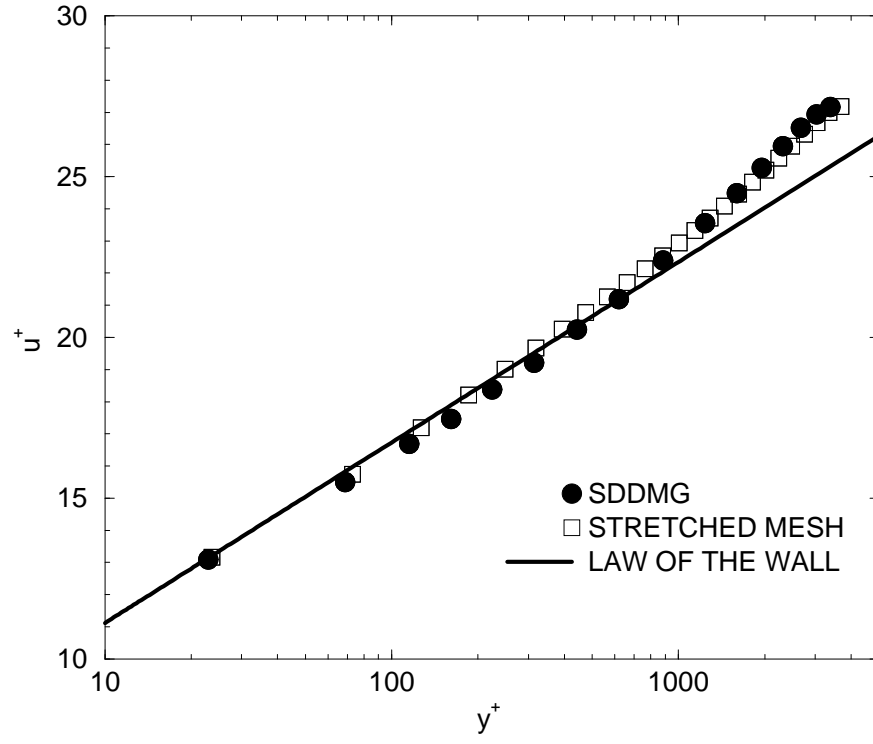


Figure 4.4. Comparison of  $u^+$  vs.  $y^+$  for SDDMG and stretched mesh with law of the wall at 90% channel length and midspan.

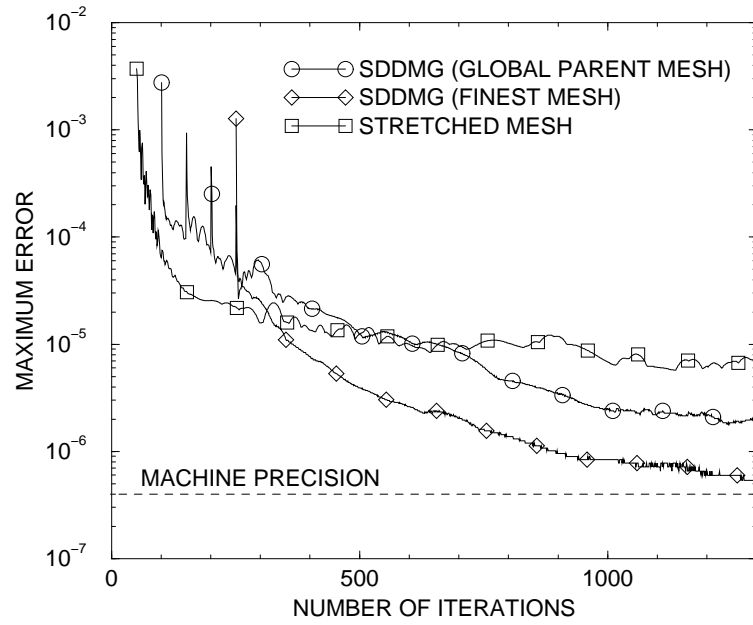


Figure 4.5(a). Convergence history of maximum error for two mesh levels of SDDMG system and stretched mesh for channel flow.

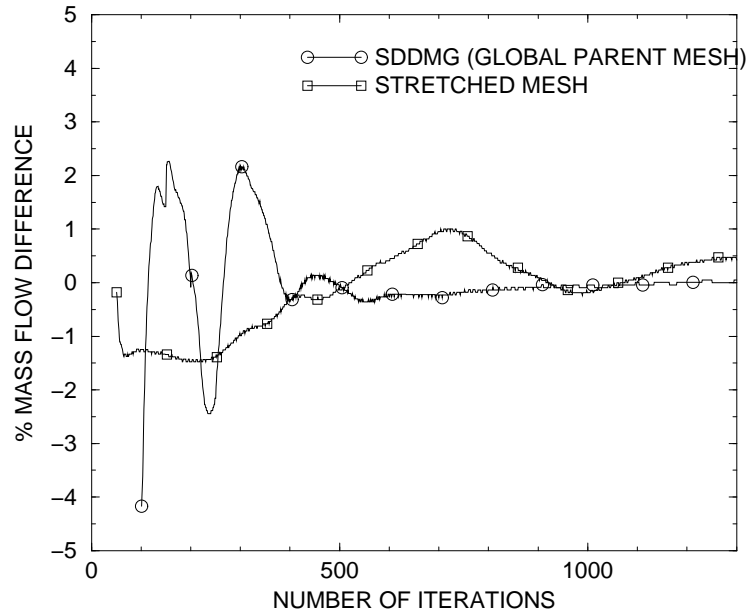


Figure 4.5(b). Convergence history of percent massflow difference for SDDMG global parent mesh and stretched mesh for channel flow.

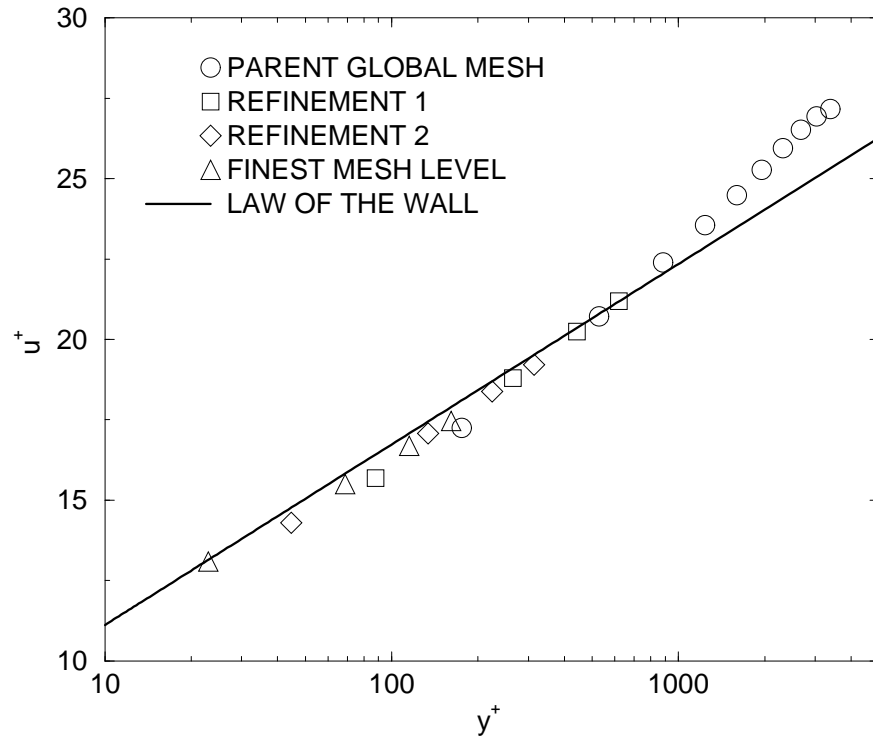


Figure 4.6. Comparison of  $u^+$  vs.  $y^+$  for each SDDMG mesh refinement for channel flow.

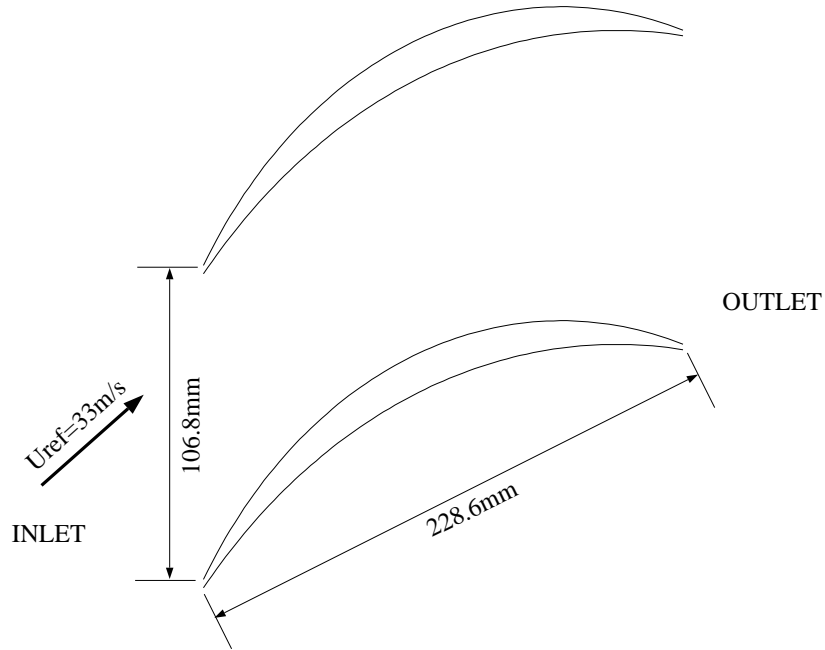


Figure 4.7. Schematic of the ARL Compressor Cascade.

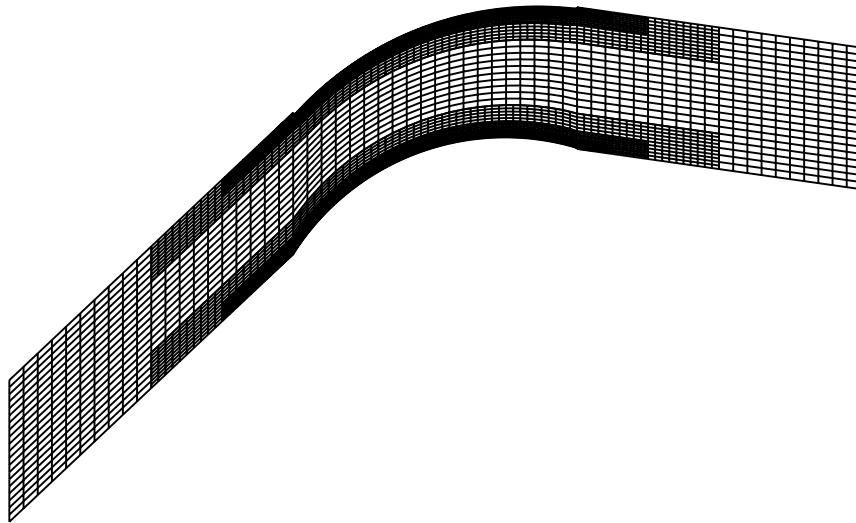


Figure 4.8. Sectional view of the ARL compressor SDDMG mesh at midspan with 7 mesh levels.



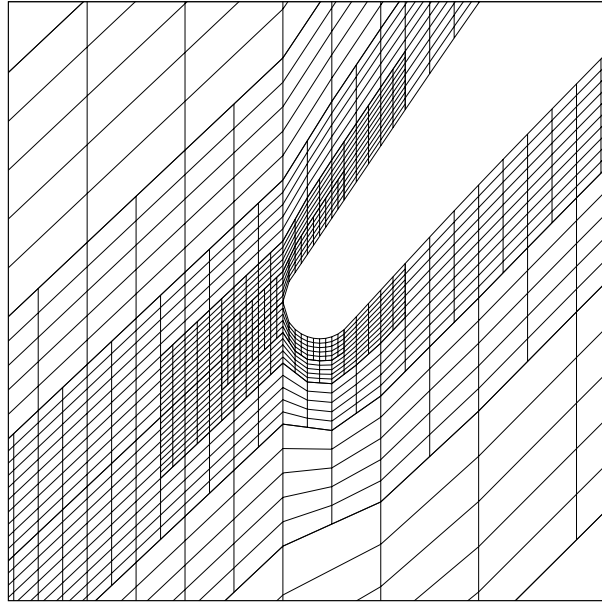


Figure 4.9(a). SDDMG mesh near the leading edge region of the ARL compressor at midspan.

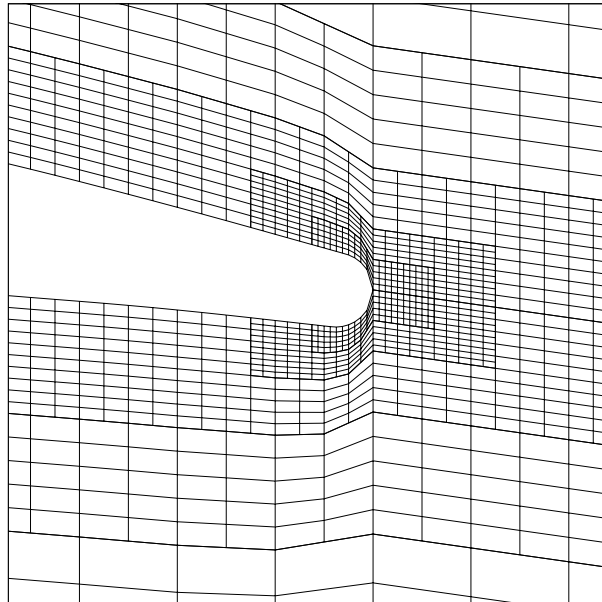


Figure 4.9(b). SDDMG mesh near the trailing edge region of the ARL compressor at midspan.

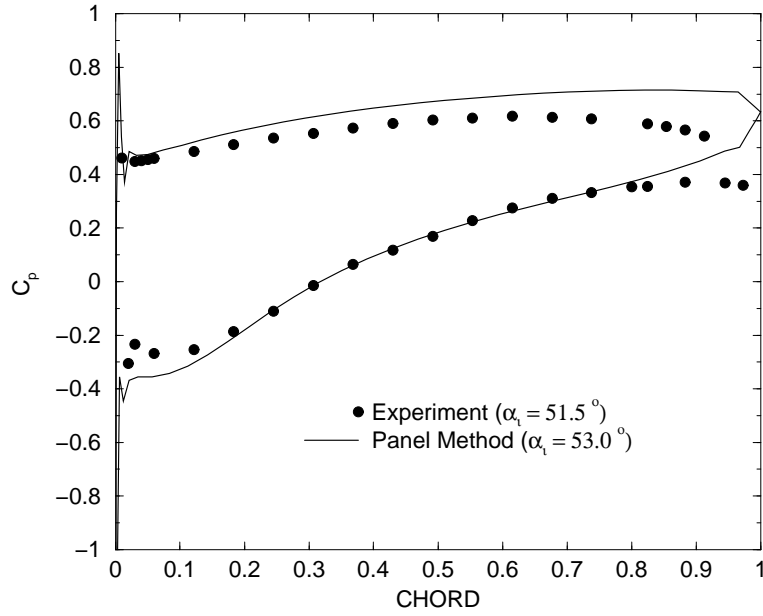


Figure 4.10. Comparison of Blade Surface  $C_p$  vs. percent chord at midspan for experimental data and panel method.

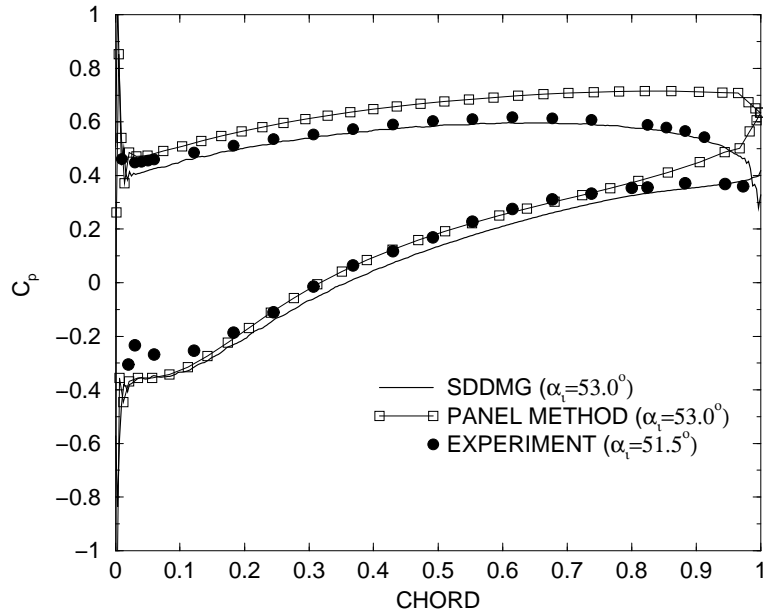


Figure 4.11. Comparison of Blade Surface  $C_p$  vs. percent chord at midspan for experimental data, panel method and SDDMG.

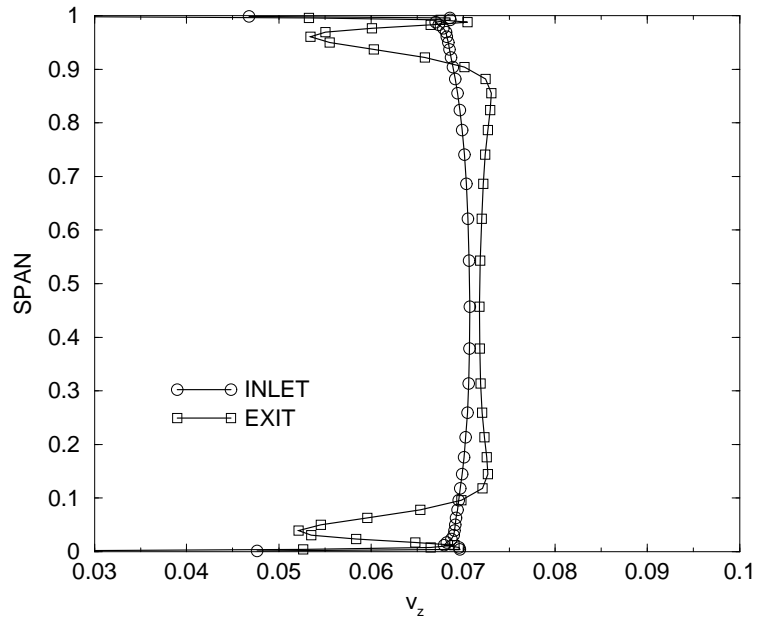


Figure 4.12. Percent span vs. axisymmetric axial velocity at inlet and outlet.

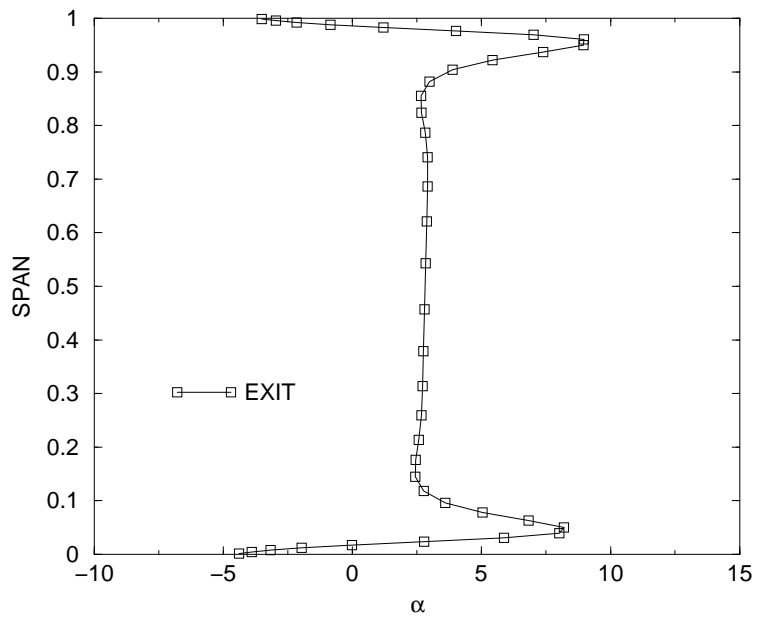


Figure 4.13. Percent span vs. axisymmetric flow angle,  $\alpha$ , at outlet.

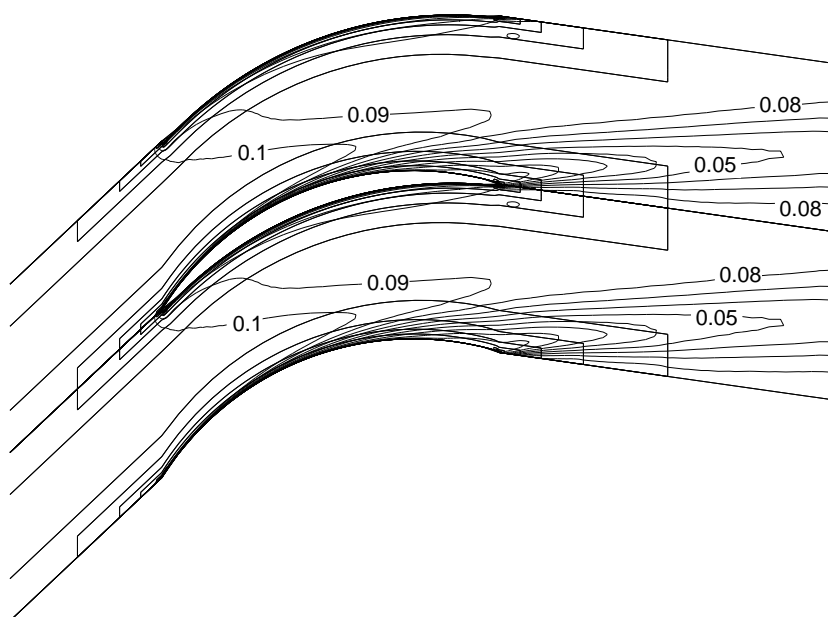


Figure 4.14. Contours of velocity magnitude at midspan.

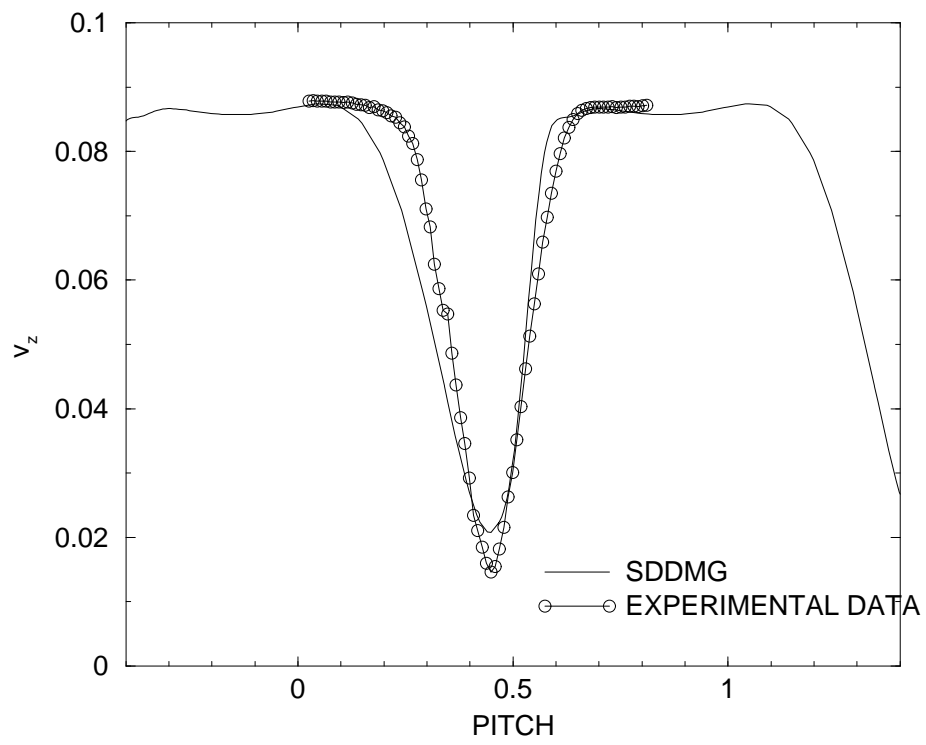


Figure 4.15. Comparison of axial velocity,  $v_z$  vs. pitch at midspan for SDDMG and experimental data.

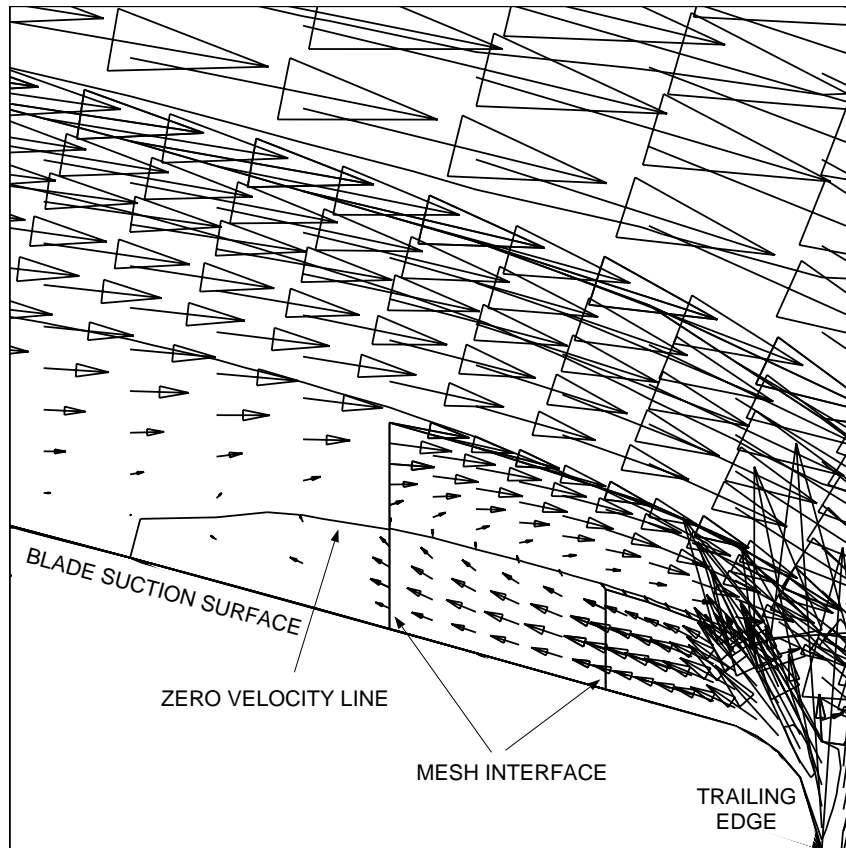


Figure 4.16. Velocity vectors near the suction surface trailing edge midspan for SDDMG.

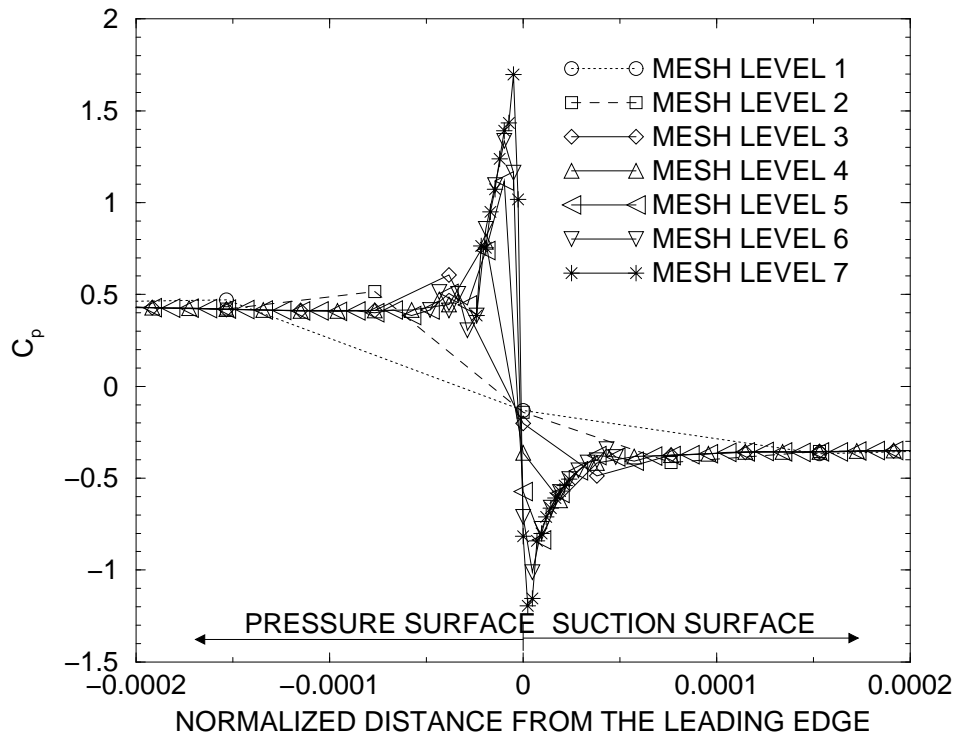


Figure 4.17. Convergence of Blade Surface  $C_p$  for all the SDDMG mesh levels vs. normalized distance from the leading edge at midspan.

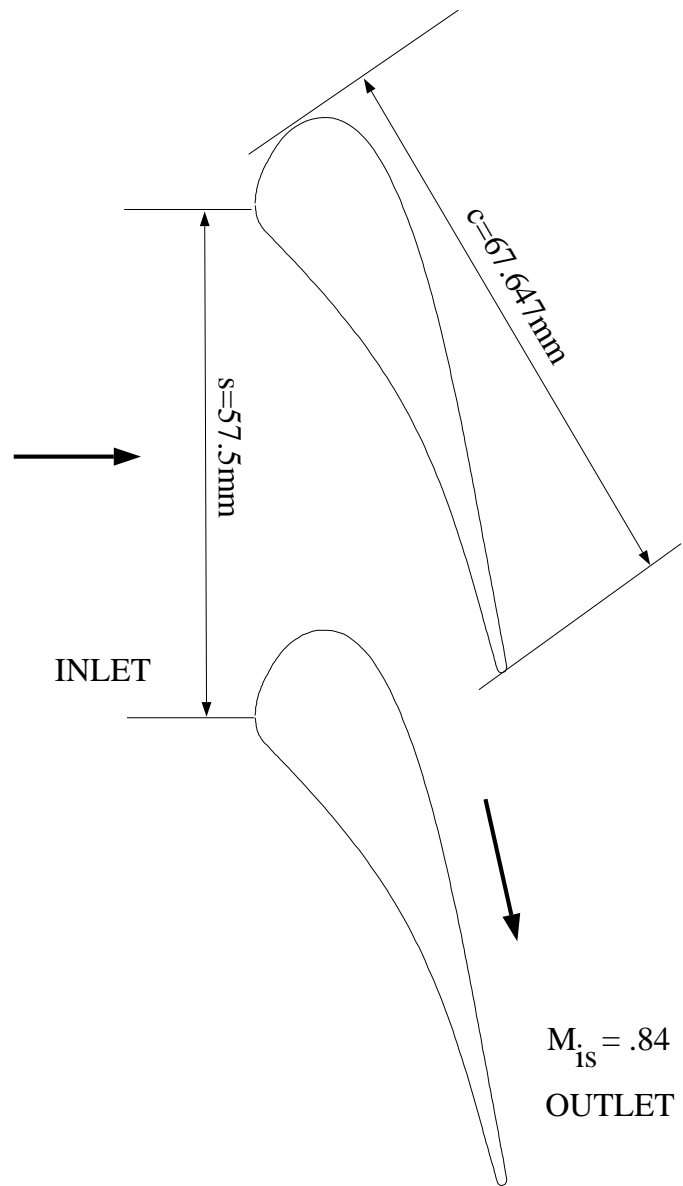


Figure 4.18. Schematic of VKI Turbine Cascade.



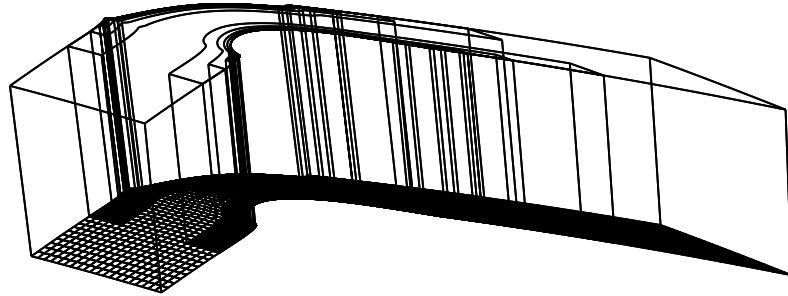


Figure 4.19. Perspective view of the VKI turbine cascade geometry with 6 mesh levels outlined.

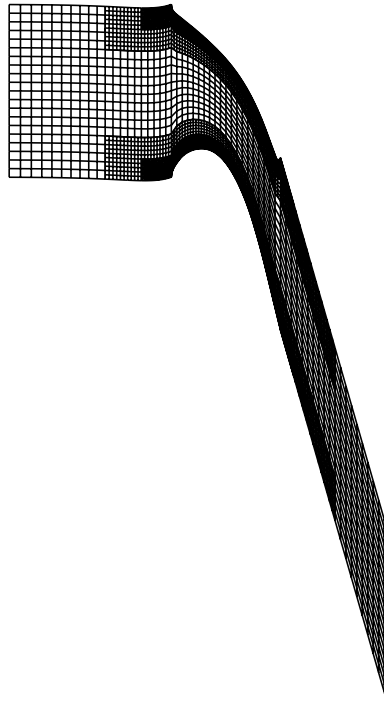


Figure 4.20. Sectional view of the VKI turbine SDDMG mesh at midspan with 6 mesh levels.

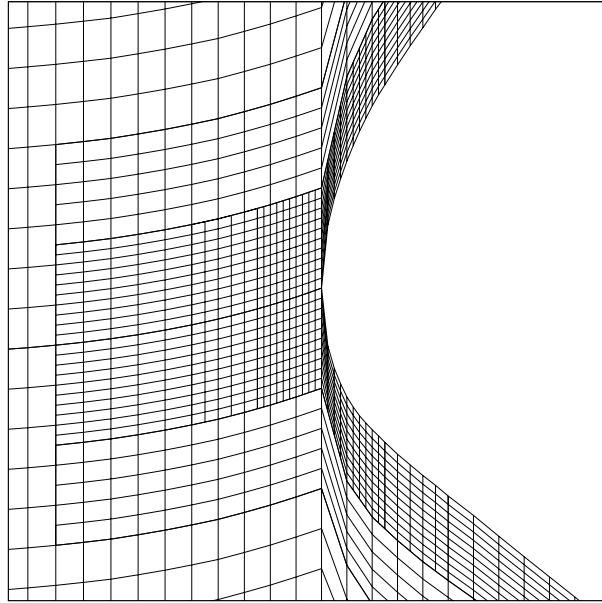


Figure 4.21(a). Two passages of the SDDMG mesh near the leading edge of the VKI turbine at midspan.

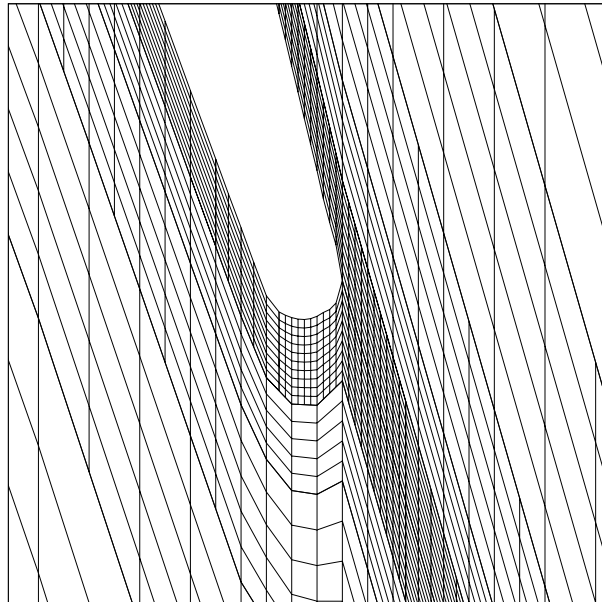


Figure 4.21(b). Two passages of the SDDMG mesh near the trailing edge of the VKI turbine at midspan.

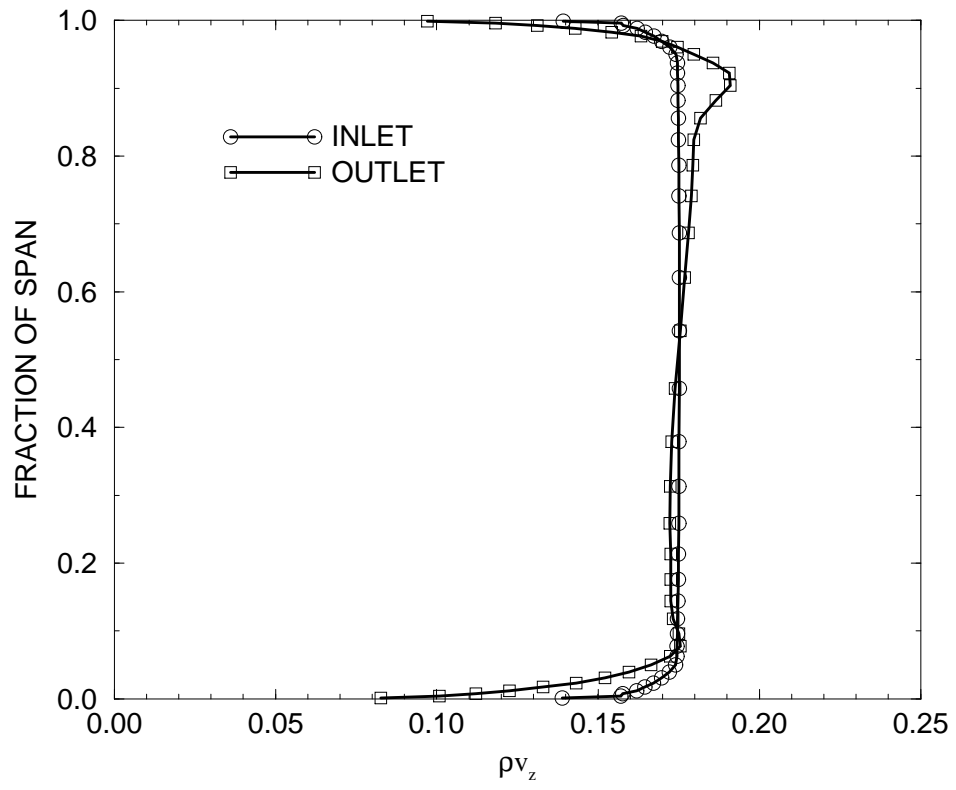


Figure 4.22. Axisymmetric axial momentum versus fraction of span at the inlet and exit.

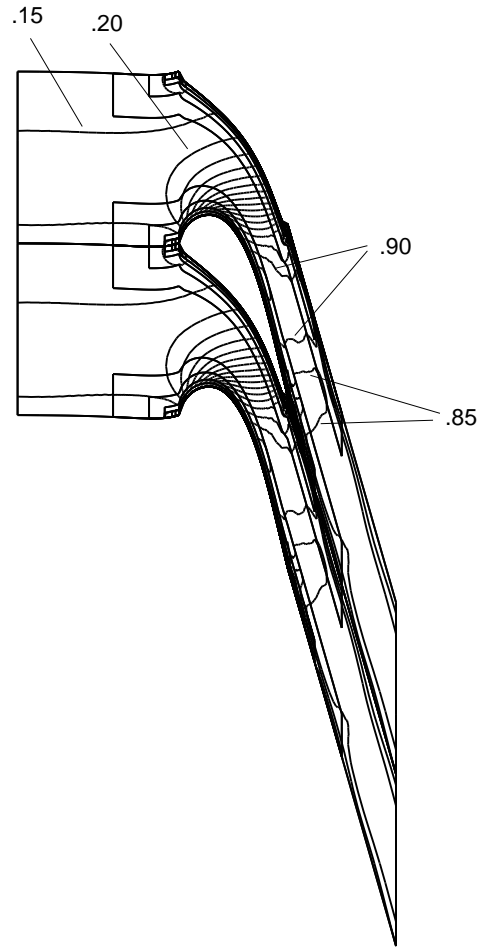


Figure 4.23. Mach Number contours at midspan in increments of 0.05. VKI turbine cascade.

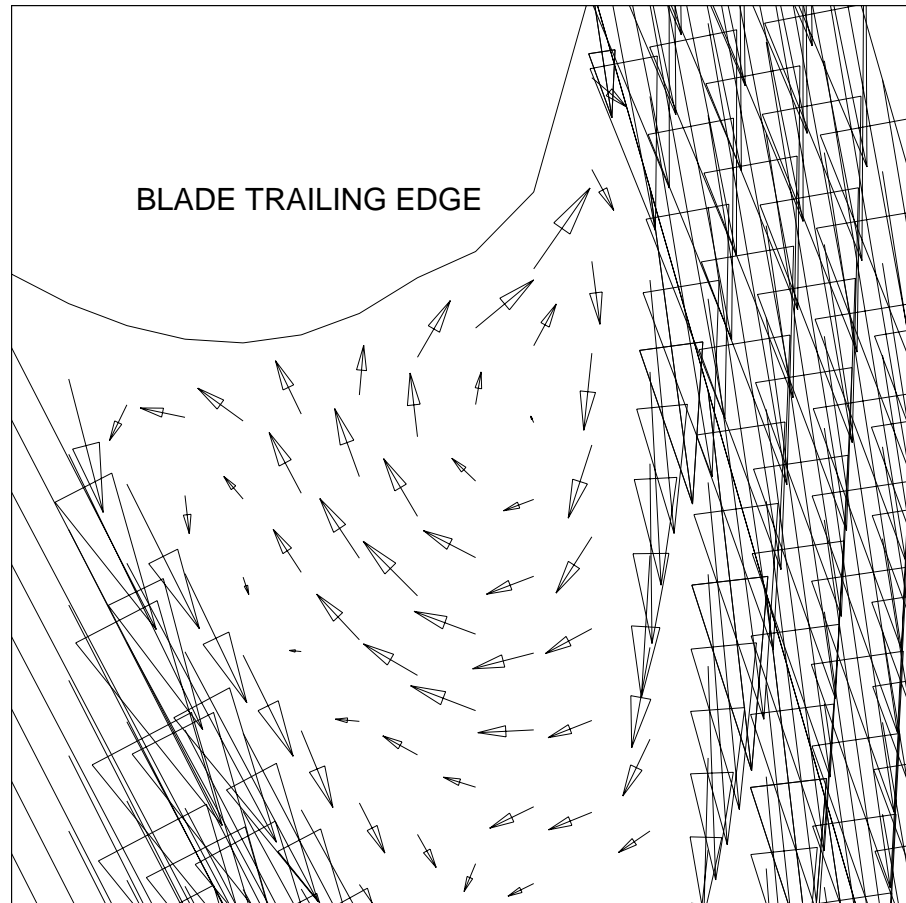


Figure 4.24. Flow vectors near the trailing edge at midspan. VKI turbine cascade.

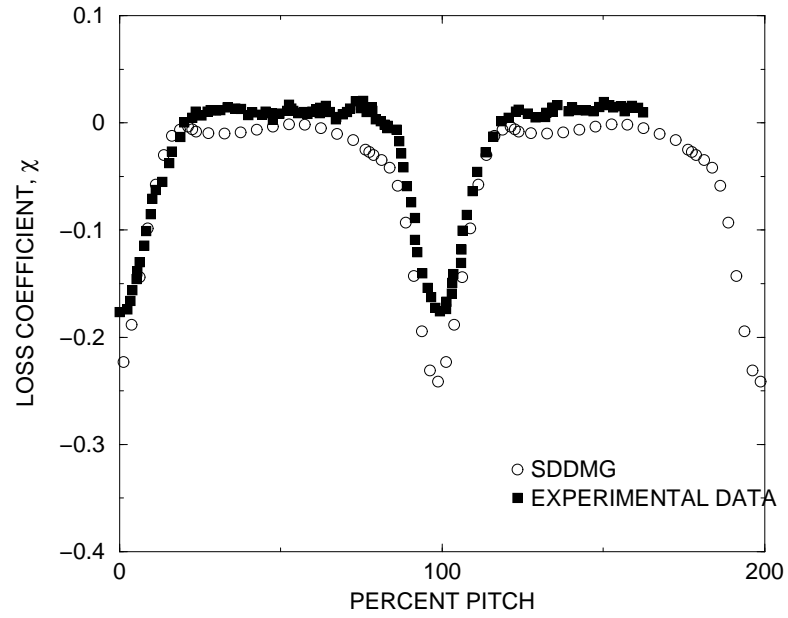


Figure 4.25. Loss coefficient at 143% chord at midspan. VKI turbine cascade.

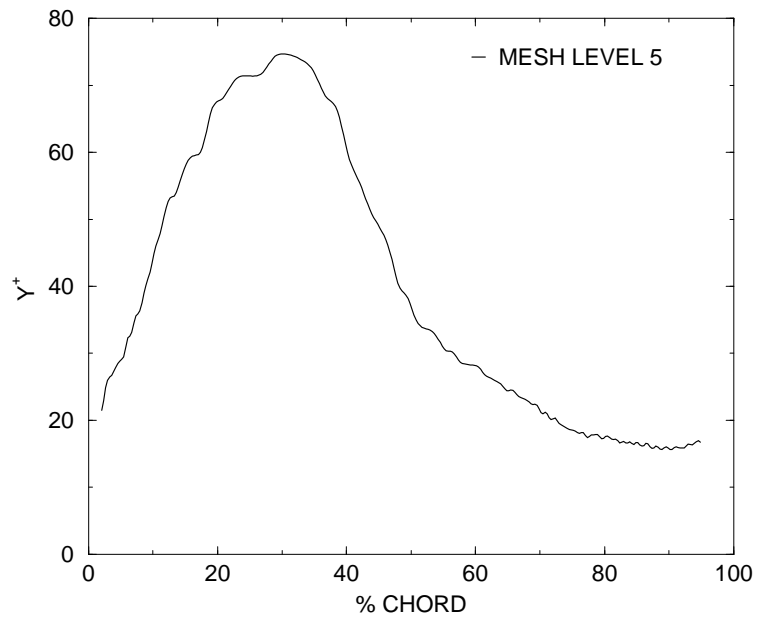


Figure 4.26.  $y^+$  versus % chord on suction surface at midspan. VKI turbine cascade.

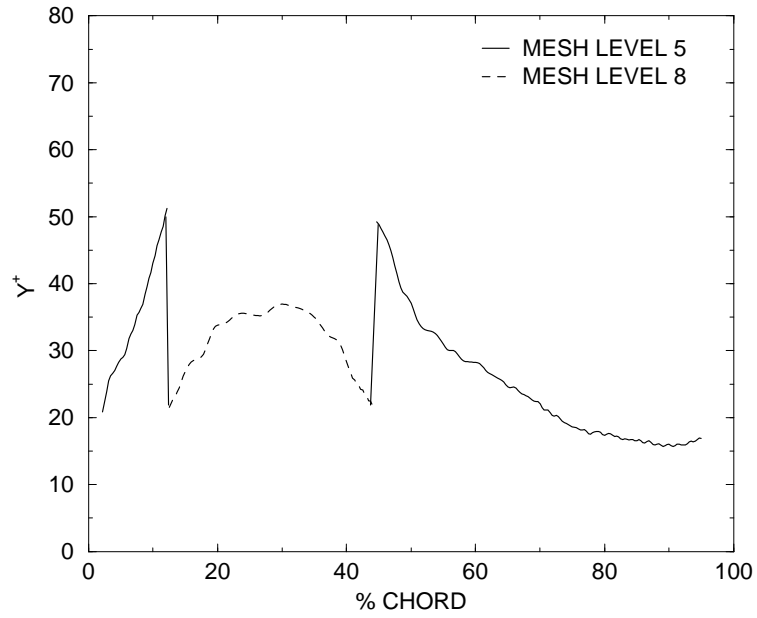


Figure 4.27.  $y^+$  versus % chord on suction surface at midspan. VKI turbine cascade.

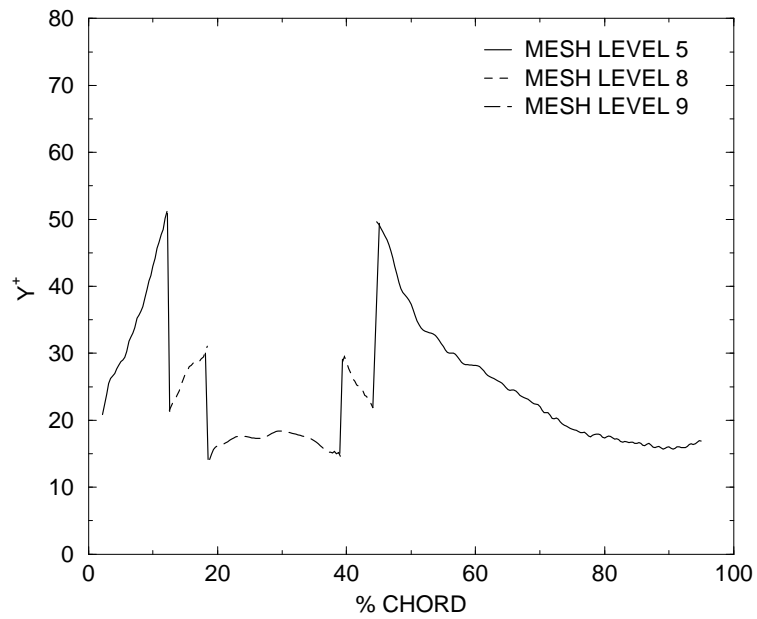


Figure 4.28.  $y^+$  versus % chord on suction surface at midspan. VKI turbine cascade.

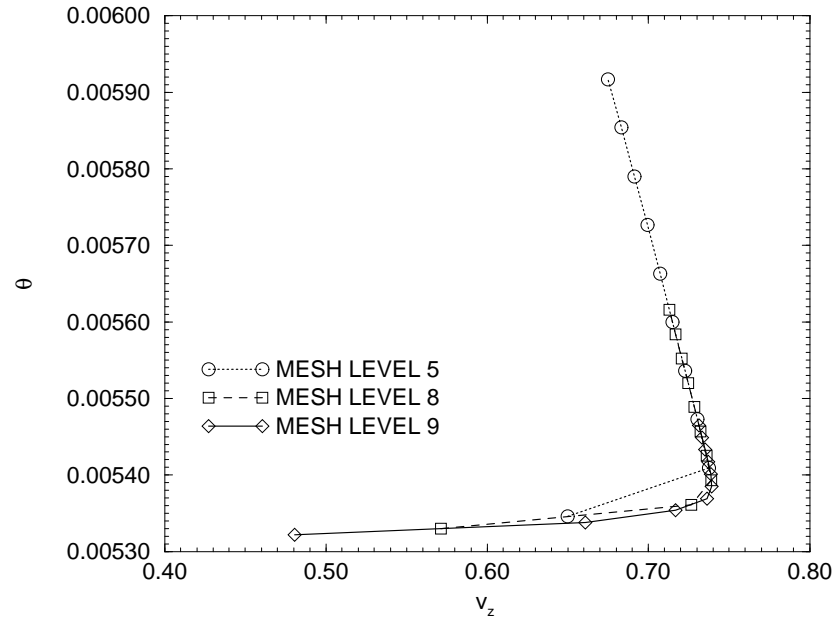


Figure 4.29.  $v_z$  versus  $\theta$  on suction surface at midspan. VKI turbine cascade.

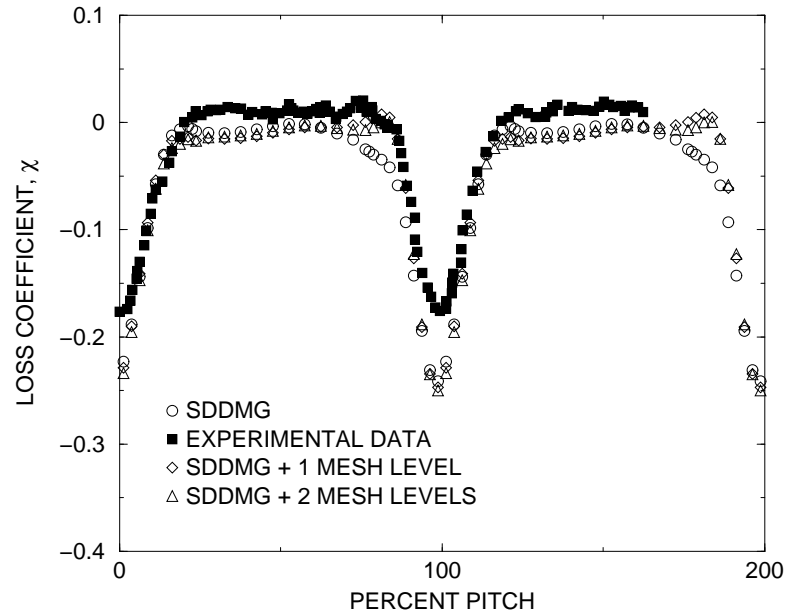
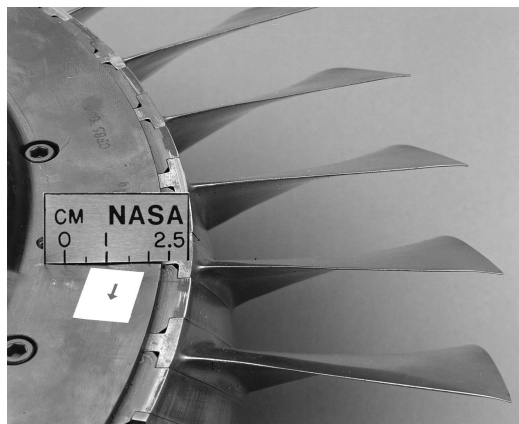


Figure 4.30. Comparison of Loss coefficient at 143% chord at midspan. VKI turbine cascade.

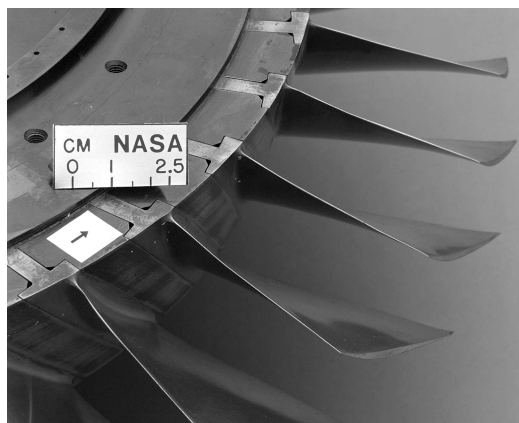




a) Wheel Assembly



b) Channel View  
Looking Downstream



c) Channel View  
Looking Upstream

Figure 4.31. Rotor 37.

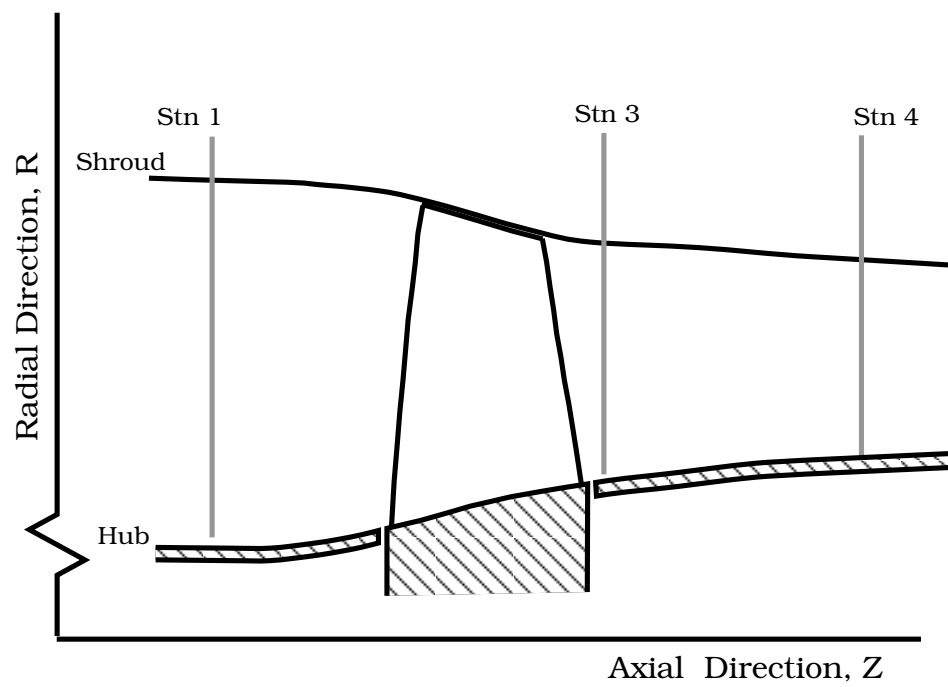


Figure 4.32. Rotor 37 flowpath indicating locations of Stations 1, 3 & 4.

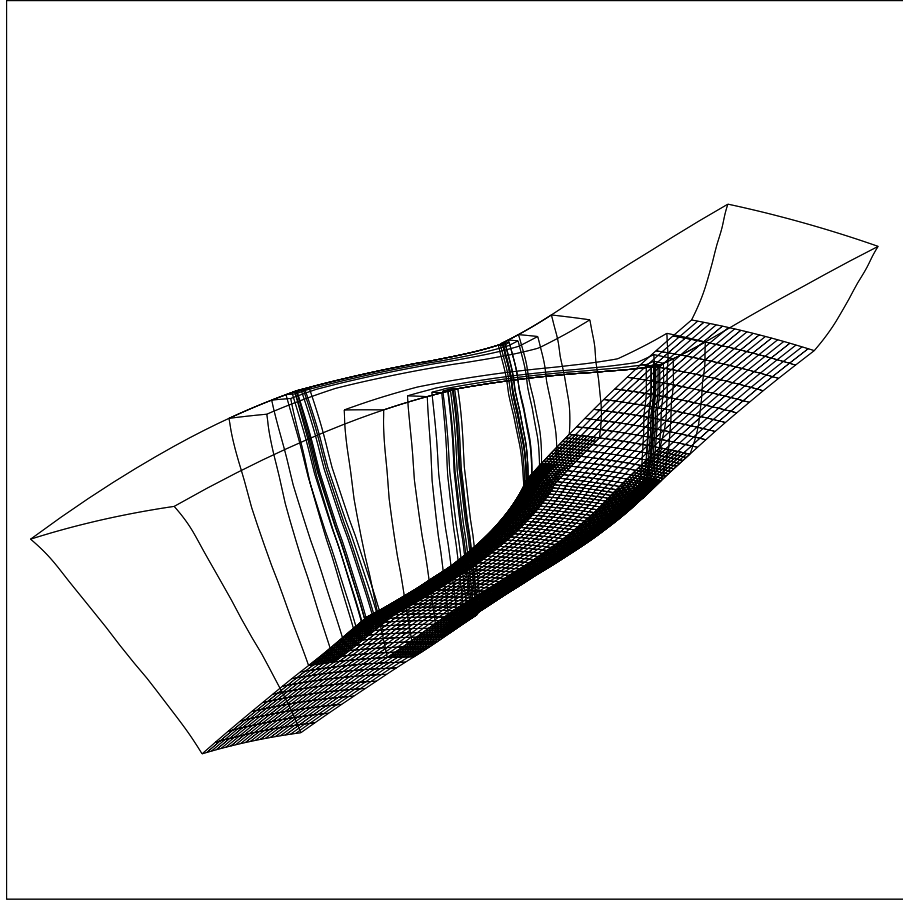


Figure 4.33. Perspective view of the grid outline for a passage of Rotor 37.

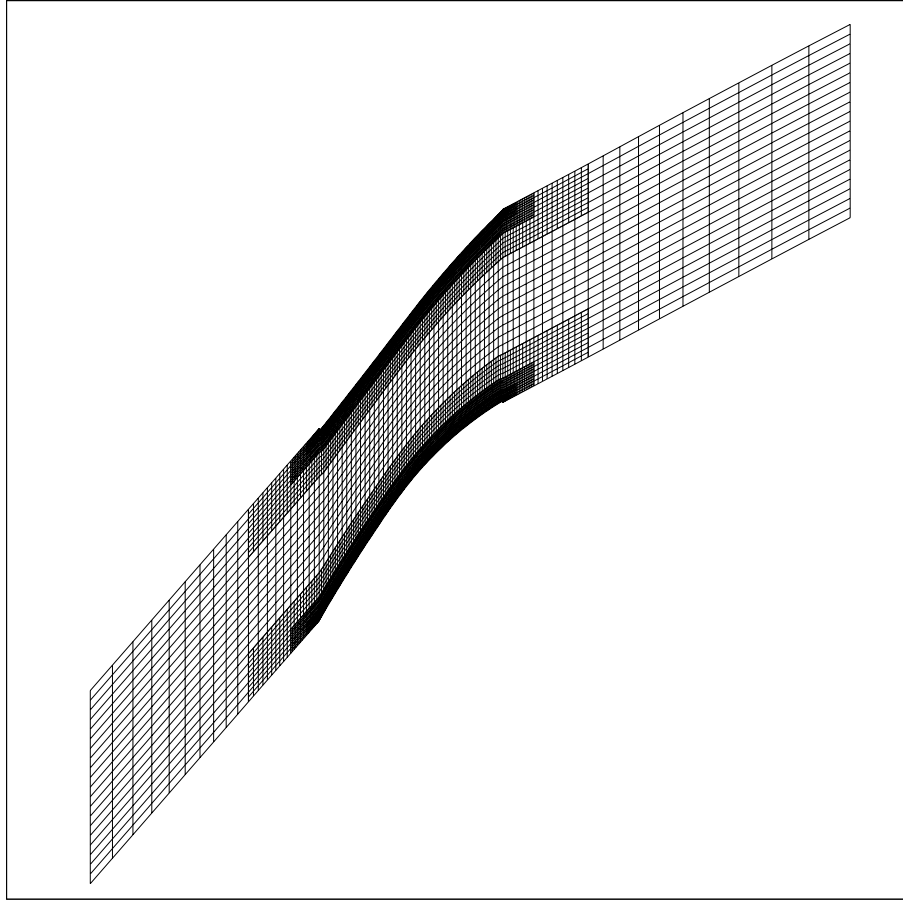


Figure 4.34. Sectional view of the Rotor 37 SDDMG mesh at midspan.

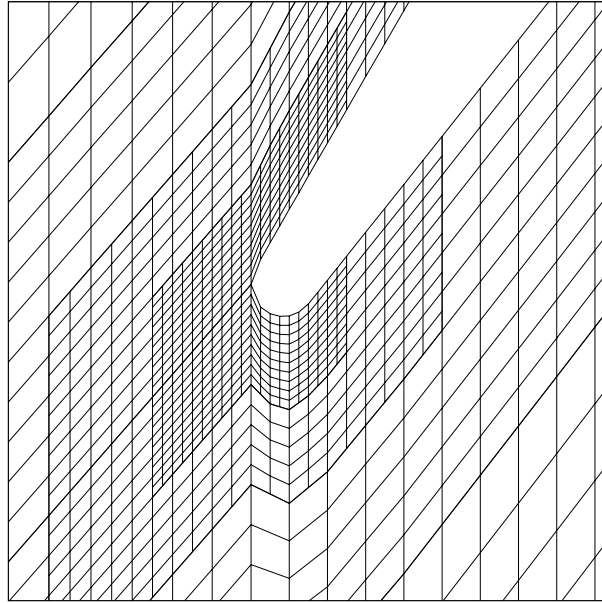


Figure 4.35(a). Two passages of the SDDMG mesh near the leading edge of the rotor at midspan.

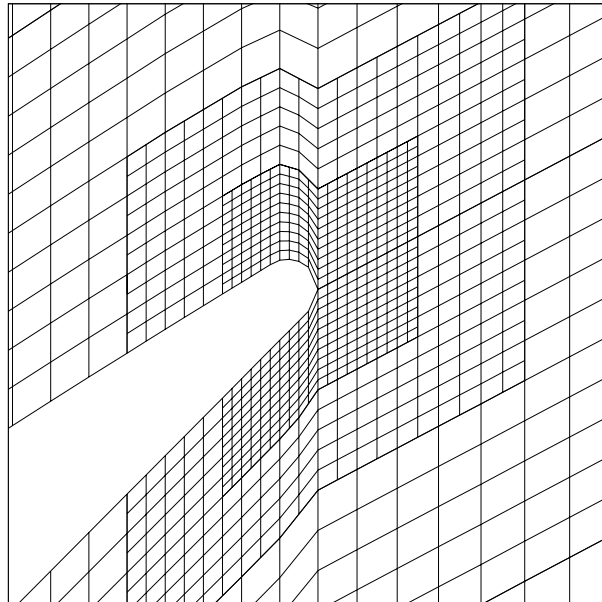


Figure 4.35(b). Two passages of the SDDMG mesh near the trailing edge of the rotor at midspan.

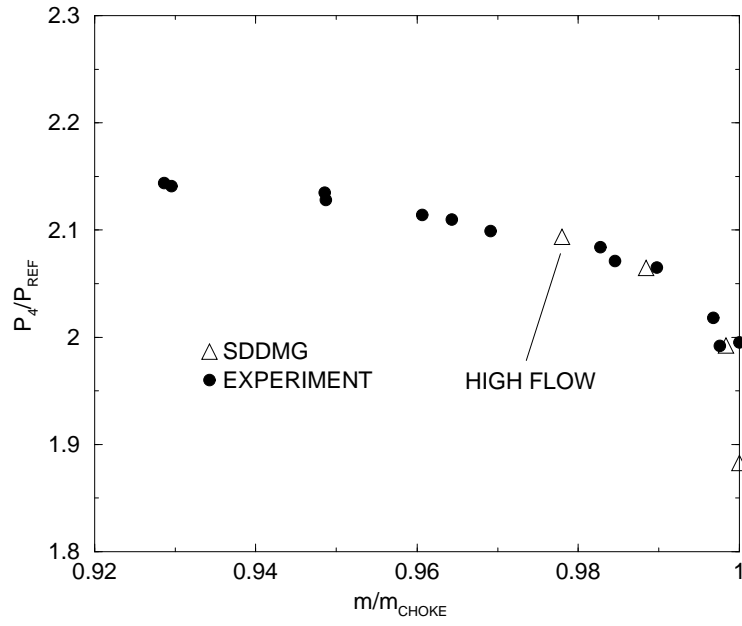


Figure 4.36(a). Mass flow ratio vs. Pressure Ratio at design speed for Rotor 37.

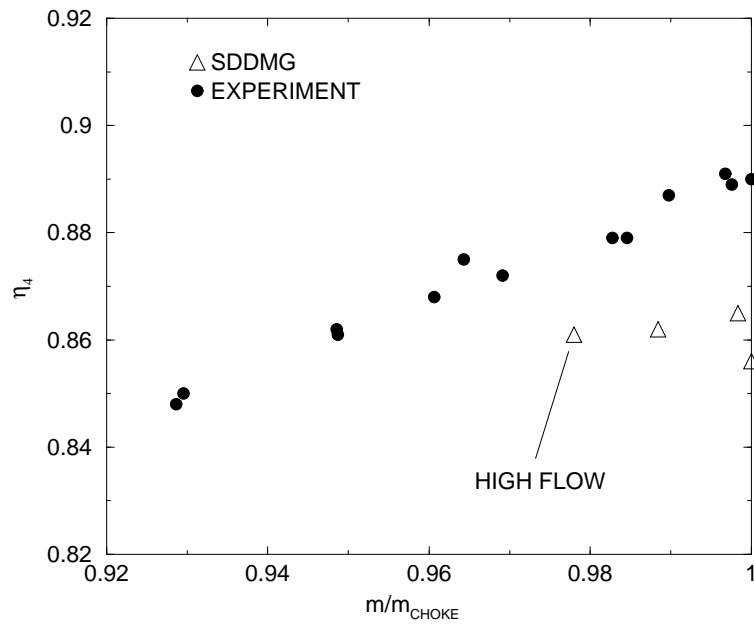


Figure 4.36(b). Mass flow ratio vs. Efficiency at design speed for Rotor 37.

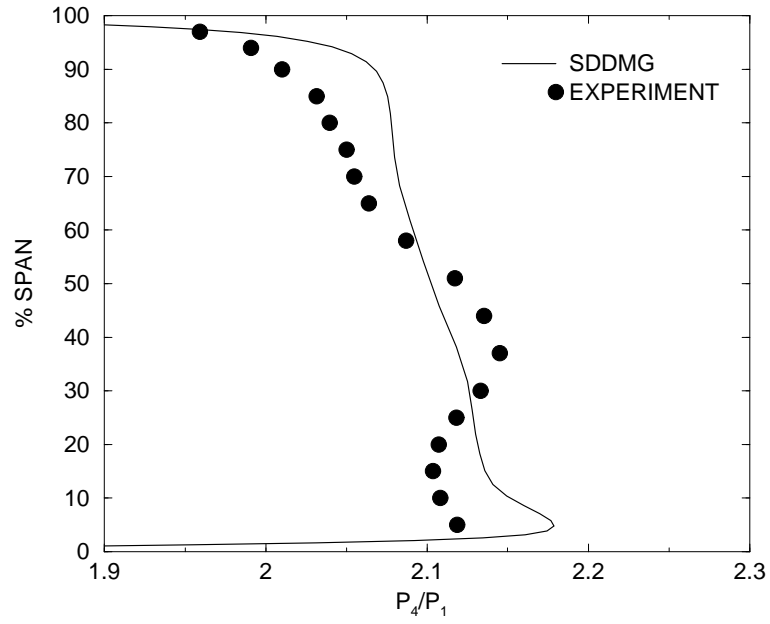


Figure 4.37(a). Mass-averaged pressure ratio vs. percent span at Station 4 for Rotor 37.

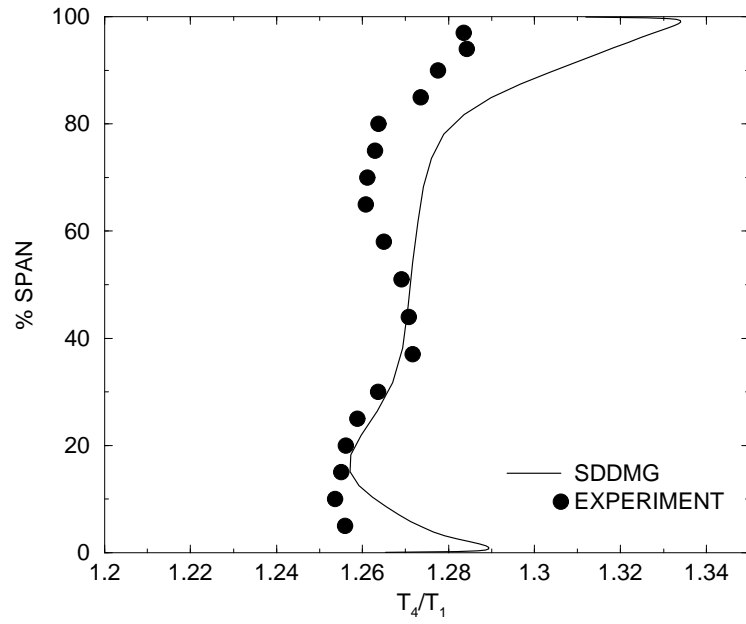


Figure 4.37(b). Mass-averaged temperature ratio vs. percent span at Station 4 for Rotor 37.

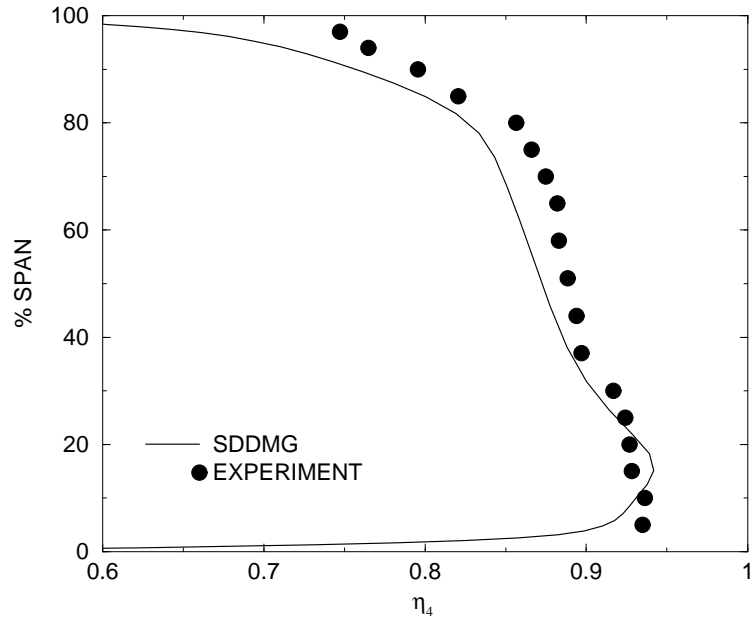


Figure 4.37(c). Efficiency vs. percent span at Station 4 for Rotor 37.

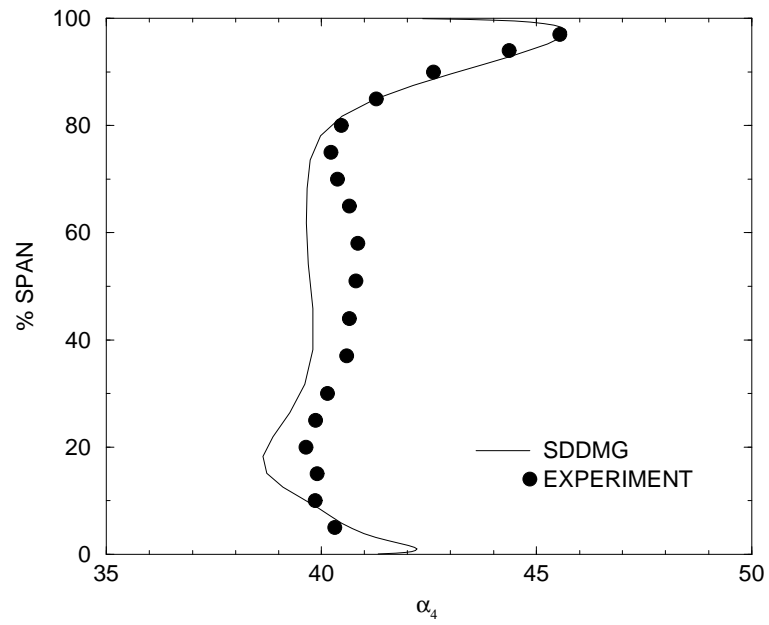


Figure 4.37(d). Absolute flow angle,  $\alpha$ , vs. percent span at Station 4 for Rotor 37.



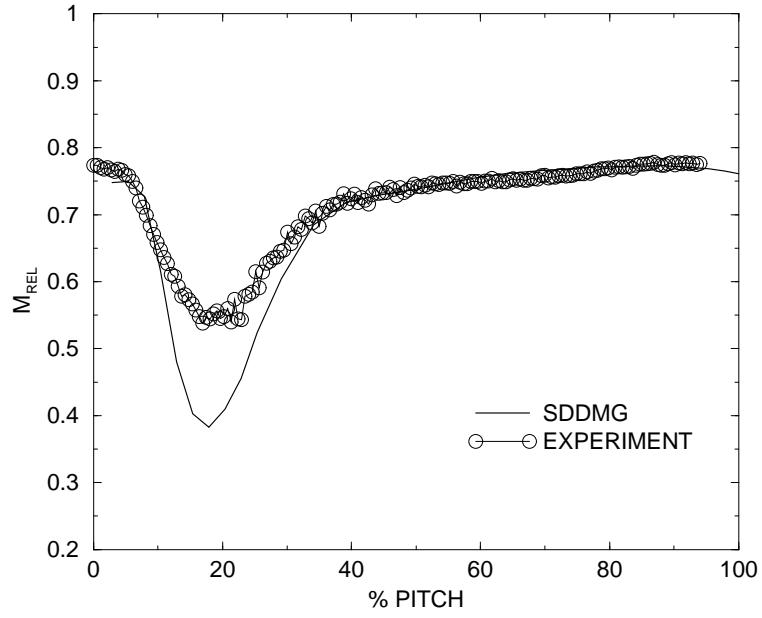


Figure 4.38. Percent pitch vs. relative Mach number at midspan at Station 3 for Rotor 37.

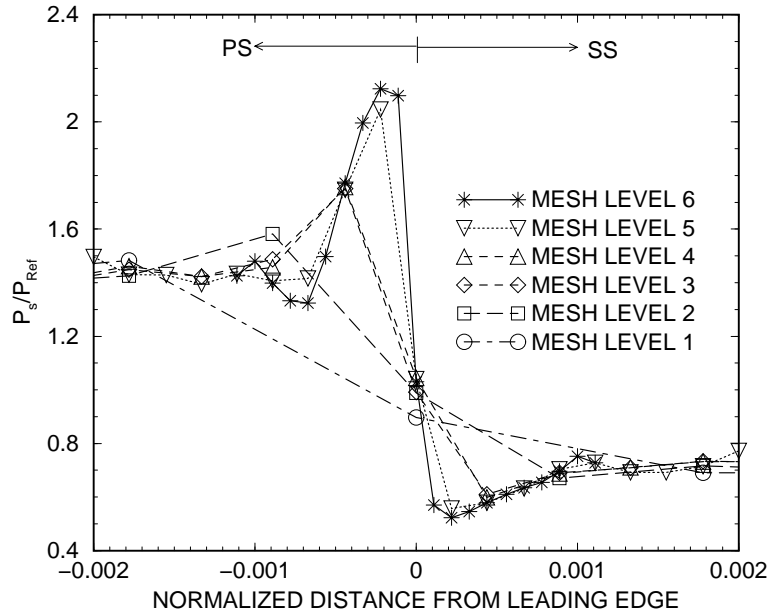


Figure 4.39. Solution convergence of static pressure on mesh levels at leading edge for Rotor 37 at midspan.

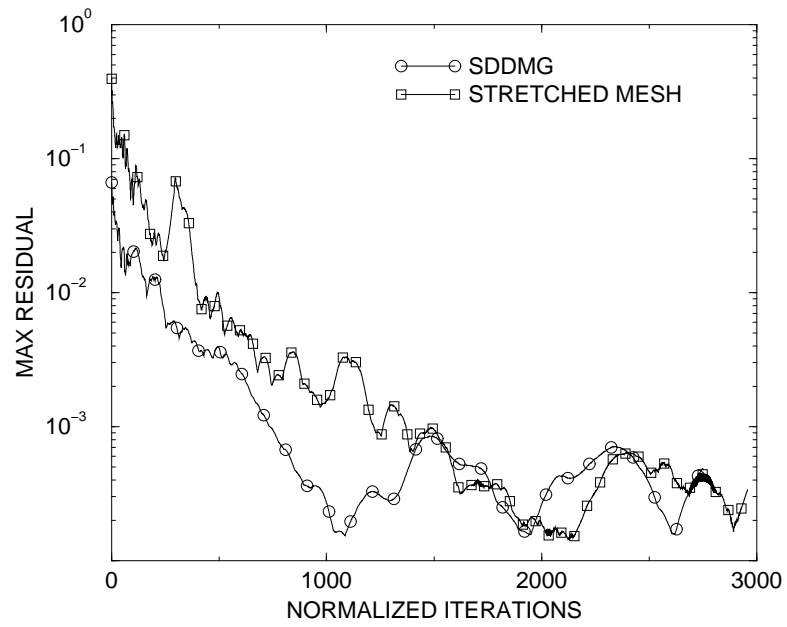


Figure 4.40(a). Convergence History of Maximum Error of SDDMG and single block stretched mesh for Rotor 37.

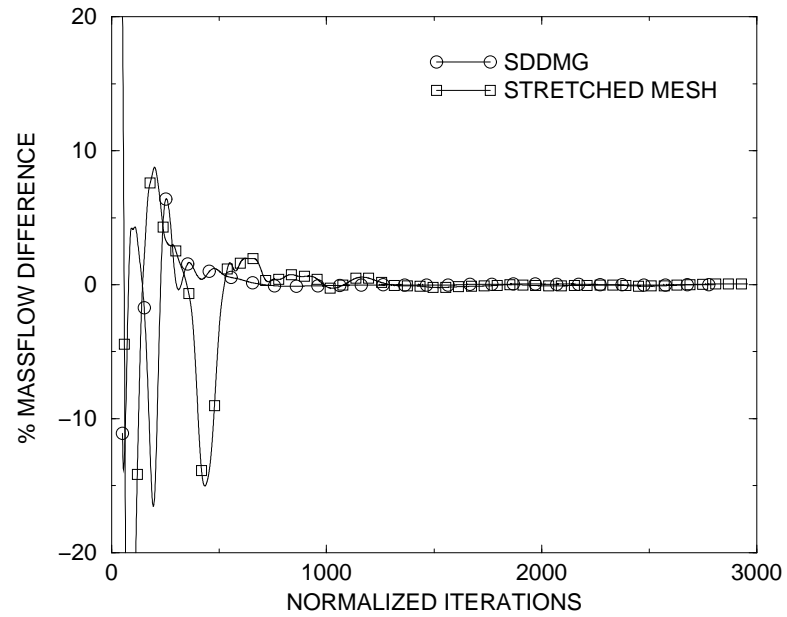


Figure 4.40(b). Convergence History of Percent Massflow Difference of SDDMG and single block stretched mesh for Rotor 37.

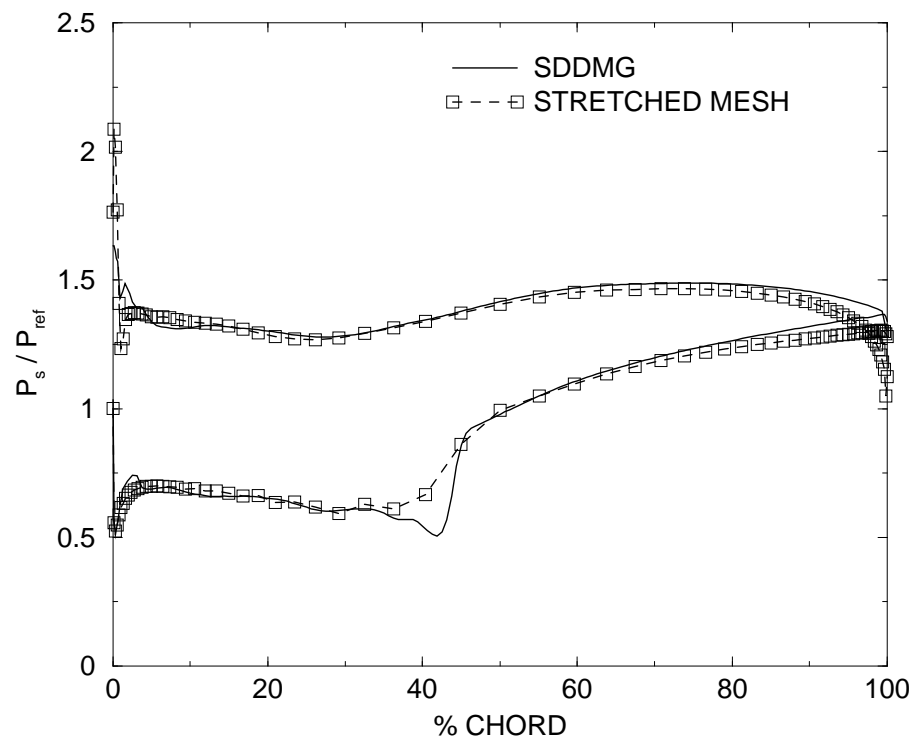


Figure 4.41. Comparison of blade loading of SDDMG and single block stretched mesh at midspan for Rotor 37.

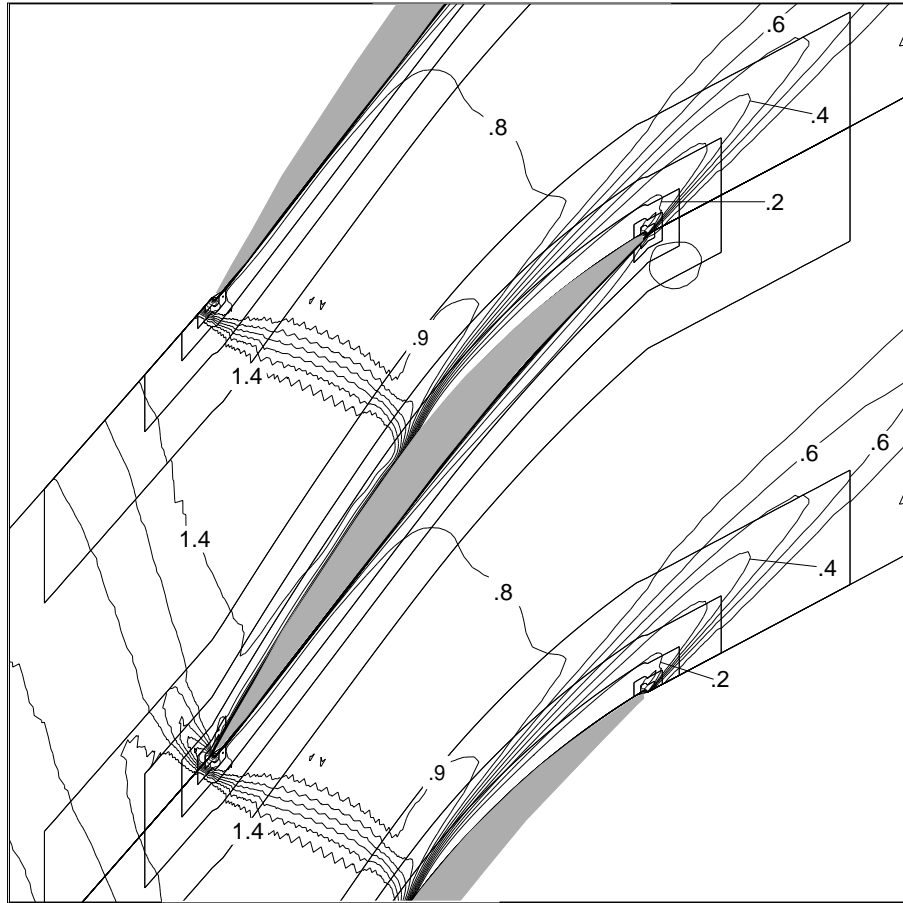


Figure 4.42(a). Contours of Relative Mach Number from the SDDMG solution at midspan for Rotor 37.

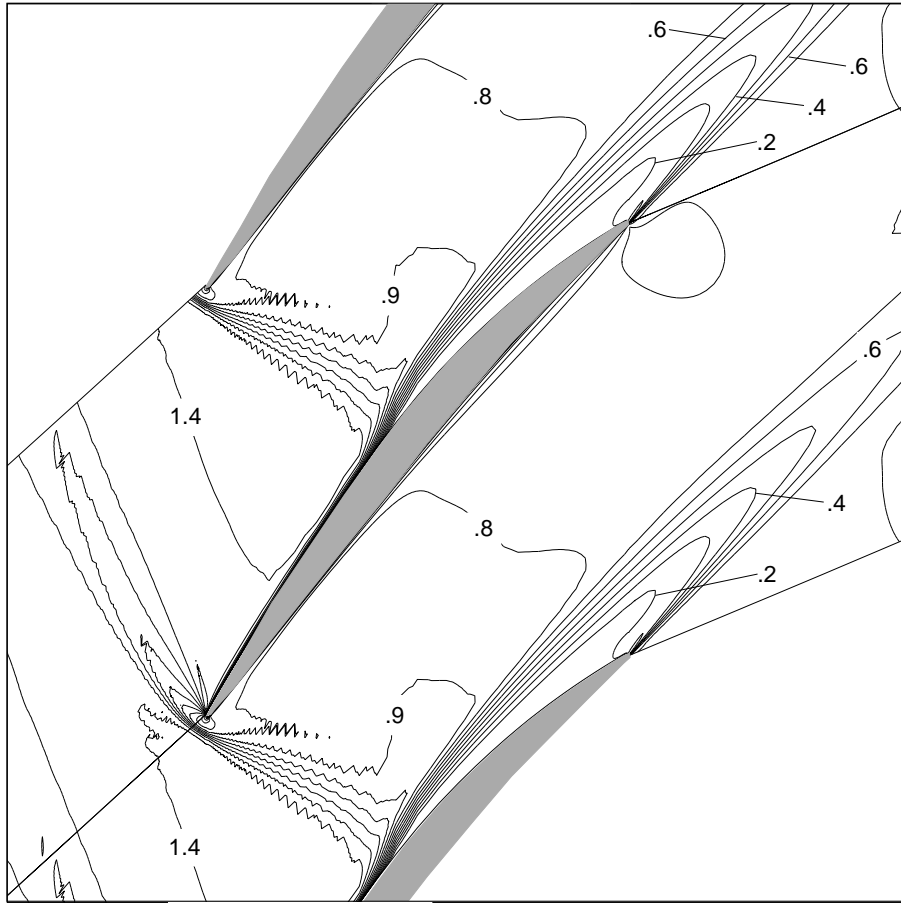


Figure 4.42(b). Contours of Relative Mach Number from the single block stretched mesh solution at midspan for Rotor 37.

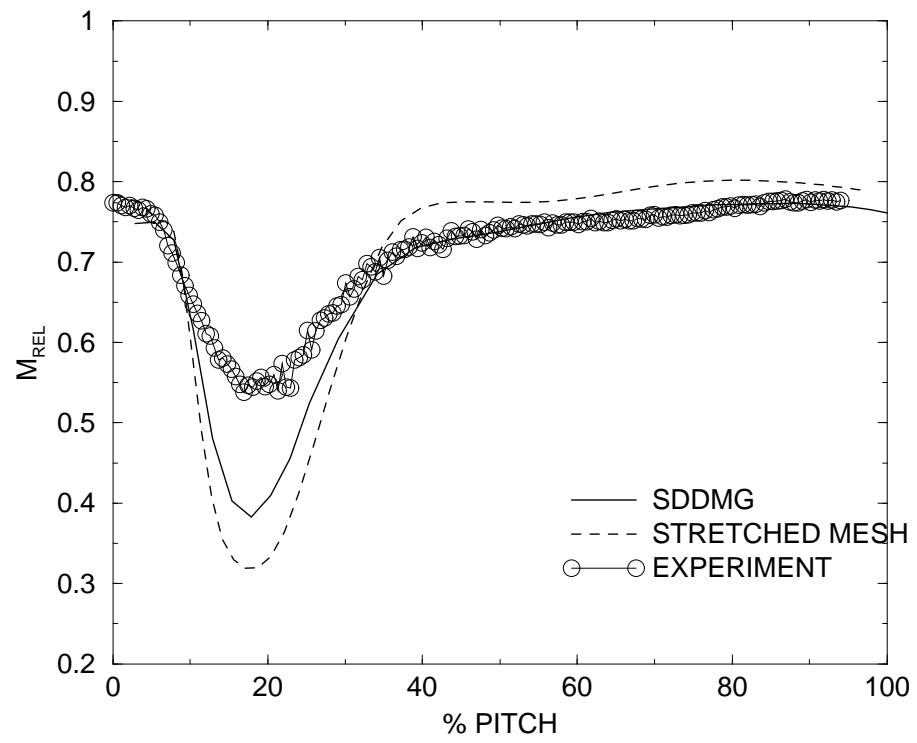


Figure 4.43. Comparison of wake profiles of SDDMG and single block stretched mesh at Station 3 at midspan for Rotor 37.

## CHAPTER 5

### SUMMARY, CONCLUSIONS AND FUTURE WORK

#### 5.1 Summary and Conclusions

Accurate predictions of the the aerodynamic performance of turbomachinery blading requires resolution of flow features near the leading and trailing edges of blades as well as resolution of the blade boundary layers. To obtain this flow detail, grid cells are clustered near the leading and trailing edges of blades as well as in the pitch-wise direction near the blade surface. For stretched meshes, this clustering leads to grids with large aspect ratios cells. For Runge-Kutta based schemes, mesh stretching and large aspect ratio cells may lead to convergence problems. An investigation of mesh stretching and aspect ratio was carried out on a model problem. The results of the analysis using a Runge-Kutta based scheme showed that mesh stretching and high aspect ratio cells adversely affect the spatial stability of numerical schemes. To circumvent these stability problems, a segmented domain decomposition multigrid (SDDMG) procedure was developed that employs uniform cells.

The SDDMG procedure solves the Reynolds-Averaged Navier-Stokes equations using a cell-centered finite volume formulation and is applied to geometries typical of those found in turbomachinery.

The main elements of the SDDMG procedure are:

1. development of a multiblock structure;
2. implementation of multigrid;
3. development of a grid strategy;
4. implementation of a consistent set of boundary conditions;
5. development of interface conditions.

The role of each of these elements to develop the SDDMG procedure is described

in the following section.

The multiblock structure allows information to be passed from one grid level to the next. The multigrid procedure accelerates the convergence of the fine grid and allows regions of the flow described by the coarse grid equations to be solved to the same accuracy as the regions of the flow described by the fine grid. The grid strategy used a well-known mesh generation code to develop a base grid from which all coarse and fine mesh levels were generated. In addition, a strategy for further mesh refinement was discussed. The boundary conditions were developed and implemented consistent with the multigrid procedure. The interface conditions across local fine grid refinements, were developed from conservation laws. With the above elements, the implementation of the SDDMG procedure was discussed beginning with coarse mesh iterations, interpolating to the fine mesh refinements and how each of the mesh levels exchange information both to finer and coarser mesh levels.

The procedure resulting from this work was applied to channel flow and three geometries which are typical of a turbomachine. The first two turbomachine geometries were a compressor cascade and a turbine cascade with experimental data. The final test case was a three-dimensional transonic compressor rotor. Each case was used to show a different feature of the SDDMG system.

Channel flow was used to compare the SDDMG procedure to a stretched mesh by refining the pitch-wise direction only. Both systems used identical axial and radial spacings. The comparison of the two systems showed that for a small stretching ratio, the stretched mesh and SDDMG systems achieved comparable results. However, the SDDMG method was able to obtain a solution with less effective grid cells and converged in less iterations. Effective cells are defined as the cells that are pertinent to the final solution. The convergence history of the maximum error associated with the continuity equation showed that the SDDMG system achieved a better convergence rate and attained a lower level of error than the stretched mesh. The SDDMG system also reduced the difference in the integrated massflow at inlet



and exit faster than the stretched mesh.

The results of the simulations of the turbomachine geometries and comparisons to experiment showed that the SDDMG procedure is predicting solutions that are consistent with the experimental trends. With the establishment of credible solutions, the strength of the SDDMG procedure to give grid independent solutions was shown for the ARL compressor and Rotor 37. The resolution of the leading edge through grid refinement provided a more detailed wall static pressure. The VKI cascade was used to show that incorporating additional refinement levels could be done after a solution had already been obtained with a given set of mesh levels. The additional refinement was needed to further resolve the very thin boundary layer on the suction surface of the blade. The additional mesh levels showed the convergence of the velocity profile as well as an improvement in the resulting wake profile.

Finally, a single block stretched mesh was generated for the Rotor 37 test case based upon the cell sizes of the coarsest and finest mesh levels in the pitch-wise direction from the SDDMG system. A grid density in the axial direction could not be similarly defined because the total number of grid points in the stretched mesh would be three times larger than the total number of grid points in the SDDMG system. Thus, the axial direction was stretched in similar fashion to the pitch-wise direction. The grid density for the radial direction was identical in both systems. A solution was obtained for both systems at the same mass flow to choke mass flow ratio. The comparisons of convergence based on maximum error of the continuity equation and percent differences in mass flow showed that the SDDMG system achieves convergence in half the number of iterations as the single block stretched mesh. The iteration count was normalized to the differences in the average CPU time per iteration. The comparison of blade loadings at midspan showed that the SDDMG system defines the shock over fewer grid points due to the higher axial grid density along the blade. Downstream of the shock, the SDDMG system predicts more diffusion than the stretched mesh as a result of less blockage in the SDDMG

system. Comparison of the wake profile downstream of the blade confirmed that the diffusion is better predicted in the SDDMG system due to less blockage. Blockage is a key parameter in axial compressors that affect their work output and mass flow capacity. Predicting the right level of blockage is critical in the design of axial compressors.

## 5.2 Future Work

**Parallelization.** In developing this procedure, it becomes apparent that the process of local mesh refinement lends itself very naturally to parallelization. The refinement of a parent mesh into a number of child meshes can be treated as a single processor calculation (the parent process) that spawns a number of (child) processor calculations. Likewise, this can be continued with each successive refinement. Information need only be transferred to different mesh levels during the multigrid restriction and prolongation steps. These steps take up a fraction of the processing time as compared to the local solution procedure resulting in a large reduction of CPU time for two child meshes spawned from a parent mesh.

**Mesh Strategy.** The meshes used in this work were restricted to sheared H-meshes only. This means that the axial and radial values of a grid line in the blade-to-blade direction were the same. This is adequate for thin blades with small stagger angles and axial inlet flow but this can cause convergence problems for thick blades with larger stagger angles such as those typically found in turbines. Figures 4.21 show the mesh at the leading and trailing edge of the VKI turbine. The mesh defining these regions of the blade are very skewed. For these geometries, the sheared H-mesh is probably the worst choice to use but was still able to obtain credible results. It would be preferable to allow the blade-to-blade grid lines to become more normal to the geometry at the surface. This type of mesh would probably converge better than a sheared H-mesh and allow for better definition of the leading edge and trailing edge region with less grid points.

**Full Domain Decomposition.** The problem with full domain decomposi-

tion is the difficulty of determining where to refine. In the case of blade-to-blade decomposition, this work concentrated on resolving the thin viscous region near the blade surfaces. For a general three-dimensional turbomachinery flow with mixing, tip clearance, bleeds, leakages and a host of other physical processes developing, it is not as clear where to refine a mesh. The complexity of refining two additional surfaces in a full three-dimensional decomposition procedure is added to the problem along with the juncture of two surfaces. In addition to trying to resolve blade surface boundary layers, and stagnation points, the additional complexity of radial flow migration, horseshoe vortices, and tip clearance flow exacerbate the refinement problem. However, based on the results of this work, the effort may be worthwhile in order to obtain some very fine flow details that require the flexibility of arbitrary mesh refinement.

## BIBLIOGRAPHY

- [1] Denton, J. D., "An Improved Time-Marching Method for Turbomachinery Flow Calculations," *J. of Engr. for Power*, V. 105, N. 3, July, 1983, pp. 514-524.
- [2] Hah, C. "A Navier-Stokes Analysis of Three-Dimensional Turbulent Flows Inside Turbine Blade Rows At Design and Off-Design Conditions," ASME 83-GT-40, 1983.
- [3] Katsanis, T., "Calculation of Three-Dimensional, Viscous Flow Through Turbomachinery Blade Passages by Parabolic Marching", AIAA-85-1408 1985.
- [4] Dawes, W. N., "A Numerical Analysis of the Three-Dimensional Viscous Flow in a Transonic Compressor Rotor and Comparison with Experiment," *J. of Turbomachinery*, V. 109, N. 1, January, 1986, pp. 83-90.
- [5] Chima, R. V., and Yokota, J. W., "Numerical Analysis of Three-Dimensional Viscous Internal Flows," AIAA-88-3565, 1988.
- [6] Subramanian, S. V., "Three-Dimensional Multigrid Navier-Stokes Computations for Turbomachinery Applications", AIAA-89-2453 1989.
- [7] Adamczyk, J. J., "Model Equation for Simulating Flows in Multistage Turbomachinery," ASME 85-GT-226 1985.
- [8] Adamczyk, J. J., Celestina, M. L., Beach, T. A., Barnett, M., "Simulation of Three-Dimensional Viscous Flows Within a Multistage Turbine," *J. of Turbomachinery*, V. 112, N. 3, July, 1990, pp. 370-376.
- [9] Adamczyk, J. J., Celestina, M. L., Greitzer, E. M., "The Role of Tip Clearance in High-Speed Fan Stall", *J. of Turbomachinery*, V. 115, N. 1, January, 1993, pp. 28-38.
- [10] Brandt, A., "Multi-Level Adaptive Solutions to Boundary-Value Problems," *Mathematics of Computation*, V. 31, N. 138 April, 1977, pp. 333-390.
- [11] Srinivasan, K., Rubin, S. G., "Segmented Domain Decomposition Multigrid Solutions for Two and Three-Dimensional Viscous Flows", *Proceedings Symposium on Adaptive, Multilevel and Hierarchical Computational Strategies*, ASME Winter Annual Meeting, Nov. 1992.
- [12] Srinivasan, K., "Segmented Multigrid Domain Decomposition Procedure for Two and Three-Dimensional Viscous Flows", *Ph.D. Dissertation*, Department of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, 1994.
- [13] Srinivasan, K., Rubin, S. G., "A Local Grid Refinement Procedure Through Segmented Multigrid Domain Decomposition For Three-Dimensional Flows", *Proceedings of 12th AIAA CFD Conference*, June, 1995.
- [14] Launder, B. E., and Spalding, D. B., "The Numerical Computation of Turbulent Flows," *Comp. Math. Appl. Mech. Engr.* V. 3, pp. 269-289.

- [15] Holmes, D. G., and Tong, S. S., "A Three-Dimensional Euler Solver for Turbomachinery Blade Rows", ASME 84-GT-79, 1984.
- [16] Shabbir, A., Zhu, J., and Celestina, M. L., "Assessment of Three Turbulence Models in a Compressor Rotor", ASME 96-GT-198, 1996.
- [17] Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes", AIAA-81-1259, AIAA 14th Fluid and Plasma Dynamics Conference, June, 1981.
- [18] Jameson, A., and Baker, T. J., "Solution of the Euler Equations for Complex Configurations", AIAA-83-1929, AIAA 6th CFD Conference, 1983.
- [19] Berger, M. J., and Oliger, J., "Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations", *Journal of Computational Physics*, **53**, pp. 484-512, 1984.
- [20] Berger, M. J., and Jameson, A., "Automatic Adaptive Grid Refinement for the Euler Equations", *AIAA Journal*, Vol. 23, No. 4 1985.
- [21] Berger, M. J. and Colella, P., "Local Adaptive Mesh Refinement for Shock Hydrodynamics", *Journal of Computational Physics*, **82**, pp. 64-84, 1989.
- [22] Steger, J., and Benek, J. A., "On the Use of Composite Grid Schemes in Computational Aerodynamics", *Computer Methods in Applied Mechanics and Engineering*, Vol. 64, pp. 301-320, 1987.
- [23] Steger, J., "Thoughts on the Chimera Method of Simulation of Three-Dimensional Viscous Flow", *Computational Fluid Dynamics Symposium on Aeropropulsion*, pp. 1-10, 1991.
- [24] Himansu, A. and Rubin, S. G., "Multigrid Acceleration of a Relaxation Procedure for the Reduced Navier-Stokes Equations", *AIAA Journal*, Vol. 26, No. 9, pp. 1044-1051, 1988.
- [25] Himansu, A. and Rubin, S. G., "Three-Dimensional RNS Computations with Multigrid Acceleration", AIAA-91-0105, 1991.
- [26] Swanson, R. C., and Turkel, E., "Part I, Multistage Central Difference Schemes for the Euler and Navier-Stokes Equations," Incomplete Reference.
- [27] Jentink, T. N., and Usab, W. J., "Formulation of Boundary Conditions for the Multigrid Acceleration of the Euler and Navier-Stokes Equations", *Final Technical Report for NASA Research Grant No. NAG 3-875*, School of Aeronautics and Astronautics, Purdue University, 1990.
- [28] Mulac, R. A., "A Multistage Mesh Generator for Solving the Average-Passage Equation System," NASA CR-179539, 1986.
- [29] Zierke, W. C., and Deutsch, S., "The Measurement of Boundary Layers on a Compressor Blade in Cascade, Volume 1," NASA CR-185118, 1989.

- [30] Arts, T., Lambert de Rouvroit, M., and Rutherford, A. W., "Aero-thermal Investigation of a Highly Loaded Transonic Linear Turbine Guide Vane Cascade," VKI Technical Note 174, 1990.
- [31] McFarland, E.R., "Solution of Plane Cascade Flow Using Improved Surface Singularity Methods," *J. of Engr. for Power*, V. 104, July, 1982, pp. 668-674.
- [32] McFarland, E. R., "A Rapid Blade-to-Blade Solution for Use in Turbomachinery Design," *J. of Engr. for Gas Turbines and Power*, V. 106, April, 1984, pp. 376-382.
- [33] McNally, W. D., "Fortran Program for Calculating Compressible Laminar and Turbulent Boundary Layers in Arbitrary Pressure Gradients," NASA TN D-5681, 1970.
- [34] Crawford, M. E., Kays, W. M., "STAN5 - A Program for Numerical Computation of Two-Dimensional Internal and External Boundary Layer Flows," NASA CR-2742, 1976.
- [35] Reid, L., Moore, R. D., "Design and Overall Performance of Four Highly Loaded, High-Speed Inlet Stages for an Advanced High-Pressure Ratio Core Compressor," NASA TP 1337, 1978.
- [36] Moore, R. D., Reid, L., "Performance of Single-Stage Axial-Flow Transonic Compressor with Rotor and Stator Aspect Ratios of 1.19 and 1.26, Respectively, and With Design Pressure Ratio of 2.05," NASA TP 1659, 1980.
- [37] Denton, J. D., "Lessons Learned from Rotor 37," Presented at the 3rd International Symposium on Experimental and Computational Aerothermodynamics of Internal Flows (ISAIF), Beijing, China, 1996.
- [38] Suder, K. L., Celestina, M. L., "Experimental and Computational Investigation of the Tip Clearance Flow in a Transonic Axial Compressor Rotor," ASME 94-GT-365, 1994.
- [39] Shabbir, A., Celestina, M. L., Adamczyk, J. J., Strazisar, A. J., "The Effect of Hub Leakage Flow on Two High-Speed Axial Flow Compressor Rotors," ASME 97-GT-346, 1997.
- [40] Van Zante, D. E., Adamczyk, J. J., Strazisar, A. J., Okiishi, T. H., "Wake Recovery Performance Benefit in a High-Speed Axial Compressor," ASME 97-GT-535, 1997.
- [41] Kirtley, K. R., Beach, T. A., Adamczyk, J. J., "Numerical Analysis of Secondary Flow in a Two-Stage Turbine," AIAA-90-2356, 1990.
- [42] VanZante, D. E., Strazisar, A. J., Wood, J. R., Hathaway, M. D., "Recommendations for Achieving Accurate Numerical Simulation of Tip Clearance Flows in Transonic Compressor Rotors," To be presented at the ASME IGTI Congress, Indianapolis, Indiana, 1999.
- [43] Jennions, I. K., Adamczyk, J. J., "Evaluation of the Interaction Losses in a

- Transonic Turbine HP Rotor / LP Vane Configuration,” ASME 95-GT-299, 1995.
- [44] Atkinson, K. E., “An Introduction to Numerical Analysis,” John Wiley & Sons, 1978.
  - [45] Srinivasan, K., Rubin, S. G., “Segmented Multigrid Domain Decomposition Procedures for Two- and Three-Dimensional Viscous Flows,” *J. of Fluids Eng.*, V. 115, N. 4, Dec., 1993, pp. 608-613.
  - [46] Srinivasan, K., Rubin, S. G., “Solution Based Grid Optimization Through Segmented Multigrid Domain Decomposition,” *J. Comp. Phys.*, V. 136, N. 2, Sept., 1997. pp. 467-493.
  - [47] Hussain, A. K. M. F., Reynolds, W. C., “Measurements in Fully Developed Turbulent Channel Flow,” *J. of Fluids Eng.*, Dec., 1975, pp. 568-578.
  - [48] Suder, K. L., “Experimental Investigation of the Flow Field in a Transonic, Axial Flow Compressor With Respect to the Development of Blockage and Loss,” *Ph.D. Dissertation*, Department of Mechanical and Aerospace Engineering, Case Western Reserve University, 1996.

## APPENDIX A

To investigate the effects of mesh stretching and aspect ratio more fully, the two-dimensional advection equation is analyzed with periodic boundary conditions. An artificial dissipation term is incorporated much like that used in many flow solvers. The discrete points in the  $x$ -direction are uniformly spaced and the discrete points in the  $y$ -direction are either uniform with constant cell aspect ratio or are stretched exponentially. In addition, the wave speeds are assumed constant and since the interest of this work is in obtaining the time-asymptotic solution, local time-stepping or time-marching at a constant CFL number is assumed.

### A.1 Model Equation

The two-dimensional advection equation can be written as

$$\gamma \frac{\partial f}{\partial t} + \alpha \frac{\partial f}{\partial x} + \beta \frac{\partial f}{\partial y} = 0 \quad (A.1)$$

where  $\alpha$ ,  $\beta$  and  $\gamma$  are constants. The boundary conditions are periodic. Local time-stepping (i.e., marching at a constant CFL number) is used since the emphasis is on the time-asymptotic solution. The discrete points in the  $x$ -direction are uniformly spaced in the interval  $[0, 1]$  and the discrete points in the  $y$ -direction are either uniform with constant cell aspect ratio ( $AR = \frac{\Delta x}{\Delta y}$ ) or stretched exponentially.

The equations are transformed to general, curvilinear coordinates yielding

$$\gamma \frac{\partial(fJ)}{\partial t} + \tilde{\alpha} \frac{\partial f}{\partial \xi} + \tilde{\beta} \frac{\partial f}{\partial \eta} = 0 \quad (A.2)$$

where  $J = x_\xi y_\eta(y)$ ,  $\tilde{\alpha} = \alpha y_\eta(y)$ , and  $\tilde{\beta} = \beta x_\xi$ . This analysis is restricted to stretching in the  $y$ -direction only, i.e.,  $x_\eta = y_\xi = 0$ ,  $y_\eta = y_\eta(y)$ , and  $x_\xi = \text{constant}$ .

Discretizing the governing equation in space using central differences yields:

$$\gamma \frac{d(fJ)}{dt} + \tilde{\alpha} \frac{\delta_\xi f}{\Delta \xi} + \tilde{\beta} \frac{\delta_\eta f}{\Delta \eta} + Q = 0 \quad (A.3)$$



$Q$  is the artificial viscosity term used to stabilize the differential-difference equation and is given as

$$Q = \mu_{4_\xi} \frac{\delta_{\xi\xi\xi\xi} f}{\Delta\xi} + \mu_{4_\eta} \frac{\delta_{\eta\eta\eta\eta} f}{\Delta\eta} \quad (A.4)$$

The coefficients  $\mu_{4_\xi}$  and  $\mu_{4_\eta}$  are scaled by the wave speed in that direction such that,

$$\mu_{4_\xi} = \nu \tilde{\alpha}$$

$$\mu_{4_\eta} = \nu \tilde{\beta}$$

where  $\nu$  is a constant chosen to maintain stability.

Multiply Eq. (A.3) by  $\frac{1}{\gamma J}$  and choosing  $\gamma$  as  $\frac{\tilde{\beta}}{J}$  yields

$$\frac{df}{dt} + \frac{\tilde{\alpha}}{\tilde{\beta}} \frac{\delta_\xi f}{\Delta\xi} + \frac{\delta_\eta f}{\Delta\eta} + \nu \left[ \frac{\tilde{\alpha}}{\tilde{\beta}} \frac{\delta_{\xi\xi\xi\xi} f}{\Delta\xi} + \frac{\delta_{\eta\eta\eta\eta} f}{\Delta\eta} \right] = 0 \quad (A.5)$$

Eq. (A.5) is a differential-difference equation that incorporates the effects of local time-stepping and eigenvalue scaling in the artificial viscosity.

The ratio  $\frac{\tilde{\alpha}}{\tilde{\beta}}$  can be written in terms of the local aspect ratio. Since  $y_\eta \sim \frac{\Delta y}{\Delta\eta}$  and  $x_\xi \sim \frac{\Delta x}{\Delta\xi}$  and setting  $\Delta\eta = \Delta\xi = 1$ . Then

$$\frac{\tilde{\alpha}_j}{\tilde{\beta}} = \frac{\alpha y_{\eta_j}}{\beta x_\xi} = \frac{\alpha \Delta y_j}{\beta \Delta x} = \frac{\alpha}{\beta} \frac{1}{\text{AR}_j}$$

where  $\text{AR}_j = \frac{\Delta x}{\Delta y_j}$ .

To isolate the effects of high aspect ratio and stretching we are seeking a solution of Eq. (A.5) which is periodic in space. Hence, we assume a discrete Fourier expansion for the spatial dependence. The Fourier expansion for  $f_{i,j}$  and  $(\frac{\tilde{\alpha}_j}{\tilde{\beta}})$  are

$$\hat{f}_{m,k} = \frac{1}{M} \sum_{m=0}^{M-1} \frac{1}{N} \sum_{k=0}^{N-1} f_{i,j} \exp(-I2\pi jk/N) \exp(-I2\pi im/M) \quad (A.6)$$

and

$$\hat{\alpha}_k = \frac{1}{N} \sum_{k=0}^{N-1} \left( \frac{\tilde{\alpha}_j}{\tilde{\beta}} \right) \exp(-I2\pi jk/N) \quad (A.7)$$

where  $I = \sqrt{-1}$ ,  $M$  is the number of Fourier modes in the  $\xi$ -direction and  $N$  is the number of Fourier modes in the  $\eta$ -direction. Introducing these expansions into Eq.

(A.5) yields:

$$\begin{aligned} \frac{d\hat{f}_{m,k}}{dt} + \sum_{n=0}^{N-1} \left[ I \sin\left(\frac{2\pi m}{M}\right) + 16\nu \sin^4\left(\frac{\pi m}{M}\right) \right] \hat{\alpha}_{k-n} \hat{f}_{m,n} \\ + \left[ I \sin\left(\frac{2\pi k}{N}\right) + 16\nu \sin^4\left(\frac{\pi k}{N}\right) \right] \hat{f}_{m,k} = 0 \end{aligned} \quad (\text{A.8})$$

For each  $m$ , Eq. (A.8) can be written in matrix notation as

$$\frac{d\bar{f}_m}{dt} + G_m \bar{f}_m = 0 \quad (\text{A.9})$$

where

$$\bar{f}_m = [\hat{f}_{m,0}, \hat{f}_{m,1}, \dots, \hat{f}_{m,N-1}]^T$$

and  $G_m$  is the spatial stability matrix which can be written as the sum of a circulant matrix ( $C_m$ ) and a diagonal matrix ( $D$ ).

$$G_m = C_m + D \quad (\text{A.10})$$

where

$$C_m = \gamma_m \begin{pmatrix} \hat{\alpha}_0 & \hat{\alpha}_{-1} & \dots & \hat{\alpha}_{-(N-1)} \\ \hat{\alpha}_1 & \hat{\alpha}_0 & \dots & \hat{\alpha}_{-(N-2)} \\ \vdots & \ddots & \ddots & \vdots \\ \hat{\alpha}_{-1} & \dots & \hat{\alpha}_1 & \hat{\alpha}_0 \end{pmatrix}$$

and

$$\begin{aligned} D &= \begin{pmatrix} \gamma_0 & 0 & \dots & 0 \\ 0 & \gamma_1 & 0 & \dots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \gamma_{N-1} \end{pmatrix} \\ \gamma_m &= 16\nu \sin^4\left(\frac{\pi m}{M}\right) + I \sin\left(\frac{2\pi m}{M}\right) \\ \gamma_k &= 16\nu \sin^4\left(\frac{\pi k}{N}\right) + I \sin\left(\frac{2\pi k}{N}\right) \end{aligned}$$

Note that the entries of matrix  $C_m$  contain terms from the  $\xi$ -direction only while the entries of matrix  $D$  contain terms from the  $\eta$ -direction. Also, because periodic boundary conditions were chosen, the Fourier coefficients  $\hat{\alpha}_{k-n}$  are periodic about  $N$  and thus have the property  $\hat{\alpha}_1 = \hat{\alpha}_{-(N-1)}$ .

## A.2 Exponential Stretching

A stretching function was chosen representative of the stretching function employed in many 3-D viscous production codes. It's form is

$$\Delta y_j = \Delta y_0 \sigma^j \quad \text{for } 0 \leq j < \frac{N}{2} \quad (A.11)$$

$$\Delta y_j = \Delta y_0 \sigma^{N-j} \quad \text{for } \frac{N}{2} \leq j \leq N-1$$

where  $N$  is an even integer. The value of the parameter  $\sigma$  lies in the interval  $(0, 1]$ . It's value sets the amount of stretching. For  $\sigma = 1$ , there is no stretching, i.e., a uniform distribution of points is obtained. The Fourier coefficients associated with Eq. (A.11) can be evaluated in closed form using the functional form of the geometric series

$$\sum_{i=0}^{M-1} z_i^M = \frac{1 - z_i^M}{1 - z_i}. \quad (A.12).$$

They are

$$\hat{\alpha}_{k-n} = \frac{1}{N} \frac{\tilde{\alpha}_0}{\tilde{\beta}} \left[ \frac{(1 - \sigma^2)[1 - (-1)^{k-n} \sigma^{\frac{N}{2}}]}{1 - 2\sigma \cos(\frac{2\pi(k-n)}{N}) + \sigma^2} \right] \quad (A.13)$$

If we assume  $N$  is large and define  $\sigma = 1 - \delta$  where  $0 \leq \delta \leq 1$ , then  $\hat{\alpha}_{k-n}$  can be approximated as

$$\hat{\alpha}_{k-n} = \frac{1}{N} \frac{\tilde{\alpha}_0}{\tilde{\beta}} \left[ \frac{\delta(2 - \delta)[1 - (-1)^{k-n} \exp(\frac{-N\delta}{2})]}{2(1 - \delta)[1 - \cos(\frac{2\pi(k-n)}{N})] + \delta^2} \right] \quad (A.14)$$

The stretching parameter  $\delta$  determines the values of the off-diagonal elements of the spatial stability matrix  $G_m$ . Examining Eq. (A.14) further, if  $\frac{\tilde{\beta}}{\tilde{\alpha}_0} > 1$  and  $\delta = 0$  (no stretching), then  $\hat{\alpha}_{k-n} = 0$  for  $k \neq n$  (off-diagonal entries) and  $\hat{\alpha}_0 = \frac{\tilde{\alpha}_0}{\tilde{\beta}}$  for  $k = n$ . The spatial stability matrix  $G_m$  is then diagonal and strictly dependent on the aspect ratio. Again, this corresponds to specifying a uniform aspect ratio mesh with no stretching.

Let us now look at how a  $k^{th}$  Fourier mode is affected by stretching. For  $\delta \neq 0$ ,  $\hat{\alpha}_0$ , the diagonal entry of  $C_m$ , is less than  $(\frac{1}{AR})$ . The off-diagonal elements become

nonzero but decay as  $|k - n|$  increases. This may be viewed locally as increasing the aspect ratio in the stretched direction. As the stretching parameter increases, the value of the diagonal term decreases and the off-diagonal elements increase. The stability of any time integration scheme is dependent on the eigenvalues of  $G_m$ . Since the stretching parameter,  $\delta$ , determines the entries of  $C_m$ , it will affect the eigenvalues of  $G_m$  and, in turn, affect the stability of the time integration scheme chosen. For zero stretching, the spatial stability matrix  $G_m$  is diagonal and the eigenvalues are known. The eigenvectors are simple Fourier modes. With stretching, the eigenvalues of  $C_m$  move from the position with no stretching. Some might cause the scheme to become unstable. The individual eigenvectors of  $G_m$  for  $\delta \neq 0$  are also not simple Fourier modes but a superpositioning of these modes.

Three limits of  $N\delta$  were examined with  $N$  large: (a)  $N\delta \sim O(\epsilon)$ ,  $\epsilon$  a small parameter; (b)  $N\delta \sim O(1)$ ; and, (c)  $N\delta \sim O(N^p)$ , for  $0 < p < 1$ .

For case (a), the unstretched case should be recovered as the leading term. To show this, we begin by furthering simplifying  $G_m$ . Let

$$\begin{aligned}\gamma_k^{(R)} &= 16\nu \sin^4\left(\frac{\pi k}{N}\right) \\ \gamma_m^{(R)} &= 16\nu \sin^4\left(\frac{\pi m}{M}\right) \\ \gamma_k^{(I)} &= \sin\left(\frac{2\pi k}{N}\right) \\ \gamma_m^{(I)} &= \sin\left(\frac{2\pi m}{M}\right) \\ \widehat{\text{AR}} &= \frac{\tilde{\beta}}{\tilde{\alpha}_0}\end{aligned}$$

Thus

$$\begin{aligned}\gamma_m &= \gamma_m^{(R)} + I\gamma_m^{(I)} \\ \gamma_k &= \gamma_k^{(R)} + I\gamma_k^{(I)}\end{aligned}$$

In terms of the entries of  $G_m$ , denoted as  $\gamma_{k,n}^m$ :

$$\gamma_{k,n}^m = \gamma_m \hat{\alpha}_{k-n} + \gamma_k \bar{\delta}_{k-n} \quad (\text{A.15})$$

where  $\bar{\delta}_{k-n} = 1$  for  $k = n$  and  $\bar{\delta}_{k-n} = 0$  for  $k \neq n$ . The first term on the right hand side are the elements of matrix  $C_m$  and the second term are the elements of the diagonal matrix  $D$ .

Since we know the eigenvalues for the case of uniform aspect ratio with zero stretching, we can obtain estimates of the eigenvalues for the stretched mesh using the Gerschgorin Theorem [44]. This Theorem bounds the eigenvalues of a non-diagonal matrix by a disk centered at the diagonal element. The radius is given by the sum of the absolute value of the off-diagonal terms. If the off-diagonal terms are small, the eigenvalues of the matrix will be contained in a small neighborhood of the diagonal entries.

From Gerschgorin's Theorem [44], the  $k^{th}$  eigenvalue of  $G_m$ ,  $\lambda_k^m$ , falls within a disk given by

$$|\lambda_k^m - \gamma_{k,n=k}^m| \leq \sum_{\substack{n=1 \\ n \neq k}}^{N-1} |\gamma_{k,n}^m| \quad (A.16)$$

For the unstretched case,  $\delta = 0$ , and  $\widehat{\text{AR}}$  large, the eigenvalues for the  $m^{th}$  Fourier mode are the components of the vector

$$\lambda_k^m = \lambda_k^{m(R)} + I \lambda_k^{m(I)} \quad (A.17)$$

where

$$\begin{aligned} \lambda_k^{m(I)} &= \gamma_k^{(I)} + \frac{\gamma_m^{(I)}}{\widehat{\text{AR}}} \\ \lambda_k^{m(R)} &= \gamma_k^{(R)} + \frac{\gamma_m^{(R)}}{\widehat{\text{AR}}} \end{aligned}$$

The estimate of the eigenvalues for the three limits will be expressed in terms of  $\lambda_k^{m(I)}$  and  $\lambda_k^{m(R)}$ . The real and complex part of  $\lambda_k^m$  will be denoted as  $\lambda_{k(R)}^m$  and  $\lambda_{k(I)}^m$ .

(a)  $N\delta \sim O(\epsilon)$ . The Gerschgorin disk for this limit is

$$\begin{aligned} & \left[ \left( \lambda_{k(R)}^m - \lambda_k^{m(R)} + \frac{\gamma_m^{(R)}}{\widehat{\text{AR}}} \left\{ (N\delta) + \frac{\delta}{2} \right\} \right)^2 + \right. \\ & \left. \left( \lambda_{k(I)}^m - \lambda_k^{m(I)} + \frac{\gamma_m^{(I)}}{\widehat{\text{AR}}} \left\{ (N\delta) + \frac{\delta}{2} \right\} \right)^2 \right]^{1/2} \leq \\ & \left[ \frac{(\gamma_m^{(R)})^2 + (\gamma_m^{(I)})^2}{\widehat{\text{AR}}^2} \right]^{1/2} \left\{ \frac{7(N\delta)}{2\pi^2} - \frac{5\delta}{N} \right\} \end{aligned} \quad (A.18)$$

Figure A.1 depicts pictorally the movement of the disk from the case of zero stretching on the complex plane. The leading term corresponds to zero stretching. This is obviously an exaggerated view of the eigenvalue movement but does show the effect of stretching on the spatial stability of a numerical scheme. The inward movement, as will be shown later, suggests a decrease in stability.

Physically, the dissipation, necessary for the stability of the scheme, and the convection in the  $x$ -direction are reduced by a combination of the high aspect ratio and the stretching.

**(b)**  $N\delta \sim O(1)$ . The disk now becomes

$$\begin{aligned} & \left[ (\lambda_{k(R)}^m - \lambda_k^{m(R)} + \frac{\gamma_m^{(R)}}{\widehat{\text{AR}}} \{1 + \frac{2(1 - \exp(\frac{-N\delta}{2}))}{N\delta}\})^2 + \right. \\ & \quad \left. (\lambda_{k(I)}^m - \lambda_k^{m(I)} + \frac{\gamma_m^{(I)}}{\widehat{\text{AR}}} \{1 + \frac{2(1 - \exp(\frac{-N\delta}{2}))}{N\delta}\})^2 \right]^{1/2} \leq \\ & \quad \left[ \frac{(\gamma_m^{(R)})^2 + (\gamma_m^{(I)})^2}{\widehat{\text{AR}}^2} \right]^{1/2} \left\{ 1 + \exp(-\frac{N\delta}{2}) \right\} \end{aligned} \quad (A.19)$$

For this case, the leading term is order one. The exaggerated view of the eigenvalue movement illustrated in Figure A.2 shows that stretching in this range predicts a disk radius less than the distance between the disk center and that of the unstretched eigenvalue. This was also seen in limit (a). It is expected that choosing a stretching parameter falling within limits (a) or (b) will not have an appreciable impact on the stability.

**(c)**  $N\delta \sim O(N^p)$ . For this worst case, the Gerschgorin disk is

$$\begin{aligned} & \left[ (\lambda_{k(R)}^m - \lambda_k^{m(R)} + \frac{\gamma_m^{(R)}}{\widehat{\text{AR}}} \{1 + \frac{2}{N\delta}\})^2 + \right. \\ & \quad \left. (\lambda_{k(I)}^m - \lambda_k^{m(I)} + \frac{\gamma_m^{(I)}}{\widehat{\text{AR}}} \{1 + \frac{2}{N\delta}\})^2 \right]^{1/2} \leq \\ & \quad \left[ \frac{(\gamma_m^{(R)})^2 + (\gamma_m^{(I)})^2}{\widehat{\text{AR}}^2} \right]^{1/2} \left\{ 1 - \frac{3\delta}{N(2 - \delta)} \right\} \end{aligned} \quad (A.20)$$

The disk is represented pictorally in Figure A.3 and shows the disk radius encompassing the shift of the eigenvalues from the unstretched location to the disk center.

The spatial stability matrix,  $G_m$ , from the analysis, predicts a loss of diagonal dominance. Whether this is significant is not known. More importantly, the disk center is much further from the unstretched eigenvalue than in the first two cases.

For the stretched case, the matrix  $G_m$ , is not diagonal. Since  $N$  is large, finding the eigenvalues of the spatial stability matrix in closed form is very difficult. In the interest of obtaining an approximation to the amplification matrix, it is assumed that the eigenvalue coincides with the disk center, i.e., the diagonal entry. This, in essence, replaces the matrix  $G_m$  with a diagonal matrix. The largest term is always the diagonal entry as is evident from Eq. A.14. However, the contribution of the off-diagonal terms should be more important as the stretching parameter becomes larger. However, as we shall see, the diagonal entry yields the dominant parameter. This assumption uncouples the differential-difference equation and the stability criteria is reduced to a scalar equation. The diagonal entry of  $G_m$  becomes

$$\lambda_k^m = \left[ \frac{\gamma_m^{(R)}}{\widehat{\text{AR}}} \left( \frac{2}{N\delta} \right) + \gamma_k^{(V)} \right] + I \left[ \frac{\gamma_m^{(I)}}{\widehat{\text{AR}}} \left( \frac{2}{N\delta} \right) + \gamma_k^{(I)} \right] \quad (\text{A.21})$$

for the worst case of  $N\delta \sim O(N^p)$ . The factor  $N\delta$ , with this assumption, is similar to the  $\widehat{\text{AR}}$  in that it reduces the dissipation and convection in the  $x$ -direction.

### A.3 The Time Integration Scheme

Based on the insight provided by this analysis, Eq. (A.1) was integrated in time using a four-stage Runge-Kutta time integration scheme,

$$\begin{aligned}
\bar{f}_m^{n+\alpha_1} &= \bar{f}_m^n - \alpha_1 \Delta t G_m \bar{f}_m^n \\
\bar{f}_m^{n+\alpha_2} &= \bar{f}_m^n - \alpha_2 \Delta t G_m \bar{f}_m^{n+\alpha_1} \\
\bar{f}_m^{n+\alpha_3} &= \bar{f}_m^n - \alpha_3 \Delta t G_m \bar{f}_m^{n+\alpha_2} \\
\bar{f}_m^{n+1} &= \bar{f}_m^n - \Delta t G_m \bar{f}_m^{n+\alpha_3}
\end{aligned} \tag{A.22}$$

to provide numerical results to support and enhance the analysis presented in the previous section. The coefficients of the scheme are taken as  $\alpha_1 = \frac{1}{4}$ ,  $\alpha_2 = \frac{1}{3}$ , and  $\alpha_3 = \frac{1}{2}$ . Eq. (A.22) can be further simplified to

$$\bar{f}_m^{n+1} = A_m \bar{f}_m^n \tag{A.23}$$

where

$$A_m = I - \Delta t G_m + \alpha_3 (\Delta t G_m)^2 - \alpha_2 \alpha_3 (\Delta t G_m)^3 + \alpha_1 \alpha_2 \alpha_3 (\Delta t G_m)^4$$

and is known as the amplification matrix. The stability criteria is  $\|A_m\| < 1$ .

First consider the case of zero stretching ( $\delta = 0$ ). The spatial stability matrix  $G_m$  is diagonal and uncouples Eq. (A.23) resulting in  $N \times M$  scalar equations. From Eq. (A.17), the diagonal entry of  $G_m$  is

$$\lambda_k^m = \left[ \frac{\gamma_m^{(R)}}{\widetilde{\text{AR}}} + \gamma_k^{(R)} \right] + I \left[ \frac{\gamma_m^{(I)}}{\widetilde{\text{AR}}} + \gamma_k^{(I)} \right] \tag{A.24}$$

The local timestep, from a stability analysis of Eq. (A.3) in which the temporal integration is a four-stage Runge-Kutta scheme with uniform spacing, is given as

$$\Delta t = \frac{CFL\gamma J}{\tilde{\alpha} + \tilde{\beta}} \tag{A.25}$$

For large cell aspect ratio,  $\tilde{\beta} \gg \tilde{\alpha}$ , and the timestep becomes

$$\Delta t = \frac{CFL\gamma J}{\tilde{\beta}} \tag{A.26}$$



$\Delta t$  from Eq. (A.25) can be written as  $\text{CFL} \cdot \widehat{\text{AR}}/(1 + \widehat{\text{AR}})$  for the Runge-Kutta scheme. The amplification matrix  $A_m$  also reduces to a diagonal matrix whose entry can be written as

$$a_k^m = 1 - \left( \frac{\text{CFL} \cdot \widehat{\text{AR}} \lambda_k^m}{1 + \widehat{\text{AR}}} \right) + \frac{1}{2} \left( \frac{\text{CFL} \cdot \widehat{\text{AR}} \lambda_k^m}{1 + \widehat{\text{AR}}} \right)^2 - \frac{1}{6} \left( \frac{\text{CFL} \cdot \widehat{\text{AR}} \lambda_k^m}{1 + \widehat{\text{AR}}} \right)^3 + \frac{1}{24} \left( \frac{\text{CFL} \cdot \widehat{\text{AR}} \lambda_k^m}{1 + \widehat{\text{AR}}} \right)^4 \quad (\text{A.27})$$

For large  $\widehat{\text{AR}}$ , this can be further reduced by approximating the term  $\widehat{\text{AR}}/(1 + \widehat{\text{AR}})$  as unity. From Eq. (A.24), as  $\widehat{\text{AR}} \rightarrow \infty$ ,  $\lambda_k^m \rightarrow \gamma_k^{(V)} + I\gamma_k^{(I)}$ . The spatial stability of the scheme and the amplification factor  $a_k^m$  of Eq. (A.27) are dependent only on the  $y$ -direction. For large cell aspect ratio at all wave numbers in the  $x$ -direction and low wave numbers in the  $y$ -direction, the scheme, at worst, is marginally stable.

For the stretched case, the matrix  $G_m$  is not diagonal and thus  $A_m$  is not diagonal. Stability is then determined by the eigenvalues of the amplification matrix  $A_m$ . Given the estimate of the eigenvalues of the spatial stability matrix, the movement of a diagonal element of  $G_m$  towards the origin corresponds to a movement of the diagonal element of  $A_m$  towards a less stable position closer to the unit circle.

## A.4 Numerical Results

The model equation was coded and the Runge-Kutta integration scheme was used to advance the equations in time. The CFL number was set to 2.8 and the artificial viscosity coefficient  $\nu$  was set to  $\frac{1}{32}$  as dictated by a linear stability analysis on a uniform mesh. The convective speeds  $\alpha$  and  $\beta$  were both set to unity. To investigate the worst case effect of  $\widehat{\text{AR}}$  and stretching, we will consider high Fourier modes in the  $x$ -direction and fixing the  $y$ -direction to its lowest Fourier mode. Thus, the initial solution is chosen as a cosine curve in the  $x$ -direction of mode  $\pi$  and uniform in the  $y$ -direction. The mesh sizes  $M + 1$  and  $N + 1$  were set to 101.

Two cases were considered to assess the impact of stretching and cell aspect ratio. First, the cell aspect ratio was varied with no stretching. This was followed by varying the stretching parameter only. The residual history of these cases will be plotted versus iteration count or cycles. The residual is defined as the change in the solution over one cycle divided by the timestep.

Figure A.4 shows the effect of varying the cell aspect ratio with no stretching. A comparison of the estimated amplification factor for the different cell aspect ratios from the simulation and analysis show excellent agreement as it should since no approximation is involved. They are listed in the following table.

$\widehat{\text{AR}}$	Predicted	Computed
1	0.4978	0.4978
10	0.8805	0.8805
100	0.9862	0.9862
1000	0.9986	0.9986

Table A.1 Comparison of Amplification Factor

As expected, the amplification factor approaches unity rapidly with increasing  $\widehat{\text{AR}}$ . As the aspect ratio of the mesh increases, the Fourier mode of  $\pi$  changes from a very stable mode to one approaching marginal stability. As was mentioned earlier, the stability characteristics of the scheme are dominated by the  $y$ -direction as the aspect ratio is increased.

As mentioned in the Appendix introduction, a typical problem mesh with stretching will have a local aspect ratio in excess of 1000. This aspect ratio is defined as

$$\overline{\text{AR}} = \frac{\Delta x}{\min(\Delta y_j)}$$

Figure A.5 shows a plot of  $\delta$  versus  $N$  for three maximum aspect ratios based on the noted definition using the exponential stretching function. These curves give an indication of the value  $\delta$  should be for a given  $N$  to model a large cell aspect ratio mesh. We will consider five stretching values in the following section. The three largest values:  $\delta = .0897$ ;  $.1315$ ; and  $.1714$ , correspond to maximum local aspect ratios of 100, 1000, and 10,000, respectively for  $N = 100$ . The latter two stretching parameters would never be used in practice, however, a typical mesh would be contained between the first two.

Figures A.6 and A.7 show the effect of varying the stretching parameter while maintaining an aspect ratio ( $\widehat{\text{AR}}$ ) of unity for  $M = N = 100$  and  $M = N = 40$ , respectively. Thus, the local aspect ratio is only dependent upon the stretching parameter. For  $N\delta \sim O(\epsilon)$ , the residual curve lies on top of the case with no stretching. The maximum local aspect ratio for this curve is 1.005. For  $N\delta \sim O(1)$ , there is still very little change in the slope of the residual curve. The maximum local aspect ratio for this curve is 1.6. Thus, very small amounts of stretching do not seem to affect the stability of the scheme. For larger values of  $N\delta$ , where the local aspect ratio varies from 100 to 10,000, a much different story develops. The residual curves are convergent but oscillate in their downward path until they eventually reach a limit cycle. Reducing the mesh size to a more realistic value of  $M = N = 40$  does not change the predicted amplification factor. Note that to obtain the same maximum local aspect ratio, the stretching parameter,  $\delta$ , has increased by more than a factor of two in Figure A.7. Aside from the larger stretching parameter, the only other difference evident between Figures A.6 and A.7 is the smaller amplitude of the oscillation in the residual curves for the smaller mesh. Both grids tend to the same limit cycle. This limit cycle is round-off error. The computations for  $\delta = .0897$

and .1714 were computed using Cray single (64 bit) and Cray double (128 bit) precision. For the double precision runs, the limit cycle is no longer evident. Using the approximation of Eq. (A.21) for the eigenvalues of  $G_m$ , we can compare the estimated value of the amplification factor based on Eq. (A.25) to that estimated for the residual history plot for the largest three values of  $\delta$  studied.

$\delta$	$N\delta$	AR	Predicted	Computed	Error (%)
.0897	8.97	1-100	.8555	.7802	9.59
.1315	13.15	1-1000	.8990	.8401	7.01
.1714	17.14	1-10000	.9216	.8857	4.05

Table A.2 Comparison of Amplification Factor.

The amplification factor estimated from the residual history was computed using the local minima of the residual curve. From the table, the approximate value of the amplification factor approaches that of the code as  $N\delta$  becomes large. This is surprising since only the diagonal term of  $G_m$  is retained to predict the amplification factor. Both, however, show the deterioration in convergence as stretching is increased.

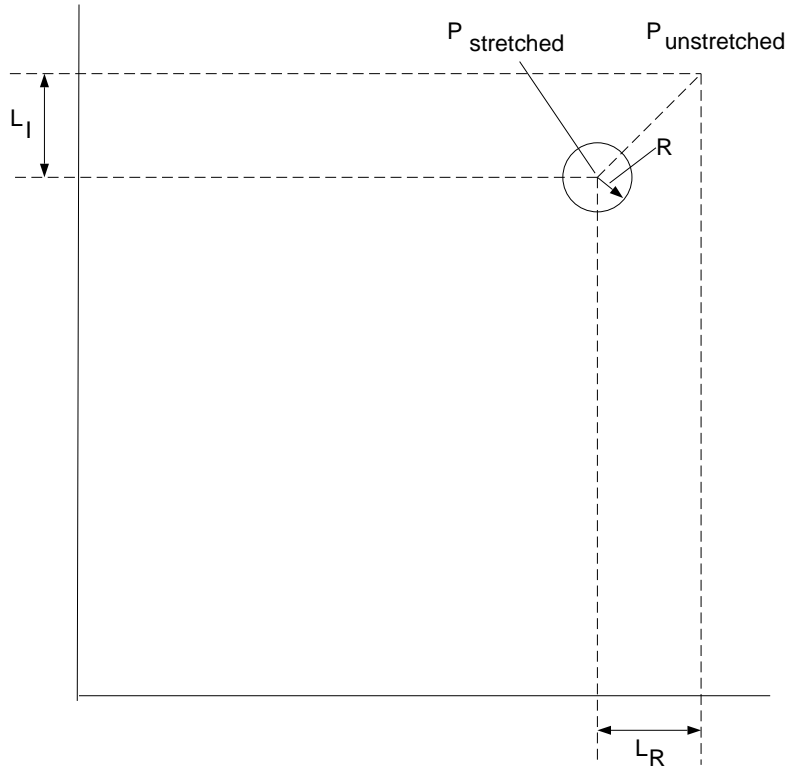
## A.5 Conclusion

A Fourier analysis of the two-dimensional time-dependent advection equation on stretched and high aspect ratio meshes was carried out. The results of the analysis show that mesh stretching and high aspect ratio cell sizes have an adverse effect on the spatial stability of a numerical scheme. An exponential function was defined to stretch the mesh. The equation in Fourier space yields a spatial stability matrix that is comprised of a circulant and diagonal matrix. To assess the impact of mesh stretching, the location of the eigenvalues of this spatial stability matrix were estimated using Gerschgorin's theorem and compared to their original (unstretched) position. The disk centers move inward toward the origin (a less stable position) as the stretching is increased. The stretching parameter times the number of cells in the stretched direction,  $N\delta$ , is a key term in the spatial stability of a scheme and

acts much like the aspect ratio term for the unstretched but uniform aspect ratio case. The amplification factor for large aspect ratio and no stretching shows that the terms in the unstretched direction are reduced by the inverse of the aspect ratio. Physically, the convective and dissipative process in the direction with the large cell size is diminished with increasing aspect ratio. With stretching,  $N\delta$  plays the same role on the unstretched direction as the aspect ratio with the added complexity that the process is now a function of the local aspect ratio. Both the aspect ratio and stretching is independent of time and would thus impact any time integration scheme chosen.

The equations were integrated in time using the four stage, Runge-Kutta scheme and the impact of aspect ratio and stretching was assessed by comparing their amplification factors. To judge the merits of the analysis, the model equation was coded and run on a Cray YMP. The resulting computed amplification factor compared quite well to the predicted values. The resulting residual history showed the stretching was locally oscillatory and was subjected to round-off error.

The underlying problem in dealing with high aspect ratio and/or stretched meshes is that they are a function of the spatial discretization and thus independent of the time integration scheme chosen. To deal with these problems, workers usually minimize the aspect ratio or stretching which causes an increase in the grid size or accept the mesh as the price for obtaining solutions in a reasonable amount of time on a mesh containing fewer nodes than is actually required. However, it has been shown that the convergence rate on high aspect ratio and stretched meshes is slowed as the inverse function of the aspect ratio and/or stretching parameter.

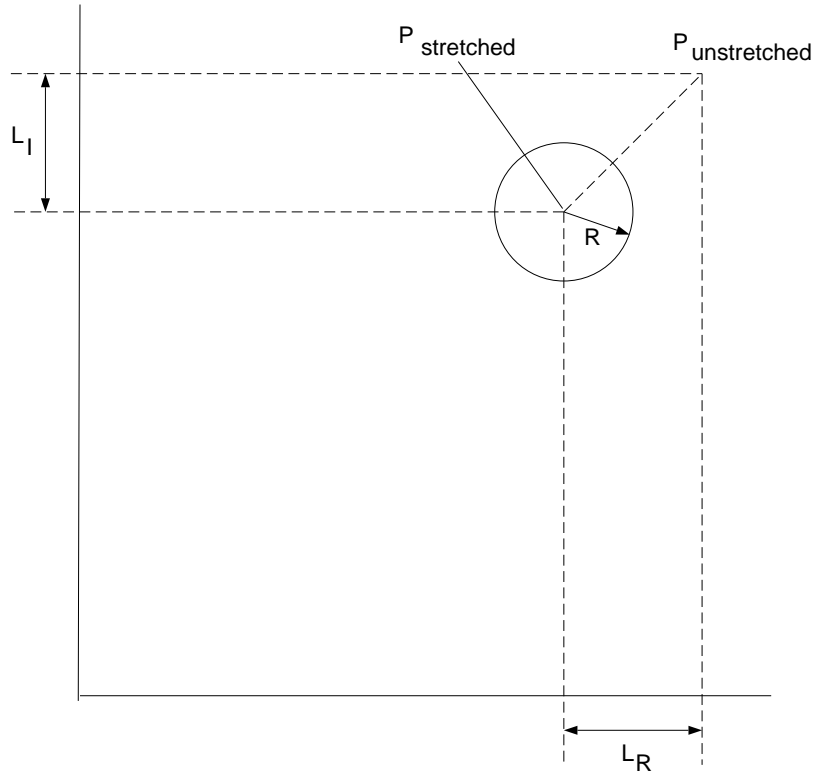


$$P_{\text{unstretched}} = (\lambda_k^{m^{(R)}}, \lambda_k^{m^{(I)}}), \quad P_{\text{stretched}} = (\lambda_k^m, \lambda_k^m),$$

$$L_R = \frac{\gamma_m^{(V)}}{\widehat{\text{AR}}} \{N\delta + \frac{\delta}{2}\}, \quad L_I = \frac{\gamma_m^{(I)}}{\widehat{\text{AR}}} \{N\delta + \frac{\delta}{2}\},$$

$$R = \left[ \frac{\gamma_m^{(V)^2} + \gamma_m^{(I)^2}}{\widehat{\text{AR}}} \right] \left\{ \frac{7N\delta}{2\pi^2} - \frac{5\delta}{N} \right\}$$

Figure A.1. Gerschgorin disk for  $N\delta \sim O(\epsilon)$  in the imaginary plane.

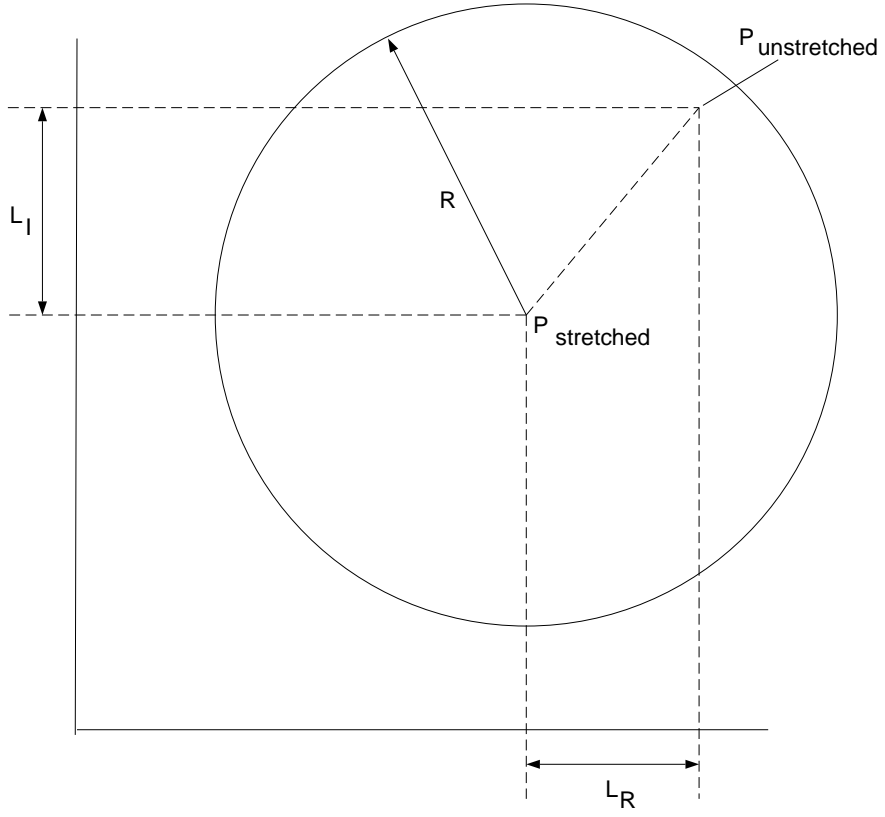


$$P_{\text{unstretched}} = (\lambda_k^{m^{(R)}}, \lambda_k^{m^{(I)}}), \quad P_{\text{stretched}} = (\lambda_k^m, \lambda_k^m),$$

$$L_R = \frac{\gamma_m^{(V)}}{\widehat{\text{AR}}} \left\{ 1 + \frac{2(1 - e^{\frac{-N\delta}{2}})}{N\delta} \right\}, \quad L_I = \frac{\gamma_m^{(I)}}{\widehat{\text{AR}}} \left\{ 1 + \frac{2(1 - e^{\frac{-N\delta}{2}})}{N\delta} \right\},$$

$$R = \left[ \frac{\gamma_m^{(V)^2} + \gamma_m^{(I)^2}}{\widehat{\text{AR}}} \right] \left\{ 1 + e^{\frac{-N\delta}{2}} \right\}$$

Figure A.2. Gerschgorin disk for  $N\delta \sim O(1)$  in the imaginary plane.



$$P_{\text{unstretched}} = (\lambda_k^{m^{(R)}}, \lambda_k^{m^{(I)}}), \quad P_{\text{stretched}} = (\lambda_k^m, \lambda_k^m),$$

$$L_R = \frac{\gamma_m^{(V)}}{\widehat{AR}} \left\{ 1 + \frac{2}{N\delta} \right\}, \quad L_I = \frac{\gamma_m^{(I)}}{\widehat{AR}} \left\{ 1 + \frac{2}{N\delta} \right\},$$

$$R = \left[ \frac{\gamma_m^{(V)^2} + \gamma_m^{(I)^2}}{\widehat{AR}} \right] \left\{ 1 - \frac{3\delta}{N(2 - \delta)} \right\}$$

Figure A.3. Gerschgorin disk for  $N\delta \sim O(N^P)$  in the imaginary plane.



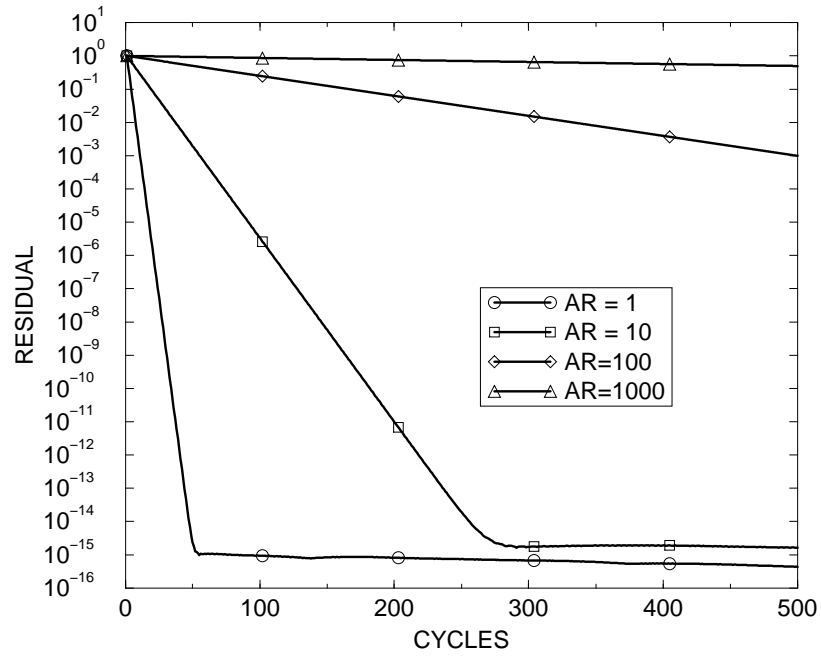


Figure A.4 Variation of AR on the model problem,  $CFL = 2.8$ ,  $\nu = \frac{1}{32}$ ,  $N = 100$ .

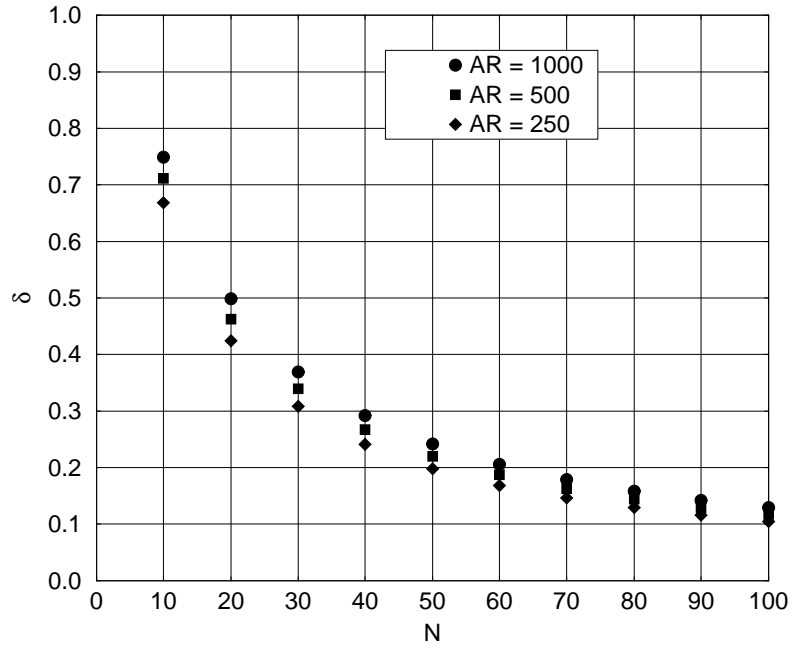


Figure A.5 Plot of  $N$  vs.  $\delta$  for a maximum AR of 250, 500, and 1000 for the exponential stretching function.

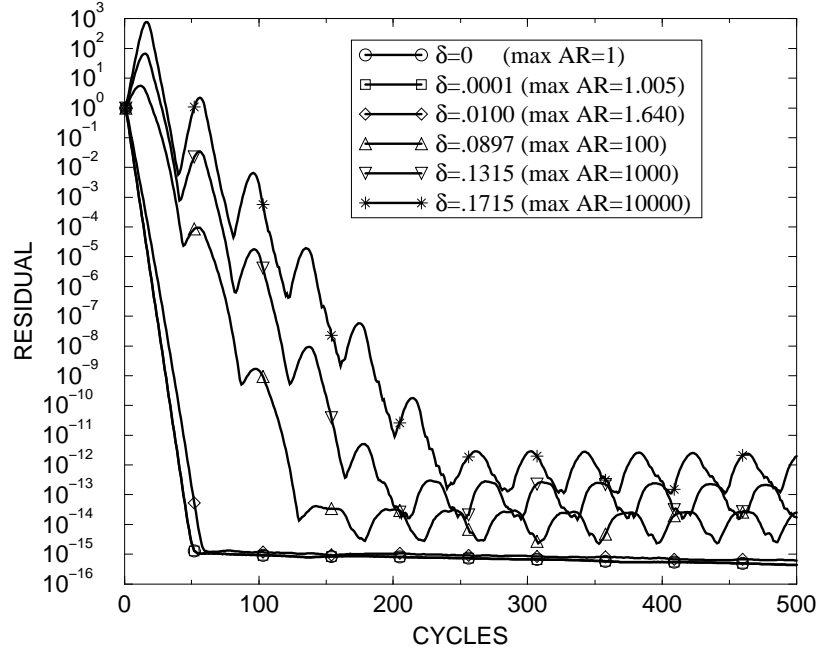


Figure A.6 Variation of stretching on the model problem,  $\text{CFL} = 2.8$ ,  $\nu = \frac{1}{32}$ ,  $N = 100$ .

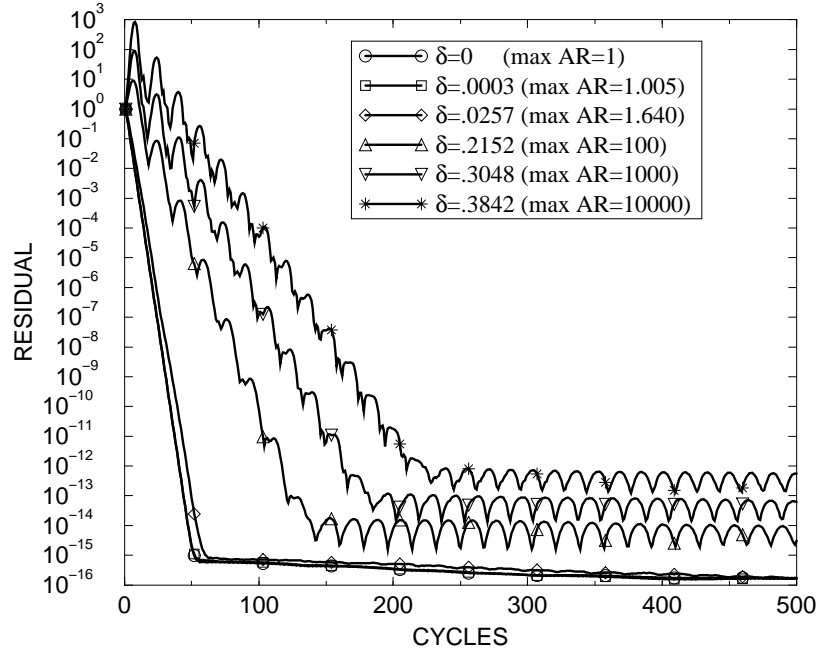


Figure A.7 Variation of stretching on the model problem,  $\text{CFL} = 2.8$ ,  $\nu = \frac{1}{32}$ ,  $N = 40$ .