

AD-A264 933



2

12

The Pennsylvania State University
APPLIED RESEARCH LABORATORY
P.O. Box 30
State College, PA 16804

AN INCOMPRESSIBLE AXISYMMETRIC THROUGH-FLOW
CALCULATION PROCEDURE FOR DESIGN AND
OFF-DESIGN ANALYSES OF TURBOMACHINERY

by

B. W. Siebert
A. M. Yocum

DTIC
ELECTE
MAY 26 1993
S A D

Technical Report No. TR 93-05
May 1993

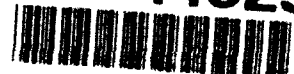
Supported by:
Space and Naval Warfare Systems Command

L.R. Hettche, Director
Applied Research Laboratory

Approved for public release; distribution unlimited

93 5 25 239

93-11829



REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE May 1993		3. REPORT TYPE AND DATES COVERED	
4. TITLE AND SUBTITLE An Incompressible Axisymmetric Through-Flow Calculation Procedure for Design and Off-Design Analyses of Turbomachinery				5. FUNDING NUMBERS N00039-88-C-0051	
6. AUTHOR(S) B. W. Siebert, A.M.Yocum				8. PERFORMING ORGANIZATION REPORT NUMBER TR#93-05	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Applied Research Laboratory The Pennsylvania State University P.O. Box 30 State College, PA 16804				10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Space and Naval Warfare Systems Command Department of the Navy Washington, DC 20362-5100				16. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES					
12a. DISTRIBUTION / AVAILABILITY STATEMENT Unlimited				12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) A computational procedure has been developed which solves the steady-state incompressible Navier-Stokes (or Reynolds) equations for axisymmetric flows through turbomachinery. The solution of the Navier-Stokes equations allows for the prediction of recirculating flow regions that may be due to either through-flow geometry or off-design operating conditions. The effects of the blade rows are represented in an approximate manner by replacing the forces they impart to the fluid in the axial, radial, and tangential directions with body forces. In order to provide an algorithm which can be applied to arbitrary turbomachinery geometries, the grid covering the flow domain is mapped into a computational plane in which the nodal spacing becomes unity. The Navier-Stokes equations are transformed so that they become a spatial function of the generalized curvilinear coordinates that describe the computational plane. Local flux conservation is ensured by expressing the system of equations in conservative form and discretizing them using a control volume approach. The Pressure Weighted Interpolation Method (PWIM), coupled with the SIMPLEC algorithm, is used to iteratively solve the discretized equations on a nonstaggered grid.					
14. SUBJECT TERMS Turbomachinery, Incompressible, Axisymmetric, Navier-Stokes, Blade-row effects				15. NUMBER OF PAGES 168	
				16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UNLIMITED		

Abstract

A computational procedure has been developed which solves the steady-state incompressible Navier-Stokes (or Reynolds) equations for axisymmetric flows through turbomachinery. The solution of the Navier-Stokes equations allows for the prediction of recirculating flow regions that may be due to either through-flow geometry or off-design operating conditions. The effects of the blade rows are represented in an approximate manner by replacing the forces they impart to the fluid in the axial, radial, and tangential directions with body forces.

In order to provide an algorithm which can be applied to arbitrary turbomachinery geometries, the grid covering the flow domain is mapped into a computational plane in which the nodal spacing becomes unity. The Navier-Stokes equations are transformed so that they become a spatial function of the generalized curvilinear coordinates that describe the computational plane. However, changing the spatial dependence of the governing equations does not alter the conserved quantities. Local flux conservation is ensured by expressing the system of equations in conservative form and discretizing them using a control volume approach. Also, expressing the governing equations in a general form allows the same program to treat flows in both axisymmetric and Cartesian coordinate systems. Modifications making the Pressure-Weighted Interpolation Method (PWIM) applicable to cylindrical based coordinate systems are presented. PWIM, coupled with the SIMPLEC algorithm, is used to iteratively solve the discretized equations on a nonstaggered grid.

A revised version of the quadratic upwind differencing scheme is presented, which allows the high level of accuracy that the scheme provides to be maintained consistently throughout the flow domain, including the areas near the boundaries. The modification entails neither additional memory overhead nor computational outlays.

Two new differencing schemes have also been developed for axisymmetric applications. These schemes reflect the coupling of the magnitude of the transported quantities with the radial location in the cylindrical coordinate system and in the corresponding computational domain. Both the lower and higher-order cylindrical differencing schemes differ from their Cartesian counterparts when the flow has a radial component. Neither scheme was found to cause any stability problems.

The computer code was thoroughly validated against analytical results for flow in a pipe and radial diffuser. Test cases were run showing the increased accuracy resulting from the ability of the cylindrical differencing schemes to account for the radial dependence of the convected quantities. The quadratic upwind differencing scheme is shown to reduce numerical diffusion, and the modifications to the scheme extend this capability to the boundaries of the flow domain.

Through-flow calculations of flow in a centrifugal impeller over a wide range of operating conditions are presented and compared against experimental results. For off-design operating conditions the computer program was found to successfully predict recirculation zones within the blade rows. Calculated velocity profiles display the same trends as the measured data at both design and off-design conditions. With the original estimated parameters in the blade model, the overall impeller performance characteristics were predicted with varied success over the range of operating conditions investigated. No attempts were made to improve the performance prediction by modifying the parameters in the blade model.

Accession For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

NTIS QUALITY INSPECTED 8

Table of Contents

List of Figures	viii
Nomenclature	xi
Acknowledgements	xvii
Chapter 1 Introduction	1
Chapter 2 Governing Conservation Equations	9
2.1 Conservation of Mass and Momentum for Steady-State Incompressible Axisymmetric Flows	9
2.2 Nondimensionalization of the Governing Equations	11
2.3 Conservative Form of the Governing Equations	13
2.4 Comparison of the Cylindrical and Cartesian Forms of the Conservation Equations	16
Chapter 3 Mapping from the Physical to the Computational Plane	18
3.1 Specification of the Mapping Operation	19
3.2 Transformation of the Governing Equations into Generalized Curvilinear Coordinates	23
Chapter 4 Numerical Solution Method	28
4.1 Control Volume and Grid Definition	33
4.2 Discretizing the General Transport Equation	37
4.3 Discretizing the Momentum Equations	41
4.4 Pressure-Weighted Interpolation of Facial Contravariant Velocities	43
4.5 Pressure Corrector Equation	47
4.6 Boundary Conditions	51
4.6.1 Momentum Boundary Conditions	51
4.6.2 QUICKR Boundary Conditions	55

4.6.3 Pressure Corrector Boundary Conditions	55
4.7 Steps in the PWIM Algorithm	56
Chapter 5 Turbulence Model	58
Chapter 6 Blade Model	64
6.1 Representation of the Blade	65
6.2 Determination of the Deviation Angles	67
6.3 Specification of the Streamline and Flow Angles in the Blade Passage	70
6.4 Determination of the Tangential Velocities and Body Forces from the Flow Angles	73
6.5 Implementation of the Blade Model within the PWIM Algorithm	78
Chapter 7 Code Validation	80
7.1 Flow in a Straight Pipe	83
7.1.1 Uniform Grid	84
7.1.2 Nonuniform Grids and Turbulence Model Test	87
7.1.3 Skewed Gridlines and Differencing Scheme Comparisons	94
7.1.4 Verification of Conservation of Angular Momentum	102
7.2 Flow in a Radial Diffuser	108
Chapter 8 Mixed-Flow Centrifugal Impeller Through-Flow Calculations	119
8.1 Comparison of Calculated and Measured Flow Fields	124
8.1.1 Design-Point Calculation	125
8.1.2 Above Design-Point Calculation	126
8.1.3 Below Design-Point Calculation	132
8.1.4 Shutoff Flow Calculation	138
8.2 Comparison of Calculated and Measured Pump Performance Parameters	138
Chapter 9 Conclusions and Recommendations	146
References	149

Appendix Differencing Schemes	152
A.1 Lower-Order Differencing Schemes	153
A.2 QUICKR Differencing	157
A.3 Differencing Schemes for Cylindrical Coordinate System Applications	165

List of Figures

1.1	Two-Dimensional Solution Surfaces	5
2.1	Elemental Segment of Control Volume Wall	14
4.1	Control Volumes in Computational Space	30
4.2	Control Volumes in Physical Space	34
4.3	Control Volumes Mapped into Unit Squares in Computational Space	35
4.4	Grid in Physical Space	36
6.1	Streamline Layout in the Transformed Blade-to-Blade Plane	72
7.1	Pipe Flow Velocity Field in the Inlet Region (Uniform Inlet Profile, Uniform Grid)	85
7.2	Effect of Inlet Profile on Axial Pressure Gradient (Uniform Grid)	86
7.3	Nonuniform Grid, Δz and Δr Changing	89
7.4	Effect of Grid Nonuniformity on Axial Pressure Gradient	90
7.5	Effect of Grid Nonuniformity on Pressure	91
7.6	Comparison of Experimental and Calculated Axial Velocity Profiles for Turbulent Flow in a Pipe	93
7.7	Uniformly Spaced Grid with Skewed ξ Gridlines	95
7.8	Pipe Flow Velocity Field on Uniformly Spaced Grid with Skewed ξ Gridlines	96
7.9	Effect of ξ Gridline Skew on Axial Pressure Gradient for Various Nodal Densities (Hybrid Differencing)	98
7.10	Effect of ξ Gridline Skew on Axial Pressure Gradient for Various Differencing Schemes	99

7.11 Comparison of Results from QUICKR (Various Grid Densities) and the Lower-Order Schemes (Finest Grid)	101
7.12 Uniformly Spaced Grid with Skewed ξ and η Gridlines	103
7.13 Effect of Gridline Skew on Axial Pressure Gradient (QUICKR Differencing)	104
7.14 Shear Driven Pipe v_θ Field on a Grid with Skewed ξ and η Gridlines	106
7.15 Effect of Gridline Skew on Radial Pressure Gradient	107
7.16 Uniformly Spaced Radial Diffuser Grid with Skewed ξ and η Gridlines	111
7.17 Radial Diffuser Velocity Field on a Uniformly Spaced Grid with Skewed ξ and η Gridlines	112
7.18 Radial Pressure Gradient for Various Nodal Densities (QUICKR Differencing)	113
7.19 Radial Pressure Gradient for Various Nodal Densities and Differencing Schemes	114
7.20 Radial Pressure Gradient for Cylindrical and Cartesian Differencing Schemes	115
7.21 Effect of Gridline Skew on Radial Pressure Gradient	117
7.22 Velocity Field for a Viscous Fluid in a Radial Diffuser	118
8.1 Centrifugal Impeller Test Facility Geometry and Boundary Conditions	120
8.2 Through-Flow Calculation Grid	123
8.3 Convergence Rate for Design-Point Calculation	127
8.4 Calculated Through-Flow Velocity Vector Field ($\Phi = 0.28$)	128
8.5 Calculated Pressure Contours ($\Phi = 0.28$)	129
8.6 Comparison of Calculated and Measured Radial and Tangential Velocity Profiles ($\Phi = 0.28$)	130
8.7 Calculated Through-Flow Velocity Vector Field ($\Phi = 0.32$)	131
8.8 Calculated Pressure Contours ($\Phi = 0.32$)	133
8.9 Comparison of Calculated and Measured Radial and Tangential Velocity Profiles ($\Phi = 0.32$)	134
8.10 Calculated Through-Flow Velocity Vector Field ($\Phi = 0.14$)	135
8.11 Calculated Pressure Contours ($\Phi = 0.14$)	136

8.12 Comparison of Calculated and Measured Radial and Tangential Velocity Profiles ($\Phi = 0.14$)	137
8.13 Calculated Through-Flow Velocity Vector Field ($\Phi = 0.00$)	139
8.14 Calculated Pressure Contours ($\Phi = 0.00$)	140
8.15 Comparison of Calculated and Measured Static Head Coefficients	142
8.16 Comparison of Calculated and Measured Efficiencies	143
8.17 Comparison of Calculated and Measured Power Coefficients	145
A.1 QUICK Differencing Stencils	158
A.2 Flow Conditions which Permit QUICK Differencing on the Control Volume Faces	160
A.3 QUICKR Differencing Stencils	162

Nomenclature

Symbols

a	Coefficient multiplying a nodal quantity in the discretized total flux term, or QUICK differencing coefficient
A	Coefficient multiplying a nodal quantity in the discretized transport equation, or Blade surface area
A^+	Turbulence model constant
b	QUICK differencing coefficient
c	QUICK differencing coefficient
C	Convective flux component factor
C_{CP}	Turbulence model constant
C_D	Coefficient of drag
C_{KLEB}	Turbulence model constant
C_{WK}	Turbulence model constant
d	QUICK differencing coefficient
D	Diffusive flux component factor, or Drag, or Impeller diameter at the inlet
f	Harmonic mean interpolation factor
$F_{KLEB}(y)$	Turbulence model function
F_m	Meridional force per unit volume
$F_{r,\theta}$	Body force per unit volume in the specified cylindrical coordinate direction
F_{WAKE}	Turbulence model factor

F^η	Flux through north/south control volume face
F^ξ	Flux through east/west control volume face
g_{jk}	Metric factor in viscous diffusion term
i	Memory array index denoting ξ location, or Incidence angle
j	Memory array index denoting η location
J	Jacobian of mapping operation
k	Turbulence model constant
K	Turbulence model constant
ℓ	Eddy viscosity mixing length
ℓ_{fp}	Length of flat plate
m	Meridional coordinate
m_o	Mass source
$\dot{m}$	Mass flow rate
N	Number of blades, or Rotational speed of the impeller
NNI	Number of grid nodes in the ξ direction
NNJ	Number of grid nodes in the η direction
p	Fluid pressure
P	Power coefficient
r	Radial coordinate
Δr	Radial separation between nodes or control volume walls
r_m	Direction cosine between radial and meridional directions
$\vec{r}$	Vector extending from a point on the grid to a point on the blade
R	Residual of linear equation
Re	Reynolds number
Re_{cell}	Local, or cell, Reynolds number
Re_x	Reynolds number based on the distance from the leading edge of a flat plate

S	Source term in conservation equations, or QUICK switching factor
S_P	Implicit source term
S_U	Explicit source term
t	Time
T	Torque
u	Horizontal Cartesian velocity component, or Velocity parallel to the wall in the turbulence model
u_{DIF}	Difference between maximum and minimum velocities within a profile
U	Scaled contravariant velocity component normal to $\xi=\text{constant}$ line, or Wheel speed
v	Vertical Cartesian velocity component
V	Scaled contravariant velocity component normal to $\eta=\text{constant}$ line, or Volume of the control volume swept around the centerline
V_m	Meridional velocity
v_r	Radial velocity component
v_z	Axial velocity component
v_θ	Tangential velocity component
w	Velocity relative to the blade
x	Horizontal Cartesian coordinate, or Distance from the leading edge of a flat plate, or Coordinate in the conformal plane corresponding to the angular coordinate
y	Vertical Cartesian coordinate, or Normal distance from the wall in the turbulence model, or Coordinate in the conformal plane corresponding to the meridional coordinate
y^+	Scaled normal distance from the wall in the turbulence model
z	Axial coordinate
Δz	Axial separation between nodes or control volume walls
z_m	Direction cosine between axial and meridional directions
β'	Blade angle

β	Flow angle
γ	Ratio of relative velocities
Γ	Coefficient of diffusivity in the general transport equation
δ	Deviation angle
η	Cross-stream computational domain coordinate, or Efficiency
$\Delta\eta$	Separation in η direction between nodes or control volume walls
θ	Angular coordinate
μ	Effective fluid viscosity
μ_{lam}	Laminar fluid viscosity
μ_t	Turbulent eddy viscosity
ξ	Streamwise computational domain coordinate
$\Delta\xi$	Separation in ξ direction between nodes or control volume walls
ρ	Fluid density
σ	Slip factor
τ	Shear
ϕ	Transported quantity in general transport equation, or Flow coefficient
Φ	Global flow coefficient
ψ	Static head rise coefficient
ω	Under-relaxation factor, or Vorticity, or Angular velocity of the blade
[]	FORTRAN MAX function

β	Flow angle
γ	Ratio of relative velocities
Γ	Coefficient of diffusivity in the general transport equation
δ	Deviation angle
η	Cross-stream computational domain coordinate, or Efficiency
$\Delta\eta$	Separation in η direction between nodes or control volume walls
θ	Angular coordinate
μ	Effective fluid viscosity
μ_{lam}	Laminar fluid viscosity
μ_t	Turbulent eddy viscosity
ξ	Streamwise computational domain coordinate
$\Delta\xi$	Separation in ξ direction between nodes or control volume walls
ρ	Fluid density
σ	Slip factor
τ	Shear
ϕ	Transported quantity in general transport equation, or Flow coefficient
Φ	Global flow coefficient
ψ	Static head rise coefficient
ω	Under-relaxation factor, or Vorticity, or Angular velocity of the blade
[]	FORTRAN MAX function

Subscripts

<i>crossover</i>	Transition point between inner and outer portion of the boundary layer
<i>e</i>	Quantity related to the eastern face of the control volume
<i>E</i>	Quantity related to the node east of the central node
<i>i,j,k,l</i>	Tensor notation indices
<i>inner</i>	Inner portion of the boundary layer
<i>n</i>	Quantity related to the northern face of the control volume
<i>N</i>	Quantity related to the node north of the central node
<i>outer</i>	Outer portion of the boundary layer
<i>O</i>	Quantity related to the central node
<i>P</i>	Quantity related to the central coefficient based only on the flux discretization
<i>r</i>	Derivative with respect to the radial coordinate, r
<i>s</i>	Quantity related to the southern face of the control volume
<i>S</i>	Quantity related to the node south of the central node
<i>w</i>	Quantity related to the western face of the control volume, or Quantity evaluated on a no-slip wall in the turbulence model
<i>W</i>	Quantity related to the node west of the central node
<i>z</i>	Derivative with respect to the axial coordinate, z
η	Derivative with respect to the η computational domain coordinate
ξ	Derivative with respect to the ξ computational domain coordinate
<i>LE</i>	Leading edge of impeller blade
<i>TE</i>	Trailing edge of impeller blade
1,2,3	Measurement stations for experimental data collection

Superscripts

c	Convective flux
d	Diffusive flux
H	Hybrid, lower-order, differencing scheme
k	Inner loop iteration level
n	Outer loop iteration level
P	Pressure corrector equation
Q	QUICK, higher-order, differencing scheme
r	Quantity related to radial momentum equation
z	Quantity related to axial momentum equation
θ	Quantity related to angular momentum equation
$-$	Dimensional quantity
$-$	Reference quantity for nondimensionalization
$+$	Positive flow direction
$-$	Reversed flow direction
$'$	Correction term, or Quantity used in revised QUICK (QUICKR) differencing scheme

Chapter 1

Introduction

In order to describe the flow within a turbomachine, a number of highly complex flow phenomena must be understood. Turbulence, boundary layer separation, large recirculating flow regions, secondary flows, tip leakage, and vortex shedding are examples of flow phenomena that are difficult to quantify, yet strongly influence machine performance. The relative importance of these different phenomena can increase or decrease as the operating conditions of the turbomachine change. For example, at off-design conditions the flow incidence angles can become large enough to cause separated flows and stalled blades. Off-design conditions can also lead to large variations in the hub-to-tip blade loading, which may induce large recirculating regions within the blade rows. Turbulence and boundary layer effects also complicate the flow, even at the design point.

One of the objectives of work in the field of turbomachinery is to predict the behavior of the flow through blade passages and the resulting performance of the mechanical component. The subject of the present research is the prediction of turbomachinery performance as indicated by the flow in the through-flow, or axisymmetric, plane. Successfully predicting turbomachine performance depends on the ability to predict the significance of the complex flow phenomena and the ability to model the phenomena themselves when necessary.

The goal of this project was to develop a computer program for predicting incompressible steady-state axisymmetric flow, with special emphasis on turbomachinery applications. Only incompressible flows are considered, because there is a large family of turbomachines in which compressibility effects are negligible, and because numerical algorithms cannot be applied equally

well to both incompressible and compressible flows. Only steady-state conditions are considered, because transient behavior is relatively unimportant in a number of turbomachinery applications. Turbomachines are composed of rotating blades or vanes that either impart energy to the fluid or extract energy from the flow. In through-flow turbomachinery analyses, the effects of the blades can be introduced by estimating the forces that the blades exert on the flow and by applying these forces within the blade rows as body forces. Using this approach, the effects of the blades can be accounted for in the axisymmetric plane. Thus, the performance of the turbomachine can be calculated without having to make difficult and costly three-dimensional analyses. Making the assumption of axial symmetry implies that the variations of the flow fields in the angular direction are zero. However, the tangential velocity is nonzero and remains a function of the axial and radial coordinates.

Collecting experimental data inside high-speed rotors and small twisting fluid passages is a formidably expensive and hardware-intensive process. Therefore, relying solely on experimental evaluation can be costly and time consuming during the design and development of a turbomachine. Numerical algorithms which have been verified against experimental or analytical results can be used to supplement experimental data and to provide a means by which performance can be estimated during the design process. Throughout the development of turbomachinery design and analysis methods, numerous simplifying assumptions have been utilized to render the design problem tractable. However, modern numerical techniques have enabled many of the more complex flow phenomena to be analyzed in detail. Discrete mathematics must be used to solve for flow fields of practical interest, because the governing equations of fluid flow are coupled, nonlinear, partial differential equations and analytic solutions exist for only the most primitive geometries and flow conditions.

Numerical techniques combined with the digital computer have presented the turbomachinery field with a powerful tool. With an eventual goal of completely predicting the flow behavior

throughout the entire operating range of a turbomachine, numerous advances in the field of Computational Fluid Dynamics (CFD) have been made over the past two decades. Because of slow computer processing speeds and insufficient memory, early investigators were often required to make many simplifying assumptions regarding the flow. Tremendous increases in computer capabilities coupled with improvements in modern algorithms now allow more realistic calculations to be made.

Hardware and software limitations forced the assumption of inviscid flow upon early numerical investigators of the flow in turbomachines. Methods such as streamline curvature [1] and inviscid cascade analyses cannot predict the important contribution that viscosity makes to the flow behavior. At the design point, inviscid approximations may be acceptable for determining pressure distributions and flow angles. However, inviscid analyses cannot determine total pressure losses and the efficiency of turbomachinery components. At off-design conditions, the inviscid approximation is even less valid.

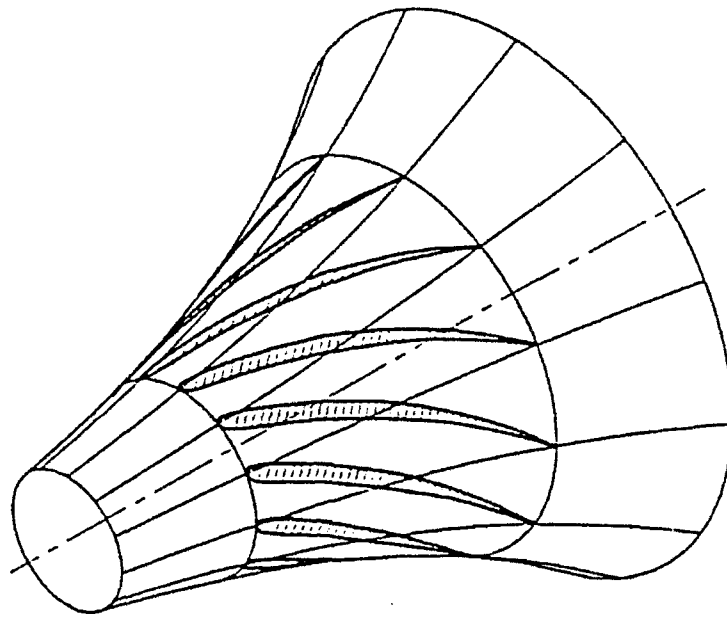
Even if the viscosity of the fluid is accounted for, there are still approximations which can be made which will greatly simplify the governing equations. The first three-dimensional calculations of a centrifugal impeller were made using the partially-parabolic Navier-Stokes equations (PPNS) [2]. Adopting the PPNS assumptions greatly restricts the applicability of the results. Off-design conditions or flows with recirculating regions in the streamwise direction cannot be predicted, because in these cases the equations are no longer parabolic in nature, but become elliptic.

Over the past few years, reports of fully elliptic three-dimensional turbomachinery calculations have been published for axial flow turbines [3], linear turbine cascades [4], axial flow compressors [5], and centrifugal impellers [6]. The results are promising, considering the coarseness of the grids that were used. Turbulence was modeled, and recirculation was predicted, but the results can only be said to be accurate in a qualitative rather than quantitative sense. In 1988, Hah *et al.* [7] stated that no successful viscous three-dimensional calculations had been made for a centrifugal impeller at off-design conditions. Their results, too, show only a fair agreement

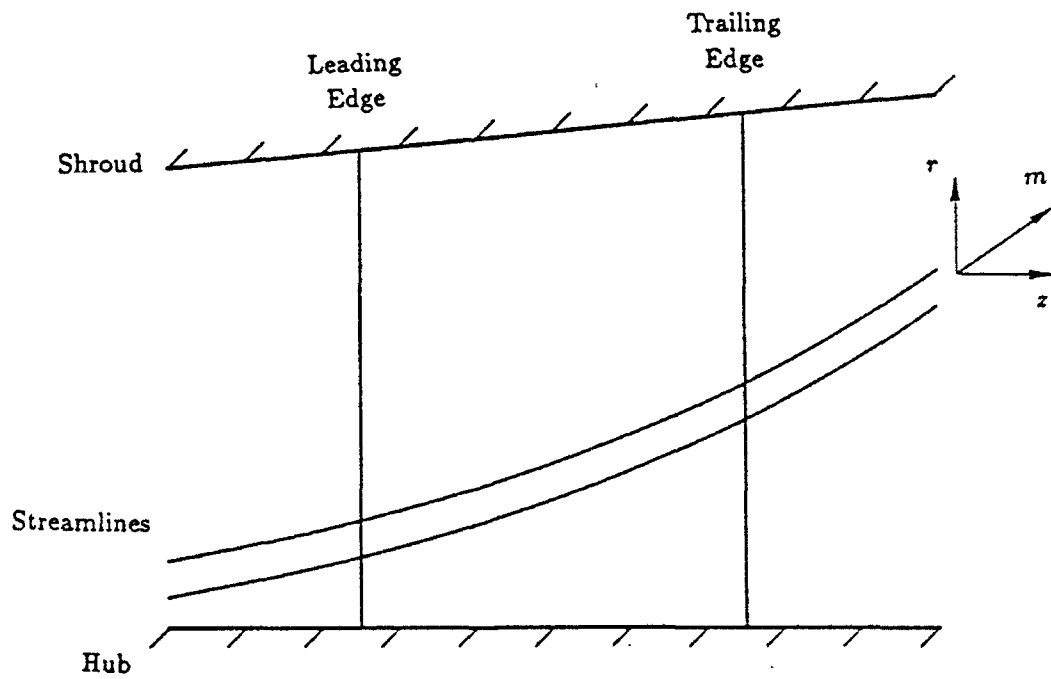
with experimental data. The errors are seemingly due to two fundamental causes. The first is an inability to accurately quantify turbulent phenomena. The second is due to the use of upwind differencing on the coarse grids. This report proposes a way of decreasing this latter error by treating the convection in a manner that is consistent with its radial dependence in cylindrical coordinate systems. Solving three-dimensional flows in a reasonable amount of time, even with a modest size grid, requires the use of supercomputers or high-speed mainframes. These, however, are too expensive and unavailable for widespread use in design applications.

Reducing spatial dimensional dependence from three to two greatly decreases the computational expense and memory requirements needed to solve the flow. In rotating turbomachinery applications, the dimensional cutback means that the flow is solved in either the through-flow plane (S_2 surface) or the blade-to-blade plane (S_1 surface), see Fig. 1.1. Only the through-flow plane, in which axisymmetry is assumed, is treated in this project because of the abundance of information pertaining to the performance of the turbomachine that flow on this surface provides. Efficiency, head rise, total pressure changes, and velocity profiles can all be determined from solutions to the axisymmetric equations, and these solutions can be found in reasonable amounts of computational time. Because viscosity strongly affects turbomachine performance at certain operating conditions, the viscous axisymmetric flow equations are solved. The result is that streamwise recirculating regions, including those within the blade rows can be calculated. To this researcher's knowledge, no one has as yet published a successful calculation of this type of recirculating zone.

In three-dimensional geometries, rotating turbomachine blades are treated using the typical no-slip boundary conditions. However, in the axisymmetric frame, the blades are not part of the flow domain boundary. The geometry for an axisymmetric flow is a pipe or annular section with walls of arbitrary shape. Nowhere in this geometry are blade passages present. The geometry is the same whether analyzing duct flow or turbomachinery through-flow. However, the flows for these two cases are vastly different. Because the geometries are identical, the effects of the blades



a. Blade-to-Blade Plane (S_1 Surface)



b. Meridional Plane (S_2 Surface)

Figure 1.1

Two-Dimensional Solution Surfaces

can only be brought into the momentum equations as external influences. Thus, by applying body forces in the regions where the blade is projected onto the axisymmetric plane, the flow will be effected as though blades are physically present. Axisymmetric flow does not restrict the fluid to motion within a single axial-radial plane, so the external effect of the blades can be thought of as determining the rate that the fluid moves relative to the blade for a given velocity vector in the axisymmetric plane. Specifying the flow angle in this manner is effectively identical to determining the amount of force that is required to turn the flow.

A variety of blade models that approximate the fluid-blade interaction have been presented by previous researchers. Novak and Hearsey [8] solve inviscid quasi-3-D flows by alternating between a through-flow calculation and a blade-to-blade analysis, using the latter to provide flow angles to the through-flow calculation. This approach involves a great deal of computational expense, because it alternately solves two types of flow problems. Multiple blade-to-blade calculations are performed on various radial surfaces of revolution, and these data are coupled to the solution in the through-flow plane. This technique is not applicable to viscous flows at off-design conditions, because strong three-dimensional blade-to-blade effects violate the assumptions made in transferring information between the blade-to-blade and through-flow solutions. Another way of accounting for the presence of the blades is to replace them by the forces they impart to the fluid. Martelli and Michelassi [9] use a single inviscid blade-to-blade calculation to provide the pressures on the suction and pressure sides of the blades. This loading distribution is then used to specify body forces for a viscous through-flow calculation. For off-design operating conditions their approach is not valid, because the assumptions that are used to arrive at the blade loading no longer hold. Jackson *et al.* [10] use a more reasonable approach in which the flow angles are prescribed. They examined flow past a stator, which simplifies the calculation since the increase of total pressure along the streamline does not need to be modeled.

The blade model developed in this investigation is related to the approach of Jackson *et al.*, in that a flow angle is extracted from an estimated mean streamline. Because the geometry of the

blade passage is known, it provides a good basis for estimating the streamline shape. To make the flow solver applicable to general turbomachines, a method for predicting the energy input along the blade row was formulated. The force required to conserve momentum in the meridional direction, in addition to creating an estimated pressure change due to the energy input of the blade, represents the effect the blade has on the conservation of axial and radial momentum.

An analysis tool must be general enough to be useful in a variety of situations. In the field of turbomachinery computational fluid dynamics (CFD), this means that the flow solver must be able to handle drastically different geometries. Towards this end, the governing equations are transformed from the physical plane into a computational plane, which is defined by a set of body-fitted or generalized curvilinear coordinates.

In the chapters that follow, the computer program that performs the through-flow calculations is explained. The governing equations for a viscous, steady-state, incompressible, axisymmetric flow are formally stated in Chapter 2. An explanation of the mapping operation is given in Chapter 3, and its use in transforming the governing equations into the computational plane is presented. Details of Rhie and Chow's [11] Pressure-Weighted Interpolation Method (PWIM) are given in Chapter 4, along with justifications for employing this algorithm in the program. Modifications to their approach have been made, and these are also discussed. The process of discretizing the governing equations is also explained in Chapter 4, along with other considerations arising from the numerical solution algorithm. Special attention is paid to the intricacies brought about by the use of the cylindrical polar coordinate system. To this author's knowledge, no researchers to date have presented modifications to PWIM that specifically adapt the algorithm to cylindrical coordinate based applications. The turbulence model used in the program is explained in Chapter 5. A full explanation of the blade model is contained in Chapter 6. The procedure for verifying the computer program is documented in Chapter 7. Additionally, the results for a number of test cases are given. Through-flow calculations of flow in a centrifugal impeller over a range of operating conditions are presented in Chapter 8. The numerically calculated velocity

profiles and overall performance characteristics are compared to experimental data. The calculated results are shown to successfully predict a number of important phenomena at off-design flow conditions. The numerical results consistently capture the trends of the velocity profiles shown in the experimental data, although overall performance parameters are not as accurately predicted. Modifications to the blade model to improve the performance predictions were not made. Conclusions supported by the research are stated in Chapter 9, and recommendations for future research are also given. The differencing schemes used in the discretization of the momentum flux terms are discussed in the Appendix. An explanation of a new procedure for extending Leonard's higher-order differencing scheme [12] all the way to the boundaries is also included in this section. Additionally, two new differencing schemes that reflect flow characteristics in the cylindrical polar coordinate system are also presented in the Appendix.

Chapter 2

Governing Conservation Equations

By accounting for inertial, pressure, and shear forces, Navier-Stokes based solvers are able to reflect the physics of the flow of a real fluid inside a turbomachine. The Navier-Stokes equations are a set of coupled, nonlinear, elliptic equations. They are also known as the equations of motion or conservation equations, because their solution yields a flow pattern that is consistent with physical conservation laws of mass and momentum. For flows in which density changes can be considered negligible, i.e. incompressible flows, the energy conservation equation is decoupled from the momentum equations. Therefore, the energy equation is not considered in this investigation. Typically, an equation of state closes the system of equations, but the assumption of incompressibility also eliminates the need for this type of relationship.

2.1. Conservation of Mass and Momentum for

Steady-State Incompressible Axisymmetric Flows

Turbomachinery flow passages involve some type of rotating mechanical component that is comprised of blades or vanes. The cylindrical polar coordinate system is naturally associated with such geometries. As the rotary device acts on the fluid media, it changes the flow properties. Unless there are an infinite number of blades, the flow properties will vary in the angular direction. If this variation is neglected or assumed negligible, ($\frac{\partial}{\partial \theta} = 0$), then the flow is independent of the orientation of the viewing plane about the centerline and is axisymmetric.

The governing equations for conservation of mass and momentum [13, 14] are shown in Eqs. 2.2-5. These are the nonconservative forms of the steady-state equations for flow that is axisymmetric and incompressible. Fluid acceleration is given by the substantial derivative, $\frac{D}{Dt}$. Dimensional quantities are denoted by a "ˆ" superscript.

Substantial Derivative:

$$\frac{D}{Dt} = \hat{v}_z \frac{\partial}{\partial \hat{z}} + \hat{v}_r \frac{\partial}{\partial \hat{r}} \quad (2.1)$$

Continuity:

$$\frac{\partial(\hat{\rho}\hat{v}_z)}{\partial \hat{z}} + \frac{1}{\hat{r}} \frac{\partial(\hat{r}\hat{v}_r)}{\partial \hat{r}} = 0 \quad (2.2)$$

z - momentum:

$$\begin{aligned} \hat{\rho} \frac{D\hat{v}_z}{Dt} = & -\frac{\partial \hat{p}}{\partial \hat{z}} + \hat{F}_z + \\ & 2 \frac{\partial}{\partial \hat{z}} \left(\hat{\mu} \frac{\partial \hat{v}_z}{\partial \hat{z}} \right) + \frac{1}{\hat{r}} \frac{\partial}{\partial \hat{r}} \left[\hat{\mu} \hat{r} \left(\frac{\partial \hat{v}_r}{\partial \hat{z}} + \frac{\partial \hat{v}_z}{\partial \hat{r}} \right) \right] \end{aligned} \quad (2.3)$$

r - momentum:

$$\begin{aligned} \hat{\rho} \left[\frac{D\hat{v}_r}{Dt} - \frac{\hat{v}_\theta^2}{\hat{r}} \right] = & -\frac{\partial \hat{p}}{\partial \hat{r}} + \hat{F}_r + \\ & 2 \frac{\partial}{\partial \hat{r}} \left(\hat{\mu} \frac{\partial \hat{v}_r}{\partial \hat{r}} \right) + \frac{\partial}{\partial \hat{z}} \left[\hat{\mu} \left(\frac{\partial \hat{v}_r}{\partial \hat{z}} + \frac{\partial \hat{v}_z}{\partial \hat{r}} \right) \right] + \frac{2\hat{\mu}}{\hat{r}} \left(\frac{\partial \hat{v}_r}{\partial \hat{r}} - \frac{\hat{v}_r}{\hat{r}} \right) \end{aligned} \quad (2.4)$$

θ - momentum:

$$\begin{aligned} \hat{\rho} \left[\frac{D\hat{v}_\theta}{Dt} + \frac{\hat{v}_r \hat{v}_\theta}{\hat{r}} \right] = & \hat{F}_\theta + \\ & \frac{\partial}{\partial \hat{z}} \left(\hat{\mu} \frac{\partial \hat{v}_\theta}{\partial \hat{z}} \right) + \frac{\partial}{\partial \hat{r}} \left[\hat{\mu} \left(\frac{\partial \hat{v}_\theta}{\partial \hat{r}} - \frac{\hat{v}_\theta}{\hat{r}} \right) \right] + \frac{2\hat{\mu}}{\hat{r}} \left(\frac{\partial \hat{v}_\theta}{\partial \hat{r}} - \frac{\hat{v}_\theta}{\hat{r}} \right) \end{aligned} \quad (2.5)$$

Axial symmetry removes the tangential component of velocity from the continuity equation, but v_θ is still coupled to the other velocity components through the momentum equations. Symmetry also removes the explicit effects of pressure from the angular momentum equation, but v_θ remains coupled to the pressure field through the centripetal acceleration term (v_θ^2/r) in the radial momentum equation. The bulk viscosity does not appear in the momentum equations

because the divergence of the velocity is zero for incompressible flows. Spatially varying viscosity is permitted with the form of the equations given. This is necessary since turbulence modeling is implemented by supplementing the laminar viscosity with a local turbulent viscosity (Chapter 5). Body forces per unit volume, F in the respective directions, provide a means by which the presence of external entities, such as turbomachine blades, can be introduced into the momentum equations. A complete discussion of the model which replaces the blades with the forces they impart to the fluid is given in Chapter 6.

2.2 Nondimensionalization of the Governing Equations

It is beneficial to express the governing equations in terms of nondimensional groups so that the flow may be characterized by standard parameters. Dimensionless variables are also advantageous from a numerical standpoint, because the values are usually scaled so that they lie between zero and one, thereby reducing floating point errors associated with very large or small numbers.

The first step in nondimensionalizing the equations is to select reference values. Buckingham's Pi theory allows for the specification of four reference quantities. All other terms are scaled by combinations of these values. The "--" superscript in Eq. 2.6 indicates a characteristic dimensional quantity.

$$\hat{z}_i = \bar{z} z_i \quad (2.6a)$$

$$\hat{v}_i = \bar{v} v_i \quad (2.6b)$$

$$\hat{\rho} = \bar{\rho} \rho \quad (2.6c)$$

$$\hat{\mu} = \bar{\mu} \mu \quad (2.6d)$$

$$\hat{p} = \bar{p} \, p = (\bar{\rho} \bar{v}^2) \, p \quad (2.6e)$$

$$\hat{F}_i = \bar{F} \, F_i = \left(\frac{\bar{\rho} \bar{v}^2}{\bar{r}} \right) F_i \quad (2.6f)$$

$$Re = \frac{\bar{\rho} \bar{v} \bar{r}}{\bar{\mu}} \quad (2.6g)$$

The flow geometry determines the reference length, and the flow conditions set the appropriate characteristic velocity. For incompressible flow, the reference density is equal to the fluid density, which is a constant. Therefore, the nondimensional density is unity, but it is retained in the equations for clarity. The fluid's molecular viscosity is the most convenient reference value for the viscosity. Thus, for laminar flow the nondimensional viscosity is also one.

The Reynolds number, Re , which represents the ratio of inertial to viscous forces, appears naturally in the momentum equations when they are nondimensionalized. Turbomachines operate in the high Reynolds number regime. Thus, to minimize floating point errors during the numerical solution, it is better to divide the viscous terms by Re rather than multiply the inertial terms by the parameter. The resulting nondimensional equations are listed in Eqs. 2.8-11.

Substantial Derivative:

$$\frac{D}{Dt} = v_z \frac{\partial}{\partial z} + v_r \frac{\partial}{\partial r} \quad (2.7)$$

Continuity:

$$\frac{\partial(\rho v_z)}{\partial z} + \frac{\partial(\rho r v_r)}{\partial r} = 0 \quad (2.8)$$

z - momentum:

$$\begin{aligned} \rho \frac{Dv_z}{Dt} &= -\frac{\partial p}{\partial z} + F_z + \\ &\frac{1}{Re} \left[2 \frac{\partial}{\partial z} \left(\mu \frac{\partial v_z}{\partial z} \right) + \frac{1}{r} \frac{\partial}{\partial r} \left(\mu r \left(\frac{\partial v_r}{\partial z} + \frac{\partial v_z}{\partial r} \right) \right) \right] \end{aligned} \quad (2.9)$$

r - momentum:

$$\begin{aligned} \rho \left[\frac{Dv_r}{Dt} - \frac{v_\theta^2}{r} \right] &= -\frac{\partial p}{\partial r} + F_r + \\ &\frac{1}{Re} \left[2 \frac{\partial}{\partial r} \left(\mu \frac{\partial v_r}{\partial r} \right) + \frac{\partial}{\partial z} \left(\mu \left(\frac{\partial v_r}{\partial z} + \frac{\partial v_z}{\partial r} \right) \right) + \frac{2\mu}{r} \left(\frac{\partial v_r}{\partial r} - \frac{v_r}{r} \right) \right] \end{aligned} \quad (2.10)$$

θ - momentum:

$$\rho \left[\frac{Dv_\theta}{Dt} + \frac{v_r v_\theta}{r} \right] = F_\theta + \frac{1}{Re} \left[\frac{\partial}{\partial z} \left(\mu \frac{\partial v_\theta}{\partial z} \right) + \frac{\partial}{\partial r} \left(\mu \left(\frac{\partial v_\theta}{\partial r} - \frac{v_\theta}{r} \right) \right) + \frac{2\mu}{r} \left(\frac{\partial v_\theta}{\partial r} - \frac{v_\theta}{r} \right) \right] \quad (2.11)$$

The form of the continuity equation, Eq. 2.8, is not altered by the nondimensionalization because, it contains only nonviscous terms.

2.3 Conservative Form of the Governing Equations

The fundamental difference between the governing equations in cylindrical and Cartesian coordinates is that surface area is dependent upon the radial displacement in the cylindrical frame of reference. When rotating an area in the $r - z$ plane about the centerline, the volume swept out is a function of the radius of the centroid of the area. In control volume (CV) formulations, this is manifested as a radial dependence of the surface area through which the flux is carried into the control volume.

Consider an elemental line segment whose midpoint is displaced from the centerline by r , as shown in Fig. 2.1. When rotated about the centerline, the line segment forms a frustum of a cone. The mass flow rate across the segment is equal to:

$$\dot{m} = 2\pi\rho r(v_z dr - v_r dz) \quad (2.12)$$

The factor of 2π is neglected in further discussions, since it will appear in the integral of every axisymmetric term and can, therefore, be factored out.

In the control volume formulation, four such segments are joined together to form a quadrilateral which serves as a control volume. Thus, the mass flux into the CV is dependent only on the facial properties and their radial locations. This presents a problem concerning the form in

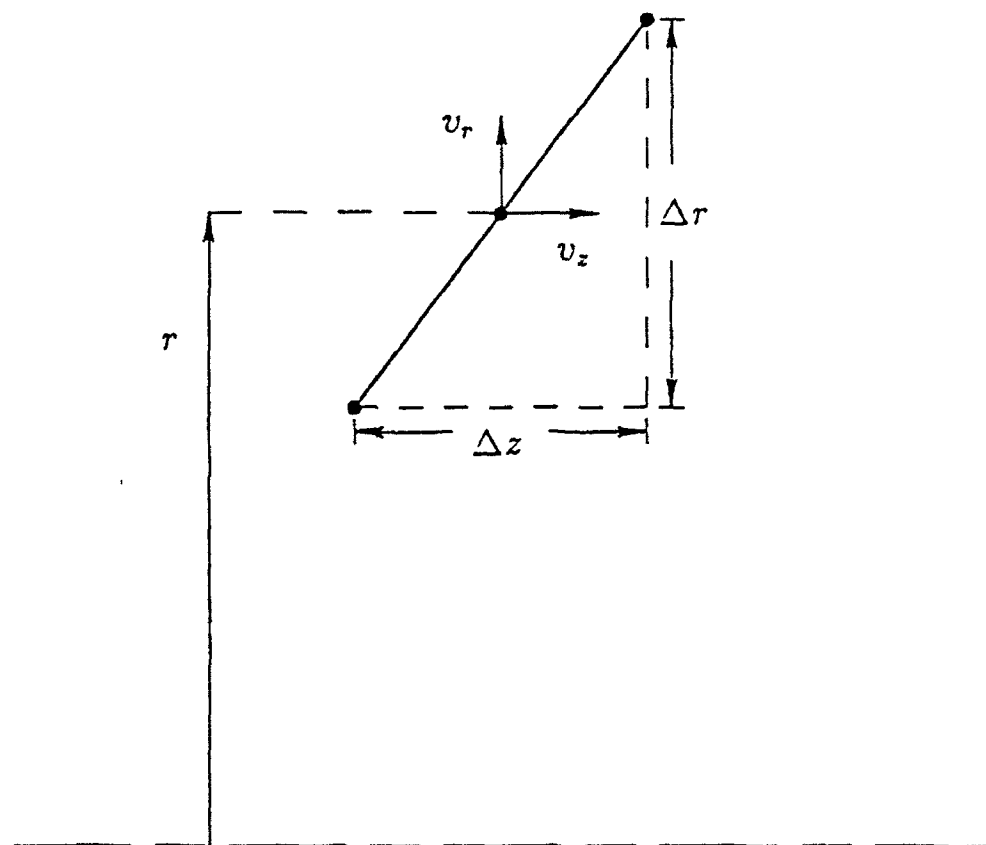


Figure 2.1

Elemental Segment of Control Volume Wall

which the conservation equations are posed. The inertial terms in Eqs. 2.8-11 do not contain the convective velocities multiplied by the radius, but these terms can be algebraically manipulated to yield the desired form. These new expressions are described as being conservative. An equation is said to be in conservative form if the coefficients of the derivatives are either constant, or if variable, their derivatives do not appear anywhere in the equation [15, p. 16]. Algebraically, the equations are identical regardless of their form; however, the solution to the equations given in Section 2.2 is not found using exact analytical techniques. Discrete mathematics is used to solve these nonlinear equations, and the final solution is dependent upon the form of the discretized governing equations. By solving the equations in their conservative form, the flux crossing one control volume face will be the same for each of the adjacent control volumes that share that face. Global, as well as local, flux conservation is ensured by expressing the equations in conservative form. Flux conservation is one aspect of control volume formulations which yields meaningful results, even on coarse grids [16, p. 31].

The essential relationships for changing from nonconservative to conservative form are shown in Eq. 2.13. Because the cylindrical coordinate system is orthogonal, r is independent of the axial coordinate and may be brought inside the derivatives with respect to z .

$$\frac{\partial v_r}{\partial r} + \frac{v_r}{r} = \frac{1}{r} \frac{\partial(rv_r)}{\partial r} \quad (2.13a)$$

$$\frac{\partial v_z}{\partial z} = \frac{1}{r} \frac{\partial(rv_z)}{\partial z} \quad (2.13b)$$

Eqs. 2.14-17 present the conservative form of the governing equations. Not only are the convection terms written in conservative fashion, but several viscous terms have been restructured into this more convenient format. Each equation has been multiplied through by r to remove the $1/r$ factors which are present in Eq. 2.13. Continuity was used in conjunction with Eq. 2.13 to eliminate the substantial derivative from the momentum equations.

Continuity:

$$\frac{\partial(\rho rv_z)}{\partial z} + \frac{\partial(\rho rv_r)}{\partial r} = 0 \quad (2.14)$$

z - momentum:

$$\begin{aligned} \frac{\partial(\rho v_z v_z)}{\partial z} + \frac{\partial(\rho v_r v_z)}{\partial r} &= -r \frac{\partial p}{\partial z} + r F_z + \\ \frac{1}{Re} \left[\frac{\partial}{\partial z} \left(\mu r \frac{\partial v_z}{\partial z} \right) + \frac{\partial}{\partial r} \left(\mu r \frac{\partial v_z}{\partial r} \right) \right] &+ \frac{1}{Re} \left[\frac{\partial}{\partial z} \left(\mu r \frac{\partial v_r}{\partial z} \right) + \frac{\partial}{\partial r} \left(\mu r \frac{\partial v_r}{\partial r} \right) \right] \end{aligned} \quad (2.15)$$

r - momentum:

$$\begin{aligned} \frac{\partial(\rho v_z v_r)}{\partial z} + \frac{\partial(\rho v_r v_r)}{\partial r} - \rho v_\theta^2 &= -r \frac{\partial p}{\partial r} + r F_r - \frac{1}{Re} \frac{2\mu v_r}{r} + \\ \frac{1}{Re} \left[\frac{\partial}{\partial z} \left(\mu r \frac{\partial v_r}{\partial z} \right) + \frac{\partial}{\partial r} \left(\mu r \frac{\partial v_r}{\partial r} \right) \right] &+ \frac{1}{Re} \left[\frac{\partial}{\partial z} \left(\mu r \frac{\partial v_z}{\partial r} \right) + \frac{\partial}{\partial r} \left(\mu r \frac{\partial v_z}{\partial r} \right) \right] \end{aligned} \quad (2.16)$$

θ - momentum:

$$\begin{aligned} \frac{\partial(\rho v_z v_\theta)}{\partial z} + \frac{\partial(\rho v_r v_\theta)}{\partial r} + \rho v_r v_\theta &= r F_\theta - \frac{1}{Re} \frac{v_\theta}{r} \frac{\partial(\mu r)}{\partial r} + \\ \frac{1}{Re} \left[\frac{\partial}{\partial z} \left(\mu r \frac{\partial v_\theta}{\partial z} \right) + \frac{\partial}{\partial r} \left(\mu r \frac{\partial v_\theta}{\partial r} \right) \right] \end{aligned} \quad (2.17)$$

2.4 Comparison of the Cylindrical and Cartesian Forms of the Conservation Equations

The two-dimensional conservation equations in Cartesian coordinates for steady-state incompressible flow [17, p. 64] can be nondimensionalized and expressed in conservative fashion using an approach similar to that given in the previous sections. In the Cartesian coordinate system there is no counterpart to the radial distance, so relationships analogous to those in Eq. 2.13 are not used. Instead, the continuity equation is enough to transform the substantial derivative into conservative form. The resulting continuity and momentum equations are given in Eqs. 2.18-20.

Continuity:

$$\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = 0 \quad (2.18)$$

x - momentum:

$$\begin{aligned} \frac{\partial(\rho u u)}{\partial x} + \frac{\partial(\rho v u)}{\partial y} &= -\frac{\partial p}{\partial x} + F_x + \\ \frac{1}{Re} \left[\frac{\partial}{\partial x} \left(\mu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial u}{\partial y} \right) \right] &+ \frac{1}{Re} \left[\frac{\partial}{\partial x} \left(\mu \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v}{\partial y} \right) \right] \end{aligned} \quad (2.19)$$

y - momentum:

$$\frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho vv)}{\partial y} = -\frac{\partial p}{\partial y} + F_y + \frac{1}{Re} \left[\frac{\partial}{\partial x} \left(\mu \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v}{\partial y} \right) \right] + \frac{1}{Re} \left[\frac{\partial}{\partial x} \left(\mu \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v}{\partial x} \right) \right] \quad (2.20)$$

Since the orientation of the plane in which the axisymmetric equations are viewed is arbitrary, the centerline is specified to run horizontally and the radial axis vertically. This is the most convenient arrangement because it aligns the directions of the cylindrical polar (z, r) and Cartesian (x, y) base vectors. The velocity components v_z and u are parallel, as are v_r and v .

Eqs. 2.14-16 and Eqs. 2.18-20 are equivalent when three conditions are met:

1. The tangential velocity is zero everywhere.
2. The radius is unity wherever it appears as a multiplicative factor.
3. The radius is infinity wherever it appears in the denominator.

Under these conditions the axisymmetric equations are identical to those in the two-dimensional Cartesian coordinate system. In the absence of blades or moving boundaries, the first condition is met by setting the inlet swirl velocity, v_θ , equal to zero. In addition, setting $r = 1$ and $1/r = 0$, enables the axisymmetric flow solver to solve flows in a Cartesian coordinate system as well. The cost of implementing this capability is one extra memory array which allows r and $1/r$ to be stored separately.

Chapter 3

Mapping from the Physical to the Computational Plane

For the complex flow geometries that are typically found in turbomachinery, it is often difficult or even impossible to generate quality orthogonal grids to cover the physical domain. Also, for a grid to provide quality solutions, nodes must be concentrated in regions where severe gradients are to be resolved. From a numerical standpoint nonorthogonality and nonuniformity each create difficulties. Only orthogonal grids possess the desirable feature of having each contravariant velocity convect fluid across only one family of gridlines. Diffusion terms are more easily evaluated when the gridlines intersect at right angles, because the gradient along one gridline is independent of the gradient along the other gridline. Solving the governing equations discretely also creates problems associated with the spacing of the nodes. Unequally spaced grids cause large increases in the computational expense and memory associated with the discretization process. Even when a control volume, rather than finite difference, approach is used, Taylor series approximations are still employed in the evaluation of gradients. Uniform grids enable high-order Taylor series approximations to be made with less computational effort than with nonuniform grids.

The benefits of uniform orthogonal grids can be achieved on a grid that is neither orthogonal nor uniform. This is accomplished by mapping the physical domain into a computational domain in which the grid is orthogonal and has unit spacing. The governing equations are actually solved on this computational grid. Since the grid on which the equations are solved lies in a new plane, the governing equations must also be transformed so that they insure conservation in the computational plane. In general, the transformation increases the complexity of the governing

equations by introducing additional terms which are not present in the governing equations expressed in the physical space. All of the information concerning the nonorthogonality and nonuniformity of the grid in the physical space appears as fixed multiplicative factors in these additional terms. The overall solution of the more complex equations in the computational space is much less involved than the solution of the simpler equations directly on the physical space grid.

In the physical coordinate system in which the grid is generated, the contours of the geometric boundaries may not conform to curves of a constant coordinate. Thus, there can exist regions where the discretized boundary lies a finite distance away from the true boundary. Solving the governing equations on this type of grid would prevent an accurate application of the boundary conditions. Thus, the body-fitted coordinates or generalized curvilinear coordinates are employed. After transforming the body-fitted coordinates into the computational domain, the boundaries of the computational grid correspond precisely to the geometric boundaries, so an accurate application of the boundary conditions is possible.

The mapping operation is defined in the next section. This is followed by an explanation of how the governing equations are transformed from the physical space into the computational domain.

3.1 Specification of the Mapping Operation

The transformation from cylindrical to generalized curvilinear coordinates is a simple chain rule expansion of the axial and radial coordinates (z, r) in terms of the ξ and η coordinates in the computational plane.

$$\begin{pmatrix} \frac{\partial}{\partial z} \\ \frac{\partial}{\partial r} \end{pmatrix} = \begin{pmatrix} \xi_z & \eta_z \\ \xi_r & \eta_r \end{pmatrix} \begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{pmatrix} \quad (3.1)$$

If the grid on which the problem is to be solved is mapped into a computational plane, the governing equations must also be transformed. After solving the system of equations, the results must be referenced back to the corresponding locations of the grid nodes in the physical space. Therefore, the mapping procedure must produce a set of points in the computational space for which there is a one-to-one correspondence with the points in the physical space. To achieve this one-to-one correspondence, the inverse transformation defined by Eq. 3.2 must exist.

$$\begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{pmatrix} = \begin{pmatrix} z_\xi & r_\xi \\ z_\eta & r_\eta \end{pmatrix} \begin{pmatrix} \frac{\partial}{\partial z} \\ \frac{\partial}{\partial r} \end{pmatrix} \quad (3.2)$$

Equating the coefficient matrix of the inverse mapping function (from Eq. 3.2) to the inverse of the mapping function coefficient matrix (from Eq. 3.1) yields the closed description of the transformation process.

$$\begin{pmatrix} z_\xi & r_\xi \\ z_\eta & r_\eta \end{pmatrix} = \begin{pmatrix} \xi_z & \eta_z \\ \xi_r & \eta_r \end{pmatrix}^{-1} \quad (3.3)$$

$$\xi_z = r_\eta / J, \quad \xi_r = -z_\eta / J$$

$$\eta_z = -r_\xi / J, \quad \eta_r = z_\xi / J \quad (3.4)$$

$$J = z_\xi r_\eta - z_\eta r_\xi \quad (3.5)$$

The Jacobian, J , of the system represents the ratio of the local physical space area to the corresponding computational plane area. As long as J is nonzero, the inverse mapping function will exist. If the value of the Jacobian vanishes, it indicates that a point in the physical space corresponds to a region in the computational domain, a violation of the one-to-one condition of the mapping. Positive area in the computational plane is assured by definition; therefore, if the Jacobian is negative, the physical grid possesses locally overlapping gridlines. This indicates that the original grid is inadequate to describe the physical system and must be changed. Otherwise, the numerical solver will inevitably fail to function or fail to yield meaningful results.

The terms in Eq. 3.4 are known as metrics. The subscript denotes partial differentiation with respect to the coordinate variable. Since the spacing between nodes in the computational grids is defined as unity, the metrics r_ξ, z_ξ, z_η , and r_η can be evaluated using finite difference approximations. Second-order accuracy in the metrics is achieved by using central differencing in the interior of the grid and one-sided differencing at the boundaries. The curvilinear coordinates are not directly known as functions of (z, r) ; therefore, the inverse formulations, Eq. 3.4, must be used to evaluate the metrics ξ_z, ξ_r, η_z , and η_r .

In an effort to minimize the complexity of the representation of the transport equations, tensor summation notation is adopted and used throughout this work. Repeated indices indicate a summation over the index from one to two, with the following equalities specifying the indicial correspondence:

$$\frac{\partial}{\partial z_1} = \frac{\partial}{\partial z}; \quad \frac{\partial}{\partial z_2} = \frac{\partial}{\partial r} \quad (3.6)$$

$$\frac{\partial}{\partial \xi_1} = \frac{\partial}{\partial \xi}; \quad \frac{\partial}{\partial \xi_2} = \frac{\partial}{\partial \eta} \quad (3.7)$$

Thus, the mapping operation is defined as follows:

$$\frac{\partial}{\partial z_i} = \frac{\partial \xi_j}{\partial z_i} \frac{\partial}{\partial \xi_j} \quad (3.8)$$

In the computational plane, it is desirable to express the terms in the governing equations in strongly conservative form. To do this, the metrics must be brought inside the derivatives. As shown in the resulting identity given in Eq. 3.11, this can be accomplished by bringing the Jacobian into the derivatives along with the metrics. In this development, the symbol ϕ represents a general transport quantity for which the derivative is to be determined. Starting with the application of the chain rule, Eq. 3.9 is obtained.

$$\frac{\partial \phi}{\partial z_i} = \frac{1}{J} \left(J \frac{\partial \xi_j}{\partial z_i} \right) \frac{\partial \phi}{\partial \xi_j} = \frac{1}{J} \frac{\partial}{\partial \xi_j} \left(J \frac{\partial \xi_j}{\partial z_i} \phi \right) - \frac{1}{J} \phi \frac{\partial}{\partial \xi_j} \left(J \frac{\partial \xi_j}{\partial z_i} \right) \quad (3.9)$$

By expanding over repeated indices, it is shown in Eq. 3.10 that the last term in Eq. 3.9 is equal to zero for z_i equaling both z and r .

$$\frac{1}{J}\phi \frac{\partial}{\partial \xi_j} \left(J \frac{\partial \xi_j}{\partial z} \right) = \frac{1}{J}\phi \left(\frac{\partial}{\partial \xi} \frac{\partial r}{\partial \eta} - \frac{\partial}{\partial \eta} \frac{\partial r}{\partial \xi} \right) = 0 \quad (3.10a)$$

$$\frac{1}{J}\phi \frac{\partial}{\partial \xi_j} \left(J \frac{\partial \xi_j}{\partial r} \right) = \frac{1}{J}\phi \left(\frac{\partial}{\partial \xi} \frac{\partial z}{\partial \eta} - \frac{\partial}{\partial \eta} \frac{\partial z}{\partial \xi} \right) = 0 \quad (3.10b)$$

After interchanging the order of differentiation, which is permissible because ξ and η are smooth, continuous functions of z and r , the terms within the parentheses cancel. Thus, the desired identity is obtained:

$$\frac{\partial \phi}{\partial z_i} = \frac{1}{J} \frac{\partial}{\partial \xi_j} \left(J \frac{\partial \xi_j}{\partial z_i} \phi \right) \quad (3.11)$$

When the conservation equations are transformed from the physical plane to the computational domain, metric invariance with respect to the order of differentiation is required so that Eq. 3.10 is valid. This can be accomplished analytically by making use of the identity shown in Eq. 3.11 during the transformation of the governing equations, or it can be done numerically by insuring the finite difference forms of the metrics satisfy:

$$\frac{\partial}{\partial \xi} \frac{\partial z}{\partial \eta} = \frac{\partial}{\partial \eta} \frac{\partial z}{\partial \xi} \quad (3.12a)$$

$$\frac{\partial}{\partial \xi} \frac{\partial r}{\partial \eta} = \frac{\partial}{\partial \eta} \frac{\partial r}{\partial \xi} \quad (3.12b)$$

Actually, Eq. 3.12 should be met regardless of the approach taken to prevent the metric invariance from effecting the governing equations, because it ensures more accurate expressions of the metrics. As long as the metrics are evaluated on a local scale, Eq. 3.12 will hold. It will not hold if the metrics are interpolated using quantities at neighboring regions. This type of interpolation destroys the meaning of the Jacobian, which, as previously stated, is the ratio of the local physical space area to that in the computational plane.

3.2 Transformation of the Governing Equations into Generalized Curvilinear Coordinates

If the governing axisymmetric equations are to be solved in the computational space, then they must be transformed to an equivalent form in that plane. A discussion of this transformation is given in this section.

On the left hand side of each of the four conservation equations (Eq. 2.14-17) there exists a convection term of the form:

$$\frac{\partial(\rho r v_j \phi)}{\partial z_j} \quad (3.13)$$

where ϕ represents the transported quantity. For the continuity equation, $\phi = 1$. For the momentum equations, ϕ is simply the corresponding velocity components. This term is transformed by reexpressing the derivatives in terms of gradients with respect to (ξ, η) . The metrics are then brought inside, using the identity in Eq. 3.11.

$$\frac{\partial(\rho r v_j \phi)}{\partial z_j} = \frac{\partial \xi_k}{\partial z_j} \frac{\partial(\rho r v_j \phi)}{\partial \xi_k} = \frac{1}{J} \frac{\partial}{\partial \xi_k} \left(\rho r \left(J \frac{\partial \xi_k}{\partial z_j} v_j \right) \phi \right) \quad (3.14)$$

The velocities and metrics within the derivative can be combined to form scaled contravariant velocities in the computational plane, (U, V) , defined by the following equations:

$$U = J(\xi_z v_z + \xi_r v_r) = r_\eta v_z - z_\eta v_r \quad (3.15a)$$

$$V = J(\eta_z v_z + \eta_r v_r) = z_\xi v_r - r_\xi v_z \quad (3.15b)$$

$$U_j = J \frac{\partial \xi_j}{\partial z_k} v_k \quad (3.15c)$$

Throughout this paper U and V are referred to as contravariant or convective velocities. Actually, they are the Jacobian times the rate the fluid particles travel in the ξ and η directions, respectively. The J scaling factor is explicitly ignored in the name, but it remains implied.

Substituting Eq. 3.15c into Eq. 3.14 yields the transformed convection term:

$$\frac{\partial(\rho r v_j \phi)}{\partial z_j} = \frac{1}{J} \frac{\partial(\rho r U_j \phi)}{\partial \xi_j} \quad (3.16)$$

To verify that U_j is a contravariant velocity, consider once again the elemental segment shown in Fig. 2.1. If it is mapped into a segment of constant ξ or η , the metrics become the ratio of the lengths. Multiplying the contravariant velocities by the radius and the density gives:

$$\rho r U = \rho r (v_z \Delta r - v_r \Delta z) / \Delta \eta \quad (3.17a)$$

$$\rho r V = \rho r (v_r \Delta z - v_z \Delta r) / \Delta \xi \quad (3.17b)$$

Recalling that the nodal spacing in the computational plane is unity, the $\Delta \xi$ and $\Delta \eta$ factors can be ignored. Comparing these relationships with those of the mass flow rate in Eq. 2.12, it can be seen that the contravariant velocities are responsible for transporting all mass across a control volume face. This is proof that U and V are contravariant rather than covariant velocities; more importantly, it demonstrates the reason for using the conservative form of the equations. The transformed continuity equation contains the $\rho r U$ and $\rho r V$ terms. When integrated over the area, the transformed continuity equation states that the net mass accumulation in the control volume is zero. Thus, the transformation to computational space has not changed the effective meaning of the continuity equation. This would not be the case if the nonconservative form of the continuity equation (Eq. 3.18) had been transformed.

$$\frac{\partial(\rho v_z)}{\partial z} + \frac{\partial(\rho v_r)}{\partial r} + \frac{\rho v_r}{r} = 0 \quad (3.18)$$

The body force terms are not affected by the transformation, since no spatial derivatives are involved. The same holds true for the centripetal acceleration term in the radial momentum equation and its counterpart in the angular momentum equation. A direct application of the mapping equations is used to transform the pressure gradients

Each momentum equation contains a diffusion term which appears in the general transport equation. In keeping with convention, ϕ is substituted for the transported quantities, and the

diffusive coefficient, Γ , replaces the viscosity. This term is transformed by applying Eq. 3.8 to the inner gradient and 3.11 to the outer gradient.

$$\frac{\partial}{\partial z_j} \left(\Gamma r \frac{\partial \phi}{\partial z_j} \right) = \frac{\partial \xi_j}{\partial z_l} \frac{\partial}{\partial \xi_j} \left(\Gamma r \frac{\partial \xi_k}{\partial z_l} \frac{\partial \phi}{\partial \xi_k} \right) = \frac{1}{J} \frac{\partial}{\partial \xi_j} \left(\Gamma r J \left(\frac{\partial \xi_j}{\partial z_l} \frac{\partial \xi_k}{\partial z_l} \right) \frac{\partial \phi}{\partial \xi_k} \right) \quad (3.19)$$

After transforming the derivatives with respect to the physical space coordinates, all that remains are gradients with respect to the computational plane coordinates and terms involving the transformation metrics. A new variable, g_{jk} , is created to represent the combination of metrics which appear in the diffusion terms. This variable is defined in Eq. 3.20 for each indicial permutation. When j and k are not equal, the resulting diffusion is known as cross-diffusion, because it arises out of the grid nonorthogonality.

$$g_{11} = \xi_z^2 + \xi_r^2 = (z_\eta^2 + r_\eta^2)/J^2 \quad (3.20a)$$

$$g_{22} = \eta_z^2 + \eta_r^2 = (z_\xi^2 + r_\xi^2)/J^2 \quad (3.20b)$$

$$g_{12} = g_{21} = \xi_z \eta_z + \xi_r \eta_r = -(z_\xi z_\eta + r_\xi r_\eta)/J^2 \quad (3.20c)$$

$$g_{jk} = \frac{\partial \xi_j}{\partial z_l} \frac{\partial \xi_k}{\partial z_l} \quad (3.20d)$$

Substituting g_{jk} for the metric combinations yields the following final form of the diffusion term:

$$\frac{\partial}{\partial z_j} \left(\Gamma r \frac{\partial \phi}{\partial z_j} \right) = \frac{1}{J} \frac{\partial}{\partial \xi_j} \left(\Gamma r J g_{jk} \frac{\partial \phi}{\partial \xi_k} \right) \quad (3.21)$$

Another term has a structure common to both the radial and axial momentum equations. It is of the form:

$$\frac{\partial}{\partial z_j} \left(\mu r \frac{\partial v_j}{\partial z_i} \right) \quad (3.22)$$

where z_i indicates which momentum equation is under consideration. This term is mapped using the same steps shown in Eq. 3.19. The metrics from the outer derivative cannot be brought

into the inner derivative and combined with v_j to create the contravariant velocities, because the indices do not conform to those appearing in the identity of Eq. 3.11.

$$\frac{\partial}{\partial z_j} \left(\mu r \frac{\partial v_j}{\partial z_i} \right) = \frac{1}{J} \frac{\partial}{\partial \xi_k} \left(\frac{\mu r}{J} \left(J \frac{\partial \xi_k}{\partial z_j} \right) \left(J \frac{\partial \xi_l}{\partial z_i} \right) \frac{\partial v_j}{\partial \xi_l} \right) \quad (3.23)$$

When expanded over repeated indices, eight separate terms are formed. The inability to reexpress this quantity in a simplified form has a negative impact from the standpoint of computational efficiency. However, the inefficiency is lessened by the appearance of some of these gradients in the cross-diffusion terms of Eq. 3.21.

The remaining terms in the momentum equations are coordinate specific, each of which is transformed by a direct application of the mapping functions. Gathering together each component term in the continuity and momentum equations and multiplying through by the Jacobian gives the conservative form of the nondimensional governing equations in generalized curvilinear coordinates.

Continuity:

$$\frac{\partial(\rho r U)}{\partial \xi} + \frac{\partial(\rho r V)}{\partial \eta} = 0 \quad (3.24)$$

z - momentum:

$$\begin{aligned} \frac{\partial(\rho r U v_z)}{\partial \xi} + \frac{\partial(\rho r V v_z)}{\partial \eta} = & r J F_z - r \left(J \frac{\partial \xi_j}{\partial z} \right) \frac{\partial p}{\partial \xi_j} + \\ & \frac{1}{Re} \left[\frac{\partial}{\partial \xi_j} \left(\mu r J g_{jk} \frac{\partial v_z}{\partial \xi_k} \right) + \frac{\partial}{\partial \xi_k} \left(\frac{\mu r}{J} \left(J \frac{\partial \xi_k}{\partial z_j} \right) \left(J \frac{\partial \xi_l}{\partial z} \right) \frac{\partial v_j}{\partial \xi_l} \right) \right] \end{aligned} \quad (3.25)$$

r - momentum:

$$\begin{aligned} \frac{\partial(\rho r U v_r)}{\partial \xi} + \frac{\partial(\rho r V v_r)}{\partial \eta} = & J \rho v_\theta^2 + r J F_r - r \left(J \frac{\partial \xi_j}{\partial r} \right) \frac{\partial p}{\partial \xi_j} + \\ & \frac{1}{Re} \left[\frac{\partial}{\partial \xi_j} \left(\mu r J g_{jk} \frac{\partial v_r}{\partial \xi_k} \right) + \frac{\partial}{\partial \xi_k} \left(\frac{\mu r}{J} \left(J \frac{\partial \xi_k}{\partial z_j} \right) \left(J \frac{\partial \xi_l}{\partial r} \right) \frac{\partial v_j}{\partial \xi_l} \right) - \frac{2 \mu J v_r}{r} \right] \end{aligned} \quad (3.26)$$

θ - momentum:

$$\begin{aligned} \frac{\partial(\rho r U v_\theta)}{\partial \xi} + \frac{\partial(\rho r V v_\theta)}{\partial \eta} = & -J \rho v_r v_\theta + r J F_\theta + \\ & \frac{1}{Re} \left[\frac{\partial}{\partial \xi_j} \left(\mu r J g_{jk} \frac{\partial v_\theta}{\partial \xi_k} \right) - \frac{v_\theta}{r} \left(J \frac{\partial \xi_j}{\partial r} \right) \frac{\partial(r \mu)}{\partial \xi_j} \right] \end{aligned} \quad (3.27)$$

Although the governing equations are transformed, the quantity which is conserved is not altered. Eqs. 3.25-27 contain the cylindrical coordinate velocity components (v_z , v_r , and v_θ) and represent conservation of momentum in the axial, radial, and tangential directions, respectively. It is possible to combine the momentum relationships to yield formulations which conserve momentum in the ξ and η directions [18], but this makes the solution algorithm much more complicated due to the fact that the mapping is carried out on a local scale and the directions of (ξ, η) with respect to (z, r) are constantly changing. The following chapter describes the numerical method used for solving Eqs. 3.24-27.

Chapter 4

Numerical Solution Method

The only type of general equation, or system of equations, that can be solved is a linear equation, or set thereof. However, the Navier-Stokes equations are nonlinear as a result of the convection terms. Thus, continuous analytic solutions to the closed system cannot, in general, be found. To get the equations into a solvable form, they are discretized via a linearization process, and solved at distinct points, called grid nodes. An iterative solution is employed to make the solution of the linear discretized equations reflect the nonlinearity of the system of equations they are describing.

Although discretization is a fundamental process in the iterative solution scheme, the particular steps in the discretization process are not determined by a given solution algorithm. Rather, the solution algorithm is dependent upon the discretization process. The Navier-Stokes equations were derived in terms of conserving a quantity within an infinitesimally small volume. Thus, it is considered that the control volume approach is more closely tied to the physics of the flow than is the finite difference approach.

For incompressible, steady-state, axisymmetric flow there are four unknown flow field variables (v_z, v_r, v_θ , and p) and four conservation equations, making the problem well posed. The three momentum equations are used to solve for the velocities, and the pressure field is extracted indirectly from the continuity equation. A fifth variable, the effective viscosity, μ , must be evaluated in turbulent flows. Chapter 5 provides an explanation of the turbulence model used to determine this quantity. If the flow is laminar, μ is simply the molecular fluid viscosity, which is a fluid property and is known independent of the flow. The solution algorithm is independent of

the means for determining the spatially varying turbulent viscosity; thus, any turbulence model may be used to approximate the effective viscosity.

In addition to the complexity associated with the nonlinearity of the momentum equations, the appearance of the first derivatives of pressure in the relations causes additional numerical difficulties. If central differencing is used to discretize the pressure gradient on a uniform grid, the pressure at the central node becomes decoupled from the approximation to the momentum equations. Referring to the uniform control volumes illustrated in Fig. 4.1, the gradients at the central node can be expressed, using second-order accurate central differencing, as:

$$\left. \frac{\partial p}{\partial \xi} \right|_O = \frac{p_E - p_W}{2\Delta\xi} \quad (4.1a)$$

$$\left. \frac{\partial p}{\partial \eta} \right|_O = \frac{p_N - p_S}{2\Delta\eta} \quad (4.1b)$$

These relationships are referred to as “2- δ ” gradients, since the component terms span two control volume widths. Because p_O does not appear in Eq. 4.1, the pressure field experiences “even-odd” decoupling, wherein a physically unrealistic checkerboard or oscillatory field can satisfy the discretized governing equations [16, pp. 115-118]. The problem stems from the need to use central differencing to evaluate the pressure gradients. One-sided differencing does not accurately represent the elliptic nature of pressure, thus it is not a viable approach for eliminating the even-odd decoupling. On a nonuniform grid the central pressure would appear in the central difference approximations to the gradients. However, the coupling this provides is very weak and only gets stronger as the grid becomes less uniform.

Staggered grids are frequently employed to alleviate the even-odd decoupling and prevent an oscillatory pressure field. Scalar and vector component fields are solved on separate grids in this type of treatment. The vector component nodes are offset in the direction of the component by one half of a control volume width from the scalar nodes. Patankar and Spalding's SIMPLE [19], Patankar's SIMPLER [20], and Issa's PISO [21] are a few of the algorithms that employ the staggered grid approach. On a nonorthogonal grid, the gridlines are not along the coordinate axes,

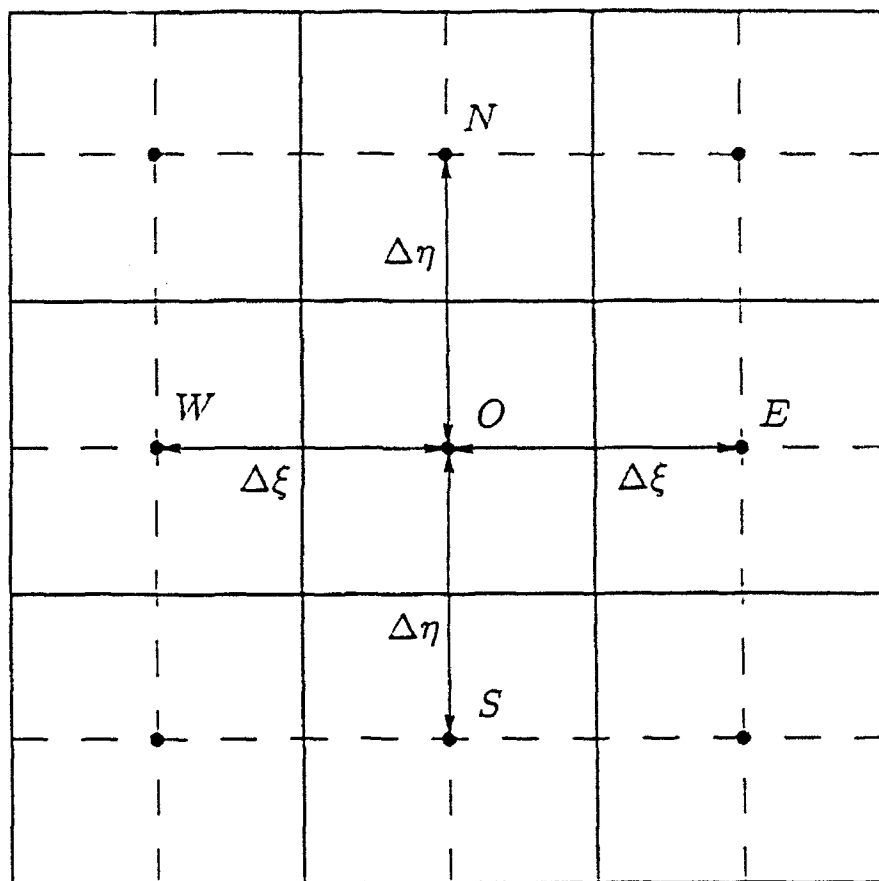


Figure 4.1
Control Volumes in Computational Space

so the nodes cannot be staggered in the coordinate directions. Therefore, staggered grids cannot efficiently prevent pressure oscillations on nonorthogonal grids [22]. In order to use staggered grids with curvilinear coordinates, Karki and Patankar [18] had to transform the governing equations to reflect conservation in the direction of the computational plane axes. This approach complicates matters without providing an exonerating computational advantage.

Rhie and Chow [11] proposed an algorithm which prevents the occurrence of an oscillatory pressure field on a nonstaggered grid. Their algorithm, dubbed "the pressure-weighted interpolation method (PWIM)" by later researchers Miller and Schmidt [23], combines " $2 - \delta$ " and " $1 - \delta$ " central differencing approximations to the pressure gradients to eliminate oscillations in the pressure field. Like SIMPLE, PWIM is a predictor-corrector scheme, in which the continuity equation is used to derive a pressure correction equation. The pressure corrector is used both to update the pressure and to correct the velocity components to satisfy continuity. When convergence is reached with this iterative solution scheme, both momentum and mass are simultaneously conserved.

The reasoning behind Rhie and Chow's development will be discussed later, but the incentive for creating this method is to eliminate the complexity and inefficiency of staggered grids. Computational fluid dynamics techniques such as curvilinear coordinates and multilevel algorithms, which have been actively researched in the years following the initial proposal of the staggered approach, are cumbersome when applied to noncollocated grids. Nonstaggered grid solvers generally require less storage than their staggered counterparts. Peric *et al.* [24] found that PWIM was usually faster than staggered grid methods, even without the SIMPLEC enhancement (see Section 4.4). Miller and Schmidt [23] demonstrated that the nonstaggered grid solver is more accurate than staggered grid algorithms. Also, the accuracy of solutions obtained using the PWIM algorithm are not reduced by interpolation errors during post-processing, since the flow quantities are already defined at the same points.

The PWIM algorithm exhibits a robustness which is further enhanced by the transformation to generalized curvilinear coordinates and the use of the finite control volume formulation to discretize the equations. The use of generalized coordinates allows the same computer code to treat the vastly different geometries that appear in various types of turbomachines and fluid passages. A control volume approach, rather than finite difference representation of the governing equations, allows the discretized form of the equations to remain conservative.

Rhie and Chow's nonstaggered solution algorithm, PWIM, was chosen as the technique best suited to solving the incompressible turbomachinery flows that are the subject of this work. However, to optimize the flow solver to the particular applications, some changes have been made. Most of them are based on the difference between cylindrical and Cartesian coordinate systems. Since the initial publication of the PWIM algorithm in 1983, several researchers have proposed minor modifications. Lapworth [22] has formulated a scheme in which pressure, rather than a pressure correction, is computed. This variation is algebraically equivalent to the original PWIM, where an error in continuity still drives the convergence of the pressure field. This approach was not incorporated into the present scheme, because a change in pressure also enables corrections to the nodal velocities to be made. Lapworth does not bother making updates to the nodal quantities, since continuity is assured by updating only the facial convective velocities. However, the minor computational expense incurred by updating nodal velocities according to the pressure correction relationships yields a more robust algorithm. It is especially helpful in preventing divergence early in the iterative process when the solution is rapidly evolving from the initial guess towards a converged solution. By updating the nodal velocities, the explicit source terms in the momentum equations are evaluated using more realistic quantities, and thus stability is enhanced. Rhie and Chow [11] use linear interpolation in the physical plane to arrive at facial values of the contravariant velocities. Lapworth [22] used simple averaging in the computational domain. Miller and Schmidt [23] developed a more accurate, albeit costlier, approach that is based on conservation of momentum. Their approach, which degenerates to simple averaging for

some flow conditions, is adopted here, because it more accurately represents the physics of the flow.

The remainder of this chapter is devoted to giving a detailed explanation of how the modified PWIM algorithm is used to solve the discretized axisymmetric equations in generalized curvilinear coordinates. The description begins with a statement of the nomenclature associated with the grid and control volumes. This is followed by a discussion of how the governing equations are discretized. To simplify the presentation of concepts which are pertinent to all three momentum equations, the discretization is first carried out in terms of a general transport equation. Then, the individual momentum equations are treated. As previously mentioned the discretization process is not unique to PWIM. After it has been presented, the way in which PWIM is specifically used to solve the discretized equations is discussed.

4.1 Control Volume and Grid Definition

Fig. 4.2 shows a typical control volume in physical space. Fig 4.3 illustrates the same control volume when it is mapped into a unit square in the computational domain. Each of these figures indicates the notation utilized throughout this work. The directions east, west, north, and south correspond to various directions with respect to the central (O) node. An uppercase directional subscript (E, W, N, S) refers to a quantity at a neighboring node. A lowercase directional subscript (e, w, n, s) corresponds to a quantity evaluated on a control volume face. Repeated directional subscripts indicate quantities in the direction of the latter subscript with respect to that indicated by the first.

An entire grid in the physical plane is shown in Fig. 4.4. When the grid is generated, only the coordinates of the control volume corners are computed. Vertical and horizontal face coordinates (denoted in Fig. 4.3 by $+$ and $\times$, respectively) are positioned at the midpoints

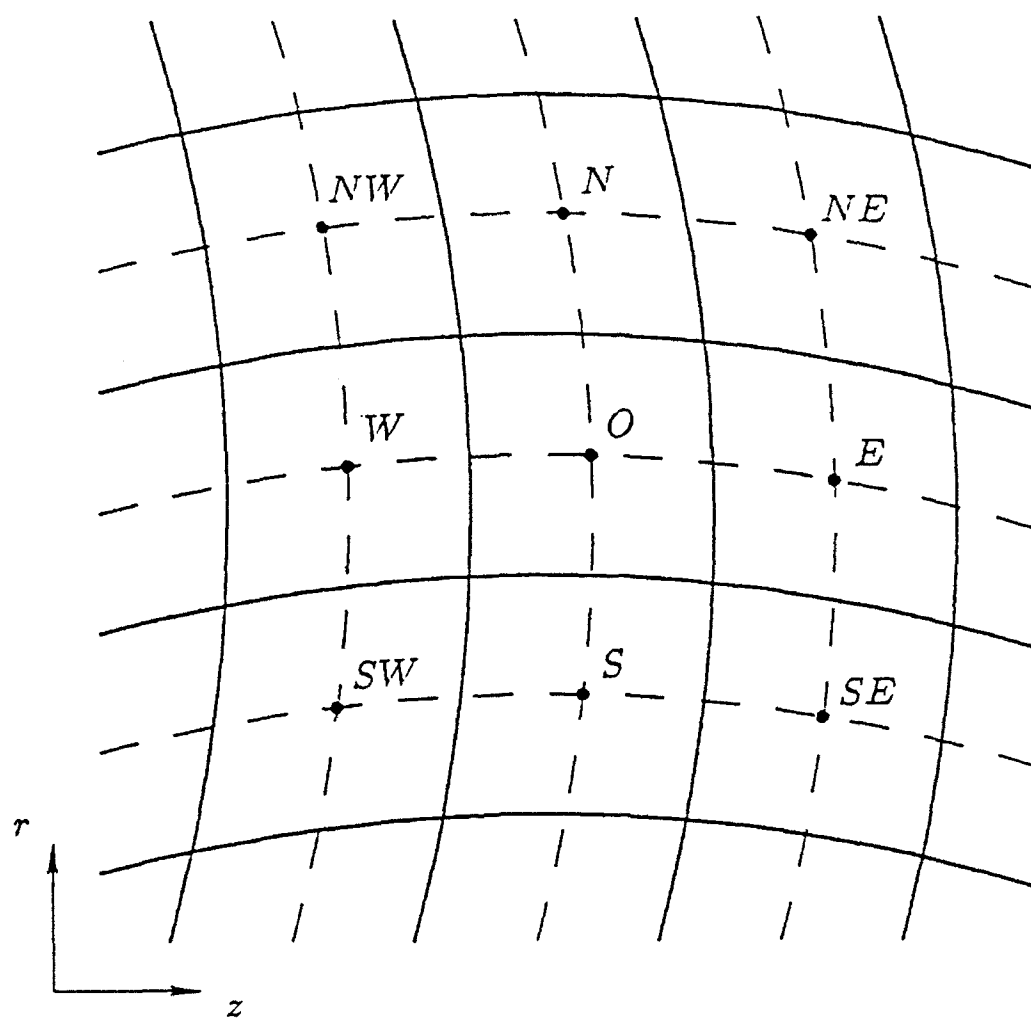


Figure 4.2

Control Volumes in Physical Space

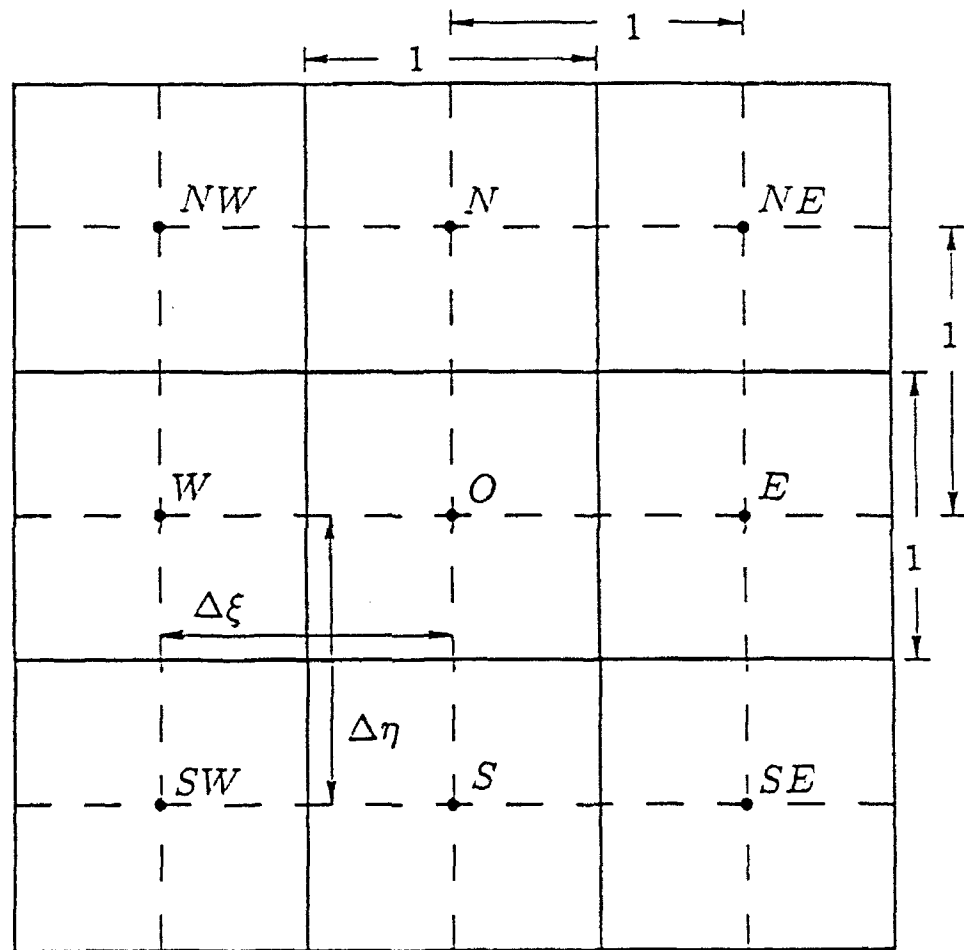


Figure 4.3

Control Volumes Mapped into Unit Squares in Computational Space

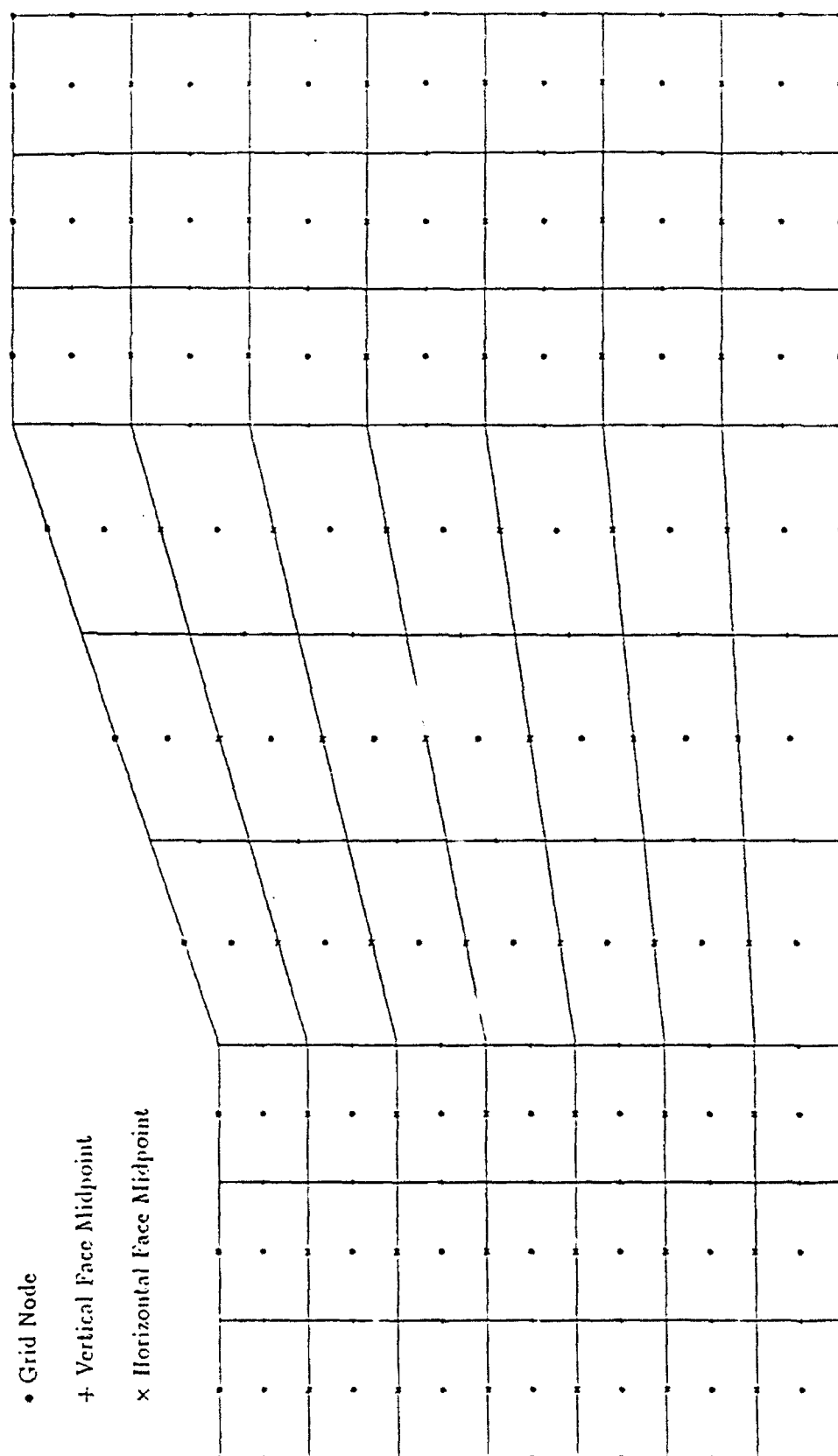


Figure 4.4
Grid in Physical Space

between adjacent corners. Nodes, indicated by circles in Fig. 4.4, are located at the intersection of the line segment connecting the vertical face coordinates with the line segment connecting the horizontal face coordinates. Additional nodes are placed on the grid boundaries for convenience in implementing the boundary conditions. The size of the grid in Fig 4.3 is 11×9 . The number of nodes in the ξ direction, NNI , is equal to 11. The number of nodes in the η direction, NNJ , is equal to 9. Nodes with indices $i = 2$ or $NNI - 1$ or $j = 2$ or $NNJ - 1$ have control volume faces which lie on the boundary of the computational domain. These points are referred to as near boundary nodes. By placing additional nodes along the boundary, iterative solvers can treat the near boundary nodes without a change in programming logic.

The geometric relationships between the nodes and control volumes corners are described, because the discretization process requires that assumptions be made concerning the flow field distribution between a node and its neighbors.

4.2 Discretizing the General Transport Equation

For two-dimensional control volume formulations, the transport equations are integrated over the control volume area. In the computational plane this corresponds to integrating over $dA = d\xi d\eta$. All of the conservation equations can be expressed in the form of a single general transport equation:

$$\frac{\partial(\rho r U_j \phi)}{\partial \xi_j} = \frac{\partial}{\partial \xi_j} \left(\Gamma r J g_{jk} \frac{\partial \phi}{\partial \xi_k} \right) + S(\xi, \eta) \quad (4.2)$$

The source term, S , contains all terms not in the form of either the convection or diffusion terms.

The differential form of the governing equation (Eq. 4.2) is integrated by assuming that quantities are constant over control volume faces.

$$(\rho r U \phi \Delta \eta)_w^e + (\rho r V \phi \Delta \xi)_s^n - (\Gamma r J g_{11} \phi_\xi \Delta \eta)_w^e - (\Gamma r J g_{22} \phi_\eta \Delta \xi)_s^n = \bar{S} \quad (4.3a)$$

$$\tilde{S} = (\Gamma r J g_{12} \phi_\eta \Delta \eta)_w^\epsilon + (\Gamma r J g_{21} \phi_\xi \Delta \xi)_e^\eta + \int_e^n \int_w^\epsilon S(\xi, \eta) d\xi d\eta \quad (4.3b)$$

Although the nodal separations, $\Delta \xi$ and $\Delta \eta$, are equal to unity, they are explicitly retained in the formulations to clarify the nature of the discretized equations. For convenience, the cross-diffusion terms are absorbed into the source term on the right hand side of the equation. The standard diffusion terms were brought over to the left hand side of Eq. 4.3a. The diffusive and convective flux can then be combined into a total flux, F , as shown in Eq. 4.4.

$$(F_e^\xi - F_w^\xi) + (F_n^\eta - F_s^\eta) = \tilde{S} \quad (4.4)$$

$$F_e^\xi = (\eta \rho r U \phi)_e - (\eta \Gamma r J g_{11} \phi_\xi)_e \quad (4.5a)$$

$$F_n^\eta = (\xi \rho r V \phi)_n - (\xi \Gamma r J g_{22} \phi_\eta)_n \quad (4.5b)$$

A specification of the total flux across the western and southern face is not given in Eq. 4.5. Due to geometric similarity there are often similar expressions which relate the eastern and western, or northern and southern, or eastern and northern, or western and southern facial quantities. Throughout this work, whenever only two relationships are formally stated, those for the remaining edges are found by evaluating the quantity using the relationship corresponding to the face of geometric similarity.

The total flux terms in Eq. 4.5 are composed of a linear combination of convective and diffusive fluxes. This form is desirable, because it allows for the solution of the transported quantity despite the presence of multiplicative factors, such as U or V , which might be a function of the transported quantity itself (this is the case for the axial and radial momentum equations, where U and V are dependent on v_z and v_r). Eq. 4.5 is a solvable linear equation in ϕ . Geometric quantities appearing in the flux terms are fixed, known functions of the grid geometry. Fluid properties are considered to be known while the transport equation is being solved. Contravariant velocities at the control volume face are known, as these quantities are used to satisfy continuity

on the nonstaggered grid (this will be further discussed in Section 4.5). Any quantities that may depend on the transported quantity are considered fixed while a particular transport equation is being solved. When the iterative process converges, all of the coupling between ϕ and the other variables is incorporated into the multiplicative factors and source terms. The remainder of this section explains how Eq. 4.5 is solved for ϕ .

The diffusive gradients are expressed with second-order accuracy on the uniform grid according to the relationships shown in Eq. 4.6.

$$\phi_{\xi}|_e = (\phi_E - \phi_O)/\Delta\xi \quad (4.6a)$$

$$\phi_{\eta}|_n = (\phi_N - \phi_O)/\Delta\eta \quad (4.6b)$$

Evaluating the convective terms is a more difficult task, because it requires the knowledge of the transported quantity at the control volume face (ϕ_e, ϕ_n). The transported quantity, however, is only defined at the nodes of the nonstaggered grid. A variety of ways of using the nodal quantities to approximate the facial value are discussed in the Appendix. The method used to express the total flux in terms of nodal quantities is known as a differencing scheme or interpolation scheme. Within the Appendix three new differencing schemes that were developed during the course of this investigation are presented. One involves a revision to Leonard's QUICK differencing scheme [12], and the other two were formulated to reflect the inherent differences between flows in cylindrical and Cartesian coordinate systems. The reader can refer to the Appendix for a complete discussion of these differencing schemes. For now, it is sufficient to say that they determine how the transported quantities at the nodes are combined to give an expression for the total flux at the control volume faces. The differencing schemes define the linear function in Eq. 4.7.

$$F_e^{\xi} = f(\phi_O, \phi_E, \dots) \quad (4.7)$$

Linear equations for the total flux for each of the terms in Eq. 4.4 are combined to yield Eq. 4.8.

$$(A_P - S_P)\phi_O = A_E\phi_E + A_W\phi_W + A_N\phi_N + A_S\phi_S + S_U \quad (4.8)$$

The particular differencing scheme determines the values of the coefficients, A_i , multiplying the nodal values. These coefficients contain the geometric, fluid property, and nonlinear factors influencing the values of ϕ at the nodes. S_U contains all of the explicitly evaluated source term integrals, while S_P corresponds to the dependent or implicit sources. Iterative solvers diverge if the system is not positive definite, meaning the coefficients do not all have the same sign. Therefore, care must be taken to make S_P negative. For stability purposes, Patankar [16, p. 49] suggests using the following relationships to break up a source term that is a simple function of the transported quantity. If $\tilde{S}$ is not a function of ϕ , then the entire source term is evaluated explicitly.

$$S_P = \frac{\partial \tilde{S}}{\partial \phi} \quad (4.9a)$$

$$S_U = \tilde{S}(\phi) - \frac{\partial \tilde{S}}{\partial \phi} \phi \quad (4.9b)$$

The five-point stencil format of Eq. 4.8 is chosen so that a variety of efficient iterative solvers may be applied to the system. It is a linear system which results in a banded matrix that can be efficiently stored and can also be solved using noniterative techniques. Gauss-Seidel or Jacobi iterative schemes are suitable for systems in which the source term dominates. Because convection transmits information in the direction of flow, the momentum equations are solved more quickly by using a tridiagonal solver to solve the field column by column or row by row sweeping from the inlet of the geometry to the outlet. The pentadiagonal system is turned into a tridiagonal system by treating the eastern and western influences explicitly when viewing a column, or by handling the northern and southern influences explicitly when isolating a row. For stiff equations, such as the pressure correction equation which will be discussed in Section 4.5, Stone's strongly implicit solver [25] can be used. The convergence rate of Stone's method does not decay as quickly as other iterative schemes as a converged solution is approached, because it treats the equations in a more implicit manner.

Setting $A_O = A_P - S_P$ give the general format of the linear system:

$$A_{Oo} \phi_O^{k+1} = \sum_O A_i \phi_i^{k+1} + \tilde{S}^n \quad (4.10)$$

The superscript on the transported quantity corresponds to the iteration or time level. The most recent, or current, level is indicated by a " $k+1$ " superscript. The last point in the iterative process where all the remaining quantities have been updated is denoted by an " n " superscript. Implicit treatment is indicated by " $k+1$." Explicit evaluation is shown by " n ." The subscript under the summation sign denotes a sum over the eastern, western, northern, and southern neighbors with respect to the central node.

The nonlinearity of the momentum equations may cause the iterative solution procedures to diverge. Therefore, under-relaxation is used to stabilize the solution. The relaxation factor ω , where $0 \leq \omega \leq 1$, specifies a fractional weighting between the new and old values, as seen in Eq. 4.11a. Eq. 4.11b shows the under-relaxed linearized equation.

$$\phi_O^{k+1} = \omega \left(\frac{\sum_O A_i \phi_i^{k+1} + \tilde{S}^n}{A_{Oo}} \right) + (1 - \omega) \phi_O^n \quad (4.11a)$$

$$\frac{1}{\omega} A_{Oo} \phi_O^{k+1} = \sum_O A_i \phi_i^{k+1} + \left[\tilde{S}^n + \frac{1}{\omega} (1 - \omega) A_{Oo} \phi_O^n \right] \quad (4.11b)$$

Now that the discretization process for the general transport equation has been discussed, the individual discretized momentum equations can be obtained by substituting the velocity components for ϕ and introducing the relevant source term.

4.3 Discretizing the Momentum Equations

The discretized momentum equations are found by integrating the transformed axisymmetric incompressible steady-state momentum equations (Eqs. 3.25-27) over the control volume according to the method outlined in Section 4.2. The multiplicative coefficients, A_i , are the same

for all of the momentum equations, because they are a function of the control volume geometry, the mass flow rate into the CV, and the fluid viscosity. All three of these properties are the same for each of the momentum equations. The under-relaxed momentum equations are expressed in standard form in Eqs. 4.12-17. All of the pressure gradients are evaluated using known pressures. Every viscous term not matching the form shown in Eq. 4.5 is evaluated explicitly as part of the source term. The centripetal and body forces are evaluated explicitly so A_O remains the same for each of the three momentum equations. The terms S^z , S^r , and S^θ are the integrals of source term with the pressure gradients extracted.

$$\frac{1}{2} A_{O_O} v_{z_O}^{k+1} = \sum_O A_i v_{z_i}^{k+1} - \Delta \xi \Delta \eta r_O ((r_\eta p_\xi)_O^n - (r_\xi p_\eta)_O^n) + [S_O^z + \frac{1}{2}(1-\omega) A_{O_O} v_{z_O}^n] \quad (4.12)$$

$$\frac{1}{2} A_{O_O} v_{r_O}^{k+1} = \sum_O A_i v_{r_i}^{k+1} - \Delta \xi \Delta \eta r_O ((z_\xi p_\eta)_O^n - (z_\eta p_\xi)_O^n) + [S_O^r + \frac{1}{2}(1-\omega) A_{O_O} v_{r_O}^n] \quad (4.13)$$

$$\frac{1}{2} A_{O_O} v_{\theta_O}^{k+1} = \sum_O A_i v_{\theta_i}^{k+1} + [S_O^\theta + \frac{1}{2}(1-\omega) A_{O_O} v_{\theta_O}^n] \quad (4.14)$$

$$\begin{aligned} S_O^z = & (r J F_z)_O \Delta \xi \Delta \eta + (\mu r J g_{12} v_{z_\eta} \Delta \eta)_w^e + (\mu r J g_{21} v_{z_\xi} \Delta \xi)_s^n + \\ & [\Delta \eta r \mu \frac{1}{2} (r_\eta (r_\eta v_{z_\xi} - r_\xi v_{z_\eta}) - z_\eta (r_\eta v_{r_\xi} - r_\xi v_{r_\eta}))]_w^e + \\ & [\Delta \xi r \mu \frac{1}{2} (-r_\xi (r_\eta v_{z_\xi} - r_\xi v_{z_\eta}) + z_\xi (r_\eta v_{r_\xi} - r_\xi v_{r_\eta}))]_s^n \end{aligned} \quad (4.15)$$

$$\begin{aligned} S_O^r = & (J \rho v_\theta^2)_O \Delta \xi \Delta \eta + (r J F_r)_O \Delta \xi \Delta \eta + \\ & (\mu r J g_{12} v_{r_\eta} \Delta \eta)_w^e + (\mu r J g_{21} v_{r_\xi} \Delta \xi)_s^n + (J 2 \mu v_r / r)_O \xi \eta + \\ & [\Delta \eta r \mu \frac{1}{2} (r_\eta (-z_\eta v_{z_\xi} + z_\xi v_{z_\eta}) - z_\eta (-z_\eta v_{r_\xi} + z_\xi v_{r_\eta}))]_w^e + \\ & (\Delta \xi r \mu \frac{1}{2} (-r_\xi (-z_\eta v_{z_\xi} + r_\xi v_{z_\eta}) + z_\xi (-z_\eta v_{r_\xi} + r_\xi v_{r_\eta}))]_s^n \end{aligned} \quad (4.16)$$

$$\begin{aligned} S_O^\theta = & -(J \rho v_r v_\theta)_O \Delta \xi \Delta \eta + (r J F_\theta)_O \Delta \xi \Delta \eta + \\ & (\mu r J g_{12} v_{\theta_\eta} \Delta \eta)_w^e + (\mu r J g_{21} v_{\theta_\xi} \Delta \xi)_s^n - [\frac{1}{2} v_\theta \Delta \xi \Delta \eta (-z_\eta (\mu r)_\xi + z_\xi (\mu r)_\eta)]_O \end{aligned} \quad (4.17)$$

Eqs. 4.12-14 each constitute a system of linear equations, where each CV contributes an equation to the system. The solution of each set of equations is referred to as an inner loop, because it is assumed that all quantities but the relevant one are fixed. In contrast, the outer loop iteration is comprised of all the steps that are taken to update each and every flow field. The remainder of this section outlines the steps comprising PWIM, listing the order in which the flow fields are updated within a particular outer loop iteration. This list is not detailed. Its purpose is to initiate the reader to the overall approach used to solve the governing equations.

1. Begin with an initial estimate for the flow fields.
2. Solve the momentum equations using the known values to construct the multiplicative coefficients and the source terms.
3. Calculate new values for the facial contravariant velocities (discussed in Section 4.4).
4. Use the failure of the velocities from Step 3 to satisfy to continuity to arrive at a correction for the pressure (discussed in Section 4.5).
5. Use the pressure correction from Step 4 to correct the facial contravariant velocities so they satisfy continuity, and to correct the nodal velocities (discussed in Section 4.5).
6. Return to Step 2 until convergence is reached.

Solving the momentum equations does not lead to the correct velocity distributions unless the correct U , V , and p fields are known. Iterating between Steps 2 and 6, referred to as cycling over the outer loop, drives the solution toward convergence. As a better pressure field is returned from Step 4, the successive velocities in Step 2 will violate continuity by a smaller degree. This continues until eventually all of the governing equations are satisfied.

4.4 Pressure-Weighted Interpolation of Facial Contravariant Velocities

It is evident from the equations in the previous sections that the values of (U, V) are needed only at the control volume faces. This section is devoted to explaining how these quantities

are evaluated. Each of these contravariant velocities is a function of v_z and v_r on the CV face. However, the cylindrical velocity components are known only at the nodes. The momentum equations are used to arrive at accurate expression for the nodal velocities, so Miller and Schmidt [23] extend this idea and state that an accurate expression for the facial velocity can be determined by formulating the momentum equation about the control volume face. The axial momentum equation expressed at the nodes surrounding the eastern face and at the face itself are shown in Eqs. 4.18-20.

$$\frac{1}{2} A_{O_o} v_{z_o}^{k+1} = \sum_o A_i v_{z_i}^{k+1} - \Delta\xi \Delta\eta r_o ((r_\eta p_\xi)_o^n - (r_\xi p_\eta)_o^n) + [S_o^z + \frac{1}{2}(1-\omega) A_{O_o} v_{z_o}^n] \quad (4.18)$$

$$\frac{1}{2} A_{O_E} v_{z_E}^{k+1} = \sum_E A_i v_{z_i}^{k+1} - \Delta\xi \Delta\eta r_E ((r_\eta p_\xi)_E^n - (r_\xi p_\eta)_E^n) + [S_E^z + \frac{1}{2}(1-\omega) A_{O_E} v_{z_E}^n] \quad (4.19)$$

$$\frac{1}{2} A_{O_e} v_{z_e}^{k+1} = \sum_e A_i v_{z_i}^{k+1} - \Delta\xi \Delta\eta r_e ((r_\eta p_\xi)_e^n - (r_\xi p_\eta)_e^n) + [S_e^z + \frac{1}{2}(1-\omega) A_{O_e} v_{z_e}^n] \quad (4.20)$$

Two types of gradients are present in these equations; $p_\xi|_e^n$ is a "1 - δ " gradient and $p_\xi|_{O,E}^n$ are "2 - δ " gradients. The appearance of both types of gradients is the key to the success of PWIM. Together, they are used to eliminate the occurrence of oscillating pressure fields that can appear on nonstaggered grids.

Eq. 4.20 shows the momentum equation expressed about a cell face, and it contains coefficients calculated at the face. These cell face coefficients are not known. Miller and Schmidt suggest using the following relationships to approximate their values.

$$\frac{\sum_e A_i v_{z_i}^{k+1}}{A_{O_e}} = \frac{1}{2} \left(\frac{1 \sum_o A_i v_{z_i}^{k+1}}{A_{O_o}} + \frac{1 \sum_E A_i v_{z_i}^{k+1}}{A_{O_E}} \right) \quad (4.21a)$$

$$\frac{1}{A_{O_e}} = \frac{1}{2} \left(\frac{1}{A_{O_o}} + \frac{1}{A_{O_E}} \right) \quad (4.21b)$$

In the cylindrical coordinate system, surface area is directly proportional to the radius, so as the radius increases, velocity must decrease in order to conserve mass. Miller and Schmidt's approximation reverts to the assumption that each nodal velocity equally contributes to the value at the face. However, as stated in the Appendix during the discussion of the differencing schemes for cylindrical flow applications, a better assumption for axisymmetric flows is to use

radial weighting to determine the relative contribution of the nodal values to the facial quantities. Thus, in the current cylindrical adaptation of the PWIM algorithm, the following approximations are made:

$$\frac{r_e \sum_e A_i v_{z_i}^{k+1}}{A_{O_e}} = \frac{1}{2} \left(\frac{r_O \sum_O A_i v_{z_i}^{k+1}}{A_{O_O}} + \frac{r_E \sum_E A_i v_{z_i}^{k+1}}{A_{O_E}} \right) \quad (4.22a)$$

$$\frac{r_e}{A_{O_e}} = \frac{1}{2} \left(\frac{r_O}{A_{O_O}} + \frac{r_E}{A_{O_E}} \right) \quad (4.22b)$$

Substituting these approximations into Eqs. 4.18-20 results in the following expression for v_{z_e} :

$$\begin{aligned} v_{z_e}^{k+1} = & (1 - \omega) v_{z_e}^n + ((r_O v_{z_O}^{k+1} + r_E v_{z_E}^{k+1}) - (1 - \omega)(r_O v_{z_O}^n + r_E v_{z_E}^n)) / (2r_e) + \\ & \frac{\omega \Delta \xi \Delta \eta}{2r_e} \left(\frac{r_O}{A_{O_O}} r_O ((r_\eta p_\xi)_O^n - (r_\xi p_\eta)_O^n) + \frac{r_E}{A_{O_E}} r_E ((r_\eta p_\xi)_E^n - (r_\xi p_\eta)_E^n) \right) - \\ & \omega \Delta \xi \Delta \eta \frac{r_e}{A_{O_e}} ((r_\eta p_\xi)_e^n - (r_\xi p_\eta)_e^n) + \\ & \frac{\omega}{A_{O_e}} S_e^z - \frac{\omega}{2r_e} \left(\frac{r_O}{A_{O_O}} S_O^z + \frac{r_E}{A_{O_E}} S_E^z \right) \end{aligned} \quad (4.23)$$

The radial momentum equation at the eastern face is handled in a similar manner.

$$\begin{aligned} v_{r_e}^{k+1} = & (1 - \omega) v_{r_e}^n + ((r_O v_{r_O}^{k+1} + r_E v_{r_E}^{k+1}) - (1 - \omega)(r_O v_{r_O}^n + r_E v_{r_E}^n)) / (2r_e) + \\ & \frac{\omega \Delta \xi \Delta \eta}{2r_e} \left(\frac{r_O}{A_{O_O}} r_O ((z_\xi p_\eta)_O^n - (z_\eta p_\xi)_O^n) + \frac{r_E}{A_{O_E}} r_E ((z_\xi p_\eta)_E^n - (z_\eta p_\xi)_E^n) \right) - \\ & \omega \Delta \xi \Delta \eta \frac{r_e}{A_{O_e}} ((z_\xi p_\eta)_e^n - (z_\eta p_\xi)_e^n) + \\ & \frac{\omega}{A_{O_e}} S_e^r - \frac{\omega}{2r_e} \left(\frac{r_O}{A_{O_O}} S_O^r + \frac{r_E}{A_{O_E}} S_E^r \right) \end{aligned} \quad (4.24)$$

The other faces are treated in a similar manner.

Substituting Eqs. 4.23 and 4.24 into Eq. 3.15 yields the expression for the contravariant velocities in terms of facial and neighboring nodal values. Making the approximation that the facial coefficients (Eq. 4.22) are a function of radius results in an expression for the contravariant velocities that is consistent with the cylindrical convection approximation that is outlined in the Appendix.

$$\begin{aligned}
U_e = & (1 - \omega)U_e^n - (1 - \omega) (r_{\eta_e}(r_O v_{i_O}^n + r_E v_{i_E}^n) - z_{\eta_e}(r_O v_{r_O}^n + r_E v_{r_E}^n)) / (2r_e) + \\
& (r_{\eta_e}(r_O v_{i_O}^{k+1} + r_E v_{i_E}^{k+1}) - z_{\eta_e}(r_E v_{r_O}^{k+1} + r_E v_{r_E}^{k+1})) / (2r_e) - \\
& \omega \Delta \xi \Delta \eta \frac{r_e}{A_{O_e}} ((z_{\eta_e}^2 + r_{\eta_e}^2) p_{\xi}|_e^n - (r_{\eta_e} r_{\xi_e} + z_{\eta_e} z_{\xi_e}) p_{\eta}|_e^n) + \\
& \omega \Delta \xi \Delta \eta \frac{r_O}{2r_e} \frac{r_O}{A_{O_O}} ((z_{\eta_e} z_{\eta_O} + r_{\eta_e} r_{\eta_O}) p_{\xi}|_O^n - (r_{\eta_e} r_{\xi_O} + z_{\eta_e} z_{\xi_O}) p_{\eta}|_O^n) + \\
& \omega \Delta \xi \Delta \eta \frac{r_E}{2r_e} \frac{r_E}{A_{O_E}} ((z_{\eta_e} z_{\eta_E} + r_{\eta_e} r_{\eta_E}) p_{\xi}|_E^n - (r_{\eta_e} r_{\xi_E} + z_{\eta_e} z_{\xi_E}) p_{\eta}|_E^n) + \\
& \underline{\frac{\omega}{A_{O_e}} (r_{\eta_e} S_e^z - z_{\eta_e} S_e^r) -} \\
& \underline{\frac{\omega}{2r_e} \left(\frac{r_O}{A_{O_O}} (r_{\eta_e} S_O^z - z_{\eta_e} S_O^r) + \frac{r_E}{A_{O_E}} (r_{\eta_e} S_E^z - z_{\eta_e} S_E^r) \right)} \quad (4.25)
\end{aligned}$$

$$\begin{aligned}
V_n = & (1 - \omega)V_n^n - (1 - \omega) (z_{\xi_n}(r_O v_{r_O}^n + r_N v_{r_N}^n) - r_{\xi_n}(r_O v_{i_O}^n + r_N v_{i_N}^n)) / (2r_n) + \\
& (z_{\xi_n}(r_O v_{r_O}^{k+1} + r_N v_{r_N}^{k+1}) - r_{\xi_n}(r_O v_{i_O}^{k+1} + r_N v_{i_N}^{k+1})) / (2r_n) - \\
& \omega \Delta \xi \Delta \eta \frac{r_n}{A_{O_n}} ((z_{\xi_n}^2 + r_{\xi_n}^2) p_{\eta}|_n^n - (z_{\xi_n} z_{\eta_n} + r_{\xi_n} r_{\eta_n}) p_{\xi}|_n^n) + \\
& \omega \Delta \xi \Delta \eta \frac{r_O}{2r_n} \frac{r_O}{A_{O_O}} ((z_{\xi_n} z_{\xi_O} + r_{\xi_n} r_{\xi_O}) p_{\eta}|_O^n - (z_{\xi_n} z_{\eta_O} + r_{\xi_n} r_{\eta_O}) p_{\xi}|_O^n) + \\
& \omega \Delta \xi \Delta \eta \frac{r_N}{2r_n} \frac{r_N}{A_{O_N}} ((z_{\xi_n} z_{\xi_N} + r_{\xi_n} r_{\xi_N}) p_{\eta}|_N^n - (z_{\xi_n} z_{\eta_N} + r_{\xi_n} r_{\eta_N}) p_{\xi}|_N^n) + \\
& \underline{\frac{\omega}{A_{O_n}} (z_{\xi_n} S_n^r - r_{\xi_n} S_n^z) -} \\
& \underline{\frac{\omega}{2r_n} \left(\frac{r_O}{A_{O_O}} (z_{\xi_n} S_O^r - r_{\xi_n} S_O^z) + \frac{r_N}{A_{O_N}} (z_{\xi_n} S_N^r - r_{\xi_n} S_N^z) \right)} \quad (4.26)
\end{aligned}$$

Because the facial source terms would require excessive computational effort to evaluate, the underlined terms are assumed to cancel. This approximation is equivalent to an assumption that the contribution of the facial source term to the contravariant velocity is equal to the average contribution of the nodal sources. Dropping only these terms is more accurate than the simple averaging used by Rhie and Chow [11] and Lapworth [22] to evaluate the facial velocities. There are both “1 - δ ” and “2 - δ ” pressure gradients forming the contravariant velocities. If

any oscillations arise in the pressure field, the difference between the gradients acts to remove them. As previously mentioned, the A_O coefficients are the same for both the axial and radial momentum equations. Therefore, they can be combined without having to distinguish them in the equations for the contravariant velocities.

Miller and Schmidt [23] have warned of the possibility that an incorrect development of the contravariant velocities can lead to a solution which is dependent upon the relaxation factor, ω . Dividing Eqs. 4.25 and 4.26 by ω and canceling terms which are equal at convergence shows that the formulations are independent of the relaxation factor.

4.5 Pressure Corrector Equation

The manner in which the velocities are determined has been explained, leaving the pressure as the only flow field variable for which a relationship must be derived. This section explains how the continuity equation is used to determine the pressure.

The discretized control volume formulation of the continuity equation is:

$$(\rho r U \Delta \eta)_w^e + (\rho r V \Delta \xi)_s^n = 0 \quad (4.27)$$

If the pressure field at a particular level during the solution procedure is incorrect, then U and V will not satisfy continuity. Therefore, a correction is needed. Corrections refer to changes in quantities between particular iteration levels. The quantities which are corrected are given in Eq. 4.28, and the levels to which the corrections correspond are indicated by the superscripts.

$$v_z^{n+1} = v_z^{k+1} + v'_z \quad (4.28a)$$

$$v_r^{n+1} = v_r^{k+1} + v'_r \quad (4.28b)$$

$$U^{n+1} = U^{k+1} + U' \quad (4.28c)$$

$$V^{n+1} = V^{k+1} + V' \quad (4.28d)$$

$$p^{n+1} = p^n + p' \quad (4.28e)$$

If the axial and radial momentum relations, Eqs. 4.12 and 4.13, are expressed at the corrected and uncorrected time levels and then subtracted, expressions for the corrections to v_{zO} and v_{rO} arise. The source terms completely cancel out of these relations, because they always involve quantities at time level n .

$$\frac{1}{2}A_{Oo}v'_{zO} = \sum_O A_i v'_z + \Delta\xi\Delta\eta r_O (-(r_\eta p'_\xi)_O + (r_\xi p'_\eta)_O) \quad (4.29)$$

$$\frac{1}{2}A_{Oo}v'_{rO} = \sum_O A_i v'_r + \Delta\xi\Delta\eta r_O (-(z_\xi p'_\eta)_O + (z_\eta p'_\xi)_O) \quad (4.30)$$

In the SIMPLE algorithm, the $\sum_O A_i v'_i$ terms are neglected. But, such a treatment requires under-relaxation of the pressure update, because the neglected terms make up a significant portion of the overall correction equations [16, p. 128]. Van Doormaal and Raithby [26] proposed the SIMPLER modification in an effort to improve the ability of p' to correct the pressure field without affecting its ability to satisfy the continuity equation. In this algorithm the magnitude of the neglected term is lessened by subtracting $\sum_O A_i v'_O$ from each side and then ignoring the resulting net summation over the neighboring nodes. The final step in formulating the velocity corrections is to make the following approximation:

$$A_{Oo} = \sum_O A_i + m_o \approx \sum_O A_i \quad (4.31a)$$

$$\frac{1}{2}A_{Oo} - \sum_O A_i = \frac{1}{2}(1 - \omega)A_{Oo} \quad (4.31b)$$

This approximation is not exact, even at convergence, when the cylindrical differencing schemes are used. However, this does not affect the final results. The velocity corrections go to zero at convergence, so it does not matter how the interim results are reached.

The corrections to the axial and radial velocity components are given in Eqs 4.32 and 33. The tangential velocity component, v_θ , does not enter into the axisymmetric continuity equation, so it requires no correction.

$$v'_{zO} = \frac{\omega\Delta\xi\Delta\eta}{(1-\omega)A_{Oo}} r_O (-(r_\eta p'_\xi)_O + (r_\xi p'_\eta)_O) \quad (4.32)$$

$$v'_{rO} = \frac{\omega\Delta\xi\Delta\eta}{(1-\omega)A_{Oo}} r_O (-(z_\xi p'_\eta)_O + (z_\eta p'_\xi)_O) \quad (4.33)$$

When the SIMPLEC modification is used, a fully implicit ($\omega = 1$) solution to the discrete linearized momentum equations cannot be made. With $\omega = 1$, the quantity that is subtracted from each side of Eqs. 4.54 and 4.55 is equal to the left hand side of the same equation. This makes the system homogeneous, and the resulting velocity correction field trivial.

Van Doormaal and Raithby report that SIMPLEC, compared to SIMPLE and SIMPLER, significantly decreases the amount of time required to solve the governing equations. The enhancement to PWIM proposed by Acharya and Moukalled [27] was not adopted, as the savings they reported were not as dramatic.

Expressions for the nodal velocity corrections have been derived, but it is the facial contravariant velocities which appear in the continuity equation. Equations for the correction of the cylindrical velocities at the faces are given in Eqs. 4.34-37. The relationships for these corrections are derived using the same logic that produced the corrections for the nodal velocities. Eq. 4.22b is used to approximate A_{O_e} and A_{O_n} .

$$v'_{z_e} = \frac{\omega \Delta \xi \Delta \eta}{(1 - \omega)} \frac{r_e}{A_{O_e}} (-(r_\eta p'_\xi)_e + (r_\xi p'_\eta)_e) \quad (4.34)$$

$$v'_{r_e} = \frac{\omega \Delta \xi \Delta \eta}{(1 - \omega)} \frac{r_e}{A_{O_e}} (-(z_\xi p'_\eta)_e + (z_\eta p'_\xi)_e) \quad (4.35)$$

$$v'_{z_n} = \frac{\omega \Delta \xi \Delta \eta}{(1 - \omega)} \frac{r_n}{A_{O_n}} (-(r_\eta p'_\xi)_n + (r_\xi p'_\eta)_n) \quad (4.36)$$

$$v'_{r_n} = \frac{\omega \Delta \xi \Delta \eta}{(1 - \omega)} \frac{r_n}{A_{O_n}} (-(z_\xi p'_\eta)_n + (z_\eta p'_\xi)_n) \quad (4.37)$$

Evaluating $p'_\eta|_e$ and $p'_\xi|_n$ is less convenient than evaluating $p'_\xi|_e$ and $p'_\eta|_n$ due to the nonstaggered nature of the grid. Therefore, the former terms are omitted from the correction equations. This omission has no effect on the converged solution, since the correction terms are zero at convergence.

Converting the cylindrical velocity components into (U, V) , via Eq. 3.15, gives the form of the contravariant velocity corrections used in the present scheme:

$$U'_e = -\frac{\omega \Delta \xi \Delta \eta}{(1 - \omega)} \frac{r_e}{A_{O_e}} (z_{\eta_e}^2 + r_{\eta_e}^2) p'_\xi|_e \quad (4.38)$$

$$V'_n = -\frac{\omega \Delta \xi \Delta \eta}{(1 - \omega)} \frac{r_n}{A_{O_n}} (z_{\xi_n}^2 + r_{\xi_n}^2) p'_\eta|_n \quad (4.39)$$

The pressure correction equation is formulated by substituting Eqs. 4.28c-e, 4.38 and 4.39 into the continuity equation.

$$A_O^p p_O' = A_E^p p_E' + A_W^p p_W' + A_N^p p_N' + A_S^p p_S' + S_U^p \quad (4.40)$$

$$A_E^p = \rho \Delta \xi \Delta \eta \left(\frac{\Delta \eta}{\Delta \xi} \right)_e \frac{\omega}{A_{O_e}} r_e^2 (z_{\eta_e}^2 + r_{\eta_e}^2) \quad (4.41a)$$

$$A_N^p = \rho \Delta \xi \Delta \eta \left(\frac{\Delta \xi}{\Delta \eta} \right)_n \frac{\omega}{A_{O_n}} r_n^2 (z_{\xi_n}^2 + r_{\xi_n}^2) \quad (4.41b)$$

$$A_O^p = A_E^p + A_W^p + A_N^p + A_S^p \quad (4.42)$$

$$S_U^p = -(1 - \omega) [(\Delta \eta \rho r U)_e - (\Delta \eta \rho r U)_w + (\Delta \xi \rho r V)_n - (\Delta \xi \rho r V)_s] \quad (4.43)$$

Continuity is satisfied at convergence, at which time the pressure correction equation becomes homogeneous and all corrections are zero.

The pressure correction equation yields a stiff linear system. Iterative solvers are slow to converge, making the solution difficult to obtain. Banded matrix solvers can be employed to find the solution, but their lower/upper decomposition has memory requirements on the order of $2 \times NN_I \times NN_J^2$, and the solution time for large grids is very long. Despite these problems, the flow solver will still converge if only a partially converged solution to the pressure corrector equation is found during the iterative process. A few sweeps of Stone's method gets the p' field going in the right direction, sending the velocities towards continuity-satisfying fields. This holds for all but the most drastic cases. If the momentum equations are utilizing an unrealistic pressure field, the resulting velocities may give large mass sources. Should these mass sources not be made sufficiently small, the perturbation to the velocity field may cause the solution to diverge. Flows in which centripetal forces come into play are especially susceptible to this phenomenon, because these terms are not lessened by a Reynolds number in the denominator nor are they differenced across a control volume. In such a situation, simply increasing the number of sweeps of the Stone solver will not necessarily alleviate the problem. This is because the late

of convergence decays as all of the high frequency errors are removed [28]. During this research it was found that turbomachinery flows, which contain centripetal acceleration effects, compounded by recirculating flow regions, and added instabilities caused by the sudden introduction of body forces in portions of the flow domain, required the use of a direct solver. This maintained realistic velocities throughout the iterative process and prevented the solution from diverging.

4.6 Boundary Conditions

Boundary conditions (BC's) make the solution to the governing equations unique, and one of the chief advantages to using body-fitted coordinates is that the BC's can be accurately enforced. Each type of equation has its own proper boundary conditions, as determined by the flow geometry and conditions. This section explains how the BC's are incorporated into the formulation of the discretized equations. Care is taken to ensure that the initial guess of the pressure and velocity fields satisfies the boundary conditions.

4.6.1 Momentum Boundary Conditions

The flux through the CV faces which lie on a boundary is evaluated by applying the boundary conditions. Thus, for near boundary nodes, the coefficient in the direction of the boundary is set equal to zero, and the differencing scheme contributions to the central coefficient come only from faces that do not lie on the boundary. The boundary conditions are enforced after the initial formulation of A_P and A_i by supplementing the central coefficient and source terms according to the particular boundary condition.

There are as many equations in the linear system as there are control volumes on the grid, but the number of unknowns present in the discretized stencils is equal to the number of nodes. The system becomes well posed by considering the boundary values to be known for a given outer loop iteration. Setting the coefficients that multiply them to zero removes the boundary nodes from the stencil, but this essentially alters the equation that is discretized. Modifying the source term and central coefficient returns the equation to proper form and allows it to reflect the proper BC.

There are four boundary conditions which arise in through-flow calculations: no-slip wall, plane of symmetry, parallel outflow, and specified velocity distribution. Each of the boundary condition (BC) modifications that follow are given for the eastern and western boundaries without loss of generality.

Two of the BC's are implemented in identical fashion. During any given outer loop iteration, both the transported quantity and convective flux are known on a no-slip wall and on a specified velocity boundary. Therefore, the convective portion of the flux is treated explicitly. The diffusive flux component is approximated using a second-order Taylor series expansion. This truncated series consists of three nodes; the central one is handled implicitly by moving its contribution into A_P , and the boundary and opposite boundary terms are placed in S_U for explicit treatment. The symbolic implementation of these BC's is given in Eqs. 4.44 and 4.45. The sign of the convective term supplementing S_U is dependent on the boundary in question, because positive flow through the face can mean either mass flow into or out of the control volume.

$$A_E = 0 \quad (4.44a)$$

$$A_P = A_P + 3D_e \quad (4.44b)$$

$$S_U = S_U - C_e \phi_E + D_e(8\phi_E + \phi_W)/3 \quad (4.44c)$$

$$A_W = 0 \quad (4.45a)$$

$$A_P = A_P + 3D_w \quad (4.45b)$$

$$S_U = S_U + C_w \phi_E + D_w(8\phi_W + \phi_E)/3 \quad (4.45c)$$

On a plane or line of symmetry, both the convective flux and the diffusive flux are known. The diffusive flux is zero, because the gradient of the transported quantity normal to the plane of symmetry is zero. The convective term is also zero since no flow may cross a plane of symmetry. Therefore, no changes in the coefficients are necessary to implement this BC.

For the exit boundary, the assumption is usually made that the streamlines of the flow are straight and normal to the exit plane. This means that the gradients in the streamwise direction are zero ($\frac{\partial}{\partial \xi} = 0$). Quantities on the outflow face are simply those at the node upstream of the boundary convected to the boundary.

$$A_E = 0 \quad (4.46a)$$

$$A_P = A_P + C_e \quad (4.46b)$$

In many flow cases the outflow boundary is a nonphysical boundary which was defined to make the computational domain either finite, or small enough to allow solutions within reasonable computational times. When this boundary is not real, it must be placed such that the chosen boundary conditions will not unduly influence the flow in the region of interest. For the turbo-machinery calculations that were made during this research, a straight extension was added to the physical domain, which allowed the flow to achieve a negligible level of streamline curvature at the exit boundary. This extension was long enough to allow the closure of any recirculating regions which may have developed. No stable BC was found that could be applied to an exit with reversed flow. Therefore, the effective viscosity had to be increased in the neighborhood of the exit to allow separated flow to reattach.

When the outflow direction has a radial component to it, the change in area makes the $\frac{\partial}{\partial \xi} = 0$ assumption invalid. Therefore, this type of boundary is considered to be a specified velocity boundary during a particular outer loop iteration. However, at high Reynolds numbers this leaves the A_P coefficient upwind of the boundary without a convection term to supplement it. Thus, the central coefficient goes to zero, which causes divide by zero errors. To prevent this error

from terminating the execution of the program, without affecting the converged results, $C_e \phi_O$ is added to each side of the equation. One side is treated explicitly and the other implicitly, according to Eq. 4.47.

$$A_E = 0 \quad (4.47a)$$

$$A_P = A_P + 3D_e + C_e \quad (4.47b)$$

$$S_U = S_U - C_e \phi_E + D_e(8\phi_E + \phi_W)/3 + C_e \phi_O \quad (4.47c)$$

After the momentum equations are solved, the outlet flow quantities are calculated so as to enforce global mass conservation. Physically, no mass may accumulate in the flow field domain, but this may not be the case for intermediate solutions. The net mass flux through the inlet, upper, and lower boundaries is calculated, as is the mass flux past the gridline just upstream of the outlet boundary. The outlet contravariant velocity is then calculated according to the relationship shown in Eq. 4.48, which ensures zero net mass flux into the domain. At convergence, the ratio of mass flux into and out of the grid is equal to unity, so $(\rho r U)_e$ equals $(\rho r U)_w$. This is consistent with the parallel outflow boundary condition, as well as the cylindrical upwind approximation that is discussed in the Appendix.

$$U_e = (\rho r_w U_w)(\dot{m}_{in}/\dot{m}_w)/(\rho r_e) \quad (4.48)$$

The cylindrical velocities at the outlet are found using the inverse of the definition of the contravariant velocities, as given in Eqs. 4.49 and 4.50. The η contravariant velocity component, V , equals zero, because the outgoing flow is normal to the exit plane.

$$v_z = (z_\xi U + z_\eta V)/J \quad (4.49)$$

$$v_r = (r_\xi U + r_\eta V)/J \quad (4.50)$$

The tangential velocity component does not appear in the continuity equation, so the angular velocity at the exit is set equal to the near outlet v_θ times the ratio of the near boundary radius

to the boundary radius. This is equivalent to conserving angular momentum in the absence of viscosity.

At a plane of symmetry, v_z is updated at the end of each outer loop iteration. A second-order accurate Taylor series expansion is used to evaluate v_z , such that the gradient of v_z normal to the plane of symmetry is equal to zero.

4.6.2 QUICKR Boundary Conditions

QUICKR (see Appendix) is treated in a manner identical to the other differencing schemes when it comes to implementing the boundary conditions. The flux through the boundary is not allowed to contribute to either the central or neighboring coefficients. However, this does not mean that the boundary coefficients are zero, since the differencing scheme for the CV face opposite the boundary is dependent on the boundary value.

Because the BC's are placed into the central coefficient for both the higher and lower-order differencing schemes, the QUICKR source term, S^Q , must be calculated before the BC's changes are incorporated into A_P . Otherwise, the BC's will be removed by the $(A_P^H - A_P^Q)$ term in S^Q .

4.6.3 Pressure Corrector Boundary Conditions

Boundary conditions for the pressure corrector equations are enforced by setting the boundary coefficients equal to zero. The mass flux at a boundary is a fixed quantity during a given outer loop iteration. This is necessary, because there are no additional conservation equation from which the flux can be computed. Thus, the mass flux evolution must lag the velocity calculations, unless it is initially specified as a given boundary condition. Consequently, the boundary

mass flux does not require a correction. If the correction at the boundary is zero, then the pressure gradient normal to the boundary must also be zero. Therefore, the boundary node does not appear in the stencil, and the pressure corrector boundary conditions are enforced by setting the boundary coefficients at the near boundary nodes equal to zero.

In incompressible flow the pressure field is unique to within an additive constant. Therefore, the system of linear p' equations is underdetermined. Iterative solvers will still converge on this type of equation set. Patankar [16, p. 131] states that iterative solvers converge faster if allowed to seek their own level than if pressure at a particular node is fixed. Direct solution procedures, such as banded matrix algorithms, cannot handle underdetermined systems, so the pressure must be specified at one node. The particular node and the pressure level it is given is immaterial since the whole field can be referenced to any pressure after the final solution is obtained.

4.7 Steps in the PWIM Algorithm

Now that the modified PWIM algorithm developed during this research has been thoroughly explained, a more detailed list of the steps in the solution process is given. This list provides the reader with an outline of the architecture of the computer program.

1. Read in a grid that covers the physical domain.
2. Map the grid into the computational plane, calculating the transformation metrics and the Jacobians at the nodes and control volume faces.
3. Make an initial guess of the velocity and pressure fields.
4. If the flow is considered to be turbulent, use the turbulence model to evaluate the effective nodal viscosities. Use the harmonic mean interpolation to calculate the facial viscosities. See Chapter 5 for details.
5. Conserve momentum.

- a. Calculate the integrated momentum source terms.
 - b. Supplement the source terms with the pressure gradients.
 - c. Evaluate the momentum equation multiplicative coefficients, using the desired differencing scheme.
 - d. Implement QUICKR differencing if desired.
 - e. If a through-flow calculation is being made, use the blade model to evaluate the flow angles and resulting body forces for the nodes within the blade region. See Chapter 6 for details.
 - f. Implement the boundary conditions.
 - g. Calculate the residuals of the momentum equations.
 - h. Under-relax the momentum equations.
 - i. Begin the calculation of the facial contravariant velocities using the current pressure and velocity values.
 - j. Solve the momentum equations to yield new velocity fields.
 - k. Complete the calculation of the facial contravariant velocities using the newly determined axial and radial velocities.
 - l. Calculate the outflow velocities.
6. Conserve mass.
- a. Calculate the pressure corrector equation coefficients.
 - b. Calculate the mass sources and the source term for the pressure corrector equations.
 - c. Solve the pressure correction equations.
 - d. Update the pressure field.
 - e. Update the facial contravariant velocities.
 - f. Update the nodal axial and radial velocity components.
 - g. Update the plane of symmetry velocities.
7. If convergence has not been reached, return to Step 4.

Chapter 5

Turbulence Model

The thin layer approximation algebraic turbulence model of Baldwin and Lomax [29] was adapted to estimate the turbulence effects in the axisymmetric flow. Originally proposed to handle separated flows in Cartesian coordinates, the model was formulated by assuming *a priori* that the layer of fully turbulent flow near a vorticity generating wall is thin. The justification for this assumption is that the number of nodes required to properly resolve the actual streamwise diffusion is so prohibitively large that the highly stretched control volumes that are typically placed about the boundaries yield to this assumption by default. In so far as the high aspect ratio control volumes prevent the resolution of the true streamwise gradients, Navier-Stokes solvers contain the thin layer approximation at high Reynolds numbers even though they are not explicitly coded in this manner.

One of the chief advantages realized by using this model is that knowledge of the boundary layer thickness is not needed. For the complex geometries and separated flows associated with turbomachinery, automating the boundary layer thickness calculation within the flow solver is exceedingly difficult. Baldwin and Lomax proposed using the distribution of vorticity as an indication of the length scale associated with the boundary layer. In convection dominated flow domains vorticity is neither created nor destroyed, but diffusion increases near the wall in physical situations, generating vorticity.

The algebraic model, while not as rigorous as the two equation $\kappa - \epsilon$ model, is more reflective of the actual physics than zero equation mixing length models. Actually, the model of Baldwin and Lomax does make use of the mixing length formulation, but the specification of the eddy size

is not done *a priori*. Because of the severe problem dependence limitations of the $\kappa - \epsilon$ model and the amount of CPU time its use entails, this scheme was dropped in favor of the simpler algebraic model. A brief survey of publications in the field of turbomachinery CFD shows that the Baldwin-Lomax scheme is a widely used turbulence model [9,22,30].

The Baldwin-Lomax model utilizes a number of nondimensional constants, the values used by the originators of the model are:

$$A^+ = 26$$

$$C_{CP} = 1.6$$

$$C_{KLEB} = 0.3$$

$$C_{WK} = 0.25$$

$$k = 0.4$$

$$K = 0.0168 \quad (5.1)$$

Schlichting [17, pp. 602-604] uses the values $A^+ = 26$ and $k = 0.4$ when calculating flow over a flat plate and when calculating axisymmetric, turbulent pipe flow. In this research this proposal is extended so that all of the parametric constants used in the Baldwin-Lomax model are the same for flows in both axisymmetric and Cartesian frames of reference.

Baldwin and Lomax modify the dual-layer algebraic eddy viscosity model of Cebeci to provide a means of determining the turbulent viscosity. Two layers, the inner and outer, are assumed to exist along the boundary, each region having a particular formulation for viscosity. Depending upon the region in which the node lies, the viscosity is calculated as:

$$\mu_t = \begin{cases} \mu_{t,inner}, & y \leq y_{crossover} \\ \mu_{t,outer}, & y > y_{crossover} \end{cases} \quad (5.2)$$

The parameter y is the distance normal from the wall to the node, and $y_{crossover}$ is the smallest value of y for which the inner and outer turbulent viscosities are equal. The value of $y_{crossover}$ cannot be determined analytically. By inspection, the inner turbulent viscosity (Eq. 5.3) is

identically zero on the wall ($y = 0$), while $\mu_{t,inner}$ (Eq. 5.11) is not. Therefore, the transition from inner to outer is marked by the point where the value of $\mu_{t,inner}$ first exceeds that of $\mu_{t,outer}$.

In the interior region, the Prandtl-Van Driest mixing length formulation is used; the nondimensional form of which is:

$$\mu_{t,inner} = Re \rho \ell^2 |\omega| \quad (5.3)$$

$$\ell = k y [1 - \exp(-y^+ / A^+)] \quad (5.4)$$

$$\tau_w = \mu_w \left| \frac{\partial u}{\partial y} \right| \quad (5.5)$$

$$y^+ = \frac{\sqrt{\rho_w \tau_w} y}{\mu_w} = y \sqrt{Re \left| \frac{\partial u}{\partial y} \right|} \quad (5.6)$$

$$|\omega| = |\vec{\nabla} \times \vec{u}| \quad (5.7)$$

The eddy viscosity mixing length is represented by ℓ . The Reynolds number appears because nondimensionalized quantities are used throughout the description of the turbulence model. A "w" subscript denotes a quantity evaluated at the wall. As previously stated, the turbulent viscosity at the wall is zero; thus, μ_w is equal to the laminar viscosity. The wall shear, τ_w , is equal to the product of the wall viscosity and the derivative of the velocity parallel to the wall, u , with respect to the distance from it. If the local inclination of the wall with respect to the horizontal is α , then the expression for this derivative is given as follows:

$$u = v_z \cos \alpha + v_r \sin \alpha \quad (5.8)$$

$$\frac{\partial u}{\partial y} = \frac{\partial z}{\partial y} \frac{\partial u}{\partial z} + \frac{\partial r}{\partial y} \frac{\partial u}{\partial r} \quad (5.9a)$$

$$= -\sin \alpha \frac{\partial u}{\partial z} + \cos \alpha \frac{\partial u}{\partial r} \quad (5.9b)$$

$$= \frac{1}{J} \frac{\partial u}{\partial \eta} (r_\xi \sin \alpha + z_\xi \cos \alpha) \quad (5.9c)$$

Since the no-slip wall is mapped into a line of constant η , the derivative of u with respect to ξ is equal to zero. Thus, it drops out of the expression as Eq. 5.9b is transformed from the physical to computational domain.

The magnitude of the vorticity, $|\omega|$, in axisymmetric flow is given by Eq. 5.10.

$$|\omega| = \sqrt{\left(\frac{\partial v_r}{\partial z} - \frac{\partial v_z}{\partial r}\right)^2 + \left(\frac{\partial v_\theta}{\partial z}\right)^2 + \left(\frac{1}{r} \frac{\partial(rv_\theta)}{\partial r}\right)^2} \quad (5.10)$$

Expressed in terms of computational space derivatives, the magnitude of vorticity is a function of eight different gradients.

$$|\omega| = \frac{1}{J} \left[\left(r_\eta \frac{\partial v_r}{\partial \xi} - r_\xi \frac{\partial v_r}{\partial \eta} - z_\xi \frac{\partial v_z}{\partial \eta} + z_\eta \frac{\partial v_z}{\partial \xi} \right)^2 + \left(r_\eta \frac{\partial v_\theta}{\partial \xi} - r_\xi \frac{\partial v_\theta}{\partial \eta} \right)^2 + \left(\frac{1}{r} \left(z_\xi \frac{\partial(rv_\theta)}{\partial \eta} - z_\eta \frac{\partial(rv_\theta)}{\partial \xi} \right) \right)^2 \right]^{\frac{1}{2}} \quad (5.11)$$

The formulation for the outer region turbulent viscosity is given in Eq. 5.12. The Klebanoff intermittency factor, $F_{KLEB}(y)$, is calculated using Eq. 5.13. The multiplicative factor, F_{WAKE} , is evaluated according to Eq. 5.14. This factor is a function of the difference between maximum and minimum total velocities within a profile, u_{DIF} . Typically, the minimum velocity is zero on the wall and u_{DIF} can be set equal to the maximum speed. A profile is defined by the group of nodes within a particular column of control volumes all closest to the same wall. Within a given profile, F_{max} is the value of $F(y)$, Eq. 5.16, at its first local maximum away from the wall. The value of y at which that maximum occurs is referred to as y_{max} .

$$\mu_{outer} = Re K C_{CP} \rho F_{WAKE} F_{KLEB}(y) \quad (5.12)$$

$$F_{KLEB}(y) = \left[1 + 5.5 \left(\frac{y C_{KLEB}}{y_{max}} \right)^6 \right]^{-1} \quad (5.13)$$

$$F_{WAKE} = \min[y_{max} F_{max}, C_{WK} y_{max} u_{DIF}^2 / F_{max}] \quad (5.14)$$

$$u_{DIF} = \sqrt{v_z^2 + v_r^2 + v_\theta^2} \Big|_{max} - \sqrt{v_z^2 + v_r^2 + v_\theta^2} \Big|_{min} \quad (5.15)$$

$$F(y) = y |\omega| [1 - \exp(-y^+ / A^+)] \quad (5.16)$$

Since the maximum value of $F(y)$ cannot be found analytically, the exact location of the transition from the inner to the outer layer cannot be evaluated before the inner and outer

viscosities are calculated. Therefore, μ_{inner} and μ_{outer} must be compared to determine the region in which the nodes lie.

As $F(y)$ can only be evaluated discretely, there is a possibility that the maxima may vary widely from one profile to the next. To smooth this variation, Baldwin and Lomax recommend using a quadratic fit to the points about the maxima and extracting F_{max} and y_{max} from the curve fit. This parabolic fit is done as long as F_{max} does not occur at the endpoints of the profile.

Using the aforementioned relationships, the net viscosity, $\mu_{lam} + \mu_t$, is found at each node. Because the discretized form of the momentum equations requires the viscosity to be known at the control volume faces, the nodal viscosities must be used to estimate these quantities. Patankar [16, pp. 44-47] suggests using the harmonic mean of the nodal values as the interface value. This assures a more accurate description of the flux across the face. The harmonic mean is calculated using Eqs. 5.17 with the relationships shown in Eqs. 5.18 used to calculate f_e and f_n in the computational plane. The total, rather than turbulent, viscosity is interpolated because it is the net viscosity that appears in the momentum equations, not μ_t by itself.

$$\mu_e = \left(\frac{1 - f_e}{\mu_O} + \frac{f_e}{\mu_E} \right)^{-1} \quad (5.17a)$$

$$\mu_n = \left(\frac{1 - f_n}{\mu_O} + \frac{f_n}{\mu_N} \right)^{-1} \quad (5.17b)$$

$$\begin{aligned} f_e &= 2\sqrt{\Delta z^2 + \Delta r^2}|_e / \sqrt{\Delta z^2 + \Delta r^2}|_E \\ &= 2\sqrt{z_\xi^2 + r_\xi^2}|_e / \sqrt{z_\xi^2 + r_\xi^2}|_E \end{aligned} \quad (5.18a)$$

$$\begin{aligned} f_n &= 2\sqrt{\Delta z^2 + \Delta r^2}|_n / \sqrt{\Delta z^2 + \Delta r^2}|_N \\ &= 2\sqrt{z_\eta^2 + r_\eta^2}|_n / \sqrt{z_\eta^2 + r_\eta^2}|_N \end{aligned} \quad (5.18b)$$

Baldwin and Lomax state that grid nodes must be placed within the inner layer, and suggest that the first node away from the wall be located such that $y^+ \sim 2$. Failure to place nodes in the inner region results in a near zero turbulent viscosity throughout the profile. Since y^+ is

proportional to the square root of the Reynolds number, the nodes must be closely clustered near the wall for high Reynolds number flows.

A simple method was used to determine how far to place the first node from the wall [31]. For fully turbulent flow on a flat plate the wall shear is proportional to $Re_x^{-1/5}$, where the length scale in Re_x is the distance from the leading edge, x . Solving for the velocity gradient in τ_w in terms of the nondimensional variables gives the empirical relationship shown in Eq. 5.19 [32].

$$\frac{\partial u}{\partial y} = 0.02885 Re / Re_x^{1/5} \quad (5.19)$$

The gradient at the wall is evaluated as the ratio of Δu to Δy , and Δy is equal to y at the first node from the wall. Re_x can be expressed as the product of the flow Reynolds number and the ratio of the distance from the leading edge, x , to the plate length, ℓ_p . The boundary layer grows slowly, at a rate proportional to $\sqrt[3]{x/\ell_p}$, so the thickness at the midspan is representative of the boundary layer thickness across the entire plate. Manipulating the equations, and substituting $x/\ell_p = 1/2$ gives the distance from the wall for the second row of nodes.

$$y = y^+ Re^{-9/10} \sqrt{\frac{\frac{1}{2}}{0.02885}} \quad (5.20)$$

To capture the true boundary layer profile, it is necessary to place a number of points within the inner region. Transition from the inner layer to the outer layer occurs at $y^+ \sim 50$ [29], so a rough estimate of the inner layer thickness is found by assuming Eq. 5.20 holds all the way to the crossover point.

Without sacrificing accuracy, computational efficiency is increased by applying the model only during intermittent outer loop iterations. This does not change the converged results, but it lessens the significant expense of implementing the turbulence model.

Chapter 6

Blade Model

The ways in which convective, diffusive, and pressure influences on the fluid motion are calculated have been explained in previous chapters. Within turbomachines, blade rows, which are external to the fluid, join these factors in determining the fluid dynamic characteristics. In the following sections, the method of quantifying how the blades influence the flow is presented. Axial symmetry has been assumed to this stage, but as will be discussed, the loading of the blade can be used to approximate nonaxisymmetric effects.

The focus of the present research is on calculating the flow in a mixed-flow centrifugal impeller. The blade model presented in the following sections reflects this type of turbomachine, but the approach is valid for axial flow as well as radial flow turbomachines. Also, the approach is not limited to use with single blade rows or single-stage turbomachines.

The effects of blade are introduced into the momentum equations via the body force terms in Eq. 4.15-17. As mentioned in Chapter 1, these forces are only applied to control volumes which lie within the region in which the blades are projected onto the axisymmetric plane. To facilitate the distribution of the body force and to help determine which control volumes are actually in the projection of the blade, referred to as control volumes that lie on the blade, a restriction is placed on the grid when calculating turbomachinery through-flows. Gridlines must be placed along the projections of the leading and trailing edges onto the axisymmetric plane. Thus, every control volume lies either entirely on or entirely off the blade. Body forces appear in the momentum equations in dimensions of force per unit volume. The volume of a CV is

calculated by combining the volumes of the conical frustums formed by the sweeping the CV faces around the centerline.

An overall outline of the blade model is given in the following list. It is provided to give the reader a sense of the approach used, resulting in a better understanding of the explanations of the individual steps in the remaining sections in this chapter.

1. Convert the blade geometry to a form usable on the grid.
2. Approximate the degree to which the blade fails to turn the fluid tangent to its surface at the trailing edge. This is used to determine the blade loading near the trailing edge.
3. Specify a mean streamline in the blade-to-blade plane. This is used to determine the amount that the blade must turn the flow.
4. Determine the tangential velocity and tangential body force required to keep the flow along the specified streamline.
5. Estimate the static pressure rise along the blade
6. Determine the radial and axial body forces required to conserve momentum in the meridional direction and provide the estimated pressure rise.

6.1 Representation of the Blade

The geometry of the blade is obviously a very important factor in determining what effect it will have on the flow. Therefore, the first step in the blade model is to numerically represent the shape of the blade.

In the direct or analysis problem, the blade geometry is known prior to the calculation of the flow. The geometry is defined in terms of coordinates of discrete points that lie on the blade surface. These points are grouped according to design sections that form airfoil or vane sections stacked up between the inner hub and outer shroud. Fig 1.1a shows the design section of each

blade at a particular radial station. The other design sections are located at the intersection of the blades and a surface of revolution referenced to a different base radius. Because the flow is considered to be axisymmetric, all of the blades must have the same geometry, and thus only one needs to be considered. A mean line or camberline can be determined along each design station by bisecting the pressure and suction surfaces. In the blade model that has been developed, the complexity of determining the proper loading distribution has been simplified by ignoring the effects of blade thickness. Thus, for this model the blade is completely defined in terms of the camberlines. The meridional direction for a particular design section is defined by the profile of the surface of revolution, as shown in Fig. 1.1b. This meridional direction, m , is coincident with the projection of the camberline onto the axial-radial plane.

One of the primary factors determining how the fluid reacts to the blade is the local blade angle, β' , which is the angle of the blade camberline with respect to the meridional direction. Since the blades are three-dimensional, the camberlines are actually curves that do not lie in any one plane. To simplify the calculation of the blade angle, the camberlines are conformally mapped, according to the operation defined in Eq. 6.1, into the transformed blade-to-blade plane which is defined in terms of two coordinates (x, y) . Because the mapping process is conformal, angular measure is conserved and the local slope of the curve in the blade-to-blade plane is equal to the negative tangent of the blade angle.

$$dx = r_{LE} d\theta \quad (6.1a)$$

$$dy = \frac{r_{LE}}{r} dm \quad (6.1b)$$

$$\beta' = \tan^{-1} \left(-\frac{dx}{dy} \right) \quad (6.2)$$

The conformal plane coordinates are scaled by the leading edge radius, r_{LE} , of the particular design section.

Once the coordinates of the camberline in the conformal plane have been determined, they must be interpolated to yield values at the grid nodes and east/west control volume faces. The

design points describing the camberline are considered to form a mesh on the axisymmetric plane. In this plane, the known axial and radial coordinates of both the grid and the blade are used to relate blade characteristics from the design points to the positions on the grid. Four neighboring design points are isolated by finding the ones that form a quadrilateral mesh cell inside which the desired grid point lies. Blade angles, conformal plane coordinates, or any other quantity known on the blade surface can be interpolated from these four points to the location on the grid.

The meridional directions along the blade are defined by the design sections, but these do not exist on the grid. Instead, the ξ gridlines are assumed to specify the meridional directions. Metrics can be used to express meridional quantities in terms of axial and radial values. The meridional velocity, V_m , is equal to the projection of the total velocity onto the meridional line, or ξ gridline.

$$\frac{\partial}{\partial m} = z_m \frac{\partial}{\partial z} + r_m \frac{\partial}{\partial r} \quad (6.3)$$

$$\frac{\partial z}{\partial m} = z_m = z_\xi / (z_\xi^2 + r_\xi^2)^{1/2} \quad (6.4a)$$

$$\frac{\partial r}{\partial m} = r_m = r_\xi / (z_\xi^2 + r_\xi^2)^{1/2} \quad (6.4b)$$

$$V_m = z_m v_z + r_m v_r \quad (6.5)$$

6.2 Determination of the Deviation Angles

Typically, the flow is not aligned with the blade as the fluid passes over the trailing edge. The angular difference between the flow angle, β , and the blade angle, β' , is known as the deviation angle, δ . This quantity is important from the standpoint of turbomachinery performance. If the deviation angle is large, then the blade may not adequately turn the flow. The deviation angle is a result of having only a finite number of blades, boundary layer development and possible

separation, secondary flows, and for centrifugal machines the relative rotation of the fluid within the blade row. The determination of the deviation angle provides a means of estimating the blade loading near the trailing edge.

It would be desirable to allow the deviation angle to evolve as intermediate flow solutions are determined in the iterative process. However, δ is an important factor in the overall body force distribution, so it is imperative that realistic values of δ be maintained. This cannot be assured if the deviation angle is calculated using potentially unrealistic intermediate solutions, so the deviation angle is calculated at the outset of the solution process and held fixed to enhance computational stability. At this stage in the understanding of the numerical modeling of blade forces this is not an unreasonable simplification. A physically realistic iterative scheme based on a uniform static pressure at the discharge is used to determine the deviation angles. This procedure also prevents the solution from being dependent upon the initial guess of the velocity field.

In the field of centrifugal turbomachines, the flow deviation from the blade direction is more commonly expressed as a slip factor, σ , rather than a deviation angle. A relationship for the slip factor, based on a simple model of the relative fluid rotation within the blade row, was developed by Stodola [33]:

$$\sigma = 1 - \frac{\frac{\pi}{N} \cos \beta'_{TE}}{1 - \phi_{TE} \tan \beta'_{TE}} \quad (6.6)$$

where N is the number of blades and ϕ is the flow coefficient defined as the ratio of the meridional velocity to the wheel speed, U . The wheel speed is equal to the angular velocity of the blade times the radius. In the impeller calculations of this investigation, the reference length used for nondimensionalizing variables is equal to the diameter of the impeller shroud at the leading edge, and the reference velocity is the wheel speed at this location. Therefore, Eq. 6.7 holds for the nondimensional variables.

$$\omega = 2 \quad (6.7)$$

There is a distinct meridional path for each row of control volumes; thus, a separate slip factor and resulting deviation angle are calculated for each row. The Stodola slip factor is only a function of the outlet conditions, thus only a V_m distribution at the trailing edge must be specified. Experimental data indicates that the inlet meridional velocity profile is nearly uniform over a wide range of operating conditions (see Section 8.3), making the leading edge V_m distribution a known function of the operating condition. The trailing edge V_m distribution is calculated by assuming that the static pressure rise, from the leading to trailing edge, is the same for each row of nodes. An initial estimate of the outlet profile is made, and this data is used to calculate the change in static pressure for each grid row according to the Euler pump equation (Eq. 6.8). The outlet v_θ (Eq. 6.9) and slip factors are calculated using known values.

$$\Delta p = \eta \rho U_{TE} v_{\theta TE} - \frac{1}{2} \rho (V_{mTE}^2 + v_{\theta TE}^2 - V_{mLE}^2) \quad (6.8)$$

$$v_{\theta TE} = \sigma (U_{TE} - V_{mTE} \tan \beta'_{TE}) \quad (6.9)$$

The efficiency of the pump, η , is evaluated by the method described in Section 6.4. No preswirl is imparted to the fluid, so the inlet tangential velocity is zero and does not appear in the equation. The average static pressure rise over all of the gridlines is determined and used to calculate a new outlet meridional velocity profile. An expression for V_{mTE} is found by manipulating Eq. 6.8. The resulting distribution of exit meridional velocities is scaled to satisfy continuity, and the process is repeated until the meridional velocity profile converges.

The slip factor resulting from the converged V_{mTE} distribution is used to calculate a deviation angle by subtracting the blade angle from the flow angle.

$$\beta_{TE} = \tan^{-1} [(U_{TE} - \sigma (U_{TE} - V_{mTE} \tan \beta'_{TE})) / V_{mTE}] \quad (6.10)$$

$$\delta = \beta_{TE} - \beta'_{TE} \quad (6.11)$$

6.3 Specification of the Streamline and Flow Angles in the Blade Passage

One effect that the blades have on the flow is to turn or deflect the flow from an axial-radial plane, thereby changing its angular momentum. In other words, the blades impart swirl to the flow. Axial symmetry is not violated by streamlines which do not lie within a single axial-radial plane, the only requirement is that these streamlines must represent some average curve that all of the flow can be considered to follow. Determining the local turning provided by the blades is the same as determining a component of the force that the blades exert on the fluid. In the blade model developed in this investigation, the amount of turning is calculated by estimating the shape of the streamline that the fluid follows as it moves through the blade passage. The blade loading is then found by calculating the force required to have the flow follow the estimated streamline.

The streamlines are laid out in conformal planes, where the slope of the path is equal to the slope in the physical space. The deviation angle, determined in Section 6.2, fixes the slope of the streamline at the trailing edge. The flow angle at the leading edge is determined from the upstream velocity vector and the known blade speed as given in Eq. 6.12.

$$\beta_{LE} = \tan^{-1} \left(\frac{U_{LE} - v_{\theta LE}}{V_{mLE}} \right) \quad (6.12)$$

The tangential velocity at the leading edge, $v_{\theta LE}$ is typically zero because there is no preswirl. However, when a recirculating flow is set up in the leading edge region, some swirl is convected out of the blade row at the leading edge, then back into it at a different radial station. Therefore, $v_{\theta LE}$ is retained in Eq. 6.12 and set equal to the tangential velocity at the node upstream of the leading edge. A separate streamline is specified for each row of control volumes, because the meridional direction follows the ξ gridline which is unique for each row of nodes.

Within the mixed-flow impeller that is the focus of this research, the behavior of the flow in the area of the leading edge is strongly dependent on the local inlet incidence angle. The incidence

angle is the angular difference between the relative flow direction and the blade camberline at the leading edge. In the trailing edge region, the fluid motion is primarily dependent on the blade shape and the deviation angle. In between these two areas, the relatively long blades turn the flow tangent to the blade. This behavior is modeled in the specification of the mean streamline curve in the blade-to-blade plane. The control point curve fitting technique, as described by Eiseman [34], is used to define a smooth streamline while specifying a minimal number of points. This curve fitting technique was chosen because the resulting curve lies in the convex hull formed by the control points, thereby preventing the occurrence of physically unrealistic oscillations in the streamline.

Fig. 6.1 shows a representative blade passage in the conformal transformed plane and the layout of the streamline. The thick, solid lines represent the blades, and the dashed line corresponds to the center of the blade passage. Six control points, marked by dots, are used to define the streamline. They are placed to assure that the flow enters and leaves the passage with the correct flow angles and to guarantee that the flow is turned tangent to the blade at the one-third chord point. The control point curve is characterized by a curve which passes through the endpoints with a slope equal to that of the line segment connecting the endpoint and the adjacent point. Thus, the first and second points are placed to yield the desired β_{LE} , and the fifth and sixth points are located to give the required β_{TE} . The control point curve intersects the line segment between consecutive interior control points with a slope equal to that of the connecting segment. Therefore, the streamline, marked by the thin, solid line in Fig. 6.1, passes tangent to the blade at the one-third chord point, because the third and fourth control points are centered about the one-third chord location along the passage centerline.

The location of the streamline within the blade passage is arbitrary, only the curvature of the streamline affects the flow. But in keeping with the axisymmetric assumption, it is reasonable to assume that the flow primarily remains centered within the blade passage. As previously stated, the blade model is capable of accounting for nonaxisymmetric effects. The leading and trailing

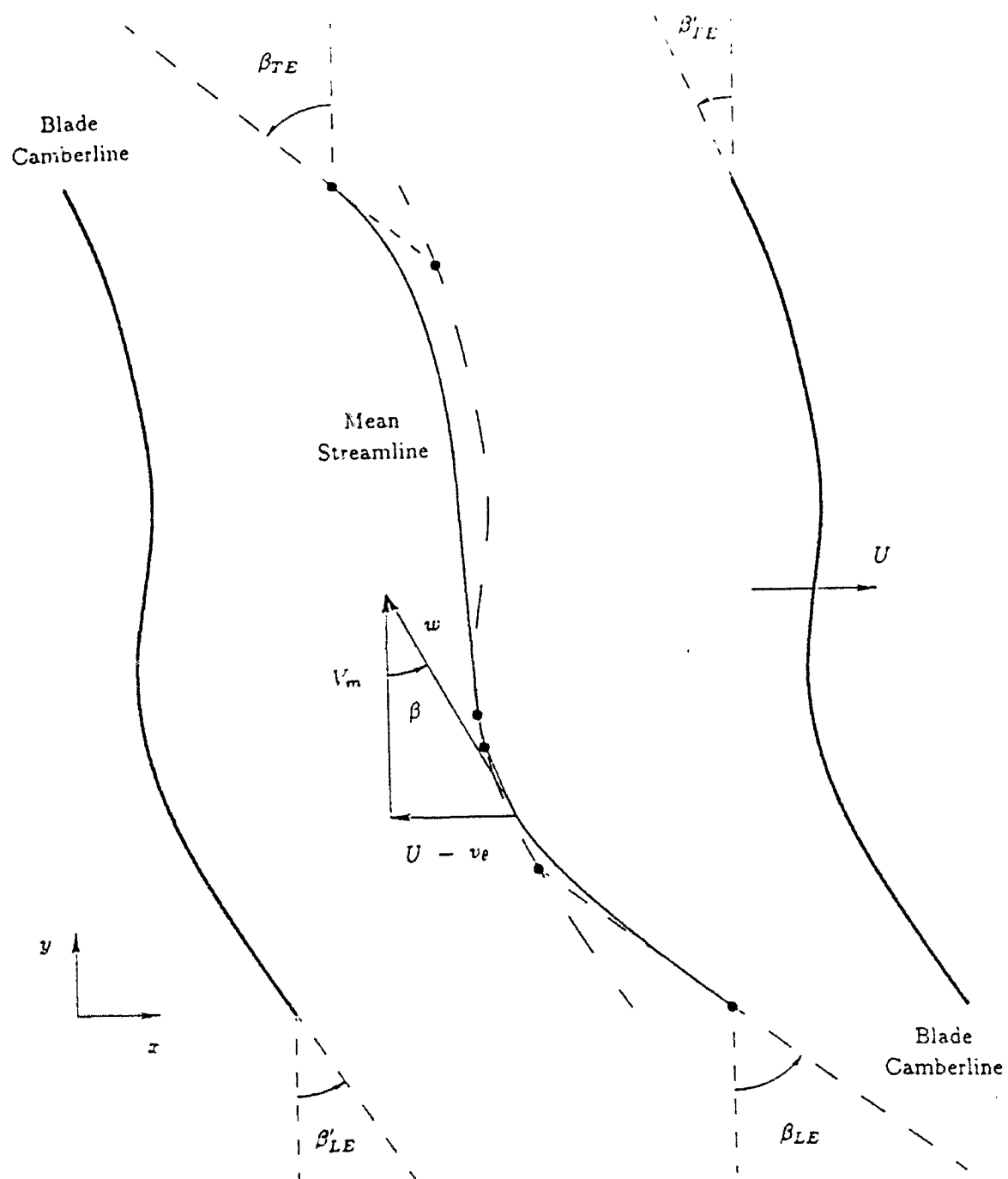


Figure 6.1

Streamline Layout in the Transformed Blade-to-Blade Plane

edges are the most sensitive regions of the blade, so the nonaxisymmetric effects will be focused in these regions. Thus, the second and fifth control points are placed on the passage centerline, making the departure of the mean streamline from the centerline minimal for the bulk of the chord length. The first and last points are meridionally located at the leading and trailing edges of the blade row and are positioned on the z axis in accordance with the inlet and outlet flow angles. The fluid cannot cross from one blade passage to another, so the first and last points are placed not farther than half of the width of the blade passage away from the centerline. The second and fifth points are repositioned closer to the blade tips if the slopes would otherwise result in the streamline crossing the channel boundary. Nominally, the second point is placed at the $1/6$ chord location and the fifth point is set at the $9/10$ chord location.

Control point curve interpolation functions are used to calculate the local slope of the streamline at each node and east/west CV face according to their meridional coordinate. The flow angle is equal to the inverse tangent of the negative slope.

$$\beta = \tan^{-1} \left(-\frac{dz}{dy} \right) \quad (6.13)$$

6.4 Determination of the Tangential Velocities and Body Forces from the Flow Angles

The streamline geometry does not directly enter into the momentum equations. Therefore, in order to place the effects of the blade into the momentum equations, quantities which do appear in the momentum equations must be extracted from the knowledge of the streamline shape.

Dividing the differential distance in Eq. 6.13 by a small change in time results in an expression of the flow angle in terms of two velocities.

$$\beta = \tan^{-1} \left(-\frac{dx/dt}{dy/dt} \right) \quad (6.14)$$

The velocity in the numerator is equal to the flow speed in the tangential direction relative to the moving blade. The speed represented in the denominator of Eq. 6.14 is simply the meridional velocity. For a known flow angle, the angular velocity in the absolute reference frame is therefore calculated according to Eq. 6.15.

$$v_\theta = U - V_m \tan \beta \quad (6.15)$$

An angular velocity is calculated at the nodes and east/west CV faces from the flow angles which have been calculated at these locations. Linear interpolation of the nodal velocities is used to calculate the meridional velocity at the control volume face.

After the tangential velocity distribution is found over the entire blade, the tangential body forces that will yield this distribution are determined. These forces are calculated for every control volume on the blade according to the simple angular momentum balance shown in Eq. 6.16. The integral of the angular momentum source term (Eq. 4.17) with the body force term removed is denoted by S^θ .

$$\Delta \xi \Delta \eta r J F_\theta = A_O v_{\theta O} - \sum_O A_i v_{\theta i} - S_O^\theta \quad (6.16)$$

Because the tangential velocities, calculated using Eq. 6.15, are used in the evaluation of F_θ , the velocity field that is returned by the solution of the discretized momentum equation will satisfy the specified streamline at convergence. During intermediate solutions the changing axial and radial velocities tend to perturb the flow from the streamline. However, as discussed in Section 6.5, the perturbation is kept to a minimum by updating v_θ and F_θ after the A_i momentum coefficients are calculated, but before the equations are under-relaxed.

Determining the radial and axial blade loading is the final step in implementing the blade model. F_z and F_r are the only remaining quantities that have not been evaluated. These forces are found by resolving a force in the meridional direction into its cylindrical components.

$$F_z = z_m F_m \quad (6.17a)$$

$$F_r = r_m F_m \quad (6.17b)$$

In the blade model, the forces normal to the meridional direction in the axisymmetric plane are assumed to be negligible. This approximation is not unreasonable as the normal force is usually small compared to the centripetal forces and radial pressure gradients. The forces normal to the meridional direction become more significant when the blades are highly skewed.

The primary effect of the meridional force is to propel the fluid through a region of adverse pressure gradient. In a centrifugal pump, there is a large total pressure rise from the inlet to the discharge of the impeller. By manipulating Euler's pump equation, the increase in total pressure can be shown to be comprised of two parts, a rise in static pressure and a rise in dynamic pressure. The dynamic head rise goes directly into changing the kinetic energy of the fluid; therefore, it is not affected by the blade efficiency. The change in static enthalpy from one meridional position to the next is the result of centrifugal forces and the diffusion of the relative velocity.

$$\Delta p = \frac{1}{2} \rho ((U_e^2 - U_w^2) + (w_w^2 - w_e^2)) \quad (6.18)$$

$$w^2 = V_m^2 + (U - v_\theta)^2 \quad (6.19)$$

The magnitude of relative velocity in the blade passage is denoted as w . The first term in this equation represents the head rise due to the centrifugal forces exerted on the fluid. Because centrifugal forces are present even in solid body rotation of a fluid (i.e. without motion of the fluid relative to the blades) it can be assumed that no losses are associated with this portion of the change in static pressure. However, total pressure losses do occur during the diffusion of the relative velocity. Therefore, an efficiency, η , is associated with this quantity. The resulting expression for the static pressure rise is shown in Eq. 6.20.

$$\Delta p = \frac{1}{2} \rho ((U_e^2 - U_w^2) - \eta(w_w^2 - w_e^2)) \quad (6.20)$$

By coupling the efficiency to the existing flow conditions, a simple loss model is incorporated into the blade model.

There is no analytic expression for the efficiency, so an empirical model must be used. Viscous losses throughout the blade passage and near the leading and trailing edges depend upon different parameters; therefore, different expressions for the efficiency are used in each section of the blade. Eq. 6.22 is used to calculate the efficiency in the first third of the blade. Eq. 6.25 is used in the latter third of the blade. In the middle third of the blade, the efficiency varies linearly as a function of chord length from the leading to trailing edge value.

The efficiency for the region near the leading edge is assumed to depend solely on the incidence angle, i . Beyond a 20° incidence, full separation of the fluid from the leading edge is assumed, and the efficiency is given a minimum value. Up to an 8° incidence, the fluid is considered to be fully attached and the leading edge is presumed to be working at maximum efficiency. A linear variation between the extremes is used for intermediate incidence values. The maximum and minimum efficiencies of 0.75 and 0.10, respectively, were selected based on typical performance values for cascades. No attempt was made in this investigation to improve these estimates.

$$i = \beta_{LE} - \beta'_{LE} \quad (6.21)$$

$$\eta_{LE} = \begin{cases} 0.75 & \text{if } |i| \leq 8^\circ; \\ 0.10 & \text{if } |i| \geq 20^\circ; \\ 0.75 - .65 \left(\frac{|i| - 8^\circ}{20^\circ - 8^\circ} \right) & \text{otherwise.} \end{cases} \quad (6.22)$$

The efficiency at the trailing edge is assumed to be a function only of the relative diffusion. If the flow at the outlet has reversed, then the efficiency is assumed to be a minimum. Otherwise, a piece-wise linear distribution is used which is a function of the ratio of the net change in relative velocities to the leading edge relative velocity. The expression corresponding to the first satisfied condition in Eq. 6.25 is used to calculate η_{TE} . The constants in Eq. 6.25 were estimated based on typical diffuser or nozzle performance.

$$\gamma = (w_{LE} - w_{TE})/w_{LE} \quad (6.23)$$

$$\gamma_{max} = (w_{LE} - (V_{mTE}^2 + U_{TE}^2)^{1/2})/w_{LE} \quad (6.24)$$

$$\eta_{TE} = \begin{cases} 0.1 & \text{if } V_{mTE} < 0; \\ 0.1 & \text{if } \gamma \geq 0.9; \\ 0.95 & \text{if } w_{TE} \geq (V_{mTE}^2 + U_{TE}^2)^{1/2} \\ 0.95 - 0.2(\gamma_{max} - \gamma)/\gamma_{max} & \text{if } \gamma \leq 0 \\ 0.75 - 0.65\gamma/0.9 & \text{if } \gamma > 0 \end{cases} \quad (6.25)$$

The meridional and tangential velocities and the wheel speeds are known at the east/west control volume faces. Therefore, the change in pressure along the meridional direction from the western to the eastern face is found using Eq. 6.20. The meridional body force required to produce this static pressure rise can now be computed by applying the momentum equations.

An expression for the meridional pressure gradient is found by combining the inviscid axial and radial momentum equations according to Eq. 6.3. Some pressure changes will occur due to the shear forces in the fluid. By neglecting these terms in the determination of the body forces, the pressure correction equation will generate a pressure field that is consistent with both the viscous and blade forces. If these terms were included, the body force would account for the viscous effects and there would be nothing left to drive the pressure corrector equation for the control volumes in the blade. The meridional body force appears in the expression of the meridional pressure gradient. The meridional momentum equation can be manipulated to form an expression for the meridional body force as shown in Eq. 6.26.

$$\begin{aligned} \Delta\xi\Delta\eta r J F_m &= \Delta\xi\Delta\eta r J \frac{dp}{dm} - \Delta\xi\Delta\eta r_m J \rho v_\theta^2 + \\ &\quad z_m [(\Delta\eta \rho r U v_z)_w^e + (\Delta\xi \rho r V v_z)_s^n] + \\ &\quad r_m [(\Delta\eta \rho r U v_r)_w^e + (\Delta\xi \rho r V v_r)_s^n] \end{aligned} \quad (6.26)$$

The meridional pressure gradient is evaluated as the ratio of the change in static pressure, as given by Eq. 6.20, to the change in meridional distance, which is equal to the denominator in Eq. 6.4. Linear interpolation of the nodal values of v_z and v_r is used to calculate the facial velocity components.

If the meridional flow is reversed within a blade row, then a streamline cannot exist entirely within a single row of control volumes. When this occurs the meridional body force is assumed to

be caused by drag, acting counter to the flow, in the meridional direction. That drag force, which is assumed to be negligible compared to the pressure rise in forward flow regimes, is calculated as follows:

$$F_m = D = \frac{1}{2} C_D w^2 A / V \quad (6.27)$$

where V is the volume of the CV, A is the surface area of the CV projected onto the blade, and C_D is the coefficient of drag. A coefficient of drag of 0.5 was found to provide enough force to prevent recirculating regions from growing unrealistically large. If the flow is reversed at the leading or trailing edges, it is assumed that there is enough swirl to allow the fluid to enter or leave the blade row with a flow angle equal to the blade angle.

6.5 Implementation of the Blade Model within the PWIM Algorithm

The remainder of the chapter explains how the blade model is implemented within the solution algorithm, while maintaining stability. As the outline in Section 4.7 shows, the blade model is called after the viscous source terms and the A_i coefficients have been calculated. This sequence is followed because the blade model also requires these quantities. The model is called before the under-relaxation is applied to the momentum equation coefficients, so that the iterative solution results in the required streamline as soon as possible. The tangential velocity is directly affected by the blade model. The blade model affects the axial and radial velocities after the momentum equations have been solved. The pressure is only affected by the presence of the blades indirectly, and because PWIM is a pressure corrector scheme, the development of the pressure field always lags that of the velocity fields. If the pressure has not had a chance to develop, physically unrealistic velocity fields will arise as there will be no centripetal pressure forces and meridional pressure rise to counter the swirl and body forces. Thus, immediately

introducing the full effect of the blade model into the momentum equations tends to destabilize the solution. A number of stability enhancing measures are taken to prevent this destabilization.

The streamline geometry is updated only during intermittent outer loop iteration, so the flow fields have a chance to stabilize while adjusting to the current streamline. The tangential velocities and body forces are calculated every iteration so that the flow remains on the streamlines. During the initial iterations of the solution, the tangential effects are estimated, but the meridional body forces are not immediately calculated. This allows a pressure field that balances the centrifugal force of fluid to develop. After a few iterations, the meridional body force is calculated and gradually introduced into the momentum equations. This permits a more stable evolution of the pressure, which will typically increase significantly through the blade row. Fourth order polynomial smoothing is used on the meridional body forces to avoid an unrealistic blade loading fluctuations which can cause the solution to diverge. The smoothing is applied row-by-row and then column-by-column. Following the smoothing process, the body forces are then scaled so that the total loading on the blade is not altered.

Chapter 7

Code Validation

Once the flow solver was written, it was thoroughly checked to insure that it correctly solves the momentum and continuity equations. The code was written in modular form so that most of the FORTRAN subroutines could be independently checked.

The first step in the solution algorithm is to map the grid from the physical space into the computational plane. This entails computing the metrics and the Jacobian of the transformation. The block of code responsible for carrying out these calculations was checked by comparing the results with hand calculations to verify that it works properly.

Both facial and nodal metrics are required. The quantities for the interior nodes are found using second-order accurate central differencing of the vertical face midpoints (denoted by "+" in Fig. 4.4) for the " ξ " gradients and of the horizontal face midpoints (denoted by "x" in Fig. 4.4) for the " η " gradients. This keeps the mapping on as local a scale as possible. Any discontinuities in the gridlines, such as those that appear in the streamwise lines in Fig. 4.4, are resolved by this treatment of the mapping. These discontinuities appear at the control volume faces, rather than at the nodes, because the grid was generated in terms of the CV corners. Quadrilateral control volumes are formed by preventing the gridlines from changing direction at the midface. Nodal coordinates are used to calculate z_ξ and r_ξ at the vertical faces and z_η and r_η at the horizontal faces. The control volume corners are used to compute the remaining facial metrics. The nodes are collocated with the facial coordinates at the boundaries, and there the metrics are evaluated using second-order accurate one-sided differencing.

Because the governing equations are solved in the computational plane, gradients with respect to ξ and η frequently appear. These gradients refer to the changes in the nodal quantities (v_x, v_r, v_θ , and p), where the changes are evaluated at the nodes (O, E, W, N, S), at the vertical faces (e, w), or at the horizontal faces (n, s). A number of gradient subroutines were written to cover the various permutations. These subroutines were checked by verifying that they returned the correct gradients on a linearly varying field.

The nonlinear system of equations that governs the fluid motion are solved iteratively as sets of linear equations. The discretized momentum and pressure corrector equations (Eqs. 4.12-14,40) each represent a system of linear equations. Four algorithms are employed to solve these sets of equations. One solver sweeps the field row by row using the common Thomas tridiagonal algorithm to solve a particular row. Another equation solver uses the same technique applied column by column. A subroutine that uses Stone's strongly implicit method [25] is available in the code. A direct, banded matrix solver is also available in the program to allow for a noniterative solution to the pressure corrector equation. This banded matrix solver is based on lower/upper matrix decomposition and was coded explicitly for pentadiagonal systems. All four procedures can be checked independently of the means by which their coefficients are generated. The equation residuals must go to zero for each linear system. Zero residuals indicate that the algorithms were properly coded.

The residual is a measure of the amount by which a field fails to satisfy the linear equations. When the solution field, ϕ , is found, the residuals are zero everywhere in the field. The residual of Eq. 4.10 is given below.

$$R = A_O\phi_O - (A_E\phi_E + A_W\phi_W + A_N\phi_N + A_S\phi_S + S_U) \quad (7.1)$$

There is a residual for each equation in the linear system. A number of ways of utilizing the residuals of the individual equations to measure overall convergence can be used. The simplest one, which is used in this work, is to monitor convergence in terms of the largest absolute value

of all of the residuals in the field. When the absolute values of R for the mass and momentum equations all drop below a specified value, the flow is considered to be solved. If the convergence limit is set low enough, then in addition to eliminating the residuals, the flow fields will stop evolving. The mass source residual is scaled by the mass flux into the domain and the momentum residuals are normalized by the incoming momentum. The normalization makes the convergence criteria independent of geometry and flow conditions.

Verifying the proper working of the pressure correction scheme can be done independently of the discretized momentum equations. The pressure corrector, p' , can be found directly using the banded matrix solver. When this solver is employed and the necessary corrections to the facial velocities are made, the mass sources must completely vanish regardless of the velocity field returned by the momentum equations. This is expected since the p' equations were formulated with the purpose of eliminating mass sources, and the direct solver assures that the set of equations are solved to a high level of accuracy.

Keeping in mind the validation procedures already discussed, there are only two primary sections left to check in the basic flow solver: the set of momentum equation solutions and the functioning of the entire code as a whole. Unfortunately, there is no easy way to check the conservation of momentum. Therefore, the code as a whole and momentum conservation are verified simultaneously.

Two approaches are possible for code validation. First, an exact solution in a known geometry may be input into the code to see if it immediately recognizes the solution as the converged flow solution, or the flow may be solved on a specified geometry starting with an arbitrary initial guess. A comparison may then be made between the calculated and known solutions. The latter method was used, because it demonstrates that the converged solution is independent of the initial guess.

Such phenomena as false diffusion, caused by the multi-dimensionality of the flow and skewness of the fluid velocity with respect to the gridlines, Taylor series truncation, arising in numerous stages of the discretization process, and floating point errors will contaminate the solution to

varying degrees. Current theory cannot determine the amount of error *a priori*; however, experience and simple numerical experiments are helpful in determining when a solution is sufficiently accurate. Floating point errors can never be completely eliminated, because digital computers store numerical values to a limited number of significant figures. However, the iterative nature of PWIM renders this type of error virtually negligible, as the errors are not compounded from one iteration to the next. Properly choosing test case geometry and grid specification make some or all of the remaining errors negligible. The following sections discuss the specific flow cases that were used to validate the code.

7.1 Flow in a Straight Pipe

Fully developed flow in an axisymmetric pipe or rectangular channel can be solved analytically. The resulting velocity is parallel to the walls, reaching a maximum at the centerline and decreasing to zero as a parabolic function of the distance from the center. The pressure field, as is always the case in incompressible flow where there is no equation of state to relate the thermodynamic variables, is unique to within an additive constant. The streamlines are parallel to the pipe, and the pressure is constant across the cross-section. A constant axial pressure gradient is set up within the pipe to overcome the viscous effects. The dimensionless pressure drop is dependent on the Reynolds number, as shown in Eq. 7.2.

$$\frac{\partial p}{\partial z} = -32/Re, \quad \text{for a circular pipe} \quad (7.2a)$$

$$\frac{\partial p}{\partial x} = -12/Re, \quad \text{for a 2-D channel} \quad (7.2b)$$

The characteristic velocity and length in the Reynolds number are the average velocity and pipe diameter (channel height), respectively.

A variety of trials using this geometry were run so the solution returned by the code could be checked against analytic results.

7.1.1 Uniform Grid

A uniform orthogonal grid was generated to cover the pipe, so that the performance of the code in the absence of numerical diffusion and truncation errors could be tested. When a parabolic, or fully developed, velocity profile is specified at the inlet, aligning the gridlines with the pipe walls eliminates the effects of numerical diffusion, because the flow is always parallel to the gridlines. Using a uniform physical space grid removes the truncation errors, because flow fields are polynomials of no higher than second-order, and the discretization process was carried out to second-order accuracy. For this type of grid, the spacing is constant, and the mapping operation is a scaling of the grid in the axial and radial directions so as to make the spacing unity. When a parabolic inlet profile was used, the resulting flow was the inlet profile convected unchanged all the way through the pipe and a constant pressure drop equal to the value given by Eq. 7.2 resulted. This was tested over a wide range of Reynolds numbers for both the cylindrical and Cartesian coordinate systems. When a uniform inlet velocity profile is used, the velocity profile evolves toward a parabolic shape, and the pressure gradient approaches the theoretical value. Fig. 7.1 shows a plot of the calculated velocity vectors in the neighborhood of the inlet for an entrance flow at $Re = 100$. The lower boundary is the plane of symmetry ($r = 0$) and the upper boundary is a no-slip wall. This convention is used throughout this work, unless otherwise stated. The radial velocity near the inlet is due to the sudden appearance of the wall. The boundary layer begins to form in the first column of control volumes, so a radial velocity is needed to remove the mass from the region of slower flow. By the time the exit is reached, a parabolic profile has formed. Fig. 7.2 shows the pressure drop along the axial line at the center

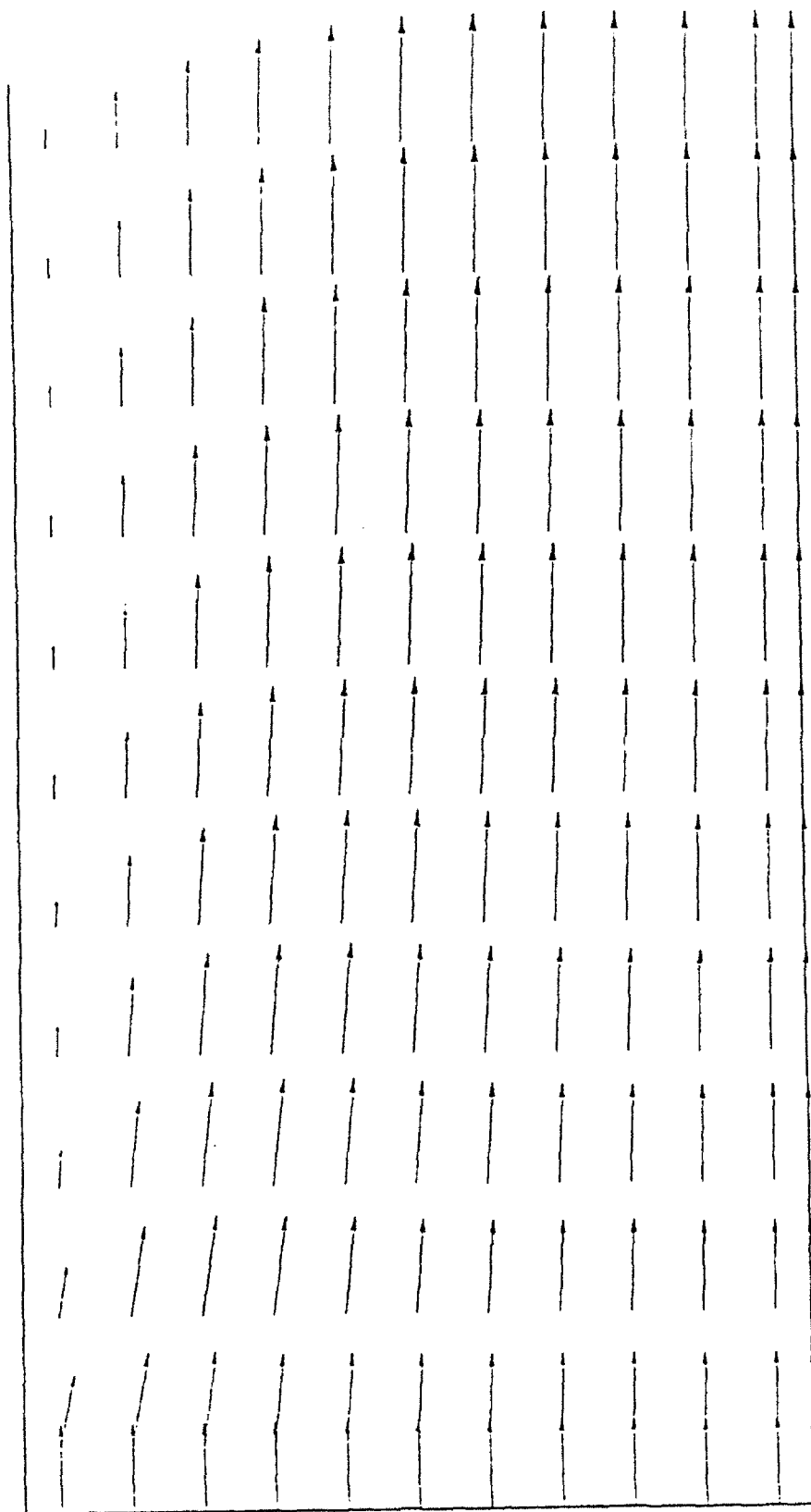


Figure 7.1
Pipe Flow Velocity Field in the Inlet Region (Uniform Inlet Profile, Uniform Grid)

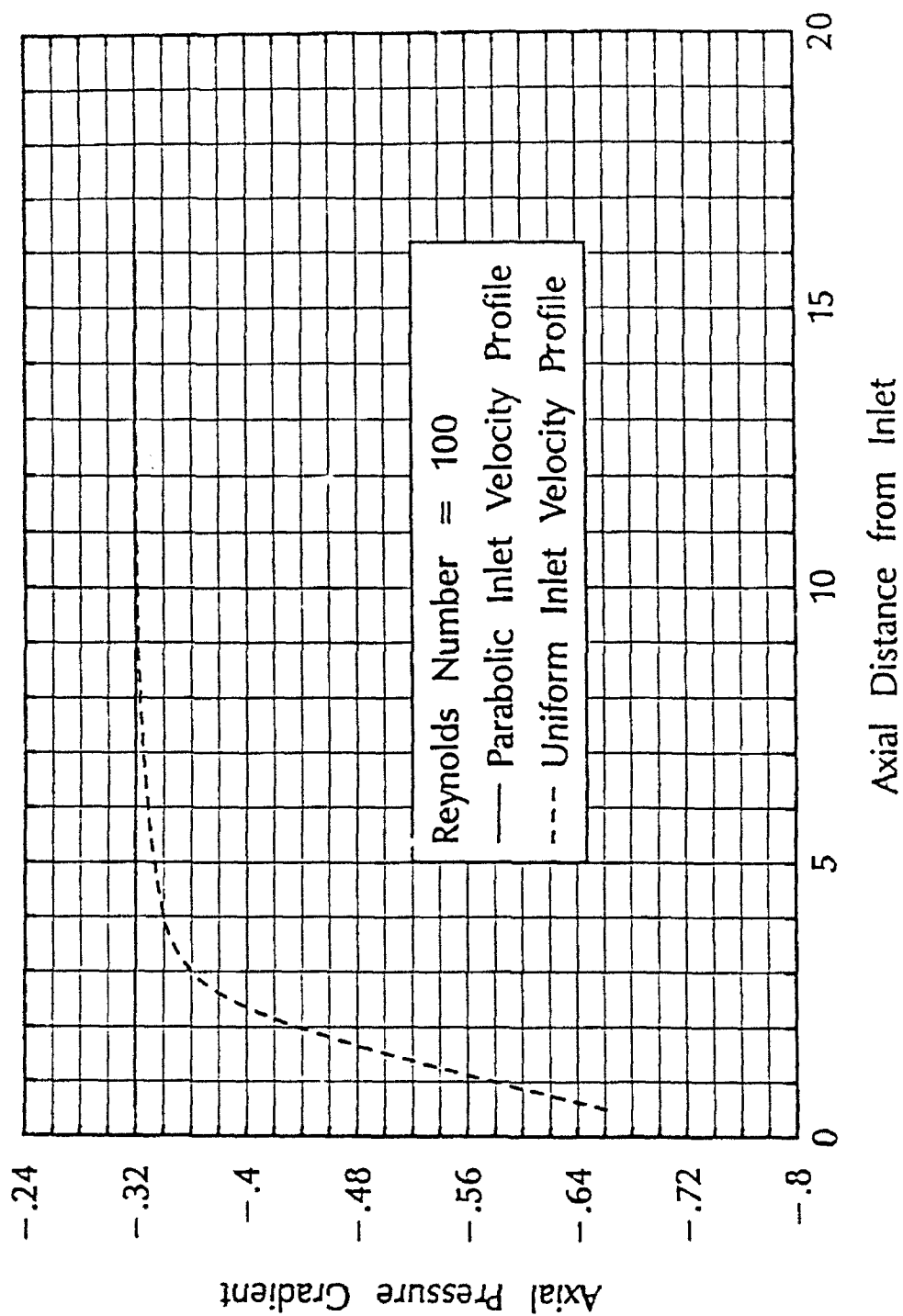


Figure 7.2

Effect of Inlet Profile on Axial Pressure Gradient (Uniform Grid)

of the physical grid ($r = 0.25$) for both parabolic and constant inlet velocity profiles. The profiles correspond to equal mass flows into the pipe, hence the pressure drops eventually attain the fully developed value. Flow from which these curves were taken is for a pipe with a nondimensional outer radius of 0.5 and a length of 5.5. The grid was a very coarse 9×7 . Despite the coarseness of the grid the analytical pressure gradient was attained, due primarily to the simplicity of the flow.

Calculating pipe flow in a uniform grid does not in itself validate the accuracy of the code, but it does demonstrate that nonorthogonal terms arising from the mapping properly vanish when the gridlines meet at right angles. This geometry also shows that all four types of boundary conditions are accurately handled. Velocity is specified at the inlet, a plane of symmetry exists at the lower boundary, the upper surface is a no-slip wall, and the flow is assumed to be constant in the axial direction and normal to the boundary at the outlet.

7.1.2 Nonuniform Grids and Turbulence Model Test

The transformed equations are simplified if the grid is uniform. To determine if the program can properly handle a grid with nonuniform spacing, flow through a grid with control volumes of alternating widths and/or heights was studied. Four test cases were run. The first test case is for flow on a uniform grid. This case serves as a baseline for the numerical experiment. The object of the experiment was to see how changing grid spacing affects the results; therefore, a standard is needed from which the comparisons can be made. The control is further maintained in each of the four trials, by using a fully developed inlet velocity profile to prevent the effects of grid spacing from being confused with those of developing flow. In the second case, the widths of the control volumes varied such that a 2, 1, 2, 1 . . . alternating pattern of neighboring CV length ratios was formed. Height, rather than width, was varied in the third trial. Both CV height and

width varied in a repeating pattern in the fourth case. Fig. 7.3 shows the grid for the last case. Because a ratio of 2 was chosen for alternate CV thicknesses, the nodes are uniformly spaced. Only the distance between the control volume faces varies.

Fig. 7.4 shows the plot of the axial gradients at the center of the grid ($r = 0.25$) for each of the four cases. The pressure gradient for the uniform grid is equal to the analytic value. When the axial spacings vary, the pressure gradient is oscillatory. This is caused by the cyclic behavior of the metrics, primarily z_ξ and J . If a plot of these functions was made as a function of the axial distance from the inlet, the curve would look similar to that of the pressure gradient curve. Central differencing is used to evaluate these metrics at the CV faces, and the Taylor series expansion that yields their second-order accurate finite difference expression is shown in Eq. 7.3.

$$\frac{\partial \phi}{\partial \xi} = \frac{\phi_E - \phi_O}{\Delta \xi} + \left(\frac{1}{3!} \left(\frac{\Delta \xi}{2} \right)^2 \frac{\partial^3 \phi}{\partial \xi^3} + \frac{1}{5!} \left(\frac{\Delta \xi}{2} \right)^4 \frac{\partial^5 \phi}{\partial \xi^5} + \dots \right) \quad (7.3)$$

For a smooth function, the higher-order derivatives are small, and the second-order central differencing approximation, which neglects the quantity in the parentheses, is accurate. However, when the widths vary, the axial coordinate is not a smooth function of ξ , nor is r a smooth function of η when the heights vary; therefore, the terms in the parentheses are no longer negligible and the truncation error contaminates the results.

A variation in CV height does not cause oscillations in the the pressure gradients, because pressure is constant across the pipe. However, it does cause the calculated pressure drop to rise slightly. This is due to the use of central differencing in the diffusive fluxes. Because the horizontal control volume faces are not located midway between the nodes, central differencing is not an accurate approximation for the diffusive gradient of v_z with respect to r .

Fig. 7.5 shows the pressure as a function of the axial location for each for the four test cases. None of the trials have an oscillatory pressure field, because the nodes are uniformly spaced even though the control volumes change in size. The pressure is very close to the analytical distribution for all four cases. This demonstrates that the solution error is caused by truncation,

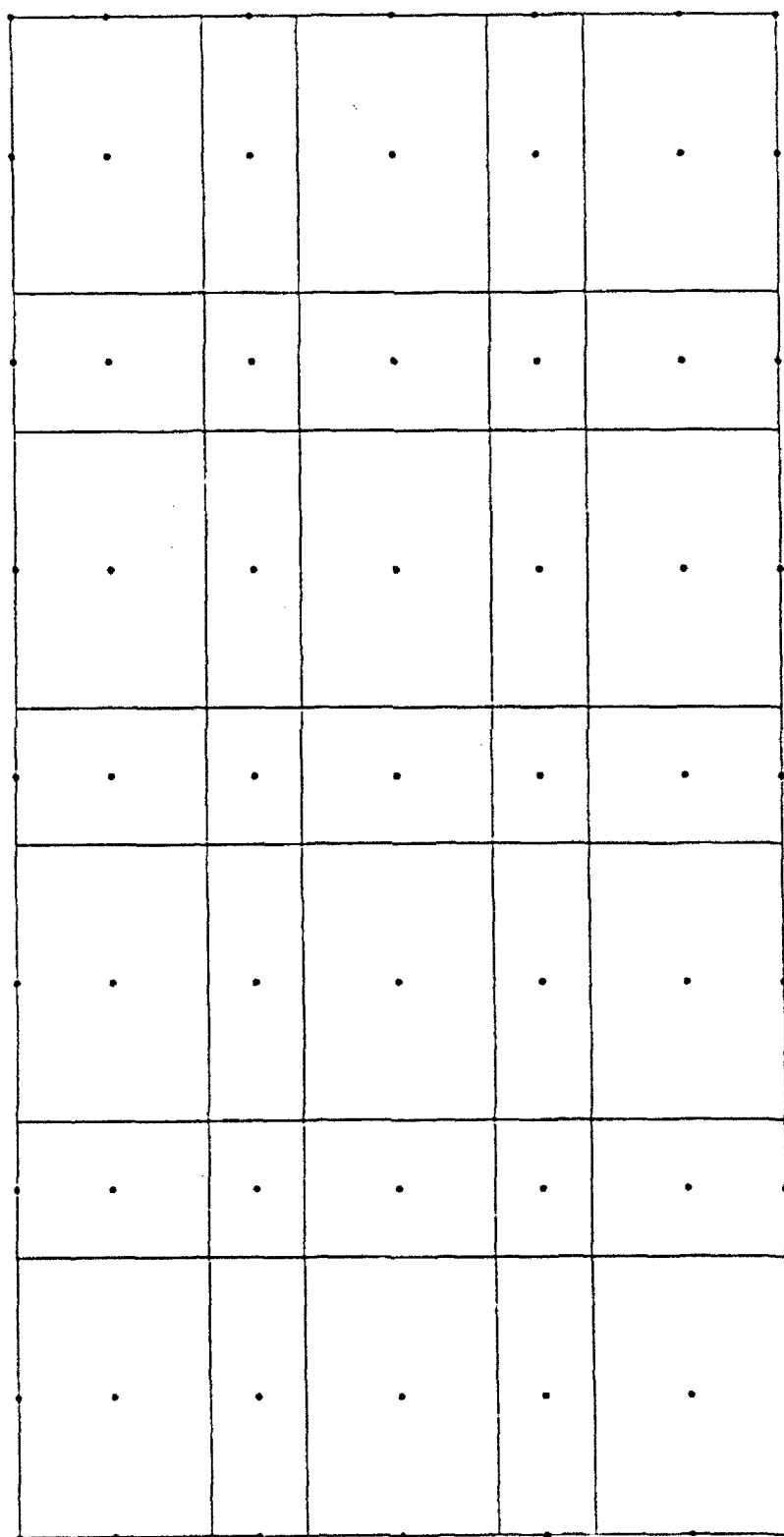


Figure 7.3
Nonuniform Grid, Δz and Δr Changing

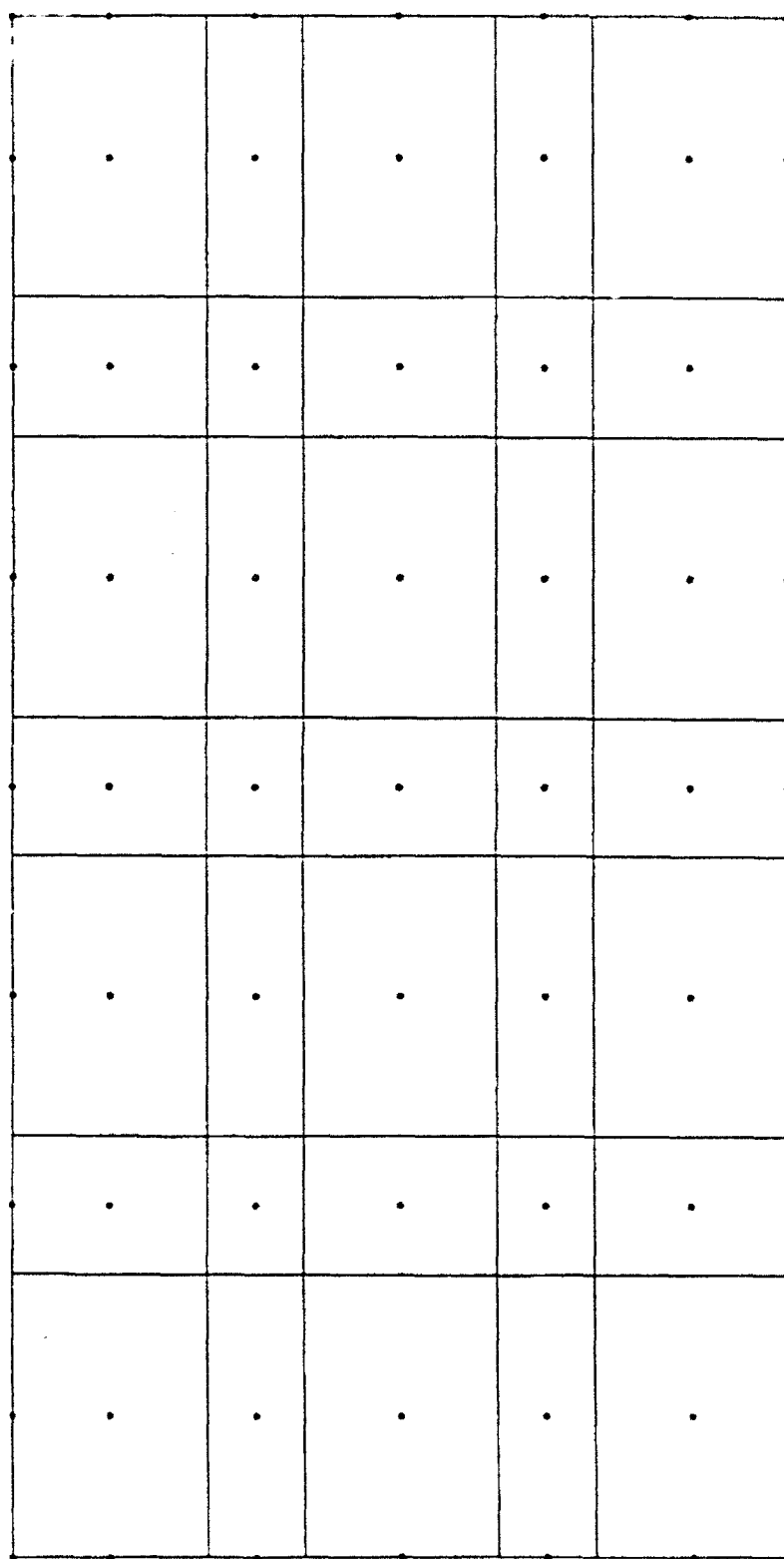


Figure 7.3
Nonuniform Grid, Δz and Δr Changing

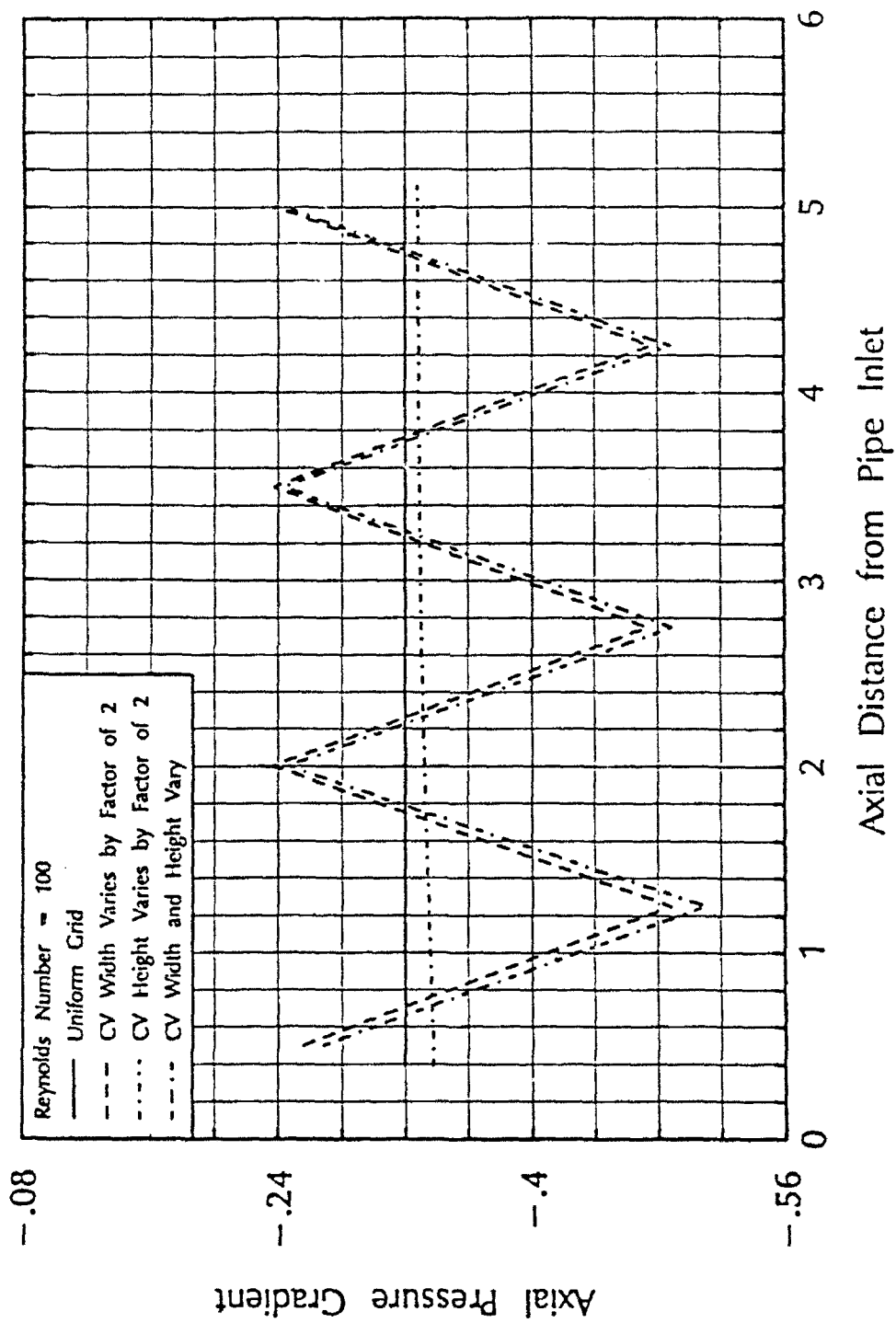


Figure 7.4
Effect of Grid Nonuniformity on Axial Pressure Gradient

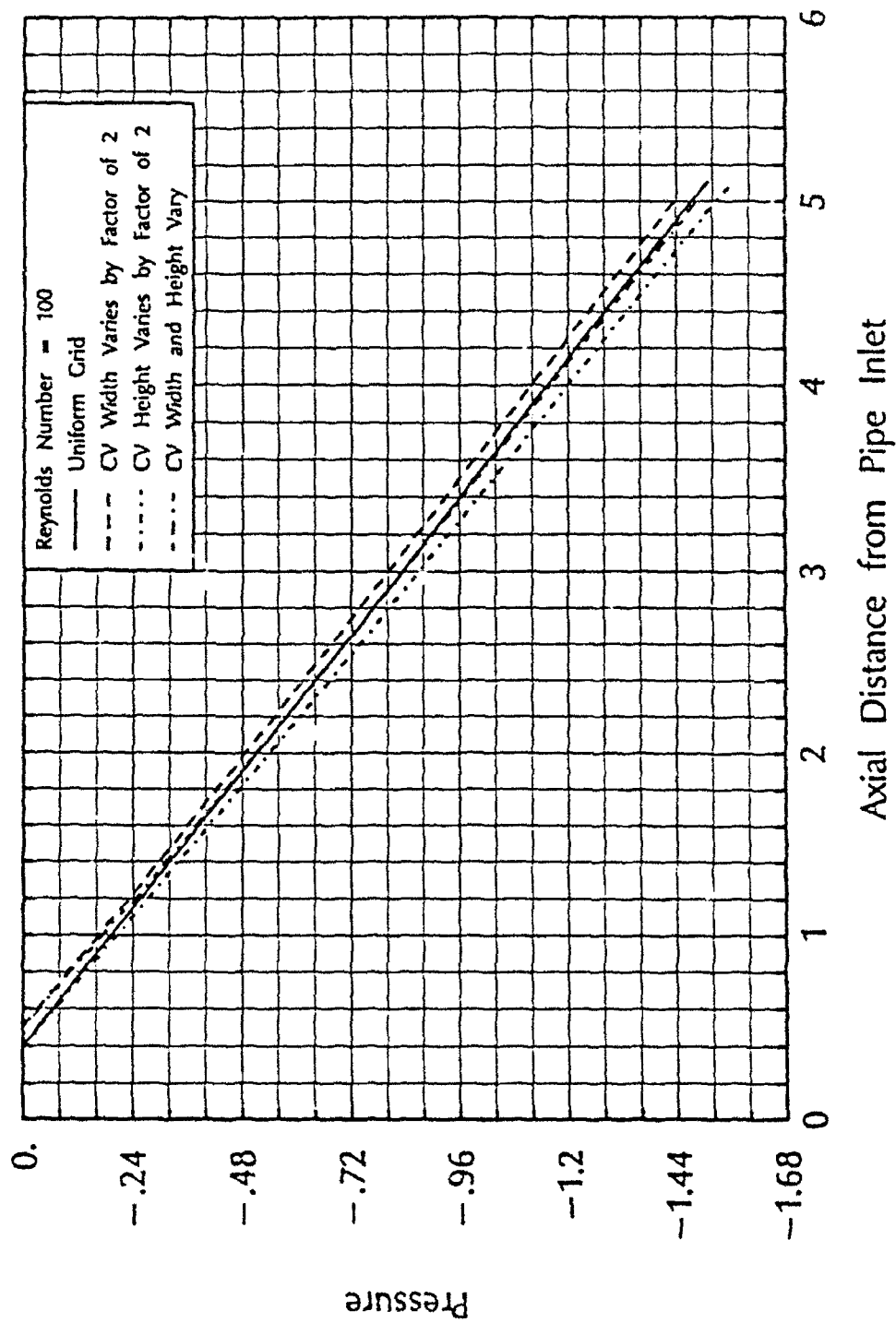


Figure 7.5
Effect of Grid Nonuniformity on Pressure

not programming error, and the test illustrates how the calculated results can be distorted if the control volume sizes vary too suddenly.

Pipe flow was also examined to check the accuracy of the Baldwin-Lomax turbulence model [29]. Trial cases were run on grids with 25 nodes across the pipe. The nodes were clustered using the technique described in Chapter 5, which upon inspection of the converged solution was seen to yield accurate estimations of y^+ at the interior layer nodes. Five nodes were placed within the inner layer.

The calculated fully developed outlet velocity profiles for laminar flow and for turbulent flow at three different Reynolds numbers are shown in Fig. 7.6. Also shown, are experimental data for the same flow conditions [17, p. 598]. As seen by the resolution of the calculated velocity profiles, a sufficient number of nodes was used for the calculations. The calculated laminar velocity profile shows that the solver is able to handle highly stretched grids. The turbulent calculations show a tendency to underestimate the velocity near the wall. The errors for the high Reynolds number cases are within acceptable limits. However, the $Re = 4000$ case shows appreciable discrepancies between the calculated and experimental results. This does not imply that there is a problem with the way the turbulence model was implemented. Considering that Baldwin and Lomax only published results for $1.33 \times 10^6 \leq Re \leq 21 \times 10^6$, it is logical to assume that the assumptions that led to the model, such as the premise that vorticity distribution is an indication of the boundary layer length scale, are invalid at Reynolds numbers near the point of transition from laminar to turbulent flow. This does not present a severe limitation as far as turbomachinery calculations go, because turbomachine Reynolds numbers are more on the order of 1×10^6 , not 1000.

It is possible that the constants used in the turbulence model for axisymmetric flow should not be equal to those used for Cartesian flows. However, in view of the ability of the model as it stands to predict the effects of turbulence, a decision was made to forego surveying the effect that changing the values of one or more parameters would have on the velocity profile. Such a task

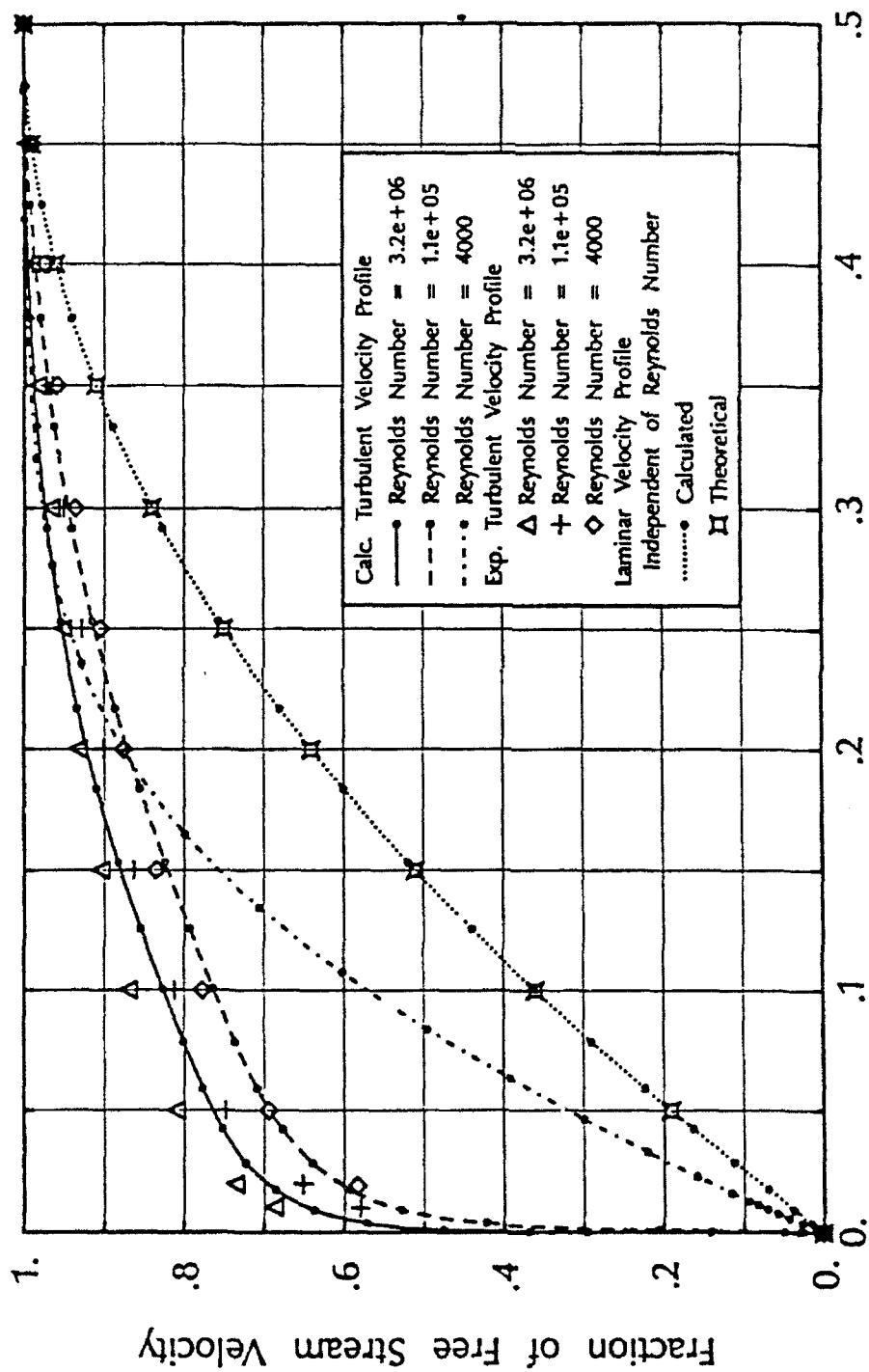


Figure 7.6
Comparison of Experimental and Calculated Axial Velocity Profiles
for Turbulent Flow in a Pipe

would require a tremendous amount of time, and there is no evidence that the modifications could be applied to yield more accurate results across a wide range of geometries and flow conditions.

7.1.3 Skewed Gridlines and Differencing Scheme Comparisons

It has been shown that the flow solver can treat orthogonal grids. The next step in the validation procedure is to examine how nonorthogonal grids are treated. Nonorthogonal grids can result from skewing either or both sets of gridlines. To help place a control on the numerical experiment, the nonorthogonality is examined by skewing one family of gridlines at a time. In the first series of tests, the ξ gridlines are skewed with respect to the flow direction. To eliminate truncation errors caused by nonuniform spacing near the boundaries, the edges of the flow domain are also skewed. Skewing the gridlines with respect to the flow direction sets up the possibility that the calculated results may be contaminated by numerical diffusion. This phenomenon is discussed in more detail in the following text.

Because the boundary lines are skewed with respect to the flow direction, those boundaries no longer correspond to the limits of the pipe, so new boundary conditions are necessary. The grid and geometry chosen to act as the test case is shown in Fig. 7.7. Its nondimensional length is 2.0. The lower corner of the inlet is located at the centerline, and the upper corner is placed at $r = 0.25$, midway to the pipe wall. The top of the outlet boundary is on the pipe wall ($r = 0.5$) and the bottom is at $r = 0.25$. Because pipe flow is axial, mass passes through the upper and lower boundaries, a different phenomenon than that which occurs in the previously used grids. The velocity is a known analytic function of the radius, so the BC is a specified velocity at all four boundaries. Fig. 7.8 shows a plot of the velocity vectors for a converged solution. Notice that each profile is parabolic, but the extrema change in response to the radial location of the

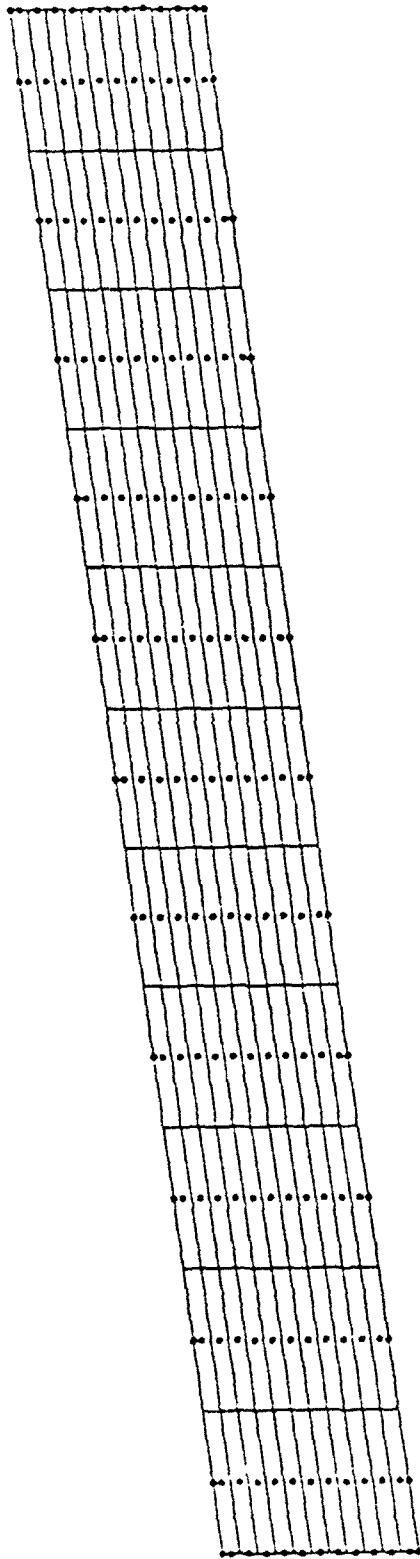


Figure 7.7
Uniformly Spaced Grid with Skewed ξ Gridlines

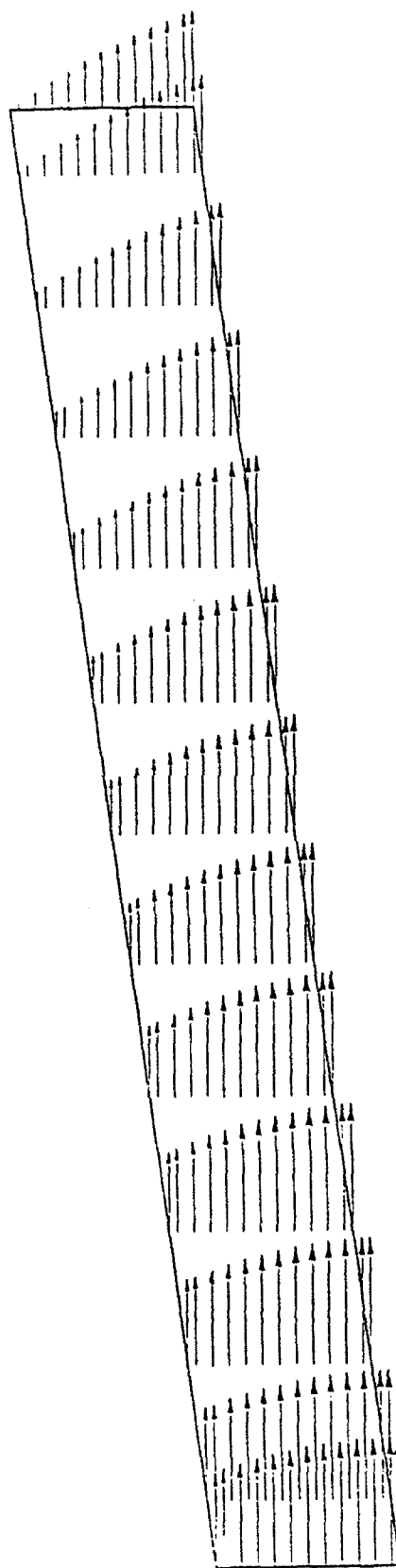


Figure 7.8
Pipe Flow Velocity Field on Uniformly Spaced Grid with Skewed ξ Gridlines

nodes. Runs were made on this geometry for a number of different nodal concentrations and using various differencing schemes.

For fully developed pipe flow the velocity profile is essentially determined by continuity, thus the behavior of the pressure field provides more insight into the workings of the momentum equations. To the eye, the velocity profiles for the various test cases are indistinguishable from one another. However, the pressure field is strongly dependent on both the differencing scheme and the number of nodes that make up the grid.

Fig. 7.9 shows a plot of the calculated pressure gradients on the skewed grid for three different grid densities. The solid line is the control case. It was calculated using the same specified velocity boundaries, except that the grid for this case extended from the centerline to the pipe wall at the inlet and outlet, removing the skew and accompanying numerical diffusion. The pressure drop for the control case is equal to the analytic value throughout the pipe. On the skewed grids, the pressure drop can no longer refer to the gradient at a particular radial location, so the curves on the graph refer to the value at the central node of each axial station. Hybrid differencing was used for these calculations, and for a cell Reynolds number greater than 2, this scheme acts like pure upwind differencing, which is very prone to numerical diffusion. Numerical diffusion causes the curves to deviate from the constant pressure drop. The error is lessened when the number of nodes in the streamwise direction is increased. Adding nodes in the cross-stream direction does not decrease the numerical diffusion.

The departure of the pressure drop from the analytic value is the greatest near the inlet and outlet, because there is a discontinuity in the accuracy of the discretized equations in these regions. The known velocity boundary conditions are specified with second-order accuracy, but the differencing schemes are only first-order accurate. This sudden change in accuracy results in larger errors in the results near the inlet and exit boundaries.

Fig. 7.10 shows a comparison of the calculated pressure gradients when hybrid differencing, QUICK differencing with hybrid about the boundaries, and the fully higher-order QUICKR

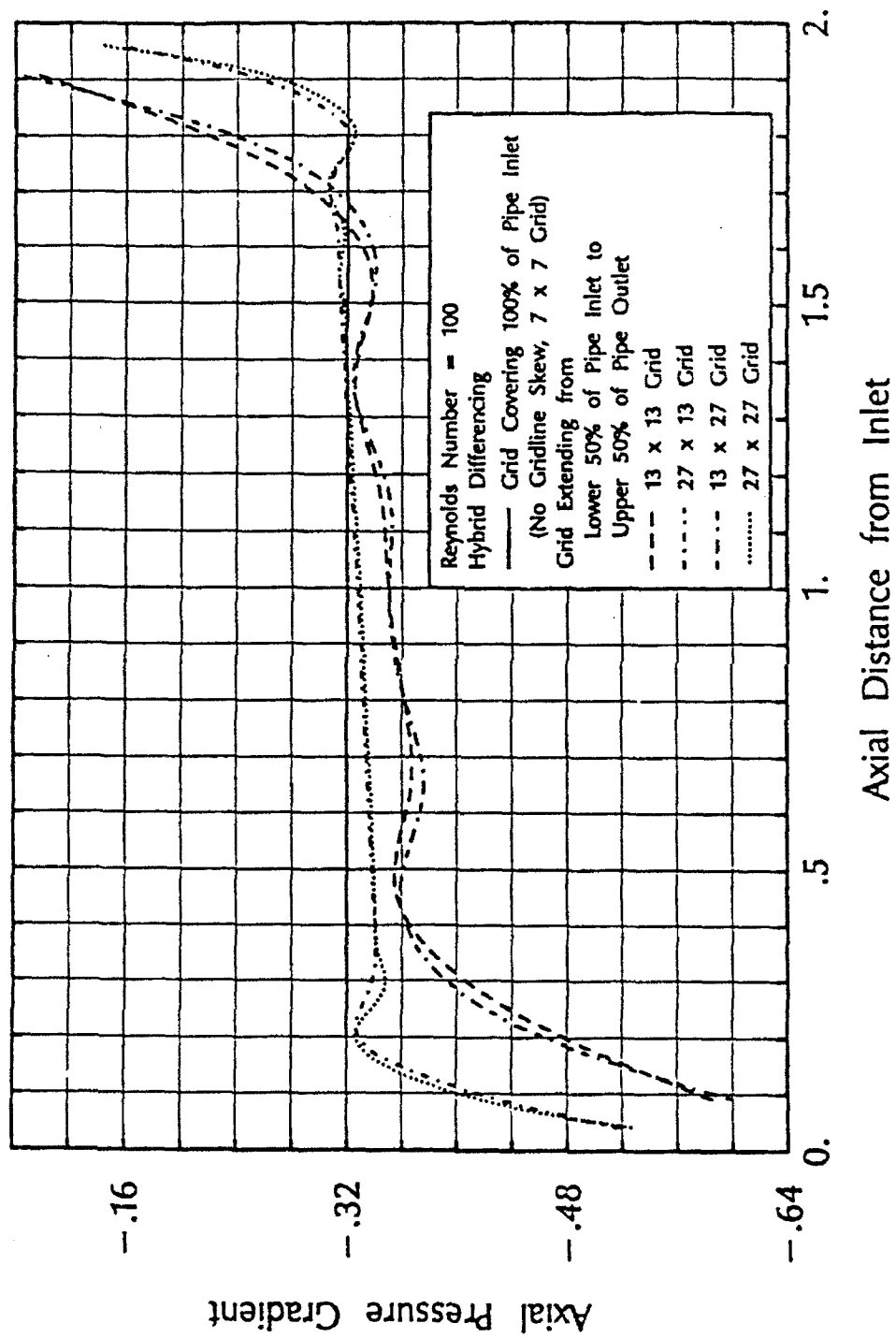


Figure 7.9

Effect of ξ Gridline Skew on Axial Pressure Gradient for Various Nodal Densities (Hybrid Differencing)

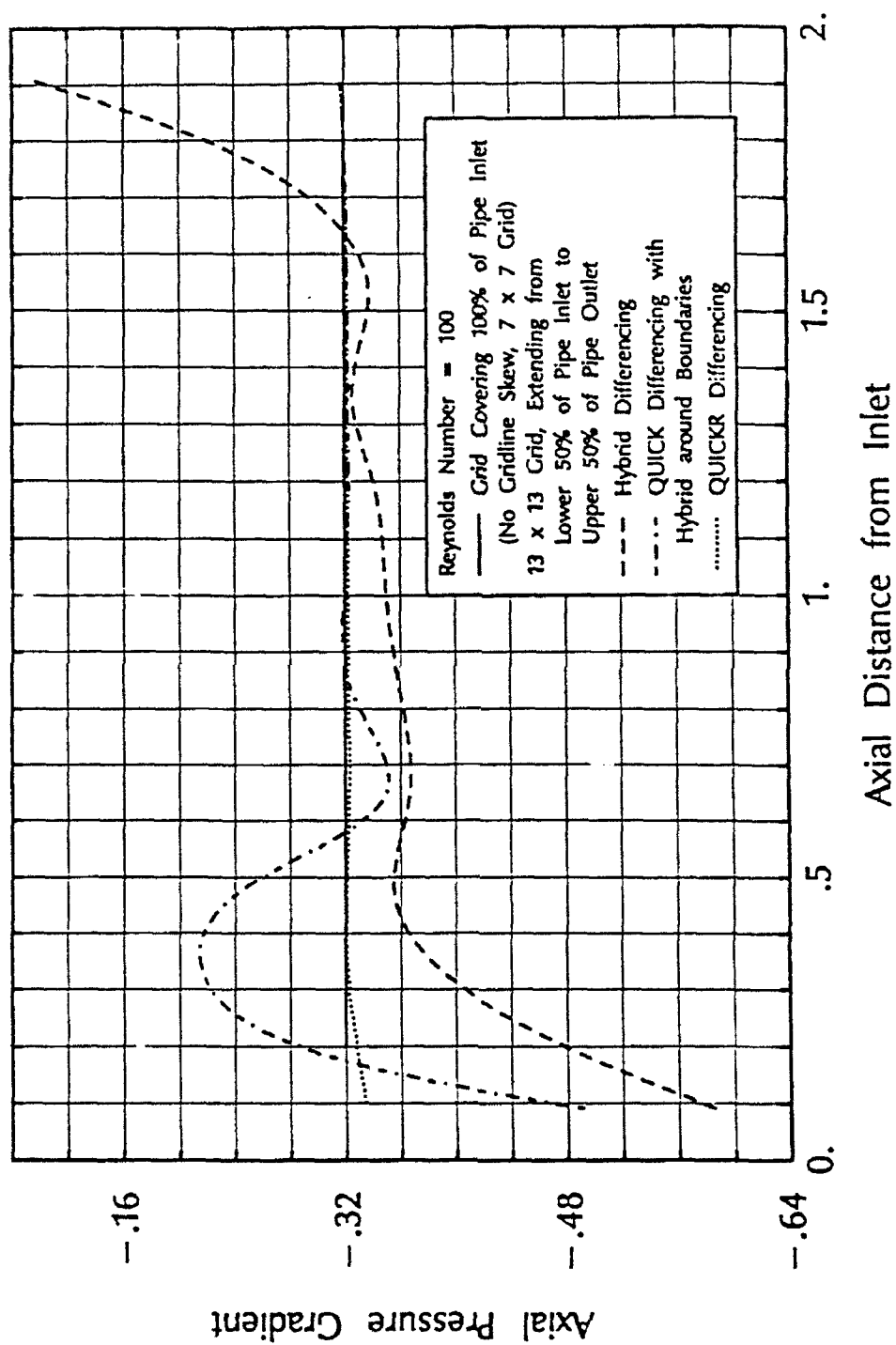


Figure 7.10

Effect of ξ Gridline Skew on Axial Pressure Gradient
for Various Differencing Schemes

differencing scheme are used. The sudden change in accuracy present when QUICK and hybrid differencing are used in conjunction causes a large perturbation in the results. The location of the perturbation lead to the initial conclusion that it was caused by an incorrect application of the BC's or an improper melding of the two differencing schemes. However, the same incongruity appeared in a one-dimensional hand calculation using the dual differencing scheme. Whenever a lower-order differencing scheme is used near the boundary, the relatively inaccurate differencing approximation is applied in the neighborhood of the precisely defined BC, resulting in a large discrepancy between analytic and calculated pressure drops. Notice how this error is not present at the outlet on the QUICK/hybrid curve. It is removed, because the positive flow direction means that QUICK is used on the control volume face just upstream of the outlet boundary. Consistently applying the higher-order differencing via the QUICKR scheme removes the large error about both the inlet and outlet regions.

Various grid densities were examined using the QUICKR differencing scheme, and Fig. 7.11 shows the results compared to the those calculated using the lower-order differencing schemes on the finest grid. The results using QUICKR on the 27×27 grid show that the pressure drop is equal to the analytic value, demonstrating that the program correctly handles the nonorthogonal terms arising from a skew of the ξ gridlines. As hand calculations of inviscid flow have shown, QUICKR completely removes numerical diffusion from the pipe flow calculation and thus enables the ability of the code to handle nonorthogonal grids to be evaluated. Because QUICKR evaluates convective fluxes with third-order accuracy, approximating the facial velocity with a second-order polynomial, it accurately resolves the parabolic axial velocity distribution. The slight error in pressure drop visible on the QUICKR curves in Fig. 7.11 is due to a failure on the part of central differencing to capture the true viscous shear. QUICKR does not remove all errors from all flow conditions, but it is much more accurate than the lower-order schemes. The answer to the question of whether or not the added expense of QUICKR provides a sufficient payoff in terms of higher accuracy is dependent on the needs of the investigator. However, it should be noted

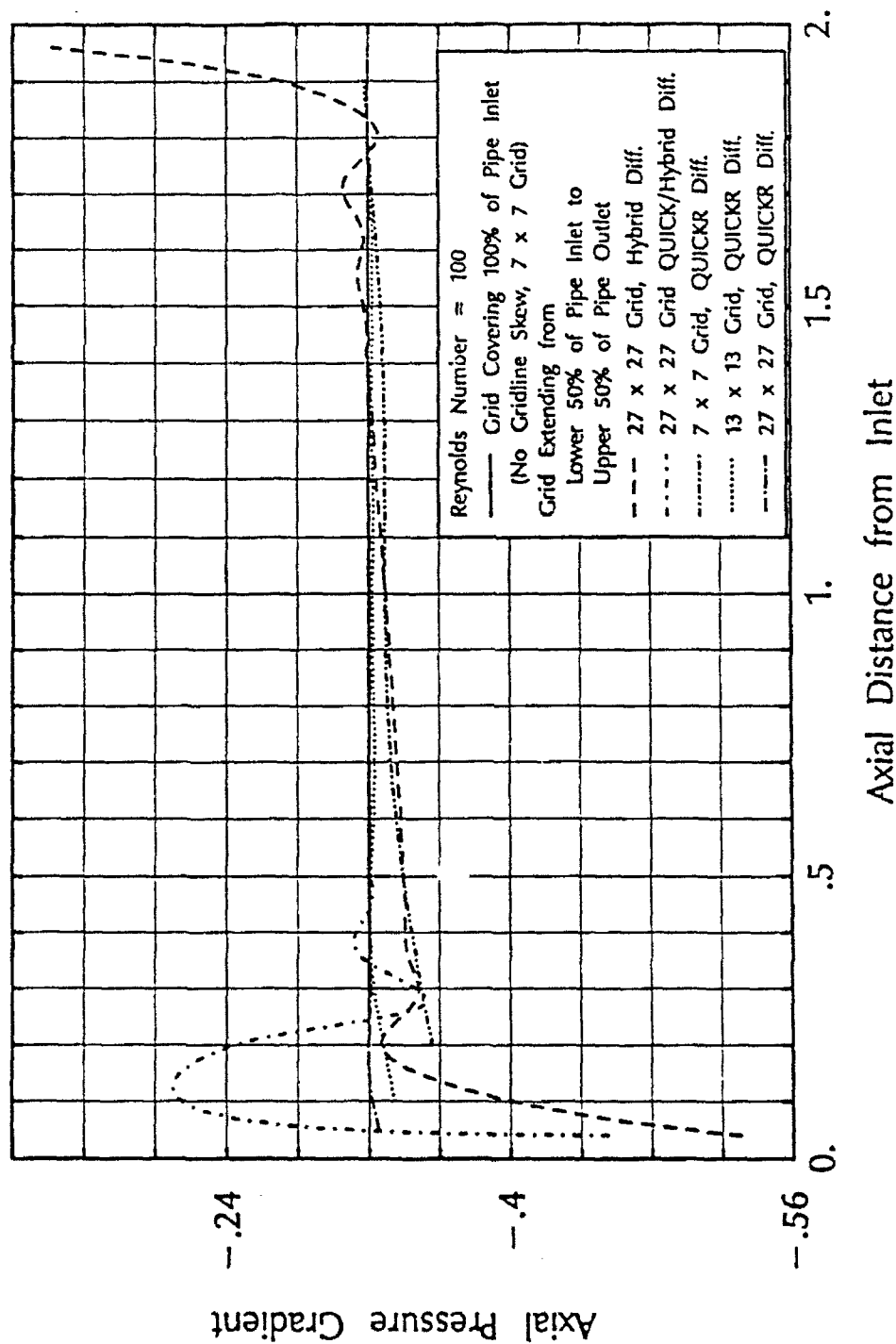


Figure 7.11

Comparison of Results from QUICKR (Various Grid Densities) and the Lower-Order Schemes (Finest Grid)

that the added accuracy could reduce the number of required grid nodes, and thereby more than make up for the expense incurred by the use of QUICKR. The results for the 7×7 grid using QUICKR are shown to be as good as those of the lower-order differencing schemes on the 27×27 grid in Fig. 7.11.

The type of geometry used to test the effects of streamwise gridline skew demonstrates the ability of the code to handle permeable boundaries. Specified boundary transpiration poses no additional problems to the flow solver, nor does it require any special treatment.

To complete the verification that nonorthogonal grids are handled correctly by the code, the η gridlines were also skewed, and then both families of gridlines were skewed. The inlet and outlet boundaries were rotated for the grids with η gridline skew. This kept the grid spacing uniform, as did rotating the upper and lower boundaries for the case of ξ gridline skew. The grid that was used to test dual gridline skew is shown in Fig. 7.12. The calculated pressure drop at the central nodes for each gridline skew permutation is given in Fig. 7.13. The skew for the η direction corresponds to the gridlines running 80° with respect to the horizontal. The ξ skew is 7.12° , the same as that used in the previously discussed trials. Because the grid is uniform and the flow is axial, the η skew does not introduce any errors into the flow. QUICKR eliminates numerical diffusion, and the only error caused by the ξ skew is due to the inability of central differencing to resolve the viscous gradients. The curves of Fig. 7.13 are further evidence that the flow solver is able to handle nonorthogonality and that the solution of the transformed conservation equations is correctly implemented.

7.1.4 Verification of Conservation of Angular Momentum

The previous section dealt only with flows that had an axial component to the velocity. Angular momentum did not enter into any of these flows. To validate the implementation of

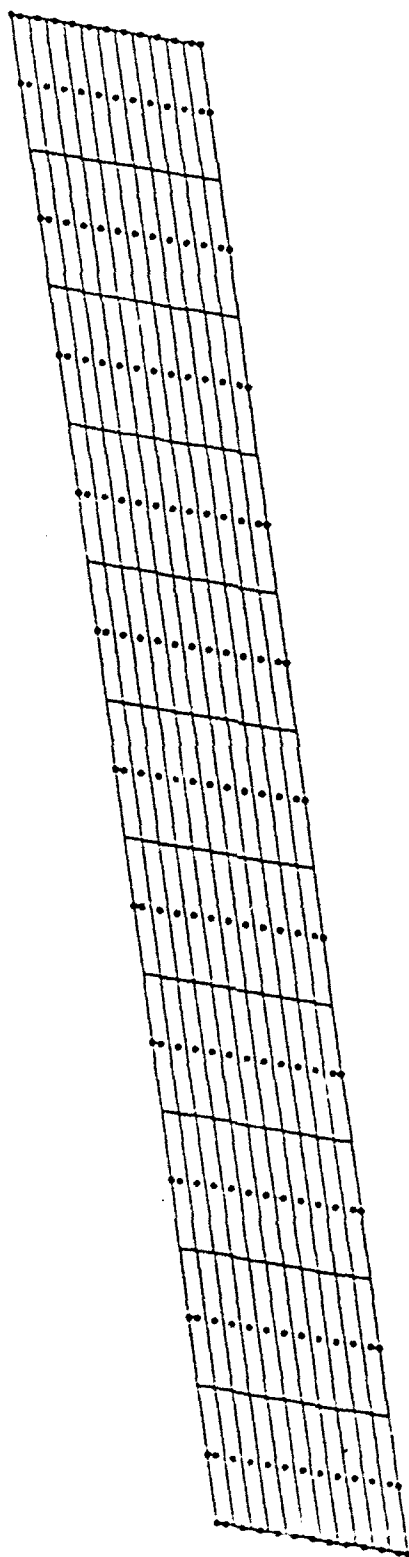


Figure 7.12
Uniformly Spaced Grid with Skewed ξ and η Gridlines

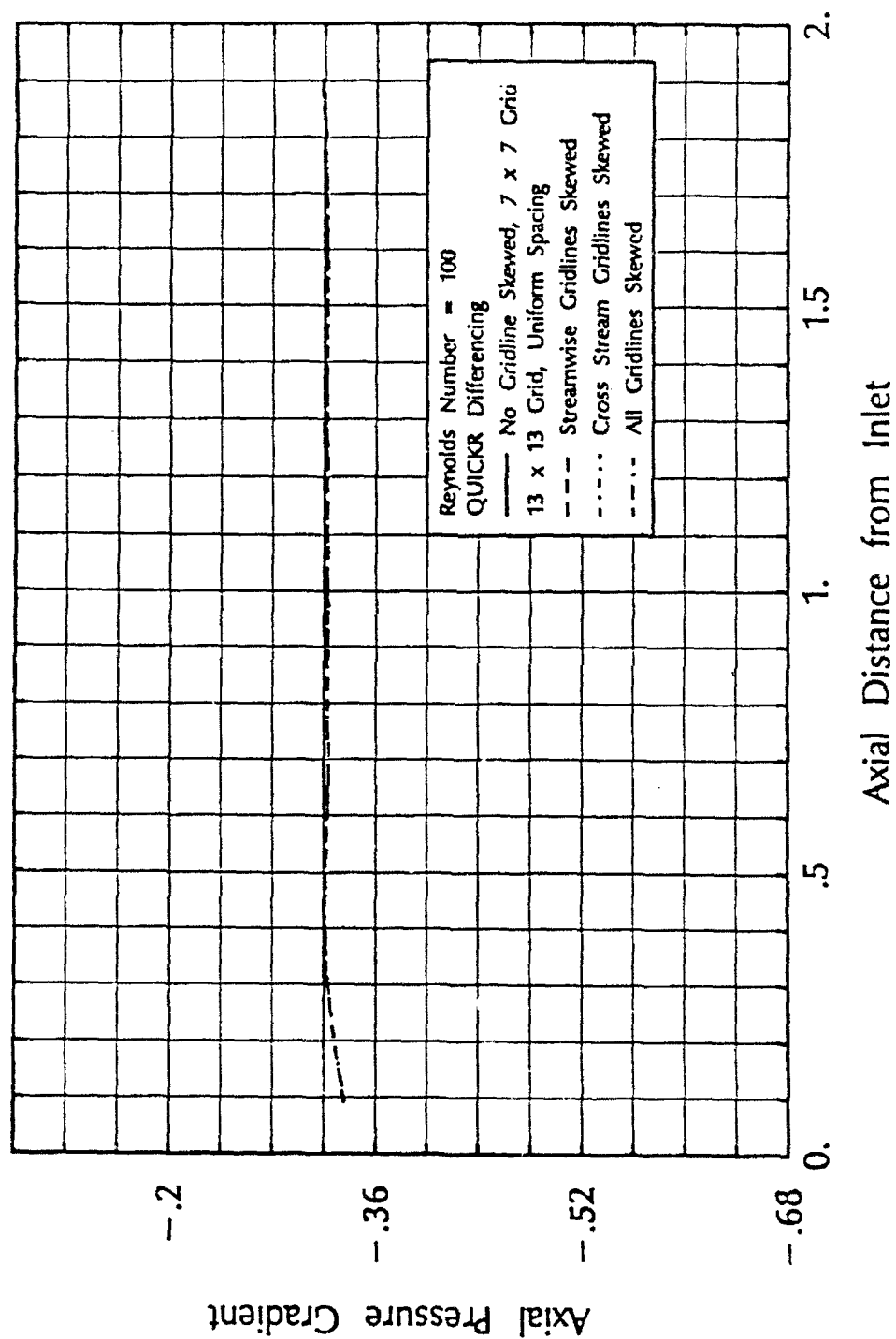


Figure 7.13

Effect of Gridline Skew on Axial Pressure Gradient (QUICKR Differencing)

the angular momentum equation, a test case analogous to the popular driven cavity is examined. Consider a pipe with capped ends, whose outer wall is set spinning about the centerline. Viscosity imparts swirl to the fluid, reaching the speed of the pipe at the wall and linearly dropping to zero at the centerline. Because the ends of the pipe are capped, there are no endwall effects. Axial and radial velocities are zero everywhere, and the conservation equations are satisfied if the pressure field meets the two conditions specified in Eq. 7.4.

$$\frac{\partial p}{\partial z} = 0 \quad (7.4a)$$

$$\frac{\partial p}{\partial r} = \frac{\rho v_\theta^2}{r} \quad (7.4b)$$

If the reference velocity is taken as the rotational speed of the wall, then the analytic solution to the flow is:

$$v_z = 0 \quad (7.5a)$$

$$v_r = 0 \quad (7.5b)$$

$$v_\theta = 2r \quad (7.5c)$$

$$\frac{\partial p}{\partial r} = 4r \quad (7.5e)$$

Unlike the other flows that have been examined, the steady state solution to shear-driven pipe flow is independent of Reynolds number.

Fig. 7.14 shows a plot of the resulting angular velocity when the flow was solved using the grid shown in Fig. 7.12. As expected, the v_θ profiles are a linear function of radius. The boundary velocities were all specified, because the analytic solution of the velocity field is a known function of the radius. Fig. 7.15 shows a plot of radial pressure gradient as a function of radius for the same grids that were used in the final series of tests in Section 7.1.3. The pressure gradients were evaluated along the column of nodes at the center of the grid. All test cases produce pressure gradients equal to the theoretical values. This is strong evidence supporting the assertion that

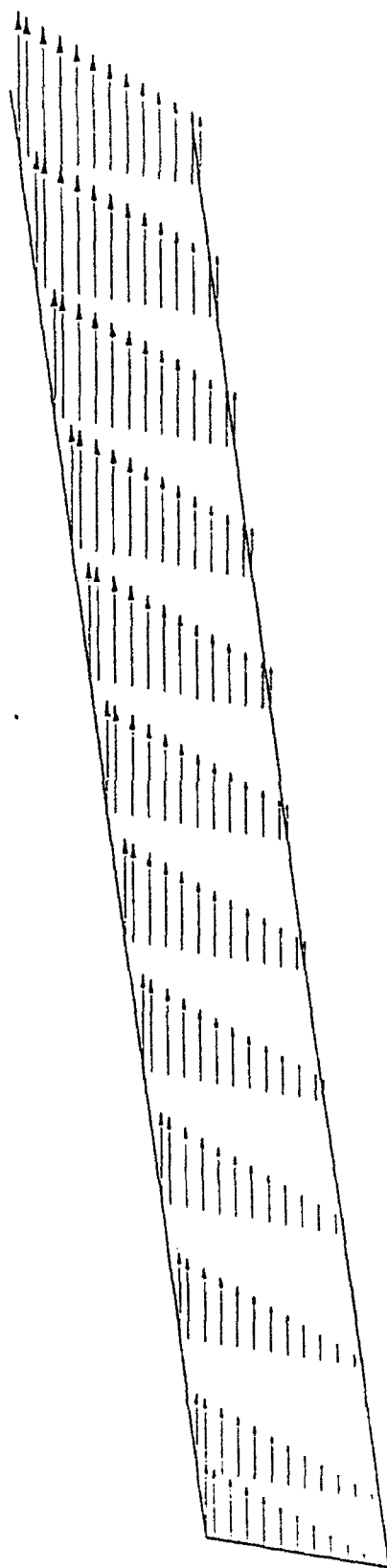


Figure 7.14
Shear Driven Pipe v_θ Field on a Grid with Skewed ξ and η Gridlines

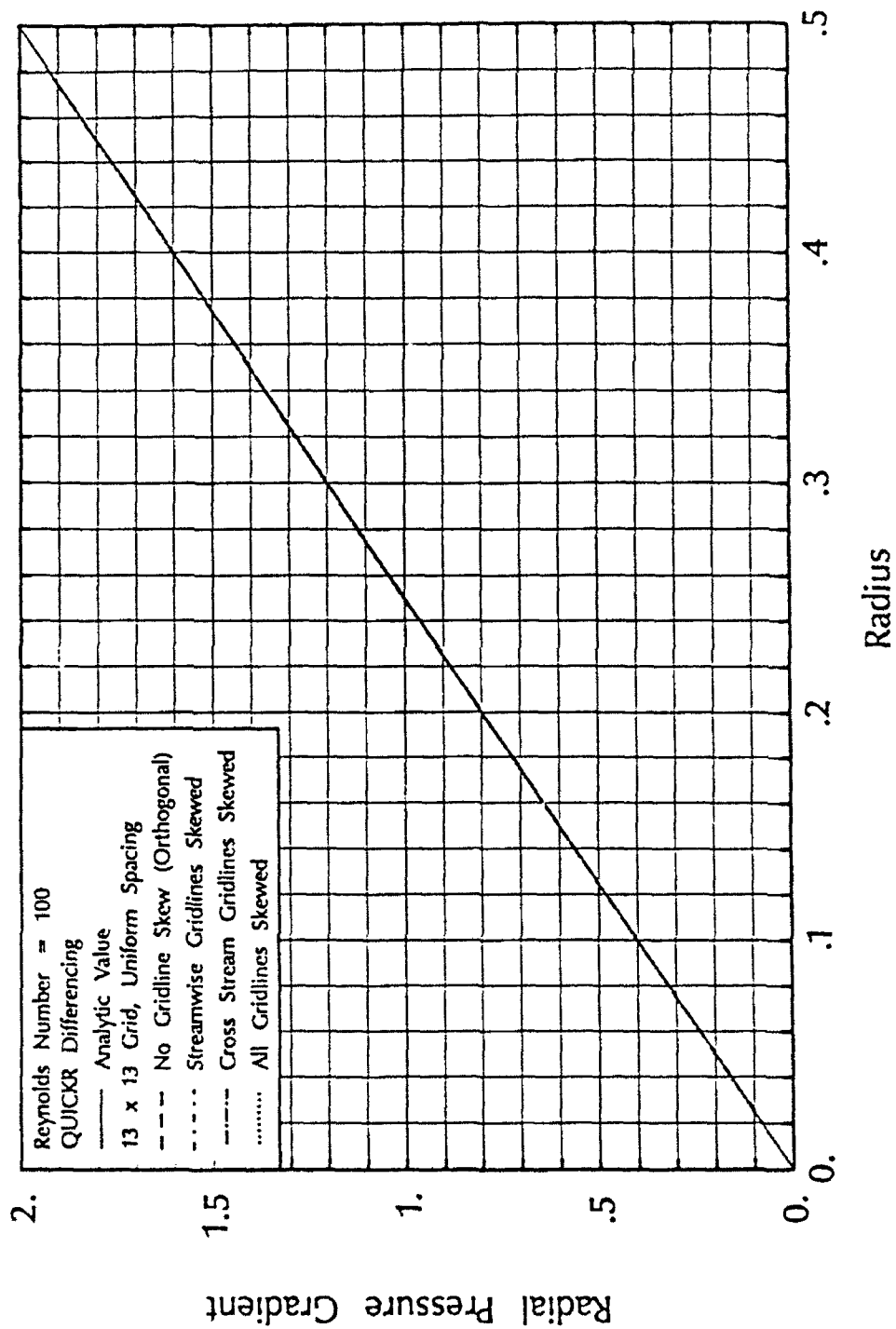


Figure 7.15

Effect of Gridline Skew on Radial Pressure Gradient

the flow solver is properly coded to yield a solution that conserves angular momentum, and that the solver is properly treating the terms arising from grid nonorthogonality.

The proper coding of the centripetal acceleration terms in the momentum equations is validated by this test case. The pressure gradients, which arise out of the radial momentum equation, are driven by the v_θ distribution that is returned by the angular momentum equation. Radial and tangential velocities are coupled by the acceleration term, $\rho v_r v_\theta$, in the angular momentum equation. Thus, the equations are discretized in a consistent manner. Otherwise, the iterative solver would not have converged to a solution which eliminates the $\rho v_r v_\theta$ term from the angular momentum equation, as the radial velocity is not zero until convergence when the proper pressure field is present.

7.2 Flow in a Radial Diffuser

Each of the previous test cases dealt with either primarily axial or purely angular flow. These trials in themselves do not sufficiently demonstrate the conservation of radial momentum, nor do they provide an adequate forum to demonstrate the importance of using the cylindrical differencing schemes. To emphasize this latter point, standard differencing schemes were used in the axial geometries, and they still provided accurate results. Flow in a radial diffuser with parallel walls was examined to help complete the validation process.

Unlike axial and angular flow, there is no geometry that is dominated by radial flow for which an analytic solution can be found for a viscous fluid. However, all of the viscous terms in the radial momentum equation, with a single exception, have a corresponding term in the axial momentum equation. The flow solver evaluates the axial and radial viscous source terms successively. For every line of code used to supplement the viscous terms in S_z , there is an equivalent line for S_r . Thus, there is little possibility that the viscous terms could be correctly

calculated for one equation, but not the other. Sections 7.1.1-3 demonstrate that the axial terms are properly handled. The exception to the axial-radial viscous correspondence is the third term on the right hand side of Eq. 2.16. This nonconservative term is present, because formulating the entire equation in strongly conservative form causes the term in Eq. 3.19 to take the following form:

$$r \frac{\partial}{\partial z_i} \left(\mu \frac{\partial}{\partial z_i} \left(\frac{\phi}{r} \right) \right) \quad (7.6)$$

Discretizing a system of equations based on this format presents several problems, so this single viscous term is kept in nonconservative format. Comparisons with hand calculations verify that this component is correctly handled by the flow solver.

A radial diffuser presents the solver with the need to deal with the radial dependence of area through which the velocity passes. As fluid passes through a diffuser, the flow is slowed to satisfy continuity. The static pressure rises as the velocity is reduced; dynamic pressure is converted to a static pressure rise. For inviscid flow and no axial velocity, the radial velocity is inversely proportional to r and the radial pressure gradient is inversely proportional to the radius cubed.

$$v_r = \frac{c}{r} \quad (7.7a)$$

$$\frac{\partial p}{\partial r} = \frac{c^2}{2r^3} \quad (7.7b)$$

The test cases for radial diffuser flows are similar to those used in Section 7.1.3. The velocity is specified along the boundary according to Eq. 7.7a. Four uniformly spaced grids were generated. One grid was orthogonal, one had skewed radial gridlines, another had skewed axial gridlines, and the fourth grid had both sets of gridlines skewed to contribute to grid nonorthogonality. For each grid, the inlet radius is unity and the outlet radius is equal to 5. The width of the diffuser is equal to unity, as is the constant c in Eq. 7.7. When axial, η , gridline skew is considered, the gridlines are oriented at 10° with respect to the horizontal. For radial, ξ , gridline skew, the inlet extends from the left corner of the diffuser to the midline, and the outlet extends from the middle of the diffuser over to the right side. A typical grid possessing dual skew is

shown in Fig. 7.16. The converged velocity field for this grid is presented in Fig. 7.17. It shows how the velocity decreases as the radius increases.

Fig. 7.18 shows a comparison of the analytic and calculated radial pressure gradients along the center of the grid for a variety of nodal densities. For this case, the flow was calculated on an orthogonal grid with uniformly spaced nodes using noncylindrical QUICKR differencing. As the number of nodes in the direction of the flow is increased, the pressure gradient approaches the theoretical curve. However, the calculated values never attain the analytic values, because central differencing cannot precisely resolve the gradients of the cubic pressure field, nor can QUICKR differencing interpolate the nodal values to give the exact facial radial velocities which vary inversely with r . This is the behavior that led to the need to develop a differencing scheme that accounts for the differences between cylindrical and rectangular coordinate spaces. The problem is further illustrated in Fig. 7.19. In this figure, the results given by QUICKR and hybrid differencing are compared. The flow is parallel to the gridlines; therefore, numerical diffusion is not present and the error can be traced back to the invalidity of the upwind approximation. Notice that the larger errors appear at the lower radii. This is where the percent change in surface area is largest and the fluid velocity is most rapidly decreased. Consequently, this is the region where the Cartesian differencing schemes are least accurate.

Fig. 7.20 shows a comparison of the results obtained using standard and cylindrical differencing schemes. The results indicate that the lower-order cylindrical differencing scheme is as accurate as the higher-order standard differencing scheme in this radial application, the former being less expensive to implement. In this case cylindrical QUICKR is as accurate as standard QUICKR differencing. For this simple flow, the errors arising from the higher-order approximations to the convective flux are much less than the truncation errors that prevent central differencing from precisely resolving the cubic pressure gradient.

No stability problems were caused by the fact that the central coefficient in the cylindrical differencing schemes is not equal to the sum of the neighboring coefficients.

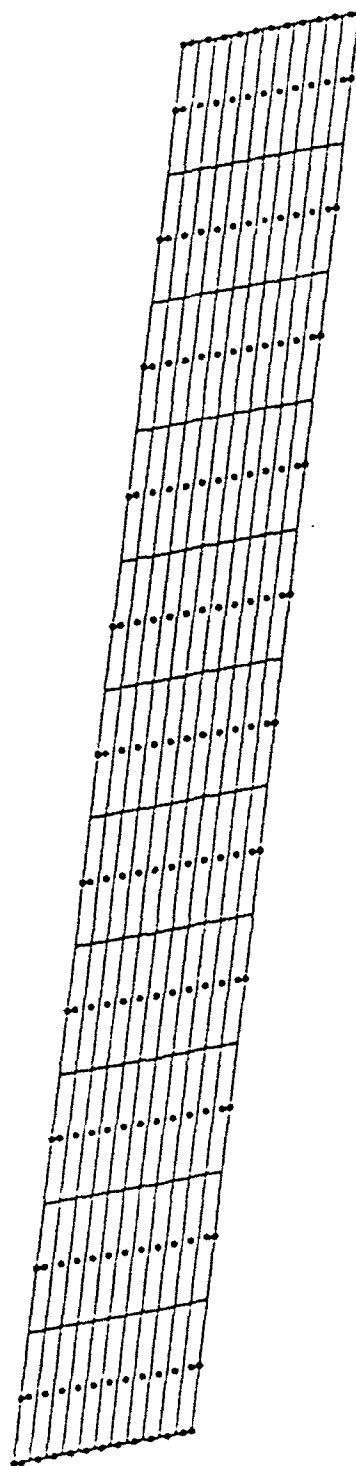


Figure 7.16

Uniformly Spaced Radial Diffuser Grid with Skewed ξ and η Gridlines

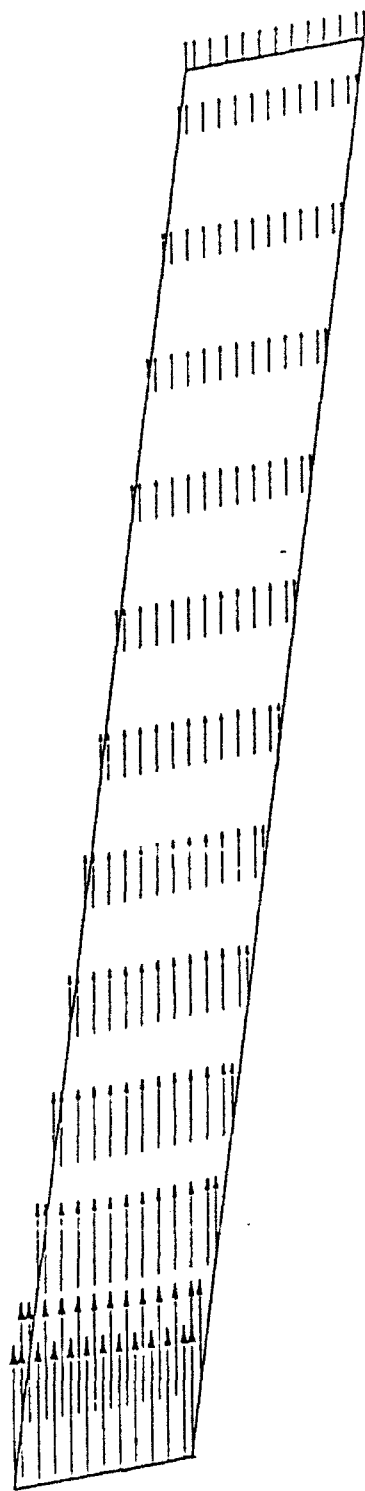


Figure 7.17

Radial Diffuser Velocity Field on Uniformly Spaced Grid with
Skewed ξ and η Gridlines

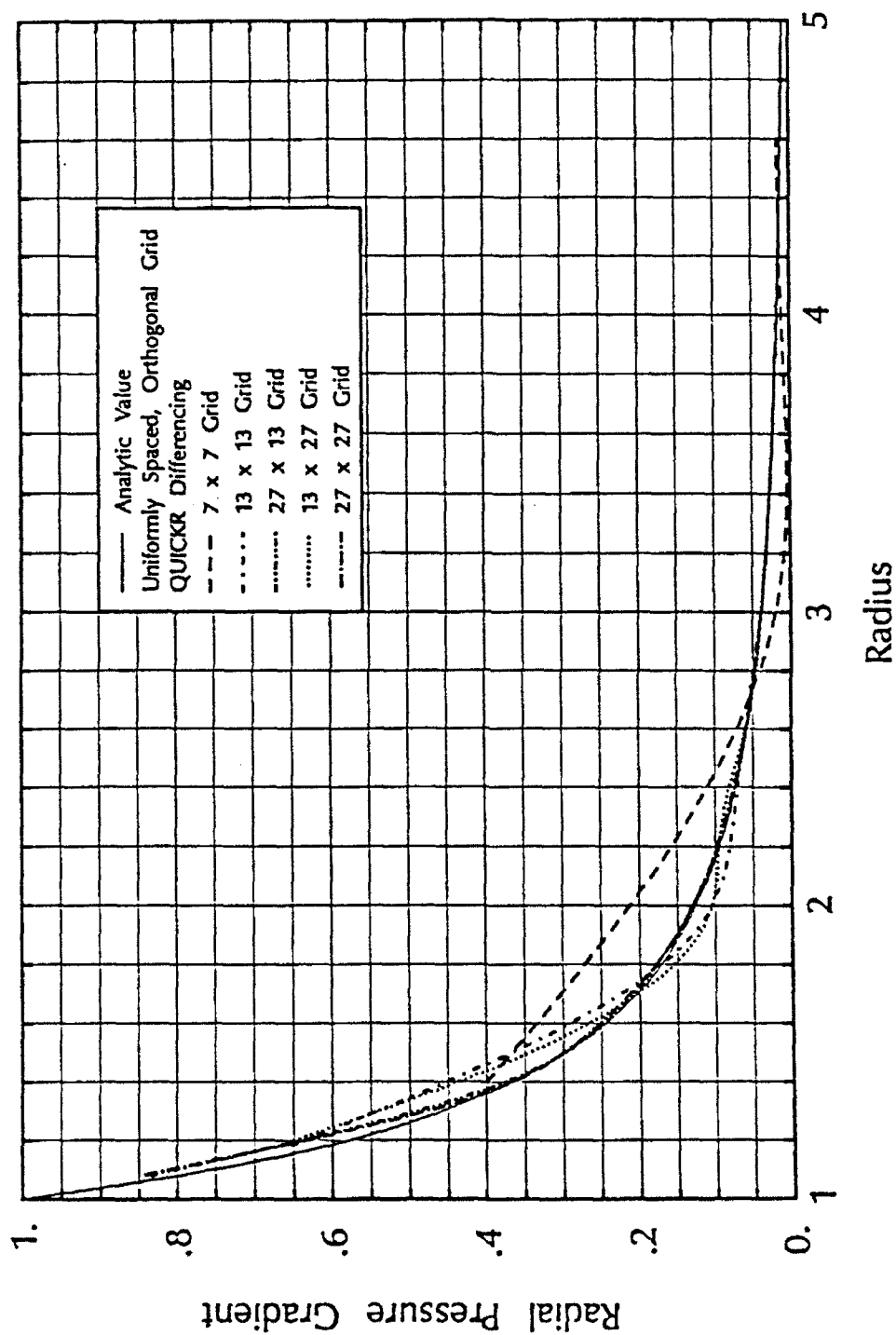


Figure 7.18

Radial Pressure Gradient for Various Nodal Densities (QUICKR Differencing)

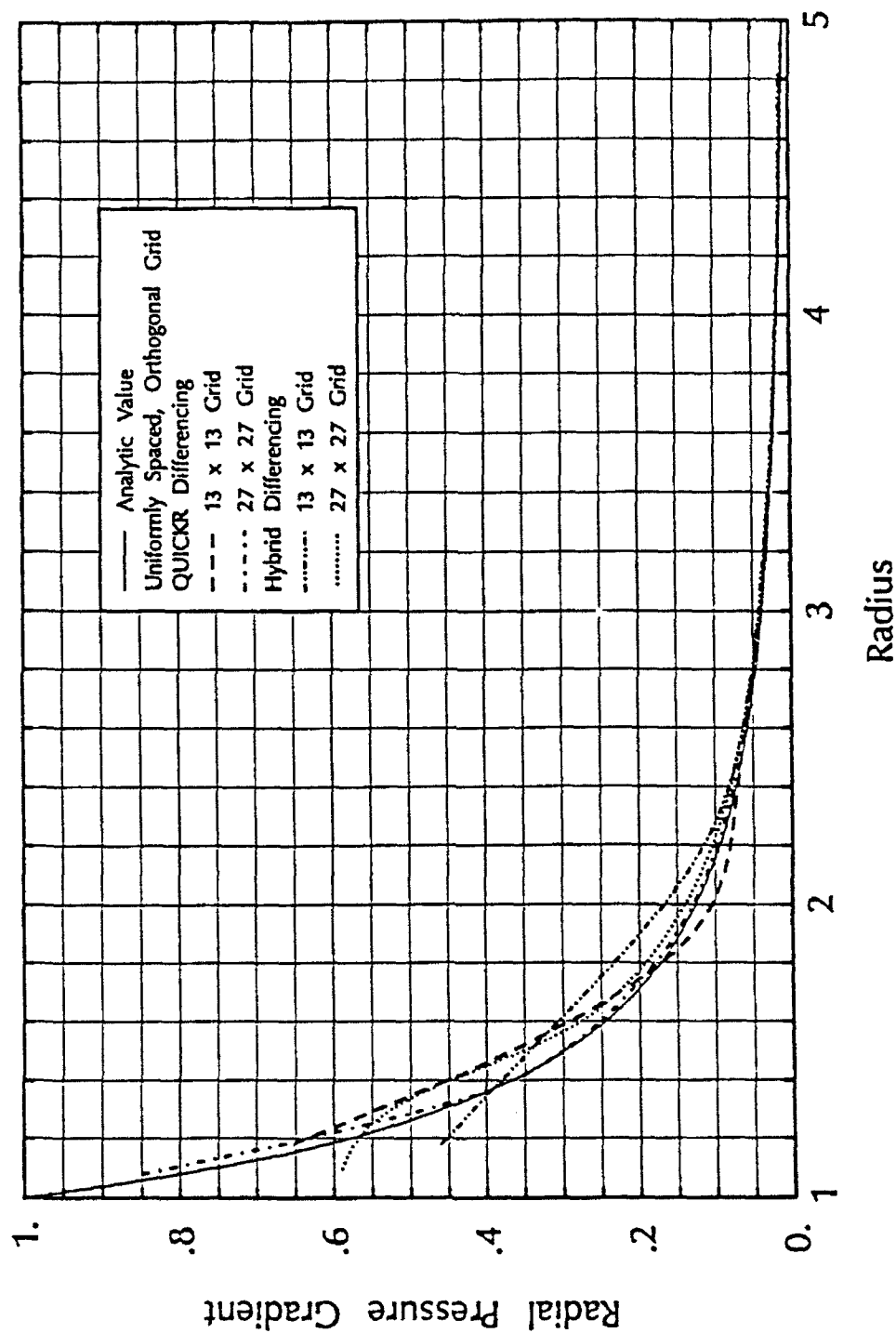


Figure 7.19

Radial Pressure Gradient for Various Nodal Densities and Differencing Schemes

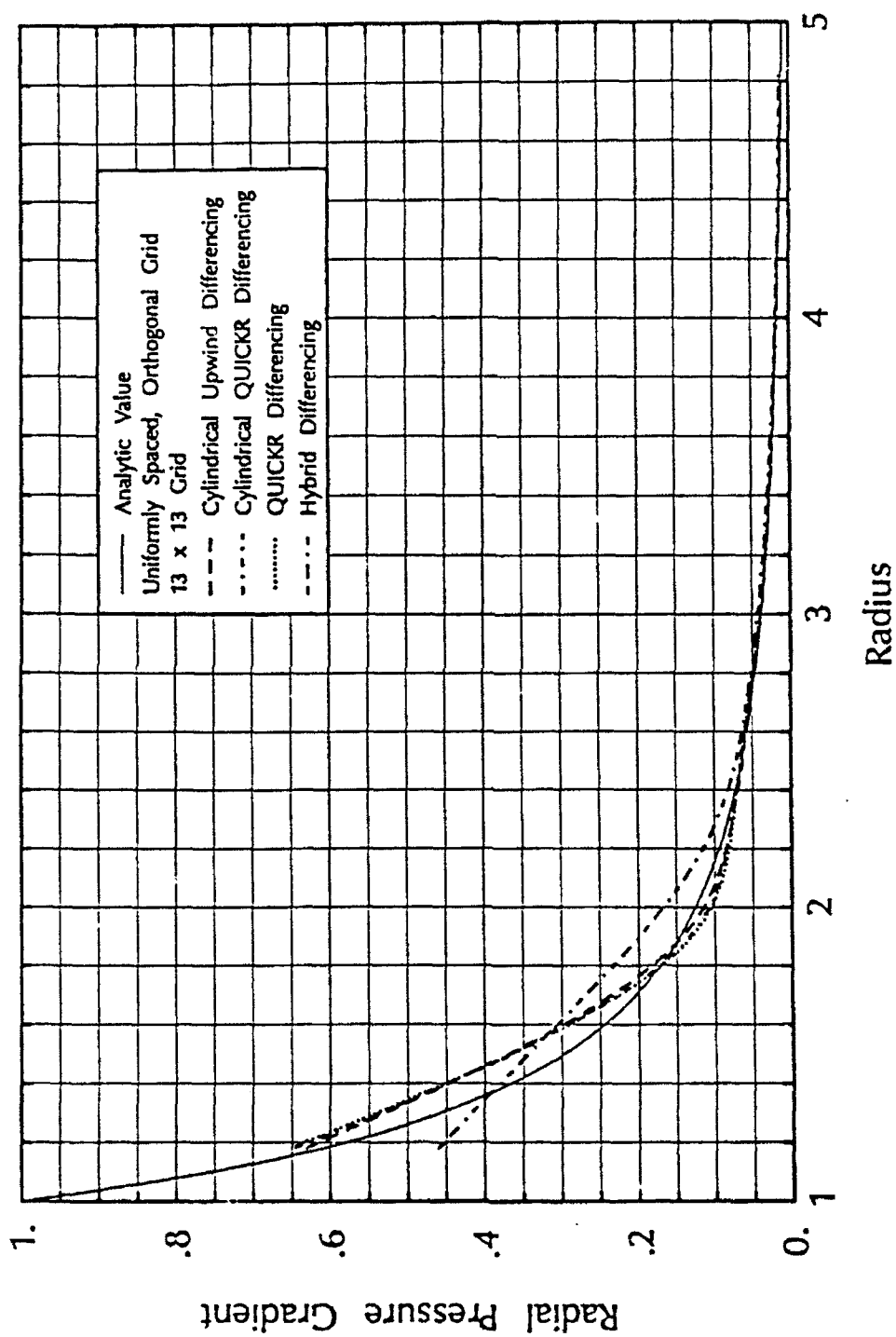


Figure 7.20

Radial Pressure Gradient for Cylindrical and Cartesian Differencing Schemes

Fig. 7.21 shows that gridline skew does not affect the ability of the code to treat nonorthogonal terms for radial flows. The absence of significant amounts of numerical diffusion is not due to an inherent ability on the part of cylindrical upwind differencing to eliminate numerical diffusion. Rather, it is due to the gradual change in the velocity fields within the radial diffuser. Numerical diffusion is not as prevalent as it is in pipe flow, where the velocity varied with the square of the radius. The slight variations in the curves are due to the fact that the truncation errors in the finite difference evaluation of the flow field gradients are dependent on radius.

A viscous fluid flowing inside a radial diffuser with parallel no-slip walls was examined. A parabolic inlet velocity profile was used, and the flow was assumed to be purely radial at the outlet. A converged solution for an orthogonal grid is shown in Fig. 7.22, displaying the capability of the flow solver to handle a radial discharge without specifying the velocity profile.

There is no visual evidence that the viscous effects are misrepresented in the program. Furthermore, a check of global radial momentum conservation was made by adding together the pressure and wall shear forces and comparing those to the change in momentum between the inlet and outlet. The net momentum loss decreased as the number of nodes increased, but the net loss was never zero. A momentum balance was not obtained, because the pressures at the inlet and outlet had to be interpolated from the interior nodes and the shear forces were known only at discrete locations. This gives further validation that the radial momentum equation is solved correctly.

Now that a substantial amount of evidence has been presented to verify that the conservation equations are correctly solved, calculations of realistic flows within a mixed-flow centrifugal impeller will be presented.

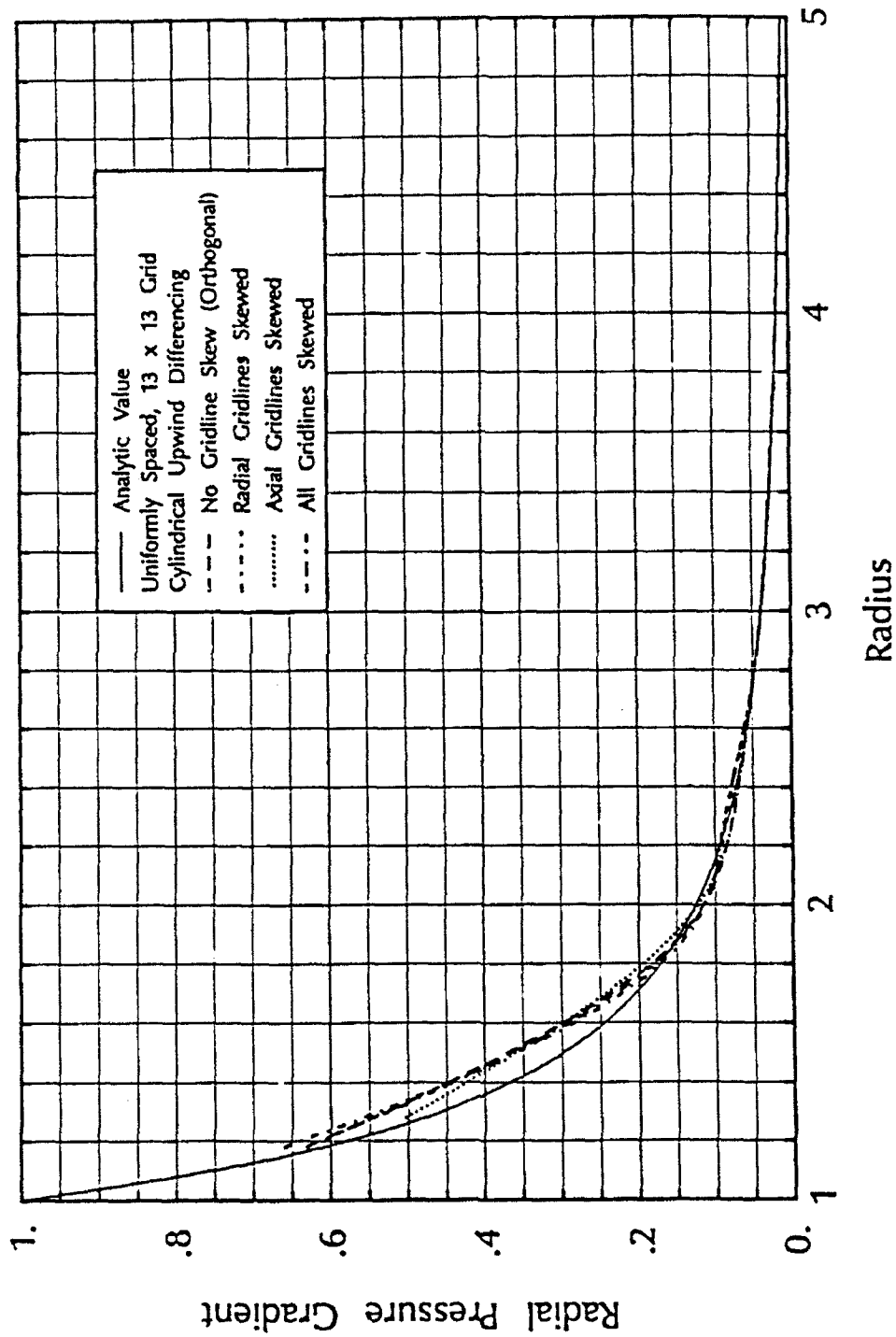


Figure 7.21

Effect of Gridline Skew on Radial Pressure Gradient

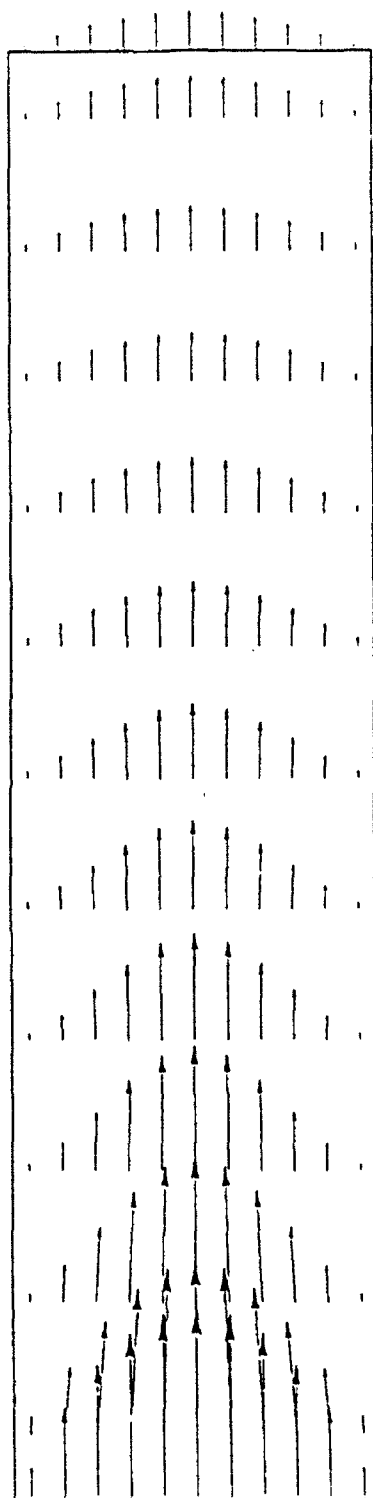


Figure 7.22

Velocity Field for a Viscous Fluid in a Radial Diffuser

Chapter 8

Mixed-Flow Centrifugal Impeller Through-Flow Calculations

The primary application of the computer code is as an analysis tool for numerically predicting fluid flow within existing and proposed turbomachinery and axisymmetric passages. In this chapter, the results for through-flow calculations in a mixed-flow centrifugal impeller over a wide range of operating conditions are presented. The impeller has been tested in the Applied Research Laboratory centrifugal impeller test facility, and the calculated results are compared to the experimental data.

The geometry of the test facility is shown in Fig. 8.1. This geometry is the physical space flow domain for the axisymmetric flow solution. The thick dark lines, h-c and g-d, mark the projection of the impeller blade leading and trailing edges onto the axisymmetric plane. The control volumes that fall within these lines are considered to lie on the blade, and are thus the CV's for which body forces are calculated via the blade model. Appropriate boundary conditions for the various sections of the geometry are indicated on the figure.

Pump performance was measured experimentally by collecting data at three different static pressure taps. The first tap was located upstream of the impeller to measure the reference inlet conditions. The second tap was located slightly past the discharge of the impeller, and the third tap was placed at the end of the radial diffuser into which the impeller expels the fluid. The locations of the pressure taps are labeled on Fig. 8.1 as stations 1, 2, and 3, respectively. Radial and tangential velocity profiles were measured using Laser Doppler Velocimetry (LDV). The data was collected along the path, referred to as the LDV traverse line, indicated by the dashed line in Fig. 8.1.

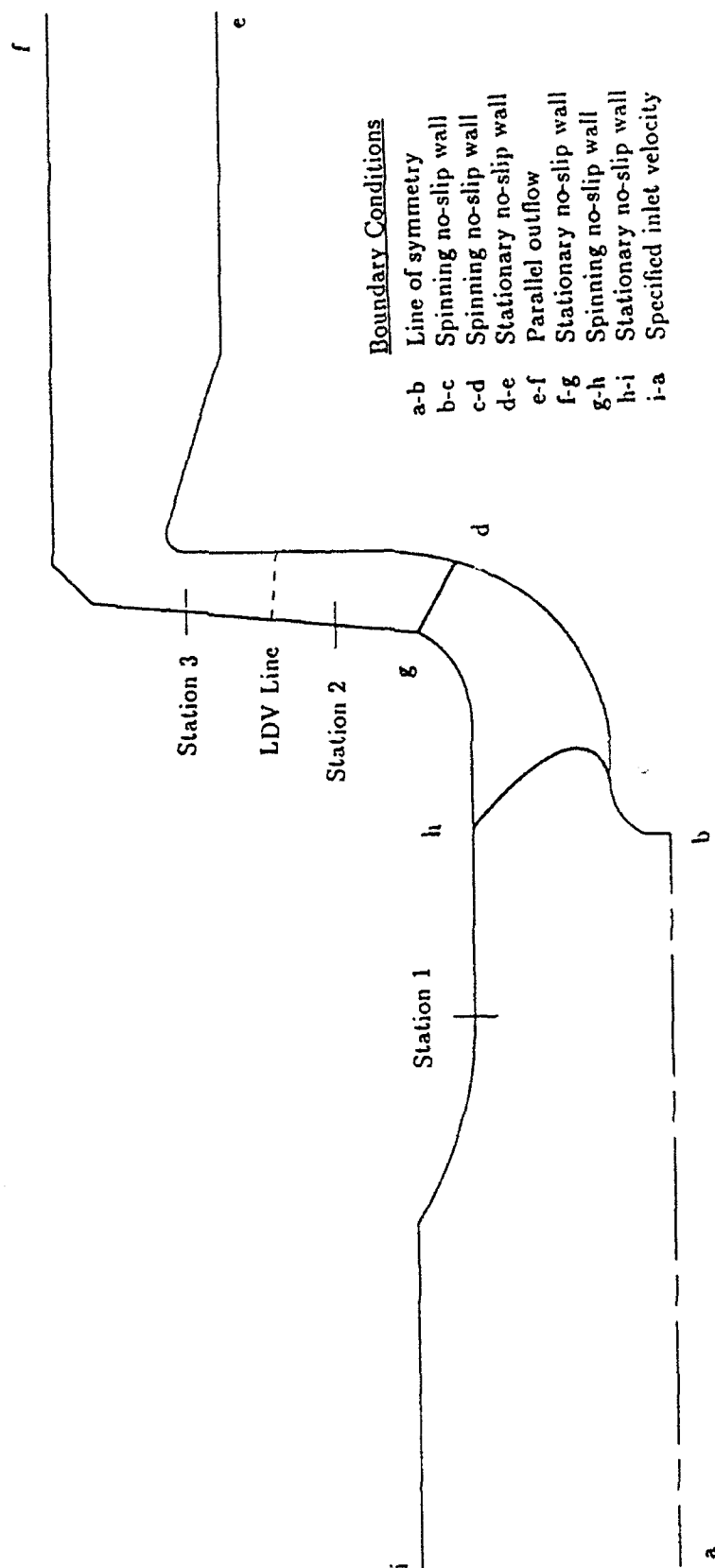


Figure 8.1
Centrifugal Impeller Test Facility Geometry
and Boundary Conditions

After the experimental data was taken, it was converted into nondimensional performance coefficients. Eqs. 8.1-8.4 give the expressions for the nondimensional pump performance coefficients. Experimental data is gathered in terms of dimensional quantities, so the first expressions are used to reduce the data. The right-most equalities in these expressions represent the dimensionless variables used in the computer program. The inlet tip speed and diameter are the reference velocity and length, respectively.

$$\Phi = \dot{V}_m / \hat{U} = V_m \quad (8.1)$$

$$\psi_{12} = \frac{\hat{p}_2 - \hat{p}_1}{\hat{\rho} \hat{U}^2} = p_2 - p_1 \quad (8.2)$$

$$\eta_{12} = \frac{\dot{m}(\hat{p}_2 - \hat{p}_1)}{2\pi N \hat{\rho} \hat{T}} = \frac{\dot{m} \psi_{12}}{2T} \quad (8.3)$$

$$P = \frac{2\pi N \hat{T}}{(2\pi N)^3 \hat{\rho} \hat{D}^5} = T/4 \quad (8.4)$$

$$T = \sum_{BLADE} r F_{\theta} Volume \quad (8.5)$$

The global flow coefficient, Φ , defines the operating condition. It is based on the net volume flow rate through the impeller, and is distinct from the local flow coefficient, ϕ . The static head rise coefficient, ψ , indicates the increase in static pressure produced by the impeller and the dynamic pressure recovery in the diffuser. The efficiency, η , is the ratio of the actual head rise to the maximum possible value. The maximum head rise is determined by the change in angular momentum of the fluid, which is equal to the torque, T , on the impeller. In the program, T is calculated as the summation of the product of the radius, angular body force, and volume for each CV on the blade. Efficiency and head rise are determined from a pressure change, where the computed pressures are taken from the locations corresponding to the experimental measurement stations. The power coefficient, P , is a dimensionless measure of the shaft power required to drive the impeller.

Computer speed, rather than memory, was the most restrictive factor on the size of the grid that could be used in the flow solver. The finest grid was used on which solutions could be

found within a reasonable computation time. The same grid was used for all of the calculations, because computational limits precluded adding nodes in regions of high flow gradients. The grid quality was high, especially when considering the intricacy of the geometry and the several slope discontinuities on the boundaries.

Fig. 8.2 shows most of the 175×32 grid that was used in the impeller through-flow calculations. An extension, equal in length to the diameter of the impeller, was added to the outlet of the grid to allow the recirculating regions to close inside the computational domain. See Section 4.6.2 for a discussion of the need for this extension. An extension was also placed at the inlet to prevent the specified inlet velocity profiles from distorting any recirculating regions originating in the blade row. Gridlines were placed along the leading and trailing edges, the LDV measurement line, and measurement stations 2 and 3.

Eiseman's control point (CP) grid generation technique [34] was used to generate the grid in the very irregular geometry of the pump facility. This algebraic method uses a sparse mesh of control points and a series of interpolations, using the control point curve functions mentioned in Chapter 6, to govern the placement and clustering of the nodes. Control point curves remain in the convex hull of the CP's, so the gridlines are smooth and free of superfluous oscillations. If necessary, however, discontinuities can be resolved by collocating control points. Gridlines can be placed along specified curves, such as the blade leading and trailing edges, by repeating the line of control points along the desired curve. The spacing of nearby CP's, over which the user has complete control, determines the amount of clustering. Grid characteristics are determined only by the local control points; therefore, grid quality can be improved in specific regions by adjusting the location of the nearest CP's. The flexibility of this technique makes it a very powerful grid generation tool, and yet the cost, in terms of computational expense and memory requirements, is minimal. No iterative solutions are required to determine the coordinates of the grid points, and the only significant memory allocations are for the arrays that hold the coordinates of the grid and control points.

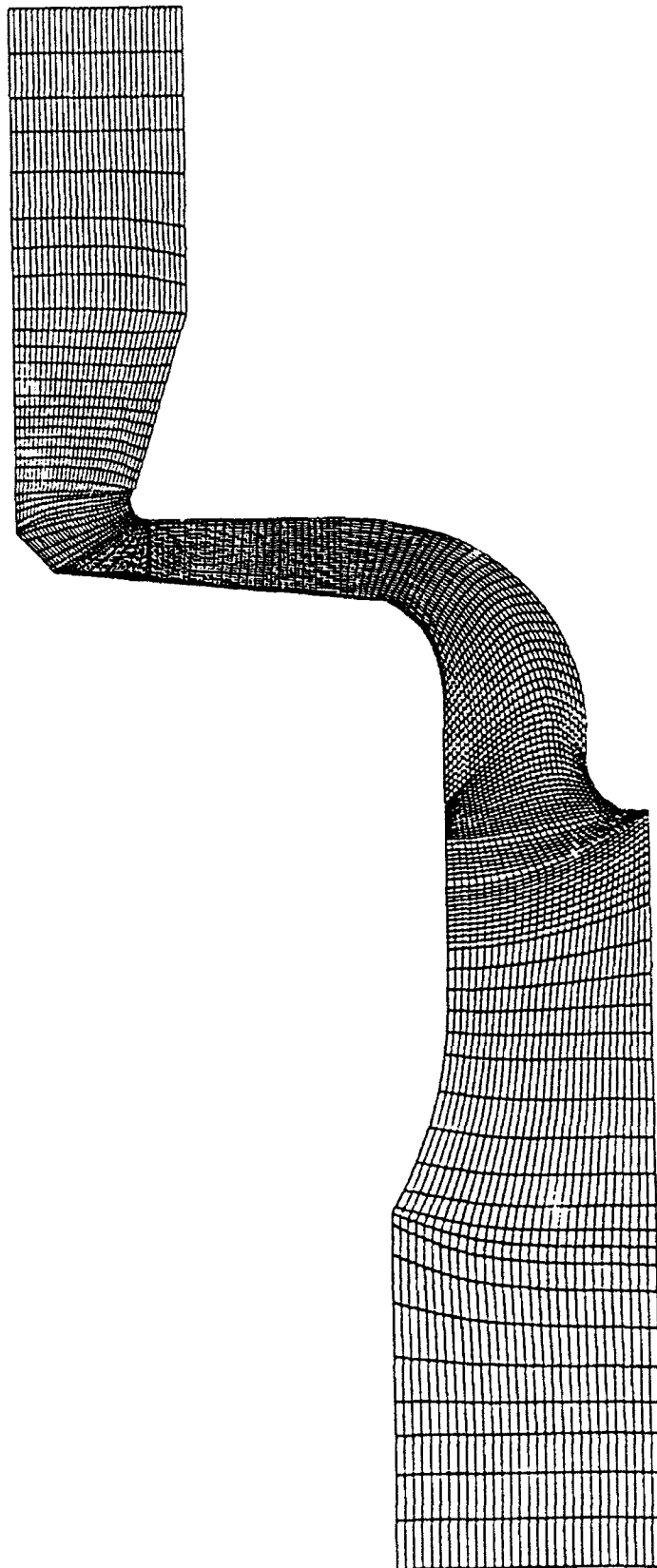


Figure 8.2
Through-Flow Calculation Grid

The remainder of this chapter is dedicated to discussing the results of the through-flow calculations. In the first section, the flow fields for each of the various operating conditions, are compared with experimental data. Overall performance predictions are compared to the measured performance in the last section.

8.1 Comparison of Calculated and Measured Flow Fields

Calculations were made at the design condition, above and below the design condition, as well as at shutoff. This allowed an assessment to be made of the flow solver and blade model over the entire range of operating conditions. The global Reynolds number was the same for each calculation, 4.486×10^6 , corresponding to a constant rotational speed in the experiments. A uniform velocity profile was specified at the inlet, the magnitude of which depended upon the flow coefficient. All results were found using single precision calculations on a VAX 8350 computer, which is significantly slower than most machines currently used in CFD applications. The memory requirements are approximately 204 bytes per node and an additional 124 bytes for every node on the blade. This does not include the additional memory requirement for the direct solver, which was discussed in Section 4.5.

A few remarks concerning the overall performance of the flow solver during the through-flow calculations are in order, before the results of the individual calculations are presented.

It was found that the use of the Baldwin-Lomax turbulence model was not appropriate for the through-flow calculations that were made. That model required an excessive number of nodes to be located close to the wall. With refined grids near the walls, convergence could not be reached when the problems of geometry, flow conditions, and low computational speed were compounded in the realistic flow cases. Therefore, a simple two-layer model was used instead. The thickness of the laminar layer was set equal to the height of the row of nodes

bordering the wall. Thus, the molecular viscosity was used within this laminar layer, whereas the turbulent viscosity throughout the remainder of the flow was modeled as a constant (35 for this investigation) times the laminar viscosity.

Because of the low computer speed, cylindrical QUICKR differencing could not be used. Its explicit implementation decreases convergence speed. Cylindrical upwind differencing was used instead.

Despite the numerous stability enhancing measures that were installed in the blade model, it was found that the solution would not converge when the estimated blade-to-blade mean streamline and blade loading were allowed to evolve with the iterative solution. Therefore, the mean streamline shape was estimated at the start of the iterative solution and held fixed. The meridional body forces were calculated at the same time and held fixed throughout the solution. This is a serious deficiency on the part of the blade model. However, despite the blade model deficiencies, useful results, that often times matched well with the experimental data, were returned by the solver.

In some of the calculations that are presented in the following sections, the residuals did not drop to machine zero. The relaxation factor was not the cause of the instabilities. Rather, the recirculating regions in the blade row caused the flow quantities in these regions to fluctuate about a mean level. The residuals of the nodes outside these regions vanished and were not affected by the unsteady region. A sufficient number of iterations were run so that the data that is presented did not change appreciably with further iterations.

8.1.1 Design-Point Calculation

The impeller was designed to perform at maximum efficiency at an operating condition of $\Phi = 0.28$. This flow condition was the one that was most readily solved, because the blade

loading did not set up any recirculating zones within the blade row. A plot of the convergence rate is shown in Fig. 8.3. The sudden increase in the residuals at iteration 90, marks the point where the meridional blade forces were introduced into the momentum equations. The total calculation time was 24,000 seconds. Over 80% of that time was used in solving the pressure corrector equations with the banded matrix direct solver.

The velocity field in the axisymmetric plane is shown in Fig. 8.4. The precursor to a separated region is present near the shroud and the midchord of the blade. This region plays a key role in the recirculating regions that develop at off-design flow conditions. Fig. 8.5 shows the pressure contours. The code confirms that the design-point criteria of a constant pressure rise at the trailing edge has been met. Fig. 8.6 shows a comparison of the calculated and measured velocity profiles. The trends in the calculated velocity profiles do not match those of the experimental data. Fig. 8.4 shows that by the time fluid reaches the radial position of the LDV measurement, it is affected by the recirculating zone at the bend. The simple turbulence model cannot accurately treat this recirculating zone, so it is possible that the discrepancies between the calculated and measured profiles are caused by an inadequate turbulence model. Near the impeller trailing edge, which is largely unaffected by the turbulence model, the calculated radial profile is similar in shape to that of the experimental data. This is further evidence in support of the supposition that the turbulence model, not the blade model, causes the poor agreement with the measured data when no recirculating zones are present in the blade row.

8.1.2 Above Design-Point Calculation

The blades were designed so that there would be zero incidence at $\Phi = 0.32$. Calculated velocity vectors in the meridional plane are shown in Fig. 8.7 for this flow coefficient. The blade midchord area near the shroud contains a very small recirculating region. Since no experimental

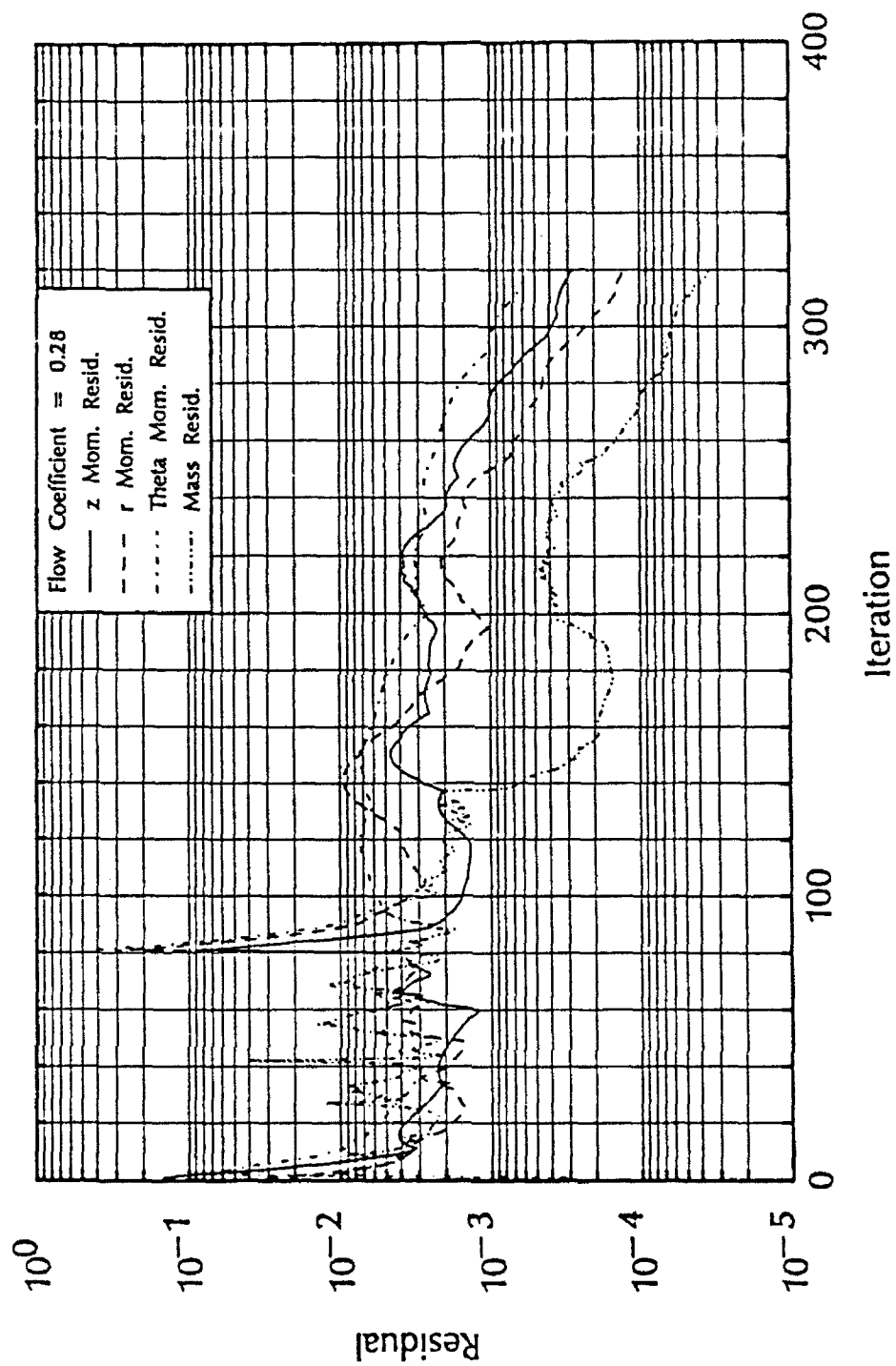


Figure 8.3

Convergence Rate for Design-Point Calculation

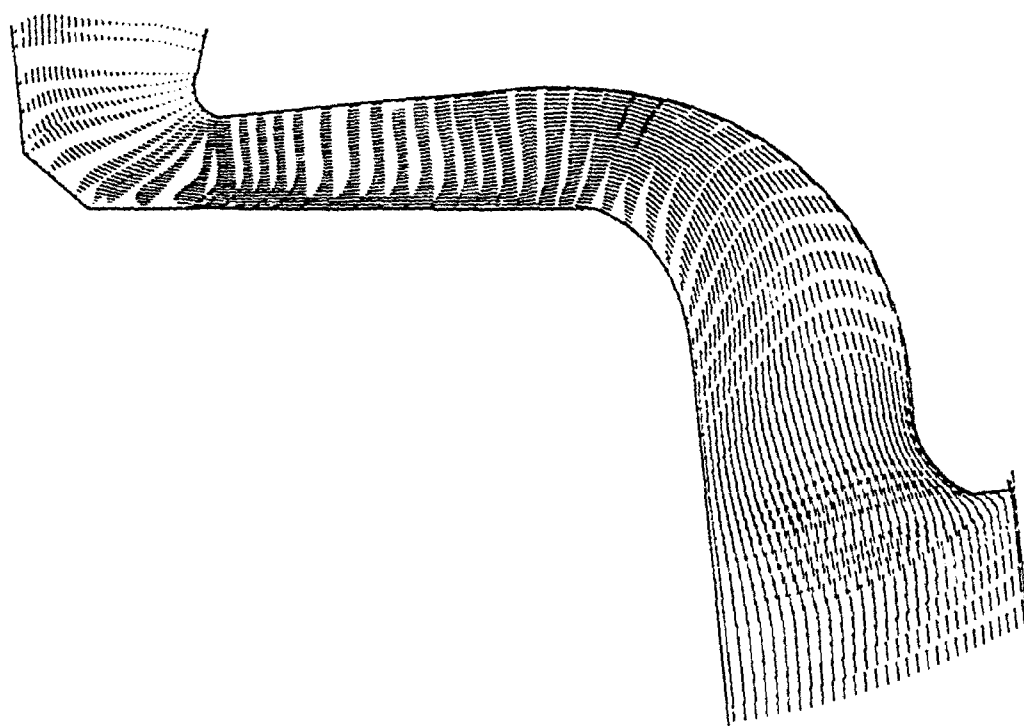


Figure 8.4
Calculated Through-Flow Velocity Vector Field ($\Phi = 0.28$)

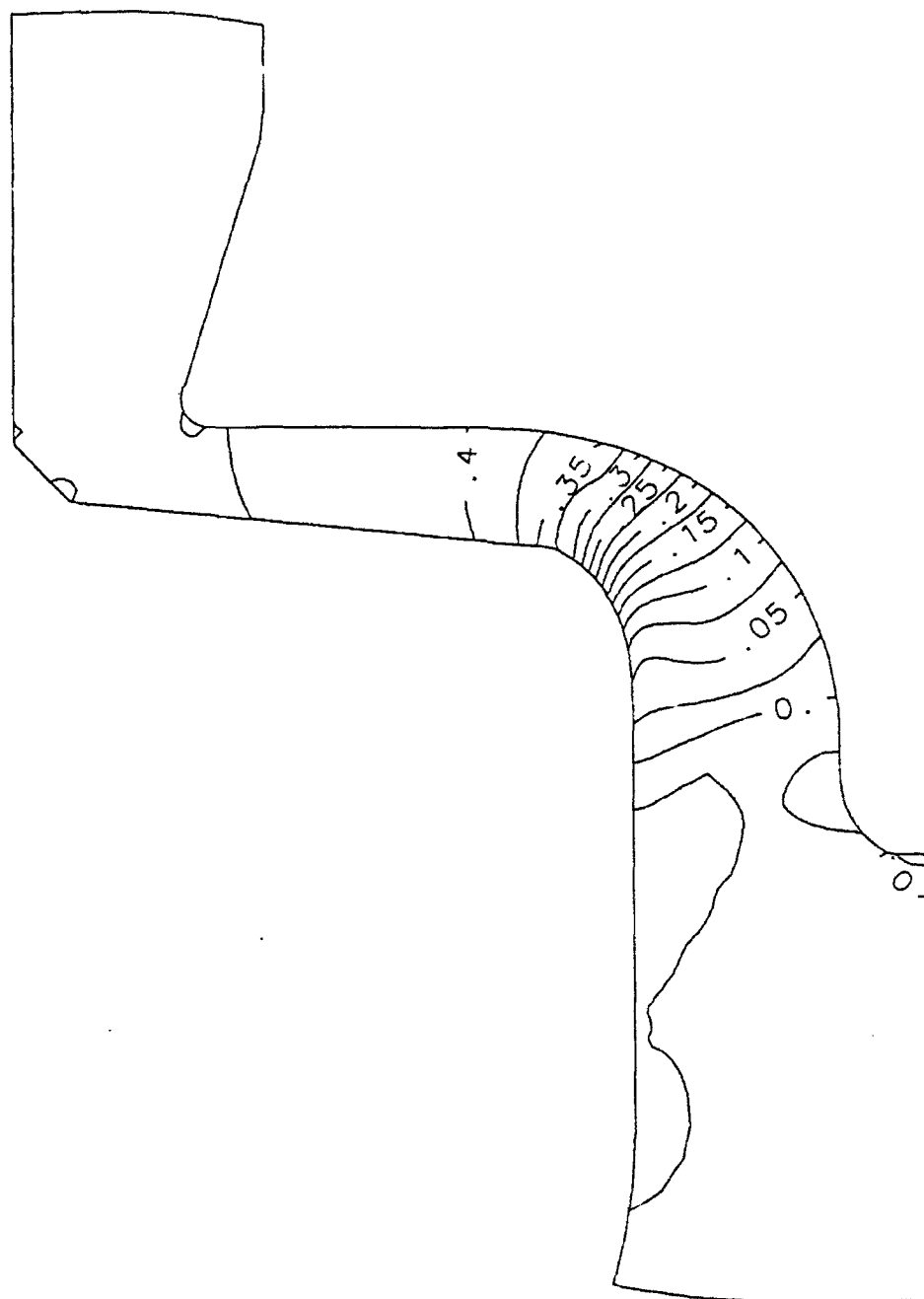


Figure 8.5

Calculated Pressure Contours ($\Psi = 0.28$)

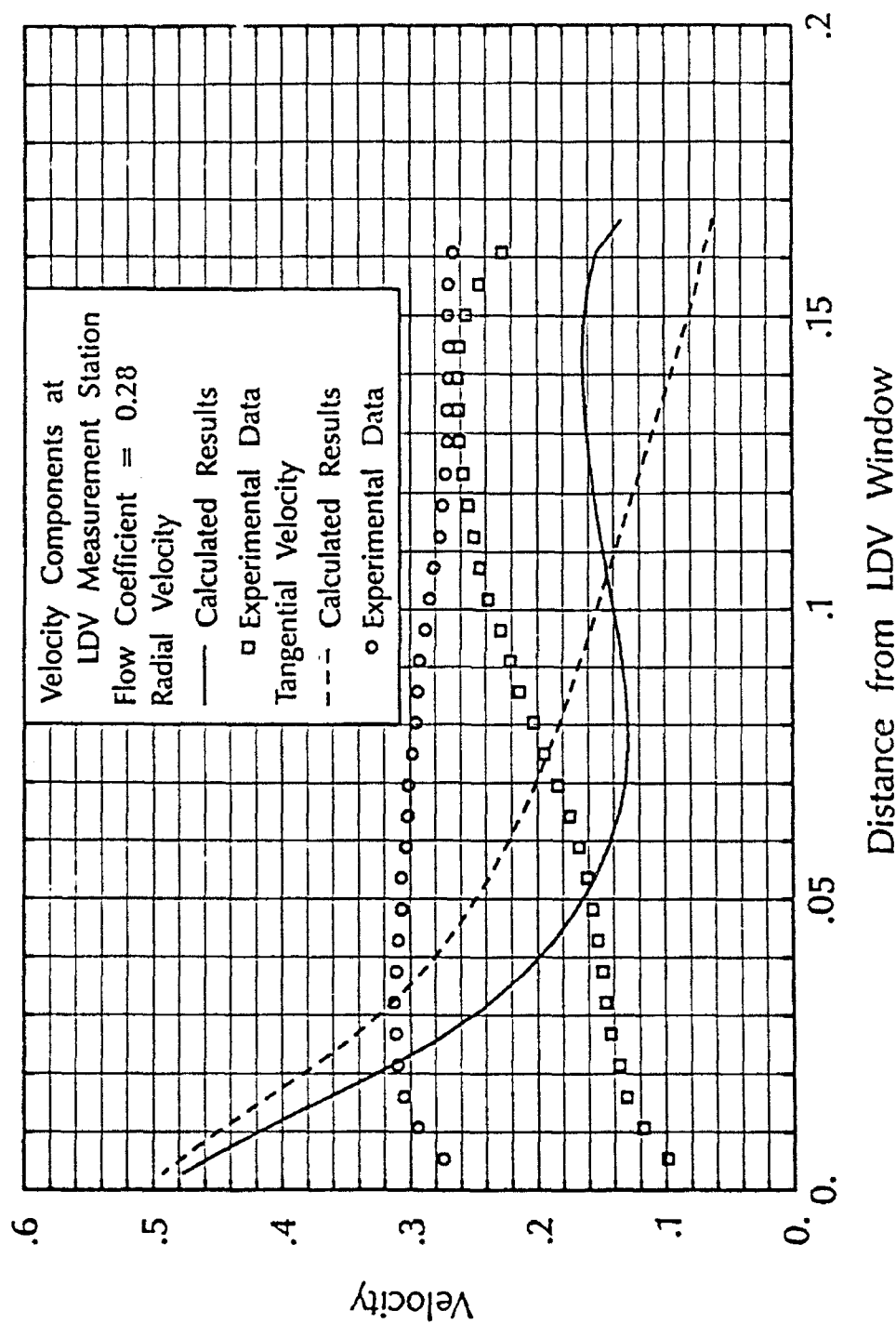


Figure 8.6

Comparison of Calculated and Measured
Radial and Tangential Velocity Profiles
($\Phi = 0.28$)

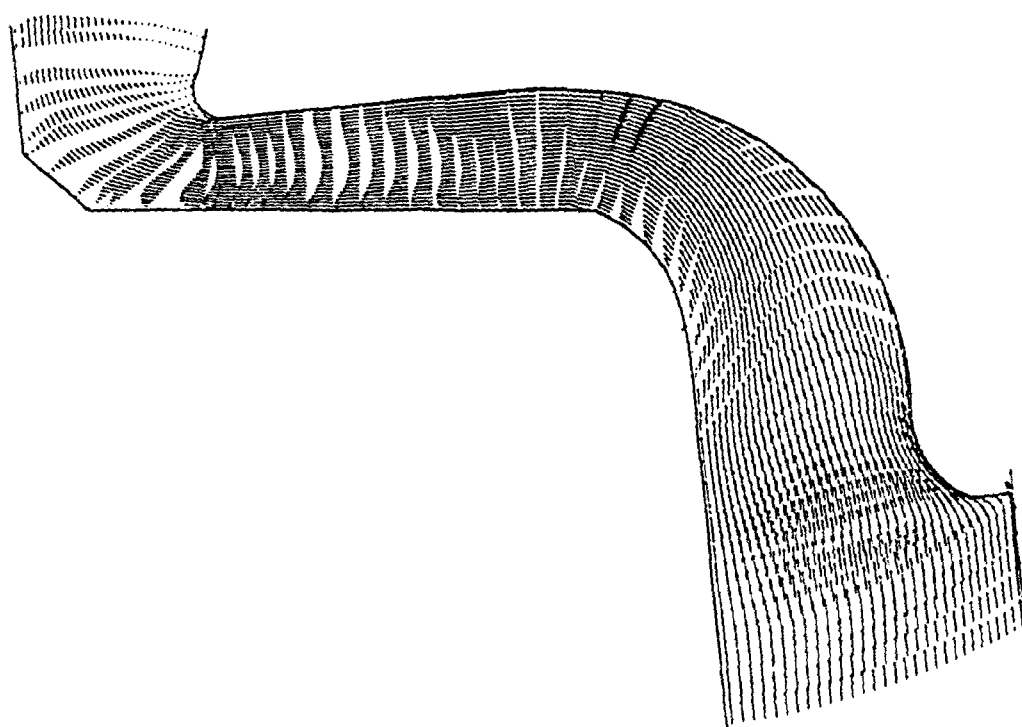


Figure 8.7
Calculated Through-Flow Velocity Vector Field ($\Phi = 0.32$)

data was collected inside the impeller, there is no experimental data to either confirm or deny the existence of this separated region in the actual impeller. After 460 iterations and 34,700 seconds of CPU time, the residuals in the recirculating region did not vanish, but the solution was not changing noticeably. The pressure contours are given in Fig. 8.8. It shows that the pressure rise is still fairly constant at the trailing edge. As indicated in Fig. 8.9, the calculated velocity profiles still fail to match the measured data. The predicted profile at the LDV station decreases in the direction from the shroud to hub walls, whereas the experimental data shows the opposite trend. As in the previous case, the predicted velocity profile at the impeller discharge shows the correct trend. It is again suspected that the inadequacies of the turbulence model destroy the validity of the solution in the diffuser.

8.1.3 Below Design-Point Calculation

The calculated through-flow velocity field for the off-design operating condition of $\Phi = 0.14$ is shown in Fig. 8.10. This solution was produced after 340 iterations and 26,000 seconds of CPU time. The pressure contours for this flow are shown in Fig. 8.11. The recirculating region predicted at the impeller inlet has been observed experimentally. To this author's knowledge, no previous numerical studies have correctly predicted a through-flow plane recirculating region that extends from the impeller to the upstream region. Fig. 8.12 shows that both the existence and width of the separated region in the diffuser were also correctly predicted. The maximum radial velocity at the impeller discharge was located on the hub side in each of the two high flow coefficient cases. This low coefficient solution correctly shows that the location of the maximum radial velocity has moved to the shroud side.

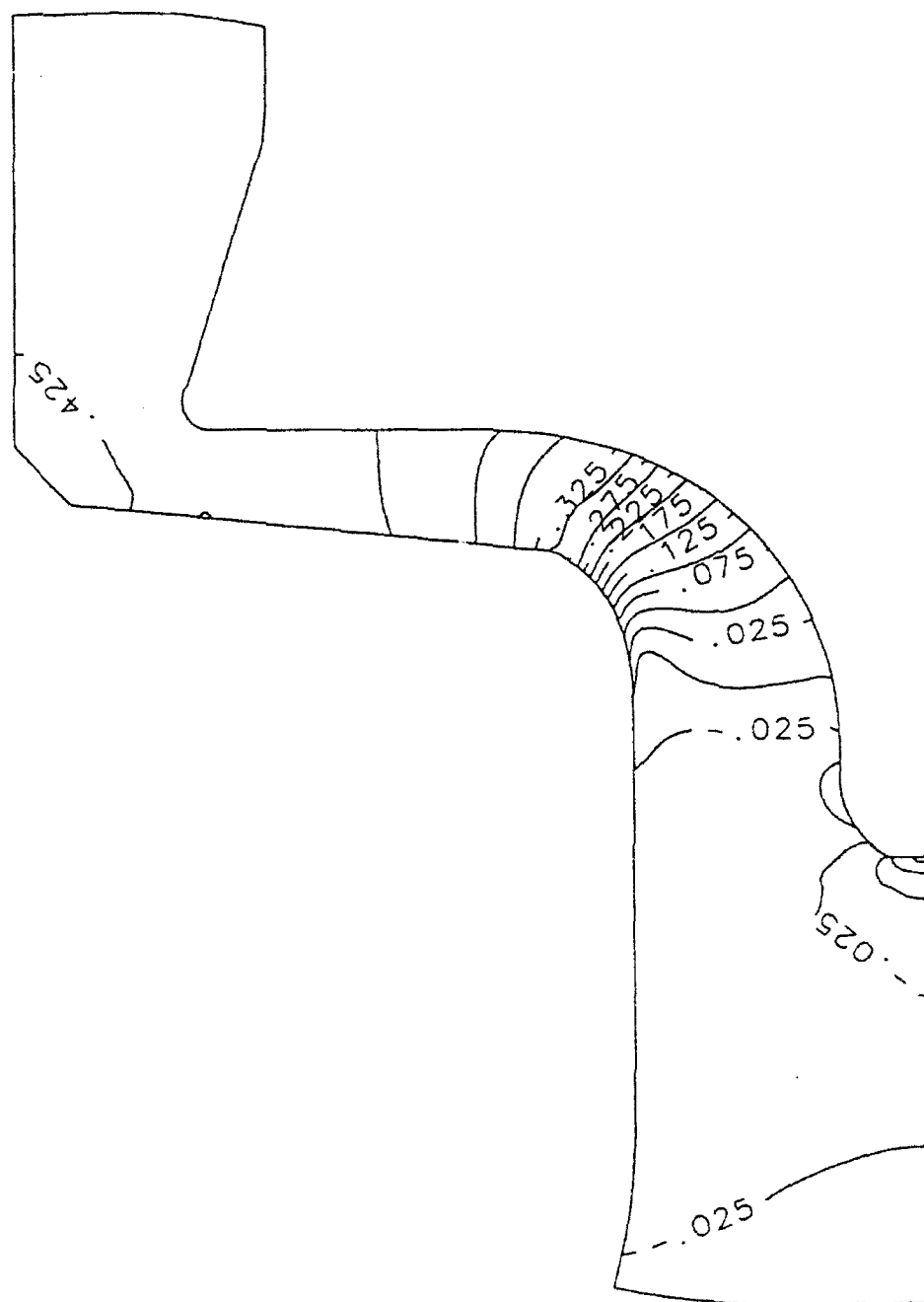


Figure 8.8

Calculated Pressure Contours ($\Phi = 0.32$)

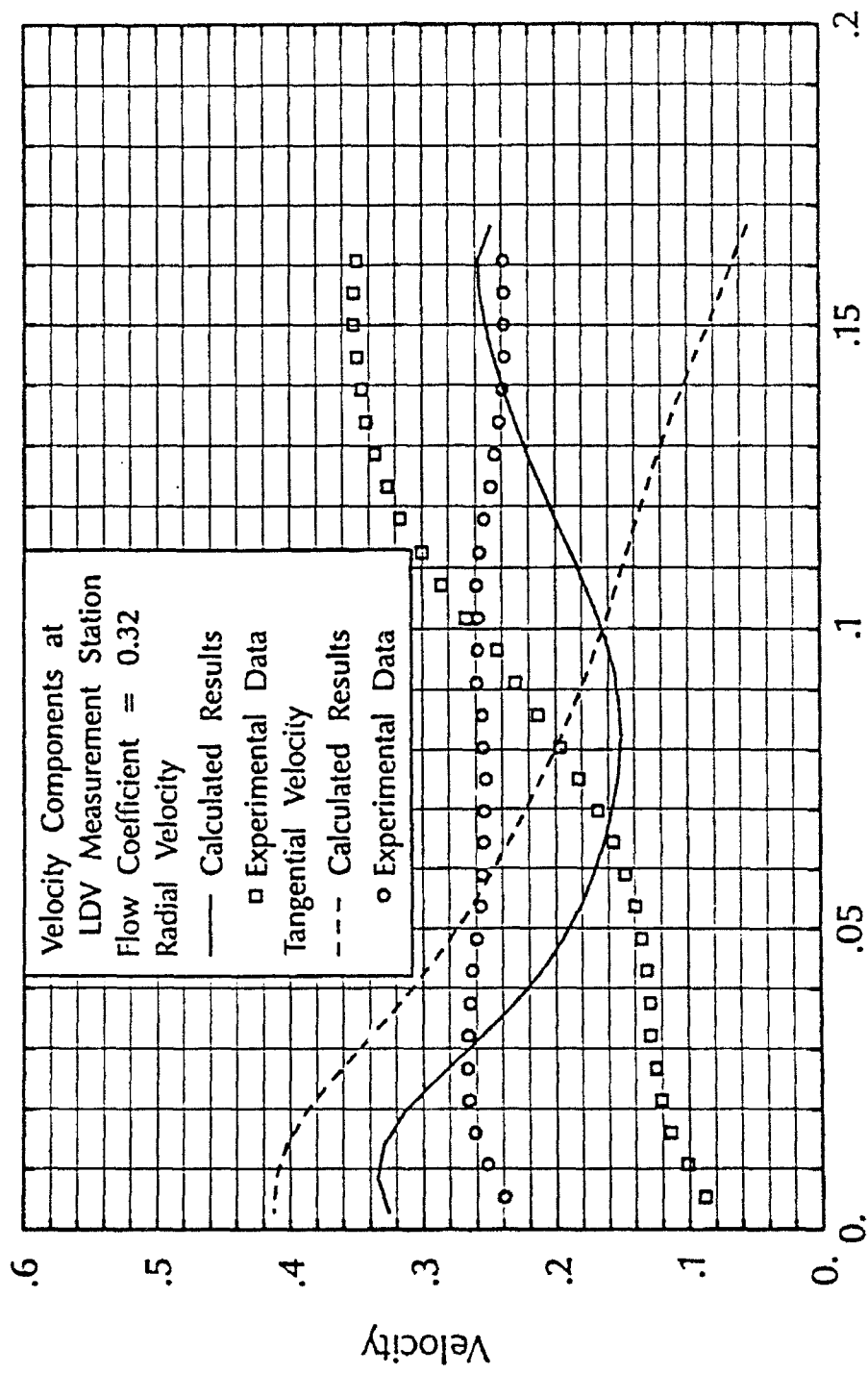


Figure 8.9
Comparison of Calculated and Measured
Radial and Tangential Velocity Profiles
($\phi = 0.32$)

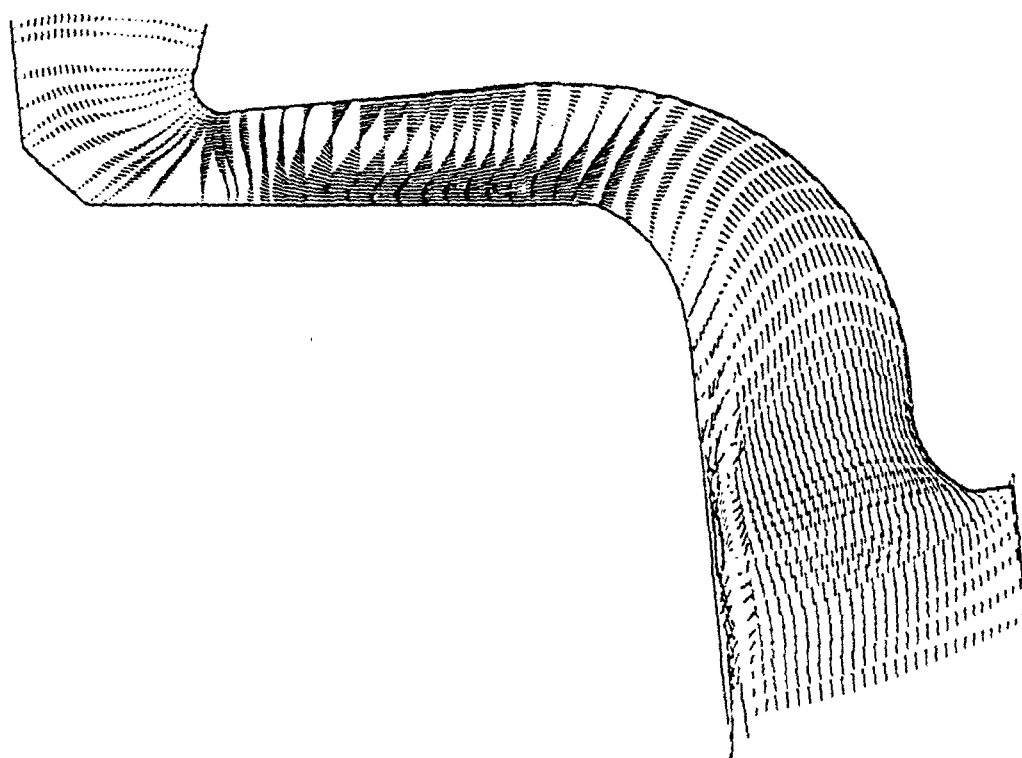


Figure 8.10
Calculated Through-Flow Velocity Vector Field ($\Phi = 0.14$)

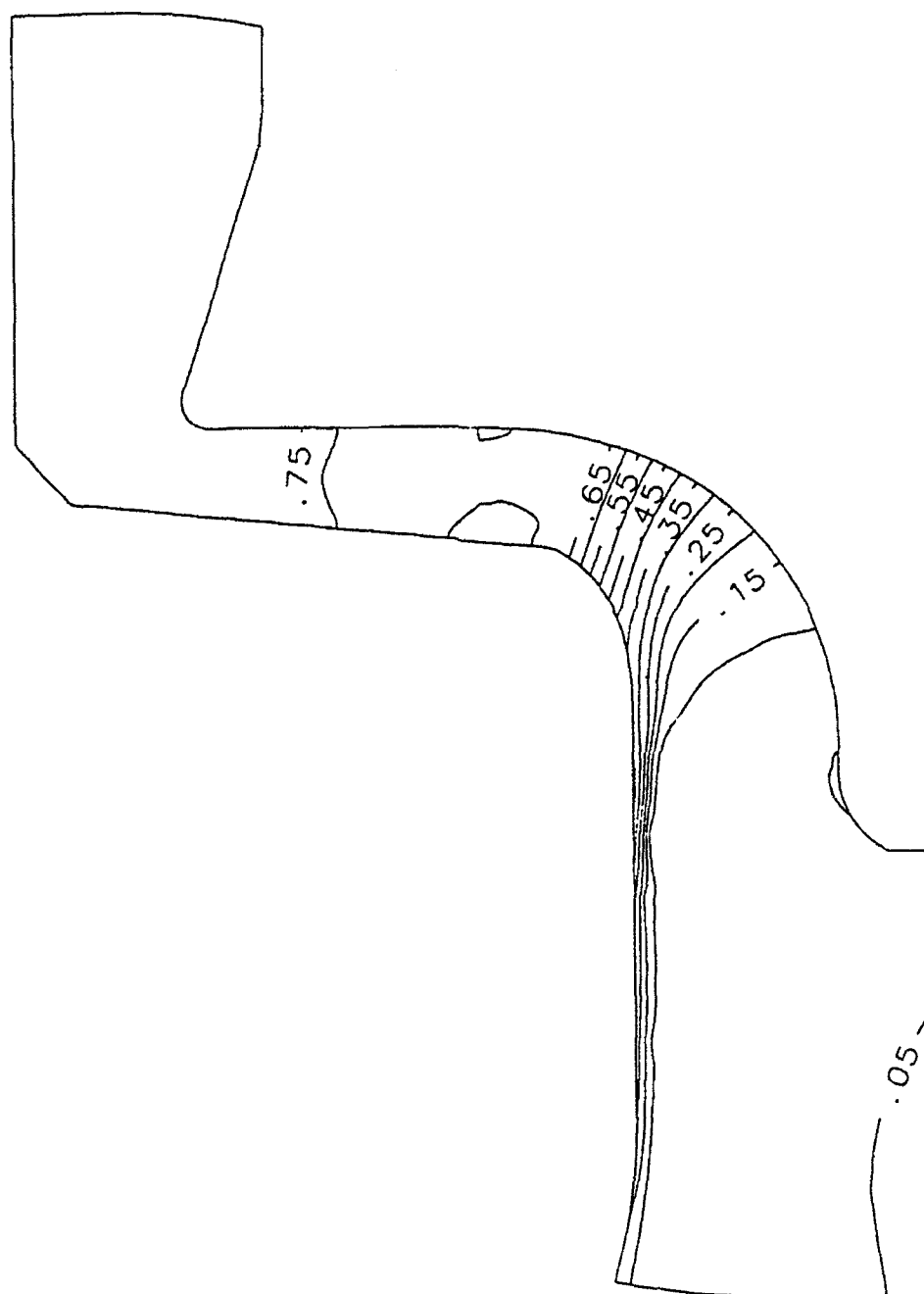


Figure 8.11

Calculated Pressure Contours ($\Psi = 0.14$)

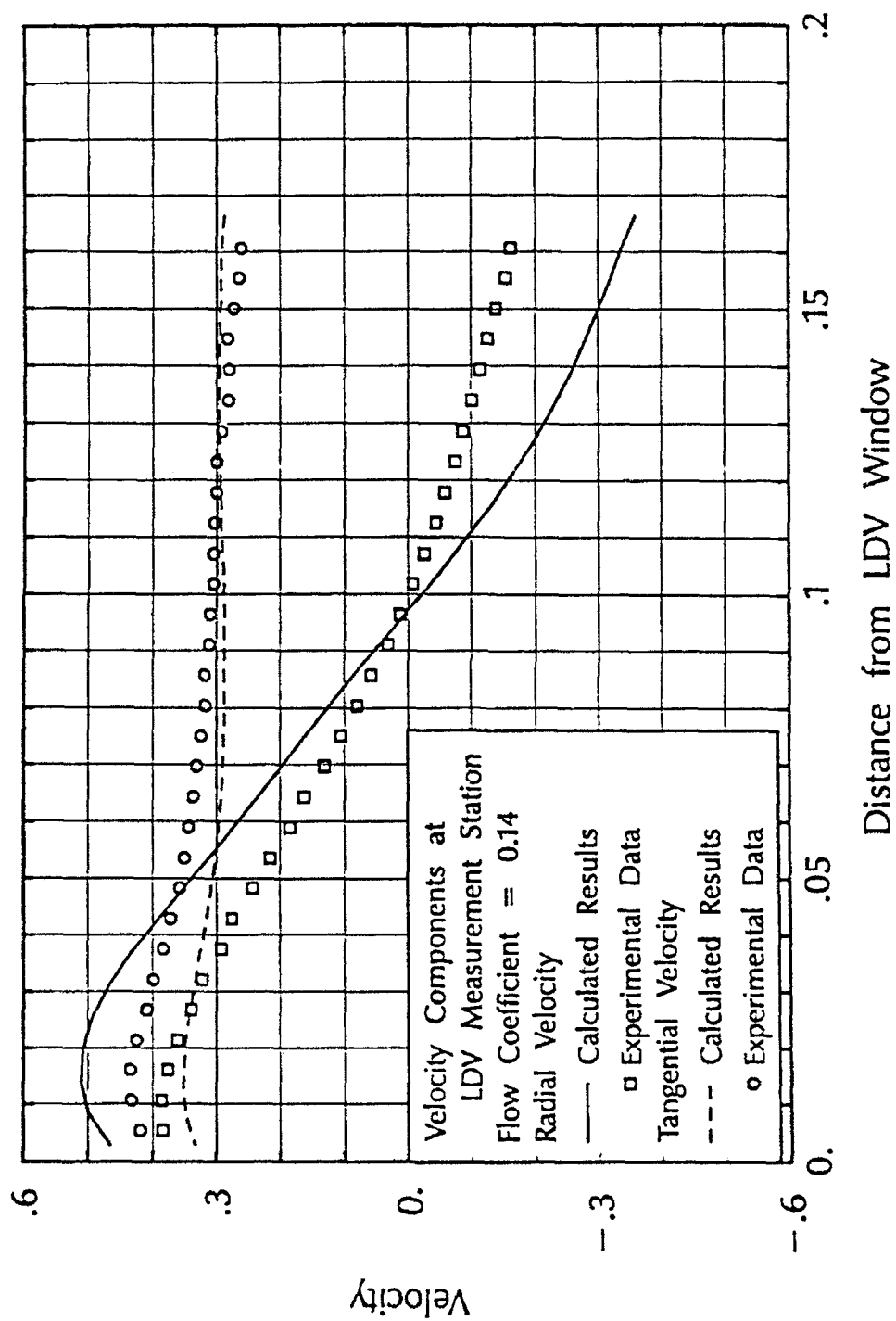


Figure 8.12

Comparison of Calculated and Measured
Radial and Tangential Velocity Profiles
($\phi = 0.14$)

8.1.4 Shutoff Flow Calculation

One of the most drastic operating conditions occurs when flow to the impeller is shutoff, while the impeller continues to be driven at a constant rotational speed. No LDV data was gathered for this type of flow, but video cameras did show a recirculating region upstream of the leading edge of the impeller. The calculated velocity fields in the axisymmetric plane, Fig. 8.13, show that the impeller sets up three recirculating regions. One passing through the leading edge, one within the impeller near the leading edge at the hub, and another one passing through the trailing edge. The flow in these regions rotate in different directions, which is consistent with the need to have a zero net tangential component of vorticity. The pressure contours are shown in Fig. 8.14. The high gradients indicate that the grid was inadequate for this calculation. This was a contributing factor to the instability of this solution. The given results only indicate the nature of the flow. The code was not able to stabilize the recirculating regions when the meridional body forces were used. When the blade model was turned off and the tangential velocities within the blade were set equal to the wheel speed, the solution did converge. Recirculating regions, similar to the two larger ones in Fig. 8.13, were then predicted.

8.2 Comparison of Calculated and Measured Pump Performance Parameters

Global performance parameters are used to specify the overall operating performance of a turbomachine. These parameters typically express the level of energy transfer which takes place and the efficiency of the energy transfer process. The performance parameters are formulated by considering the machine to be a single system, disregarding the spatial variations in the primitive variable fields. A machine can also be considered to be made up of various components, each with its own operating characteristics, acting as a subsystem of the whole. Referring to Fig. 8.1,

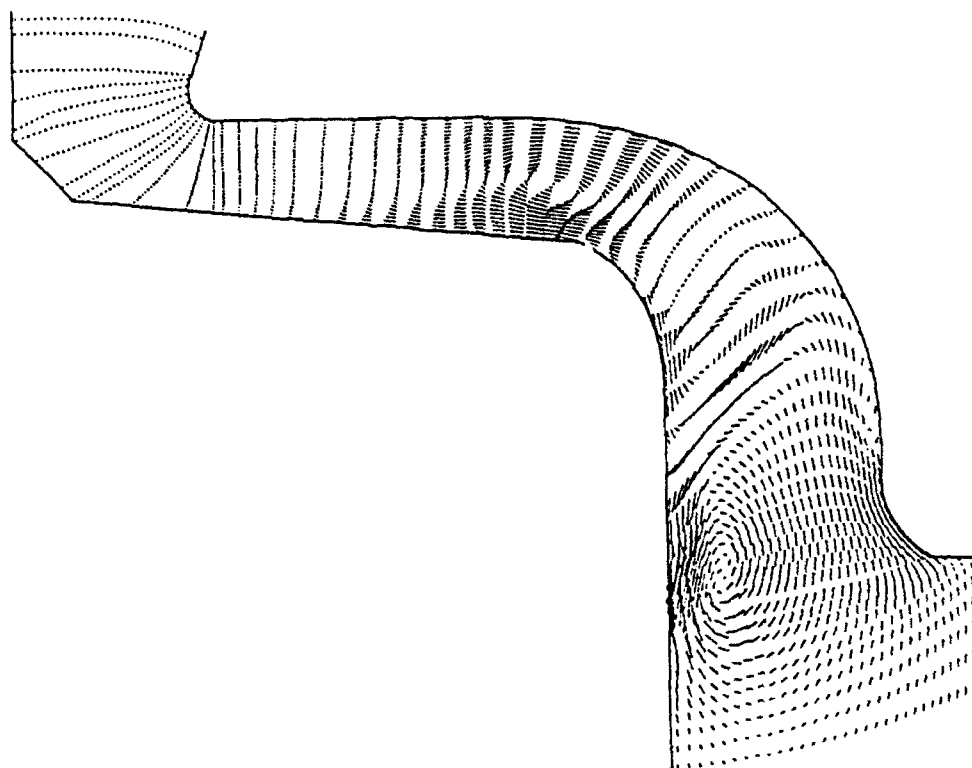


Figure 8.13
Calculated Through-Flow Velocity Vector Field ($\Phi = 0.00$)

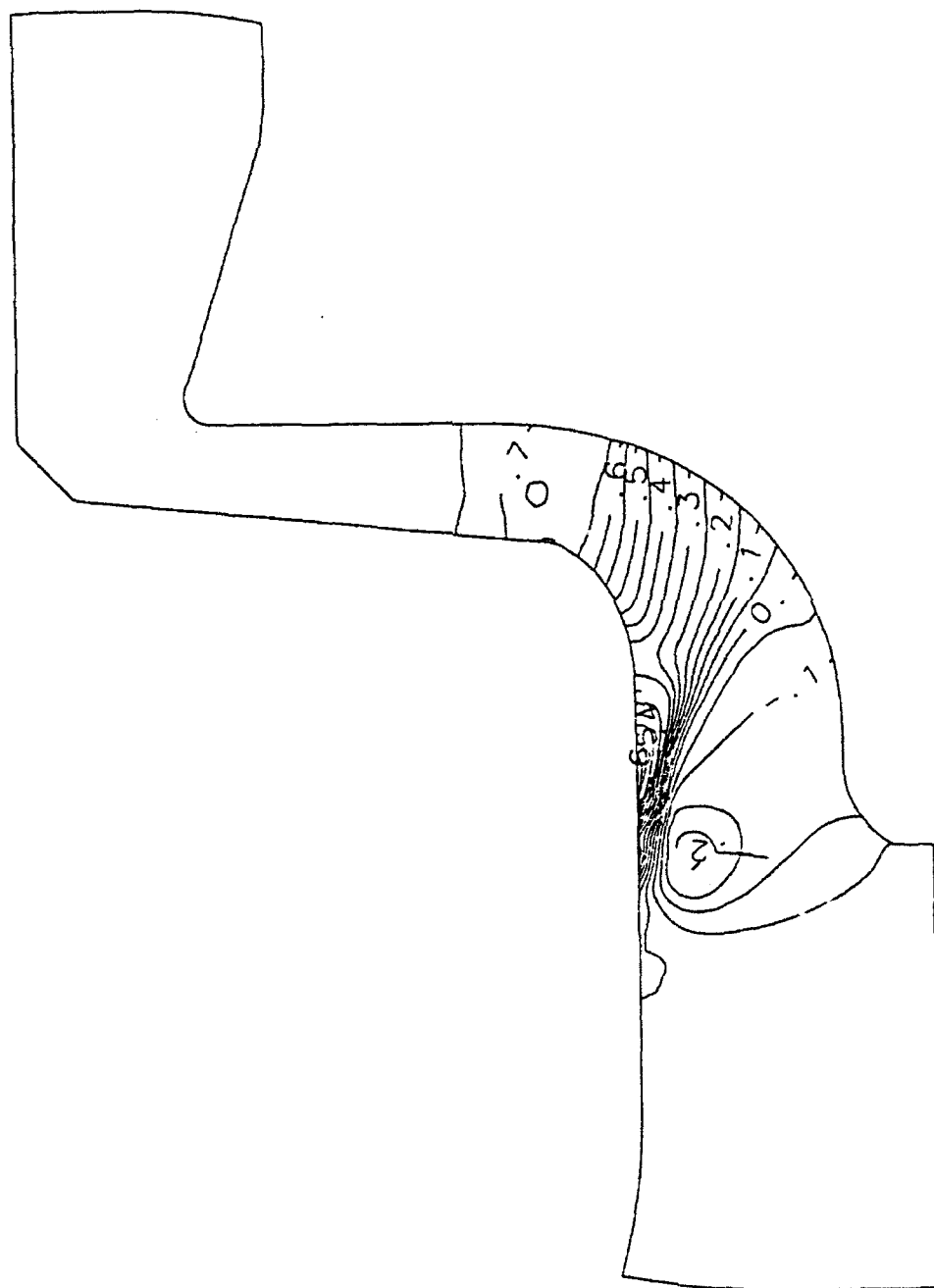


Figure 8.14
Calculated Pressure Contours ($\Phi = 0.00$)

the static head coefficient ψ_{12} reflects the performance of the impeller, while ψ_{13} indicates the overall change in the fluid as it passed through the entire pump (impeller and diffuser).

Perhaps the most important performance parameter for a pump is the static head coefficient, ψ . The primary purpose of a pump is to increase the static pressure of the fluid. The head coefficient indicates the level of head or pressure rise in dimensionless form. Fig. 8.15 shows a comparison of the calculated and measured head rises at two measurement stations. The predicted values are very close to the experimental results for the two higher flow coefficient cases. The other two static head coefficients are consistently high. For the control volumes with reversed flow, the meridional body forces (drag) are possibly too high. Since the pressure rise must counterbalance these forces, this would create a larger than expected pressure rise. The radial diffuser recovers some of the dynamic head, as indicated by the rise in static head coefficient from the second to third measuring station.

A pump works by transferring energy from the impeller to the fluid. The ratio of the useful energy to the total energy transferred to the fluid is known as the efficiency, η . Useful energy delivered by a pump is in the form of a static pressure rise. The total energy delivered to the fluid is related to the energy required to drive the impeller. This value is readily evaluated from the torque and the rotational speed of the impeller. Fig. 8.16 shows a comparison of the calculated and measured efficiencies at two measurement stations. For the efficiency, it is the two lower flow coefficient cases that show a better agreement between the predicted and the measure results. The error in ψ for the low Φ conditions would seem to indicate that the agreement between the efficiencies for these two cases is merely a coincidence. Unresolved deficiencies in the blade model still remain when a streamline, confined to a single row of nodes, is used for reversed flow. Additional static pressure rise occurs in the diffuser, as indicated by the increase in efficiency from station 2 to 3.

The power coefficient represents the dimensionless power required to drive the impeller. It can be calculated directly using the torque on the impeller or it can be evaluated indirectly from

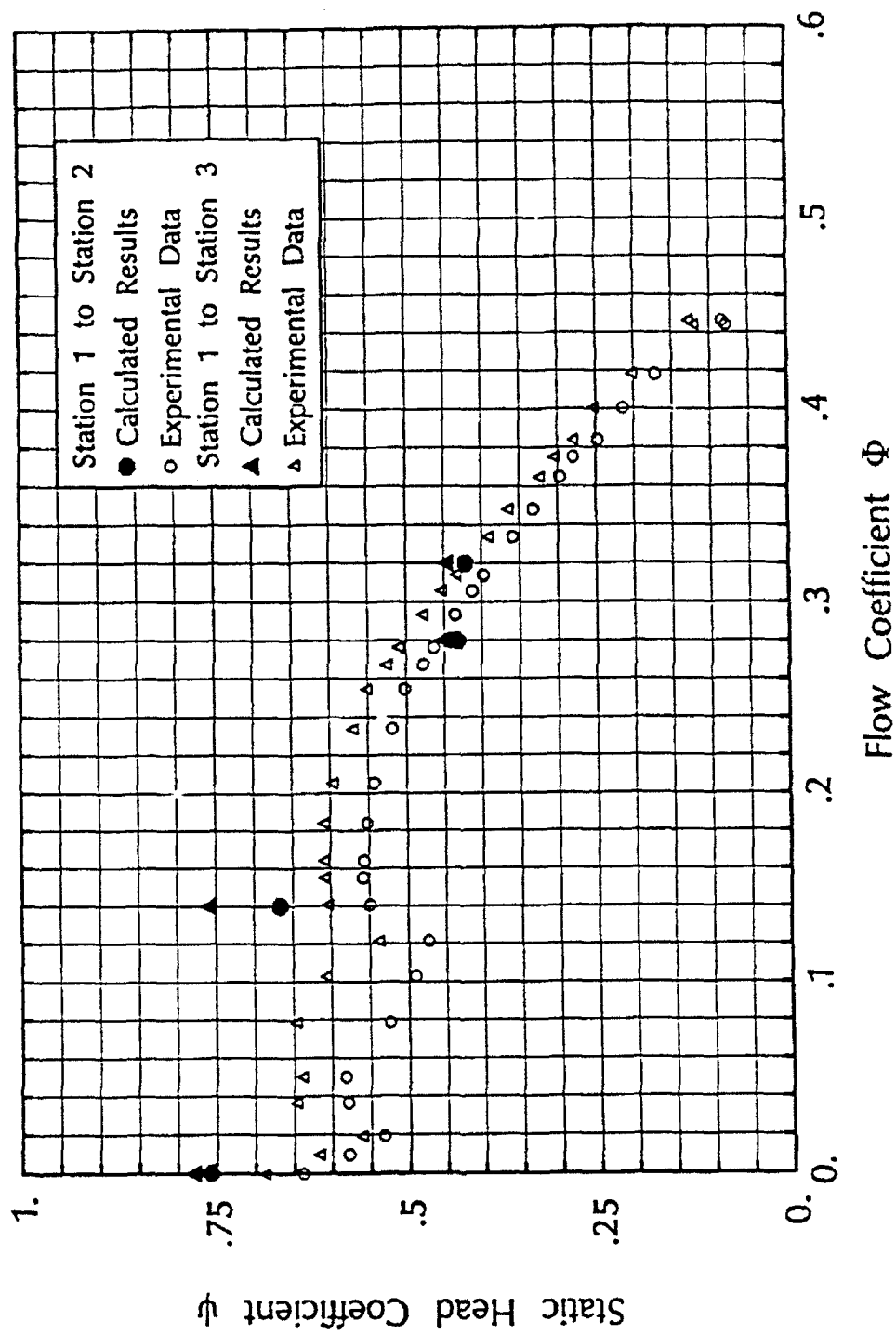


Figure 8.15
 Comparison of Calculated and Measured Static Head Coefficients

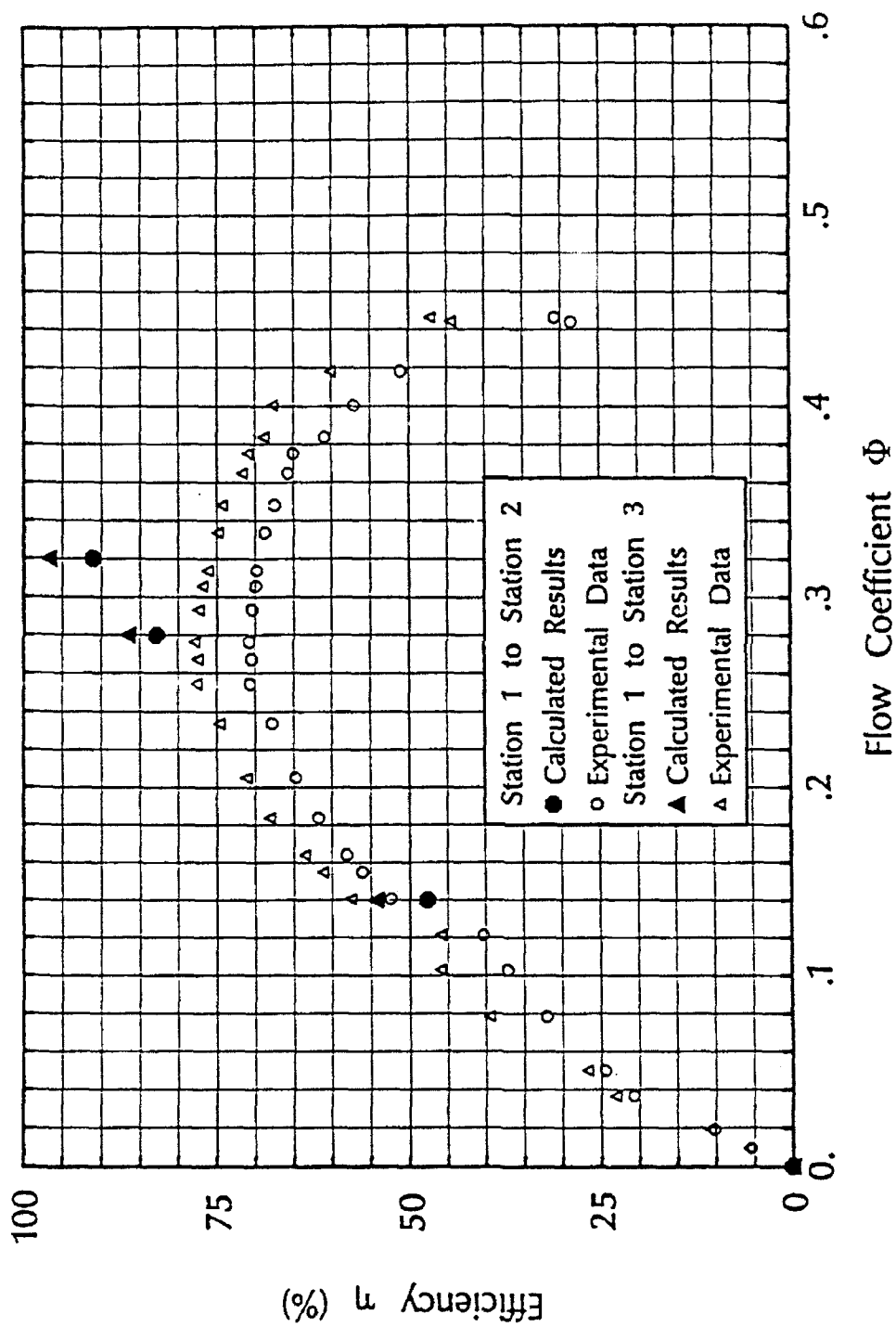


Figure 8.16

Comparison of Calculated and Measured Efficiencies

the head coefficient, flow coefficient, and efficiency. Since these coefficients have been presented in Fig 8.15 and 8.16 the power coefficient does not provide any additional information. However, the values of the power coefficient at the different operating conditions are given in Fig. 8.17 for completeness.

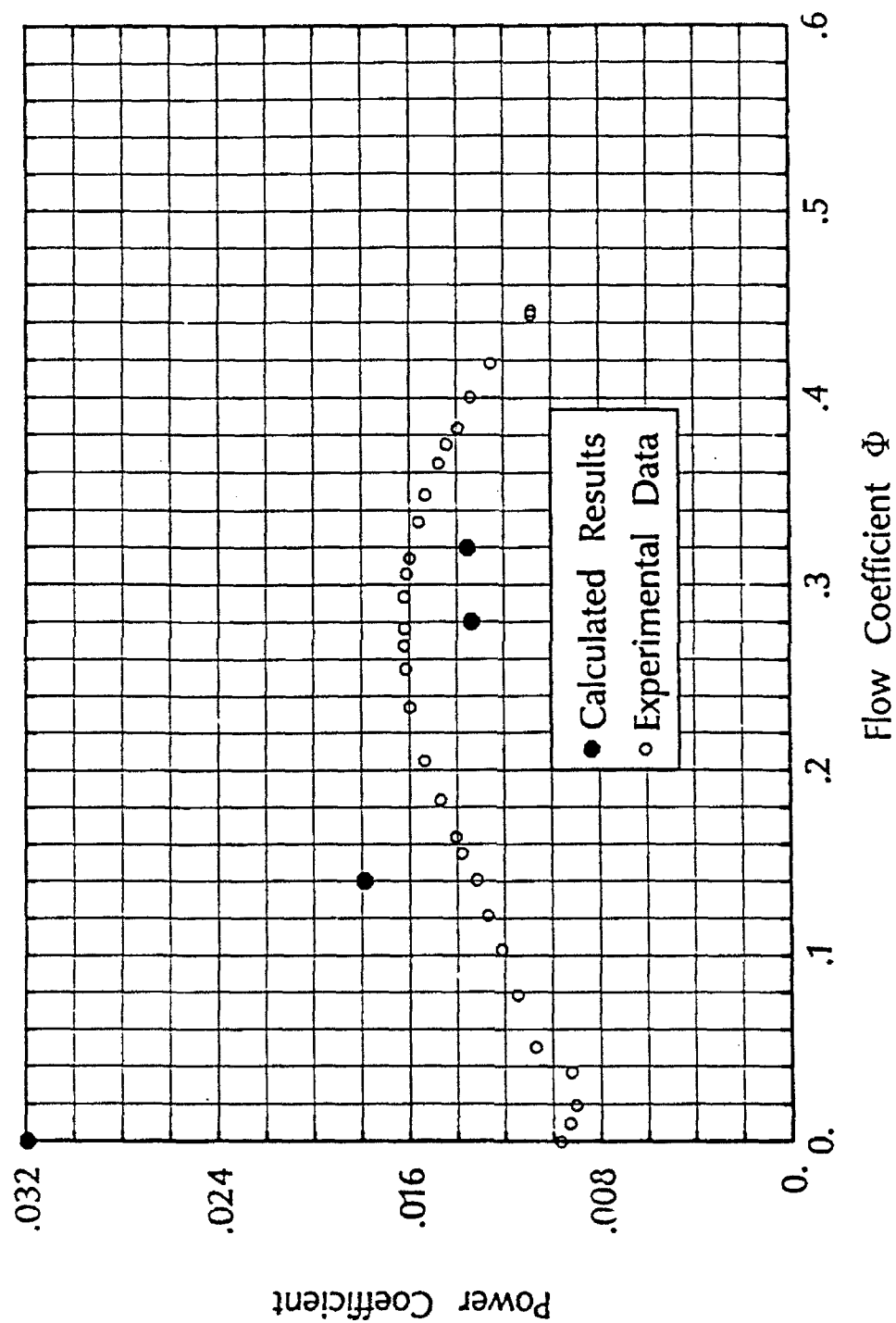


Figure 8.17

Comparison of Calculated and Measured Power Coefficients

Chapter 9

Conclusions and Recommendations

This research has illustrated the usefulness of a Navier-Stokes, axisymmetric flow solver using a non-staggered grid. The Pressure-Weighted Interpolation Method has been successfully modified for the solution of the axisymmetric, steady-state, incompressible, viscous flow equations in generalized curvilinear coordinates. A modified form (QUICKR, QUICK Revised) of Leonard's QUICK differencing has been shown to increase the accuracy of the solution by extending the higher order differencing scheme to the boundaries and, thus, covering the entire flow domain. A lower-order differencing scheme based on convective/diffusive decoupling and a higher-order scheme based on QUICKR have been formulated for use in axisymmetric flows. These schemes modify the standard upwind or convective assumption used in rectangular coordinate spaces to form an analogous assumption that is consistent with the radial dependence of surface area in cylindrical reference frames. Cylindrical upwind and cylindrical QUICKR differencing have been shown to be more accurate than standard differencing schemes for radial flows. For axial flows, they are algebraically equivalent to their Cartesian counterparts. Neither of the cylindrical differencing schemes have presented any stability problems.

The mixed-flow centrifugal impeller through-flow calculations demonstrate the usefulness of the code in indicating overall trends in the flow fields for design and off-design flow conditions. Recirculating regions passing across and out of the blade rows have been successfully calculated. The series of calculations indicates that a redesign of the impeller blades near the discharge may improve the uniformity of the flow entering the diffuser.

Integrating the experiences, both fruitful and unfruitful, undergone while carrying out the research documented herein, yields several topics which are logical extensions of the work already carried out. The most important ones are discussed in the following paragraphs.

A more robust blade model is the most pressing need in bringing the software package to the point of a complete axisymmetric turbomachine design tool. The new model should depend more strongly on the number of blades and incorporate blockage effects. A means for accounting for the blade force normal to the meridional direction in the axisymmetric plane should also be investigated. Reversed flows must also be treated more realistically.

An improved turbulence model should lead to improve results. This is a common need throughout the computational fluid dynamics field, and because a great deal of research is being done on this topic, the literature should be monitored for advances. Most of the more sophisticated turbulence models currently available would also greatly enhance the prediction capabilities.

Solving the pressure corrector equations, a stiff linear system, takes the majority of the computational effort expended in the solution process. A multilevel solver was tested on this set of equations to see if it could accelerate convergence. (Because the momentum equations are easily solved using a tridiagonal algorithm, there seems to be little advantage in using a full multilevel solver on the entire system of governing equations.) The anisotropy of the pressure corrector equation coefficients causes the multilevel solver to exhibit poor convergence characteristics. Further research may yield insight into how information should be transferred from one level to the next when the coefficient amplitudes vary widely in the different directions. Such an understanding should lead to a modified multilevel algorithm with increased convergence rates.

In 1990, Hobson and Lakshminarayana [35,36,37] presented the pressure substitution method (PSM) and presented evidence that it outperforms PWIM. Essentially, PSM treats the " $2 - \delta$ " pressure gradients in the facial contravariant velocities explicitly, and the " $1 - \delta$ " implicitly.

Thus, an expression for pressure, rather than pressure corrector, can be found, permitting the simultaneous solution of all flow fields via a block solver.

Their papers do leave some questions unanswered. The PWIM solver they used employed a Gauss-Seidel iterative solution procedure for solving the sets of equations. A more efficient iterative solver may lessen the advantage PSM has over PWIM. Neither did their PWIM use SIMPLEC, which increases PWIM's velocity-pressure coupling. Also absent from these papers was a discussion as to whether the increased coupling makes up for the added overhead associated with block solvers. This is especially relevant because the momentum equations are easily solved. Despite these questions, the potential benefits of PSM warrant further investigation.

References

1. Novak, R.A., "Streamline Curvature Computing Procedures for Fluid Flow Problems," *J. of Eng. for Power*, Trans. ASME, Series A, Vol. 89, 1967, pp. 478-490.
2. Moore, J., and Moore, J.G., "Three-Dimensional Viscous Flow Calculations for Assessing the Thermodynamic Performance of Centrifugal Compressors, Study of the Eckardt Compressor," *Proceedings, AGARD Meeting on Centrifugal Compressors, Flow Phenomena, and Performance*, Brussels, Belgium, 1980.
3. Hah, C., "A Navier-Stokes Analysis of Three-Dimensional Turbulent Flows Inside Turbine Blade Rows at Design and Off-Design Conditions," *Journal of Eng. for Gas Turbines and Power*, Vol. 106, April, 1984, pp. 421-429.
4. Moore, J., and Moore, J.G., "Performance Evaluation of Linear Turbine Cascades Using Three-Dimensional Viscous Flow Calculations," ASME Paper No. 85-GT-65, 1985.
5. Hah, C., "A Numerical Study of Three-Dimensional Flow Separation and Wake Development in an Axial Compressor Rotor," ASME Paper No. GT-84-34, 1984.
6. Lapworth, B.L., and Elder, R.L., "Computation of the Jet-Wake Flow Structure in a Low Speed Centrifugal Impeller," AIAA Paper No. 88-GT-217, 1988.
7. Hah, C., Bryans, A.C., Moussa, Z., and Tomsho, M.E., "Application of Viscous Flow Computations for the Aerodynamic Performance of a Backswept Impeller at Various Operating Conditions," *J. of Turbomachinery*, Vol. 110, July, 1988, pp. 303-311.
8. Novak, R.A., and Hearsey, R.M., "A Nearly Three-Dimensional Intrablade Computing System for Turbomachinery," *J. Fluids Eng.*, Vol. 99, 1977, pp. 154-166.
9. Martelli, F., and Michelassi, V., "Using Viscous Calculations in Pump Design," *Proceedings, 3rd Joint ASCE/ASME Fluid Mechanics Conference*, San Diego, CA, July, 1989, pp. 157-166.
10. Jackson, R., Wood, N.B., and Boston, A., "The Efficient Modeling of Blade Lean Effects Within the Turbomachinery Throughflow Method," *Int. J. Heat and Fluid Flow*, Vol. 10, No. 1, March, 1989, pp. 32-39.
11. Rhie, C.M., and Chow, W.L., "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation," *AIAA Journal*, Vol. 21, No. 11, November, 1983, pp. 1525-1532.
12. Leonard, B.P., "A Stable and Accurate Convective Modeling Procedure Based on Quadratic Upstream Interpolation," *Computer Meth. in Applied Mech. and Eng.*, Vol. 19, 1979, pp. 59-98.
13. Hughes, W.F., and Gaylord, E.W., *Basic Equations of Engineering Science, Schaum's Outline Series*, McGraw Hill Book Co., New York, NY, 1964, pp. 14-15.

14. Thompson, Philip A., *Compressible-Fluid Dynamics*, McGraw Hill Book Co., New York, NY, 1972, pp. 633-635.
15. Hoffman, Klaus A., *Computational Fluid Dynamics for Engineers*, Engineering Education System, Austin, TX, 1989.
16. Patankar S.V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Co., Washington D.C., 1980.
17. Schlichting, Hermann, *Boundary-Layer Theory*, McGraw Hill Book Co., New York, NY, 1979.
18. Karki, K.C., and Patankar, S.V., "Calculation Procedure for Viscous Incompressible Flows in Complex Geometries," *Numerical Heat Transfer*, Vol. 14, 1988, pp. 295-307.
19. Patankar, S.V., and Spalding, D.B., "A Calculation Procedure for Heat, Mass, and Momentum Transfer in Three-Dimensional Parabolic Flows," *Int. J. Heat and Mass Transfer*, Vol. 15, 1972, pp. 1787-1804.
20. Patankar, S.V., "A Calculation Procedure for Two-Dimensional Elliptic Situations," *Numerical Heat Transfer*, Vol. 4, 1981, pp. 409-425.
21. Issa, R.I., "Solution of the Implicitly Discretised Fluid Flow Equations by Operator Splitting," *Journal of Computational Physics*, Vol. 62, 1985, pp. 40-65.
22. Lapworth, B.L., "Examination of Pressure Oscillations Arising in the Computation of Cascade Flow Using a Boundary-Fitted Co-Ordinate System," *Int. J. for Num. Meth. in Fluids*, Vol. 8, 1988, pp. 387-404.
23. Miller, T.F., and Schmidt, F.W., "Use of a Pressure-Weighted Interpolation Method for the Solution of the Incompressible Navier-Stokes Equations on a Nonstaggered Grid System," *Numerical Heat Transfer*, Vol. 14, 1988, pp. 213-233.
24. Peric, M., Kessler, R., and Scheuerer, G., "Comparison of Finite-Volume Numerical Methods with Staggered and Colocated Grids," *Computers and Fluids*, Vol. 16, No. 4, 1988, pp. 389-403.
25. Stone, H.L., "Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations," *SIAM Journal of Numerical Analysis*, Vol. 5, No. 3, 1968, pp. 530-558.
26. Van Doormaal, J.P., and Raithby, G.D., "Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows," *Numerical Heat Transfer*, Vol. 7, 1984, pp. 147-163.
27. Acharya, S., and Moukalled, F.H., "Improvements to Incompressible Flow Calculations on a Non-Staggered Curvilinear Grid," *Numerical Heat Transfer*, Vol. 15, 1989, pp. 131-152.
28. Miller, T.F., and Schmidt, F.W., "Evaluation of a Multilevel Technique Applied to the Poisson and Navier-Stokes Equations," *Numerical Heat Transfer*, Vol. 13, 1988, pp. 1-26.

29. Baldwin, B.S., and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78-257, AIAA 16th Aerospace Sciences Meeting, Huntsville, AL, Jan., 1978.
30. Rubin, S.G., and Himansu, A., "Convergence Properties of High-Reynolds Number Separated Flow Calculations," *Int. J. for Num. Meth. in Fluids*, Vol. 9, 1989, pp. 1395-1411.
31. Kunz, R., Private Communication, The Pennsylvania State University, University Park, PA, Oct., 1990.
32. Fox, R.W., and McDonald, A.T., *Introduction to Fluid Mechanics*, John Wiley & Sons, New York, NY, 1985, p. 448.
33. Dixon, S.L., *Fluid Mechanics, Thermodynamics of Turbomachinery*, Pergamon Press, New York, NY, 1975, pp. 185-215.
34. Eiseman, Peter R., "A Control Point Form of Algebraic Grid Generation," *Int. J. for Num. Meth. in Fluids*, Vol. 8, 1988, pp. 1165-1181.
35. Hobson, G., and Lakshminarayana, B., "Fully Elliptic Incompressible Flow Calculations on Regular Grid by a New Pressure Substitution Method," AIAA Paper 90-0239, 28th Aerospace Sciences Meeting, Reno, NV, Jan., 1990.
36. Hobson, G., and Lakshminarayana, B., "Prediction of Cascade Performance Using an Incompressible Navier-Stokes Technique," ASME-GT-261, ASME Gas Turbine and Aeroengine Congress and Exposition, Brussels, Belgium, June, 1990.
37. Hobson, G., and Lakshminarayana, B., "Computation of Turbine Flowfields with a Navier-Stokes Code," AIAA Paper 90-2122, AIAA/SAE/ASME/ASEE 26th Joint Propulsion Conference, Orlando, FL, July, 1990.
38. Spalding, D.B., "A Novel Finite-Difference Formulation for Differential Expressions Involving Both First and Second Derivatives," *Int. J. of Num. Meth. Eng.*, Vol. 4, 1972, pp. 551-559.
39. Leschziner, M.A., "Modeling Turbulent Recirculating Flows by Finite-Volume Methods - Current Status and Future Directions," *Int. J. of Heat and Fluid Flow*, Vol. 10, No. 3, Sept., 1989, pp. 186-202.
40. Phillips, R.E., and Schmidt, F.W., "Multigrid Techniques for the Solution for the Passive Scalar Advection-Diffusion Equation," *Numerical Heat Transfer*, Vol. 8, 1985, pp. 25-43.

Appendix

Differencing Schemes

During the discussion of how the governing equations are discretized and solved using the PWIM algorithm, it was stated that the integral form of the equations must be converted to a linear, discretized equation. For a general transport quantity the equation took a five-point stencil format:

$$(A_P - S_P)\phi_O = A_E\phi_E + A_W\phi_W + A_N\phi_N + A_S\phi_S + S_U \quad (A.1)$$

The purpose of this appendix is to explain how these multiplicative coefficients, A_i , are calculated.

The total flux across a control volume face is a linear combination of convective and diffusive fluxes. The flux across a vertical face (Eq. A.2a) and a horizontal face (Eq. A.2b) were derived in Chapter 4.

$$F_e^\xi = (\eta\rho r U \phi)_e - (\eta\Gamma r J g_{11} \phi_\xi)_e \quad (A.2a)$$

$$F_n^\eta = (\xi\rho r V \phi)_n - (\xi\Gamma r J g_{22} \phi_\eta)_n \quad (A.2b)$$

Second-order accurate central differencing can be used to evaluate the diffusive flux, but the convective flux is difficult to evaluate as it is a function of the transported quantity at the control volume face. Since the transported quantity is defined only at the nodes, the nodal values must be used to approximate the facial value. The approximation process combined with the finite difference expansion of the diffusive flux is called a differencing scheme.

Differencing schemes evaluate the total flux in terms of linear functions of the nodal quantities. If only the two nodes surrounding the control volume face are used in the linear representation of the flux, the flux can be represented according to Eq. A.3. The "c" and "d"

superscripts on the coefficients multiplying the nodal quantities correspond to convective and diffusive influences, respectively.

$$F_e^t = (a_O^c \phi_O - a_E^c \phi_E) + (a_O^d \phi_O - a_E^d \phi_E) \quad (A.3)$$

Expressing the total flux through the remaining three control volume faces in a similar manner and gathering common terms (i.e. a_E^c and a_E^d) places the linear discretized equation in the following format:

$$A_P \phi_O - A_E \phi_E - A_W \phi_W - A_N \phi_N - A_S \phi_S = \tilde{S} \quad (A.4)$$

where the source terms have all been absorbed into $\tilde{S}$ and the coefficient A_P is composed of contributions from fluxes through each of the four CV faces. Moving all but the central node terms to the right hand side of the equation yields the form given in Eq. A.1. The differencing scheme can thus be described by the manner in which the coefficients A_i are formed. The remainder of the appendix is devoted to presenting various differencing schemes. The scheme that is appropriate for any given flow calculation is dependent on the geometry, fineness of the grid, and needs of the user.

Because the diffusive coefficient and the control volume over which the governing equations are integrated are identical for each momentum equation, the A_i coefficients are also identical. The presence of only one type of control volume on the nonstaggered grid results in less computational effort being expended in the discretization process, as compared to that required for staggered grids.

A.1 Lower-Order Differencing Schemes

When only two nodal values appear in the linear expression for the total flux, the differencing scheme is referred to as a lower-order scheme. Higher-order differencing schemes use three or

more nodal values to approximate the total flux through a particular CV face. These higher-order schemes are discussed in Section A.2. In the present section, a number of lower-order differencing schemes are presented in terms of the coefficients, A_i . Only the coefficients for the eastern, western, and central nodes are formally stated. Those for the northern and southern nodes are evaluated in a manner analogous to the eastern and western coefficients, respectively.

Lower-order differencing schemes are conveniently expressed in terms of the local, or cell, Reynolds number. The computational space cell Reynolds numbers are:

$$Re_{cell} = C/D \quad (A.5)$$

$$C_e = (\rho r U \Delta \eta)_e \quad (A.6a)$$

$$D_e = (\Gamma r J g_{11} \Delta \eta / \Delta \xi)_e \quad (A.6b)$$

$$C_n = (\rho r V \Delta \xi)_n \quad (A.7a)$$

$$D_n = (\Gamma r J g_{22} \Delta \xi / \Delta \eta)_n \quad (A.7b)$$

Evaluating the diffusive gradient using central differencing (i.e. $\phi_\xi|_e = (\phi_E - \phi_O)/\Delta \xi$) yields the following expression for the total flux:

$$F_e^\xi = D_e [Re_e \phi_e - (\phi_E - \phi_O)] \quad (A.8a)$$

$$F_n^\eta = D_n [Re_n \phi_n - (\phi_N - \phi_O)] \quad (A.8b)$$

By introducing the contravariant velocities and the transformation metrics, it is possible to define the cell Reynolds numbers in the computational domain as given in Eqs. 4.5-4.7. Therefore, any differencing scheme based on a physical space Re_{cell} can also be utilized in the computational plane.

Upwind Differencing:

$$A_E = D_e (1 + [-Re_e, 0]) \quad (A.9a)$$

$$A_W = D_w (1 + [Re_w, 0]) \quad (A.9b)$$

Upwind differencing [16, p. 105] states that the convected quantity is equal to the value at the upstream node. The upstream direction is determined by the intrinsic FORTRAN MAX function ([]). Upwind differencing is only first-order accurate in the sense of Taylor series truncation error. However, the truncation error is not an indication of the overall accuracy of the scheme. Instead, it indicates how quickly the error falls off when the grid size is changed. Upwind differencing accurately reflects the true physics of convection dominated flows. For convection dominated, one-dimensional flow, upwind differencing yields a solution which is much closer to the analytic solution of the convection-diffusion equation (Eq. A.10) than that produced by approximating the facial quantity using second-order accurate central differencing [16, p. 105].

$$Re\phi_{\xi} - \phi_{\xi\xi} = 0 \quad (A.10)$$

However, for multi-dimensional applications, large amounts of numerical diffusion occur if the flow is skewed with respect to the gridlines [16, pp. 105-109]. This diffusion can contaminate the results with an unacceptable amount of error.

Central Differencing:

$$A_E \approx D_e(1 - \frac{1}{2}Re_e) \quad (A.11a)$$

$$A_W \approx D_w(1 + \frac{1}{2}Re_w) \quad (A.11b)$$

Averaging nodal quantities to arrive at an expression for the convective flux is known as central differencing [16, pp. 103]. Central differencing cannot be used for high Reynolds number flows on coarse grids, because this scheme becomes unstable for cell Reynolds numbers greater than 2. The instability is a result of the A_i coefficients becoming negative. When a coefficient becomes negative, positive definiteness is destroyed, physically unrealistic variable dependencies are formed, and iterative solutions fail to converge [16, pp. 103-104]. Some differencing schemes make use of central differencing when it is stable or add artificial dissipation to enhance stability. It is included in this discussion for completeness.

Hybrid Differencing:

$$A_E = D_e \left(\left[\frac{1}{2} Re_e, 1 \right] - \frac{1}{2} Re_e \right) \quad (A.12a)$$

$$A_W = D_w \left(\left[\frac{1}{2} Re_w, 1 \right] + \frac{1}{2} Re_w \right) \quad (A.12b)$$

Hybrid differencing [38] switches the scheme between central and upwind differencing. When $Re < 2$, it is equivalent to central differencing. When central differencing would become unstable, the hybrid scheme switches to upwind differencing. Hybrid differencing is effectively a rough curve fit approximation to the analytic solution of the one-dimensional convection-diffusion equation. Hybrid differencing is a widely used scheme [23,38,39], because it is unconditionally stable, is computationally inexpensive to use, and exhibits favorable convergence properties.

Power Law Differencing:

$$A_E = D_e \left(\left[0, \left(1 - \frac{1}{10} |Re_e| \right)^5 \right] + [-Re_e, 0] \right) \quad (A.13a)$$

$$A_W = D_w \left(\left[0, \left(1 - \frac{1}{10} |Re_w| \right)^5 \right] + [Re_w, 0] \right) \quad (A.13b)$$

Power law differencing [20] is a very close curve fit to the analytic solution of the one-dimensional convection-diffusion equation. For $Re < 10$ a type of central differencing is used, and for $Re \geq 10$ upwind differencing arises. Previous researchers [20, 39] have chosen power law differencing over hybrid differencing, despite the fact that it is computationally more expensive, because it yields more accurate results.

For all of the lower order differencing schemes, the central coefficient is equal to the sum of the neighboring coefficients plus the mass source.

$$A_P = A_E + A_W + A_N + A_S + (C_e - C_w) + (C_n - C_s) \quad (A.14)$$

$$m_o = C_e - C_w + C_n - C_s \quad (A.15)$$

The mass source goes to zero at convergence, so it may also be ignored during intermediate solutions. However, retaining it prevents stability problems, because the individual mass flow

rates arise from the linearization of the convective flux terms. For the stable differencing schemes, upwind, hybrid, and power law, A_P will never drop below zero, even if the mass source is large and negative. However, A_P can become zero with some differencing schemes if physically unrealistic flow situations arise.

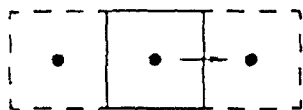
A.2 QUICKR Differencing

In an effort to maintain the description of the true physics of the flow provided by upwind differencing without incurring the corresponding numerical diffusion, Leonard [12] proposed Quadratic Upwind Interpolation of Convective Kinematics (QUICK) differencing. Three-point upwind-weighted differencing yields third-order accuracy in the expression of the facial values (ϕ_e, ϕ_n). This Taylor series expansion uses three nodal values, two upstream and one downstream. Which nodes are used depends upon the direction of the flow. Central differencing is still used to evaluate the diffusive component of the flux, so the formal accuracy of QUICK is reduced to second-order. This is not entirely indicative of the accuracy of the overall scheme, because the diffusive and convective fluxes are rarely of the same order of magnitude.

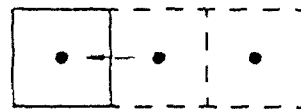
Figs. A.1a-h show which nodes are used with the QUICK differencing scheme for forward and reversed flow through the eastern, western, northern, and southern control volume faces. If the convective velocity is zero, then it is immaterial how the facial quantity is approximated, since it will be multiplied by the null mass flux. For a representative eastern face with a positive (Eq. A.16a) or negative (Eq. A.16b) convective velocity, the truncated Taylor series approximation for the facial value is:

$$\phi_e = (3\phi_E + 6\phi_O - \phi_W)/8 \quad (A.16a)$$

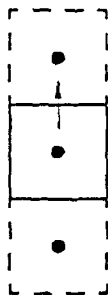
$$\phi_e = (3\phi_O + 6\phi_E - \phi_{EE})/8 \quad (A.16b)$$



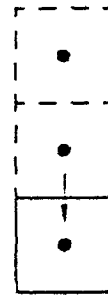
a. Positive Flow through Eastern Face



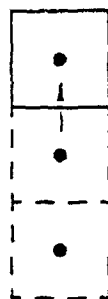
b. Negative Flow through Eastern Face



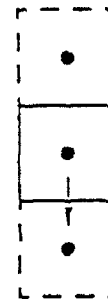
c. Positive Flow through Northern Face



d. Negative Flow through Northern Face



e. Positive Flow through Southern Face



f. Negative Flow through Southern Face



g. Positive Flow through Western Face



h. Negative Flow through Western Face

Figure A.1

QUICK Differencing Stencils

For this higher-order differencing scheme, not only are the neighboring nodes brought into the expression, but the nodes two control volumes away from the central node also appear in the stencil. Because of this, the implementation of this differencing scheme at nodes near the boundaries is more involved. Fig. A.2 shows a typical computational space grid. The CV faces on which QUICK can be used are indicated. A " + " superscript indicates that QUICK can only be used at that face if the flow is in the positive direction. A " - " superscript denotes a face for which QUICK can only be used if the flow is reversed. If QUICK can be used regardless of flow direction, no superscript is used. QUICK differencing cannot be used on the faces lying on the boundary under any flow conditions, neither can the lower-order differencing schemes. These faces are handled by an application of the boundary conditions (see Section 4.6). The problem from the standpoint of QUICK lies in treating faces opposite the boundaries. These faces, marked in Fig. A.2 as dependent on flow direction, bring in control volumes that lie outside of the computational domain. To prevent this, Phillips and Schmidt [40] suggest turning off QUICK differencing at the nodes whose stencil encompasses nodes which lie outside the computational domain and using a lower-order scheme instead. However, this causes nonconservative solutions. For example, the northern face of the grid node (3,3) is hybrid differenced, but the same face acting as the southern wall of node (3,4) is treated using QUICK. The residual for a system treated in this manner never goes to zero, making it more difficult to detect when a converged solution is reached. Also, the accuracy drops near the wall, where flow gradients are most drastic and precision most needed. Miller and Schmidt [23] report that mixing lower and higher order differencing schemes produces imperceptible error. This is not surprising, since the primary case they were studying was a driven cavity where the convective flux near all of the computational boundaries is minimal. However, this author's research has shown that this treatment causes a significant amount of error when mass flux enters the computational domain (see Section 7.1.3). The error does not appear in the data presented by Miller and Schmidt, because the error occurs in the inlet region, and their data only shows the flow field at the center of the domain.

E^-	W^-	E	W	E	W	E	W	E	W	E	W	E^+	W^+
S^+	S^+	S^+	S^+	S^+	S^+	S^+	S^+	S^+	S^+	S^+	S^+	S^+	S^+
N^+	N^+	N^+	N^+	N^+	N^+	N^+	N^+	N^+	N^+	N^+	N^+	N^+	N^+
E^-	W^-	E	W	E	W	E	W	E	W	E	W	E^+	W^+
S	S	S	S	S	S	S	S	S	S	S	S	S	S
N	N	N	N	N	N	N	N	N	N	N	N	N	N
E^-	W^-	E	W	E	W	E	W	E	W	E	W	E^+	W^+
S	S	S	S	S	S	S	S	S	S	S	S	S	S
N	N	N	N	N	N	N	N	N	N	N	N	N	N
E^-	W^-	E	W	E	W	E	W	E	W	E	W	E^+	W^+
S	S	S	S	S	S	S	S	S	S	S	S	S	S
N	N	N	N	N	N	N	N	N	N	N	N	N	N
E^-	W^-	E	W	E	W	E	W	E	W	E	W	E^+	W^+
S	S	S	S	S	S	S	S	S	S	S	S	S	S
N	N	N	N	N	N	N	N	N	N	N	N	N	N
E^-	W^-	E	W	E	W	E	W	E	W	E	W	E^+	W^+
S^-	S^-	S^-	S^-	S^-	S^-	S^-	S^-	S^-	S^-	S^-	S^-	S^-	S^-
N^-	N^-	N^-	N^-	N^-	N^-	N^-	N^-	N^-	N^-	N^-	N^-	N^-	N^-
E^-	W^-	E	W	E	W	E	W	E	W	E	W	E^+	W^+

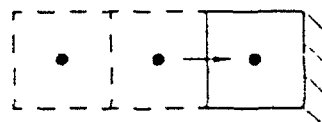
Figure A.2

Flow Conditions which Permit QUICK Differencing on the Control Volume Faces

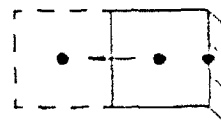
In an effort to treat the problems of QUICK differencing near the boundary, two approaches were investigated. The first was to adapt Phillips and Schmidt's suggestion so that QUICK was used wherever the flow directions allowed, and where it was prohibited, hybrid differencing was used. By basing the differencing on the control volume face rather than on the nodes, a conservative differencing algorithm arises. This approach was not entirely successful because the transition from regions using hybrid differencing into those using QUICK results a sudden change in accuracy. The resulting flow fields have a corresponding discontinuity (see Section 7.1.3 for further discussion of this phenomenon).

A second, and successful, solution to the problems caused by using QUICK differencing near the boundary was to extend Leonard's original approach so that it is applicable to CV faces that are opposite the boundary. Flow quantities are defined on the boundary, so the boundary node is used in the Taylor series approximation of the convected quantity. Whereas the standard QUICK differencing uses only nodal values centered in control volumes to approximate the facial quantity, the new differencing scheme uses the nodes on each side of the control volume face and the node on the boundary. Figs. A.3a-h show the control volumes on the east, west, north and south boundaries, and indicate the flow conditions which make use of the boundary node in the new scheme. The acronym QUICKR (for QUICK, Revised) is used to distinguish the new differencing scheme from the original QUICK. QUICKR maintains higher-order accuracy throughout the flow domain, whereas QUICK has higher-order accuracy only in the interior. A "r" is appended to the QUICKR symbols when it is necessary to distinguish the new Taylor series approximation (Eq. A.17) from the standard QUICK series (Eq. A.16). The three-point upwind-weighted Taylor series approximation that is used in QUICKR for the control volume faces opposite the boundaries is shown in Eq. A.17 for positive flow through the eastern face of the control volume that borders on the western boundary (Fig. A.3g).

$$c_e = (3c_o + c_E - c_W)/3 \quad (\text{A.17})$$



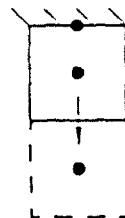
a. Positive Flow through Western Face
across from Eastern Boundary



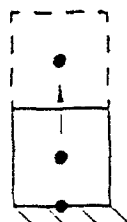
b. Negative Flow through Western Face
across from Eastern Boundary



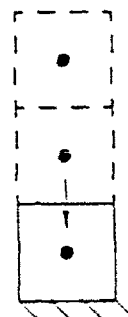
c. Positive Flow through Southern Face
across from Northern Boundary



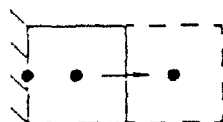
d. Negative Flow through Southern Face
across from Northern Boundary



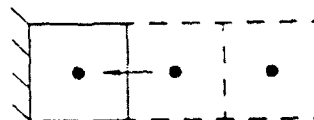
e. Positive Flow through Northern Face
across from Southern Boundary



f. Negative Flow through Northern Face
across from Southern Boundary



g. Positive Flow through Eastern Face
across from Western Boundary



h. Negative Flow through Eastern Face
across from Western Boundary

Figure A.3

QUICKR Differencing Stencils

Using this approach, high accuracy is maintained throughout the flow domain, including the regions near the walls where large flow gradients make the additional accuracy very desirable.

Implementing QUICKR's directional sensitivity is done through the use of switching factors that become either positive or zero, depending on the flow direction. Defined in Eq. A.18, these switching factors are based on the convective flux component.

$$\begin{aligned} S_e^+ &= (|C_e| + C_e)/2 & S_e^- &= (|C_e| - C_e)/2 \\ S_w^+ &= (|C_w| + C_w)/2 & S_w^- &= (|C_w| - C_w)/2 \end{aligned} \quad (A.18)$$

For flow in the positive direction, the switches with a "+" superscript equal the mass flux, and the "-" superscribed switches are zero. When the flow reverses, the former switches become zero and the latter are equal to the absolute value of the mass flux.

Discretizing the flux terms for each of the four sides and collecting common nodal factors yields the QUICKR differencing equations. For the face or faces which require the use of the modified Taylor series expansion, (a' , b' , c' , and d') of Eq. A.20 are substituted for (a , b , c , and d) of Eq. A.19. These faces are marked with either a "+" or "-" superscript in Fig. A.2.

$$\begin{aligned} a_e^+ &= \frac{3}{4} & b_e^+ &= \frac{3}{8} & c_e^+ &= -\frac{1}{8} & a_e^- &= \frac{3}{8} & b_e^- &= \frac{3}{4} & d_e^- &= -\frac{1}{8} \\ a_w^+ &= \frac{3}{8} & b_w^+ &= \frac{3}{4} & d_w^+ &= -\frac{1}{8} & a_w^- &= \frac{3}{4} & b_w^- &= \frac{3}{8} & d_w^- &= -\frac{1}{8} \end{aligned} \quad (A.19)$$

$$\begin{aligned} a_e'^+ &= 1 & b_e'^+ &= \frac{1}{3} & c_e'^+ &= -\frac{1}{3} & a_e'^- &= \frac{1}{3} & b_e'^- &= 1 & d_e'^- &= -\frac{1}{3} \\ a_w'^+ &= \frac{1}{3} & b_w'^+ &= 1 & d_w'^+ &= -\frac{1}{3} & a_w'^- &= 1 & b_w'^- &= \frac{1}{3} & c_w'^- &= -\frac{1}{3} \end{aligned} \quad (A.20)$$

$$A_E^Q = -b_e^+ S_e^+ + b_e^- S_e^- - c_w^- S_w^- + D_e \quad (A.21a)$$

$$A_W^Q = b_w^+ S_w^+ - b_w^- S_w^- - c_e^+ S_e^+ + D_w \quad (A.21b)$$

$$A_{EE}^Q = d_e^- S_e^- \quad (A.22a)$$

$$A_{WW}^Q = d_w^+ S_w^+ \quad (A.22b)$$

$$\begin{aligned}
A_O^Q &= a_e^+ S_e^+ - a_e^- S_e^- + D_e - a_w^+ S_w^+ + a_w^- S_w^- + D_w + \\
&\quad a_n^+ S_n^+ - a_n^- S_n^- + D_n - a_s^+ S_s^+ + a_s^- S_s^- + D_s
\end{aligned} \tag{A.23}$$

Unlike the lower-order differencing schemes, the central coefficient is not equal to the sum of the neighboring coefficients plus the mass source. Adding eight coefficients together does not give the central coefficient, either. Weighting the Taylor series expansion in the upwind direction is the cause of this variance. Other schemes exhibit a symmetrical weighting about the face. QUICKR, however, weights the relative importance towards the upstream nodes while retaining a smaller dependence on the downstream value.

Eq. A.24 shows the linearized equation with the nine-point stencil that arises from QUICKR.

$$\begin{aligned}
A_O^Q \phi_O &= A_E^Q \phi_E + A_W^Q \phi_W + A_N^Q \phi_N + A_S^Q \phi_S + \\
&\quad A_{EE}^Q \phi_{EE} + A_{WW}^Q \phi_{WW} + A_{NN}^Q \phi_{NN} + A_{SS}^Q \phi_{SS} + \tilde{S}
\end{aligned} \tag{A.24}$$

Inspection of the QUICKR equations shows that negative coefficients can appear. For certain flow situations diagonal dominance can be lost and the influence of the neighboring nodes will overcome that of the central one. To counter these problems, which can lead to the divergence of iterative solvers, Phillips and Schmidt [40] proposed using hybrid differencing in combination with QUICK. They treat the hybrid terms implicitly, and in the explicit terms they add the difference between the hybrid and QUICK terms. The accuracy of QUICKR is achieved via the convergence behavior of hybrid differencing by placing the QUICKR into the code as follows:

$$A_O^H \phi_O = A_E^H \phi_E + A_W^H \phi_W + A_N^H \phi_N + A_S^H \phi_S + [\tilde{S} + S^Q] \tag{A.25a}$$

$$\begin{aligned}
S^Q &= (A_O^H - A_O^Q) \phi_O + (A_E^Q - A_E^H) \phi_E + (A_W^Q - A_W^H) \phi_W + \\
&\quad (A_N^Q - A_N^H) \phi_N + (A_S^Q - A_S^H) \phi_S + \\
&\quad A_{EE}^Q \phi_{EE} + A_{WW}^Q \phi_{WW} + A_{NN}^Q \phi_{NN} + A_{SS}^Q \phi_{SS}
\end{aligned} \tag{A.25b}$$

The superscript on the nodal coefficients denotes which type of differencing is used to evaluate the factor. Any stable lower-order differencing operator, such as power law differencing, may be used in place of the hybrid scheme. At convergence $\phi^n = \phi^{n+1}$ and the hybrid terms cancel out, leaving the fully QUICKR relationship shown in Eq. A.24. Under-relaxation of the equation is not affected by the use of QUICKR if S^Q is calculated prior to under-relaxation.

The negative coefficients make QUICKR susceptible to overshoot and undershoot [12]. Such behavior during the early stage of convergence, when the flow fields are rapidly evolving, is likely to cause the solution to diverge. To prevent this, the flow solution is allowed to stabilize over a few outer loop iterations before QUICKR is turned on. QUICKR is turned off by setting $S^Q = 0$.

A.3 Differencing Schemes for Cylindrical Coordinate System Applications

In the cylindrical coordinate system and the generalized curvilinear coordinate system derived from it, the surface area is a function of radius. The transported quantity decreases as the fluid moves in the radial direction, not due to shear forces within the fluid, but due to flux conservation. In the discretization of the governing equations, this is manifested as an error in the upwind approximation when there is a difference in radii between the two points across which the differencing takes place.

An approximation, that is analogous to that used in rectangular coordinate spaces, concerning the influence convection has on the transported quantity is made for cylindrical coordinate applications. This approximation is referred to as the cylindrical convection approximation. The subtle difference between the two approaches lies in assuming that it is the product of the radius and transported quantity that is unchanged by convection, rather than the transported quantity itself. For purely convective flow in the positive direction, the facial quantity in Cartesian spaces

would be found according to equation Eq. A.26.

$$\phi_e = \phi_o \quad (A.26)$$

The cylindrical convection approximation results in the following expression for the facial value:

$$r_e \phi_e = r_o \phi_o \quad (A.27a)$$

$$\phi_e = (r_o/r_e) \phi_o \quad (A.27b)$$

This minor change increases the accuracy of the numerical model because it makes the discretized equations reflect the dependence of the concentration of the transported quantity on the radial location. Although the governing equations in cylindrical and rectangular coordinate systems are often presented as subclasses of the same set of equations, no publications addressing this characteristic of convection in cylindrical frames of reference have been found. Two differencing schemes derived from this modification are presented and discussed in the remainder of this section. The cylindrical convection approximation can be applied to any of the differencing schemes in a similar manner.

Cylindrical Upwind Differencing:

$$A_E = [-C_e, 0] r_E/r_e + D_e \quad (A.28a)$$

$$A_W = [C_w, 0] r_W/r_w + D_w \quad (A.28b)$$

$$A_O = [C_e, 0] r_o/r_e + D_e + [-C_w, 0] r_o/r_w + D_w + [C_n, 0] r_o/r_n + D_n + [-C_s, 0] r_o/r_s + D_s \quad (A.29)$$

The convective and diffusive fluxes are treated separately in cylindrical upwind differencing. Central differencing accurately reflects the physics of diffusion dominated flow, regardless of the reference space, so no change is required for the diffusion terms. Using the cylindrical convection approximation shown in Eq. A.27b, the facial quantity is approximated as the upwind nodal value

multiplied by the ratio of the nodal radius to the facial radius. The FORTRAN MAX function is used to determine the upwind node, just as it was in the upwind differencing scheme in Section A.1. Dependence of the total flux on the local Reynolds number is modeled by the relative sizes of the two component fluxes. The resulting differencing scheme reflects the dependence of the transported quantity on the nodal location, making it more accurate than the other five-point differencing schemes. However, it is formally described as a lower-order differencing scheme because the upwind approximation is still only first-order accurate, even when it is consistent with the cylindrical convection approximation.

Cylindrical QUICKR Differencing:

The higher-order differencing scheme presented in Section A.2 is formed by approximating the facial quantity according to a three-point Taylor series approximation (Eqs. A.16 and 17). The cylindrical convection approximation assumes that the quantity that is conserved by convection is $(r\phi)$ rather than ϕ . Thus, if the Taylor series expansions that form the QUICKR differencing scheme are made consistent with the cylindrical convection approximation, a higher-order differencing scheme that is suitable for cylindrical applications results. Representative modified Taylor series are given in Eq. A.30 (corresponding to Eq. A.16) and Eq. A.31 (corresponding to Eq. A.17).

$$\phi_e = (3r_E\phi_E + 6r_O\phi_O - r_W\phi_W)/(8r_e) \quad (A.30a)$$

$$\phi_e = (3r_O\phi_O + 6r_E\phi_E - r_{EE}\phi_{EE})/(8r_e) \quad (A.30b)$$

$$\phi_e = (3r_O\phi_O + r_E\phi_E - r_W\phi_W)/(3r_e) \quad (A.31)$$

The resulting scheme is referred to as cylindrical QUICKR differencing. It is implemented into the solution algorithm in the manner described for QUICKR. The expressions for the nodal coefficients are listed in the following equations.

$$A_E^Q = (-b_e^+ S_e^+ + b_e^- S_e^-)r_E/r_e - (c_w^- S_w^-)r_E/r_w + D_e \quad (A.32a)$$

$$A_W^Q = (b_w^+ S_w^+ - b_w^- S_w^-)r_W/r_w - (c_e^+ S_e^+)r_W/r_e + D_w \quad (A.32b)$$

$$A_{EE}^Q = (d_e^- S_e^-) r_{EE} / r_e \quad (A.33a)$$

$$A_{WW}^Q = (d_w^+ S_w^+) r_{WW} / r_w \quad (A.33b)$$

$$\begin{aligned} A_O^Q = & (a_e^+ S_e^+ - a_e^- S_e^-) r_O / r_e + D_e - (a_w^+ S_w^+ + a_w^- S_w^-) r_O / r_w + D_w + \\ & (a_n^+ S_n^+ - a_n^- S_n^-) r_O / r_n + D_n - (a_s^+ S_s^+ + a_s^- S_s^-) r_O / r_s + D_s \end{aligned} \quad (A.34)$$

Despite the radii in the differencing coefficients changing the diagonal dominance characteristics, no stability problems were encountered when either cylindrical differencing scheme was used. A discussion of the effects of the cylindrical convection approximation on solution accuracy and stability is given in Section 7.2. Note that the cylindrical differencing schemes revert to the Cartesian form when the radii involved are equal.