

1990

Numerical solution of the steady, compressible, Navier-Stokes equations in two and three dimensions by a coupled space-marching method

Peter Warren TenPas
Iowa State University

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>

 Part of the [Mechanical Engineering Commons](#)

Recommended Citation

TenPas, Peter Warren, "Numerical solution of the steady, compressible, Navier-Stokes equations in two and three dimensions by a coupled space-marching method " (1990). *Retrospective Theses and Dissertations*. 11224.
<https://lib.dr.iastate.edu/rtd/11224>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

5

90

3

5

1

1

9

U·M·I

MICROFILMED 1990

INFORMATION TO USERS

The most advanced technology has been used to photograph and reproduce this manuscript from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

U·M·I

University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700 800/521-0600

Order Number 9035119

**Numerical solution of the steady, compressible, Navier-Stokes
equations in two and three dimensions by a coupled
space-marching method**

TenPas, Peter Warren, Ph.D.

Iowa State University, 1990

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106



**Numerical solution of the steady, compressible,
Navier–Stokes equations in two and three dimensions
by a coupled space–marching method**

by

Peter Warren TenPas

**A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of the
Requirements for the Degree of
DOCTOR OF PHILOSOPHY**

Major: Mechanical Engineering

Approved:

Signature was redacted for privacy.

' In Charge of Major Work

Signature was redacted for privacy.

For the Major Department

Signature was redacted for privacy.

For the Graduate College

**Iowa State University
Ames, Iowa**

1990

TABLE OF CONTENTS

	Page
NOMENCLATURE	
I. INTRODUCTION	1
A. Statement of the Problem	1
B. Review of Previous Work	2
1. Non-primitive variable methods	3
2. Primitive variable methods	3
a. Elliptic segregated methods	4
b. Elliptic direct inversion methods	5
c. Elliptic time-marching methods	5
d. Parabolic space-marching methods	6
e. Elliptic segregated space-marching methods	8
f. Elliptic coupled space-marching methods	8
C. Scope of the Present Study	10
II. GOVERNING EQUATIONS	11
A. Flow Geometry and Coordinate Systems	12
B. Conservation Laws	14
1. Governing equations in physical coordinate form	14
2. Boundary conditions	15
C. Model Closure	16
1. Constitutive relations for laminar flow	16
2. Equations of state	17
3. Transport property relationships	17
D. Algebraic Simplification of the Governing Equations	18
1. Algebraic form of the energy equation	18
2. Partially parabolized Navier-Stokes approximation	18
3. Constant property viscous terms	19
4. Simplified equations for three-dimensional flow	19
5. Simplified equations for two-dimensional flow	20
E. Transformation of the Governing Equations	20
1. Two-dimensional equations in computational form	21
2. Three-dimensional equations in computational form	22

III. SOLUTION ALGORITHM	23
A. Approximate Space-Marching Solution	23
1. Coupled, primitive variable formulation	23
2. Stability of the space-marching procedure	24
3. Initialization of the space-marching procedure	25
B. Pressure Correction Procedures	26
1. Motivation for a pressure correction procedure	26
2. Definition of the pressure correction	27
3. Point relaxation method	27
4. Bulk pressure correction method	28
5. Pressure Poisson equation method	30
IV. NUMERICAL SOLUTION	33
A. Grid Layout	33
B. Finite-Difference Equations	34
1. Momentum equations	36
a. Streamwise pressure gradient terms	36
b. Transverse pressure gradient terms	39
c. Viscous stress terms	39
d. Streamwise convection terms	41
e. Transverse convection terms	41
f. Metric terms	42
2. Continuity equation	43
a. Cell face center-point flux interpolation	43
b. Control volume face areas	45
C. Numerical Boundary Conditions	46
1. Upstream boundary conditions	46
2. Side boundary conditions	47
3. Downstream boundary conditions	47
4. Rearward-facing step boundary conditions	48
5. Forward-facing step boundary conditions	48
D. Solution of the System of Discretized Equations	49
1. Newton linearization with coupling	49
2. The linear system for two-dimensional flows	50
3. The linear system for three-dimensional flows	50

E.	Global Pressure Correction Procedure	50
	1. Finite-difference form of the pressure Poisson equation	50
	2. Parabolizing assumptions	53
V.	RESULTS AND DISCUSSION	55
A.	Developing Flow in a Two-Dimensional Channel	55
B.	Channel Flow with a Symmetric Sudden Expansion	67
C.	Channel Flow with an Asymmetric Sudden Expansion	81
D.	Channel Flow with a Symmetric Sudden Contraction	82
E.	Crossflow over a Cylinder	96
F.	Developing Flow in a Square Cross-Section Duct	99
G.	Computational Requirements	109
	1. Execution speed of the space-marching solver	109
	2. Global convergence behavior	110
	3. Overall CPU and memory requirements	111
VI.	CONCLUSIONS AND RECOMMENDATIONS	119
A.	Concluding Remarks	119
B.	Recommendations for Future Work	122
VII.	REFERENCES	124
VIII.	ACKNOWLEDGMENTS	130
IX.	APPENDIX A: DEFINITION OF THE TRANSFORMATION METRICS	131
X.	APPENDIX B: TRANSFORMATION OF THE VISCOUS STRESSES	133
XI.	APPENDIX C: BOUNDARY CONDITIONS AT A FORWARD-FACING STEP	136
XII.	APPENDIX D: LINEARIZATION OF THE CONVECTIVE FLUXES	138
XIII.	APPENDIX E: GENERATION OF CLUSTERED GRIDS	141

LIST OF TABLES

	<u>Page</u>
Table 4.1. Summary of finite-difference operators for the momentum equations	36
Table 5.1. Summary of test case parameters	56
Table 5.2. Results of grid refinement for incompressible channel flow with a symmetric sudden expansion	69
Table 5.3. Results of grid refinement for incompressible channel flow with an asymmetric sudden expansion	83
Table 5.4. Separation bubble predictions for Case 4 at $Re = 40$	99
Table 5.5. Iterations required for Case 2a at $Re = 56$ and $M = 0.05$ on a 51×31 uniform grid	113
Table 5.6. Iterations required for Case 3 at $Re = 678$ and $M = 0.05$ on a 46×31 uniform grid	113
Table 5.7. Comparison of computer execution time for various methods	116
Table 12.1. Quasilinear coefficients for the adiabatic, compressible flow of a perfect gas	139
Table 12.2. Quasilinear coefficients for flow of an incompressible fluid	140
Table 13.1. Summary of clustered grid parameters	143

LIST OF FIGURES

	<u>Page</u>
Figure 2.1. Example of body-fitted coordinates in the the physical space and the computational space	13
Figure 4.1. Velocity node layout for the momentum equations at $(i+1,j,k)$	37
Figure 4.2. Pressure node layout for the momentum equations at $(i+1,j,k)$	38
Figure 4.3. Continuity control volume centered at $(i+1/2,j,k)$	44
Figure 5.1. Test case flow geometries	57
Figure 5.2. Clustered grid layout for Case 1 for $Re = 10$	60
Figure 5.3. Predicted centerline velocity distribution for Case 1	61
Figure 5.4. Comparison of NS and PPNS solutions for Case 1	62
Figure 5.5. Compressible and incompressible flow predictions for Case 1	63
Figure 5.6. Convergence of centerline velocity distribution for Case 1	64
Figure 5.7. Computed axial velocity profiles for Case 1 at $Re = 75$	65
Figure 5.8. Convergence histories for Case 1 solutions	66
Figure 5.9. Predicted centerline velocity distribution for Case 2a	70
Figure 5.10. Predicted centerline velocity distribution for Case 2b	71
Figure 5.11. Computed axial velocity profiles for Case 2a at $Re = 56$	72
Figure 5.12. Computed axial velocity profiles for Case 2b	73
Figure 5.13. Predicted wall shear stress distribution for Case 2a	74
Figure 5.14. Predicted wall shear stress distribution for Case 2b	75
Figure 5.15. Comparison of NS and PPNS shear stress distributions for Case 2b	76
Figure 5.16. Convergence histories for Case 2a solutions at $Re = 56$	77
Figure 5.17. Convergence histories for Case 2b solutions at $Re = 100$	78
Figure 5.18. Convergence histories for Case 2b fine grid solutions	79
Figure 5.19. Comparison of NS and PPNS convergence histories for Case 2b	80

Figure 5.20.	Effect of grid refinement on predicted reattachment length for Case 3	84
Figure 5.21.	Predicted wall shear stress distribution for Case 3	85
Figure 5.22.	Predicted boundary pressure distribution for Case 3	86
Figure 5.23.	Computed axial velocity profiles for Case 3 at $Re = 687$	87
Figure 5.24.	Nonorthogonal grid layout for Case 4	90
Figure 5.25.	Predicted centerline velocity distribution for Case 4	91
Figure 5.26.	Predicted boundary pressure distribution for Case 4	92
Figure 5.27.	Predicted upstream wall shear stress distribution for Case 4	93
Figure 5.28.	Predicted downstream wall shear stress distribution for Case 4	94
Figure 5.29.	Convergence histories for Case 4 solutions at $Re = 400$	95
Figure 5.30.	Nonorthogonal grid layout for Case 5 with detail view of the region near the cylinder	97
Figure 5.31.	Predicted surface pressure distribution for Case 5 at $Re = 40$	98
Figure 5.32.	Predicted centerline velocity distribution for Case 6	101
Figure 5.33.	Convergence of centerline velocity distribution for Case 6	102
Figure 5.34.	Computed axial velocity profiles at the duct mid-section for Case 6 at $Re = 100$	103
Figure 5.35.	Computed axial velocity profiles at the quadrant mid-section for Case 6 at $Re = 100$	104
Figure 5.36.	Computed fully-developed axial velocity profiles at the duct exit plane for Case 6 at $Re = 100$	105
Figure 5.37.	Computed secondary velocity vectors at the duct exit plane for Case 6 at $Re = 100$	106
Figure 5.38.	Convergence histories for Case 6 fine grid solutions	107
Figure 5.39.	Convergence histories for Case 6 solutions at $Re = 100$	108
Figure 5.40.	Effect of global pressure correction on convergence history for Case 2a	112
Figure 5.41.	Effect of α and ω on convergence rate for Case 2a	114
Figure 5.42.	Effect of α and ω on convergence rate for Case 3	115

NOMENCLATURE

a	acoustic velocity
bp	bulk pressure or average pressure
C_1	1st coefficient of the Sutherland viscosity law
C_2	2nd coefficient of the Sutherland viscosity law
C_f	skin friction coefficient, $2 \tau_w / (\rho u^2)$
C_p	pressure coefficient
c_p	coefficient of specific heat at constant pressure
D	cylinder diameter
\vec{E}	numerical flux vector in the x direction
\vec{E}'	numerical flux vector in the ξ direction
e	specific stored or internal energy
e_t	specific total energy
\vec{F}	numerical flux vector in the y direction
\vec{F}'	numerical flux vector in the η direction
\vec{G}	numerical flux vector in the z direction
\vec{G}'	numerical flux vector in the ζ direction
H	full height of channel or full width duct
h	half-height of channel or half-width of duct
k	coefficient of thermal conductivity
L	geometric reference length of the flow problem
l	reattachment length
M	Mach number based upon reference conditions, \bar{u}_r / \bar{a}_r
m	mass flowrate
n	iteration level

Pr	Prandtl number
p	static pressure
\bar{p}	assumed value of the static pressure
p'	exact value of the pressure correction
p^*	estimated value of the pressure correction
\vec{Q}	heat flux vector
\vec{q}	vector of primary unknown variables
R	gas constant
Re	Reynolds number based upon reference conditions, $\bar{q}_r \bar{u}_r \bar{L} / \bar{\mu}_r$
T	static temperature
T_0	total (or stagnation) temperature
t	time
tp	transverse pressure variation from the bulk pressure
\vec{U}	vector of conservation variables
U	contravariant velocity component
u	x component of velocity
V	contravariant velocity component
v	y component of velocity
W	contravariant velocity component
w	z component of velocity
x	1st Cartesian coordinate
y	2nd Cartesian coordinate
z	3rd Cartesian coordinate

Greek Symbols

A	parameter in global pressure correction equation
B	parameter in global pressure correction equation

Γ	parameter in global pressure correction equation
α	diffusion factor in global pressure correction equation ($\alpha = B/A = \Gamma/A$)
γ	isentropic exponent (specific heat ratio)
δ	difference between final and assumed pressure
δ	finite-difference operator
ϵ	difference between computed and assumed pressure value
ζ	3rd generalized coordinate (transverse direction)
η	2nd generalized coordinate (transverse direction)
Θ	angular coordinate
μ	fluid molecular dynamic viscosity
ν	fluid molecular kinematic viscosity
ξ	1st generalized coordinate (marching direction)
ρ	density
τ	component of the viscous stress tensor
τ_w	wall shear stress
Φ	viscous dissipation function
ω	pressure correction relaxation factor
ω_b	bulk pressure correction relaxation factor
ω_t	transverse pressure correction relaxation factor
Subscripts	
1	inlet plane
2	exit plane
b	bulk value
CL	centerline
i	mesh index corresponding to ξ direction
j	mesh index corresponding to η direction
k	mesh index corresponding to ζ direction

max	maximum value
min	minimum value
r	reference condition or reattachment point
s	separation point
t	transverse value
∞	freestream reference condition

Superscripts

n	iteration level
'	transformed vector
'	indicates exact value of the pressure correction
+	indicates estimated value of the pressure correction

Overscores

\wedge	nonlinear coefficient value
-	dimensional quantity
\rightarrow	vector quantity
-	indicates assumed value of the pressure (also indicates bulk property)

I. INTRODUCTION

A. Statement of the Problem

Fluid flow in natural and engineered systems creates very complex patterns. Understanding the distribution of velocity and pressure within these flow fields is of importance to engineers and scientists in many disciplines. The design of new engines, the specification of cooling systems for buildings or electronic components, analysis of the dispersion of pollutants in the environment and prediction of the weather all entail the solution of fluid flow problems.

The flow of a compressible, viscous fluid is mathematically described by the fundamental physical laws governing conservation of mass, momentum and energy. These equations of motion have been understood since the last century; however, in the past the complexity of the equations has limited the application of theoretical methods. Due to the recent widespread availability of powerful computers, numerical simulation of fluid flow has become an established technology. Current research is directed toward the development of progressively more powerful tools for detailed analysis, study and prediction of fluid motion.

Numerical models of fluid flow are based upon approximation of the continuous equations of motion with sets of discrete equations. The discrete equations are solved for the unknown flow properties at a sufficient number of points within the flow domain to satisfactorily resolve the fluid motion. Practical simulation methods must be both accurate and computationally efficient. This requires identification of the simplest representative mathematical model, accurate and consistent discretization of the continuous equations at all points within the flow field, proper specification of boundary conditions and utilization of robust algorithms for solving extremely large sets of equations.

The equations of motion for laminar flow of a Newtonian fluid are a set of partial differential equations known as the Navier–Stokes equations [1]. In general, the mathematical character of these equations depends upon the flow conditions; however, for steady flow at subsonic velocity the equations are classified as elliptic in the spatial coordinates. A number of different computational methods have been developed to numerically solve the Navier–Stokes equations for steady, subsonic flows. Several of these procedures are reviewed below, and although these approaches have been successful to some extent, a number of difficulties remain. Recently, improvements to existing methods as well as new approaches have been proposed. It is not yet clear which of the algorithms is superior. In general, ample motivation appears to exist for the development of advanced procedures

for modeling subsonic flows, especially for those methods which are applicable to three-dimensional flows and hold promise for extension to the transonic and supersonic flow regimes.

In the present study, an accurate, stable and efficient algorithm for the simulation of steady, laminar, subsonic flow has been developed. The specific objective of this research was to develop and evaluate a new, coupled, space-marching technique for this class of flow problems. The procedure is an extension of earlier space-marching procedures that have been used to solve the boundary-layer equations and the partially-parabolized Navier-Stokes (PPNS) equations.

The new algorithm incorporates features that enhance computational efficiency and extend the range of application of the space-marching method. First, the method may be easily and accurately used to model flow in arbitrary geometries as the primitive variable form of the equations have been discretized on a regular grid in generalized curvilinear coordinates. Secondly, the compressible formulation using pressure, rather than density, as a primary variable is applicable to incompressible as well as compressible flow regimes. Thirdly, the Navier-Stokes equation formulation is valid for laminar flow at all Reynolds numbers. Finally, a new pressure acceleration scheme has been developed that significantly improves the convergence rate for a number of test cases in the incompressible flow regime.

B. Review of Previous Work

A number of different methods have previously been developed to solve the Navier-Stokes equations for steady, subsonic flows. Many of the procedures are described in the texts by Roache [2], Anderson et al. [3], and Patankar [4]. Detailed exposition of several well established algorithms is presented in the handbook edited by Minkowycz et al. [5]. Algorithms may be categorized as either primitive variable methods or non-primitive variable methods, depending upon the nature of the variables in which the equations are cast. The primitive variables are the common physical properties of the flow, such as velocity, density, pressure and temperature. Mathematically derived quantities such as the vorticity, dilatation, stream function and velocity potential are examples of non-primitive variables. Since the method chosen for this research uses primitive variables, non-primitive variable techniques will be briefly reviewed first, followed by a more extensive survey of primitive variable algorithms.

1. Non-primitive variable methods

The Navier–Stokes equations can be expressed in many forms using mathematically derived variables that represent kinematic or dynamic properties of the flow. Two such formulations are discussed below.

For two–dimensional flow the continuity and momentum equations can be manipulated to form a transport equation for the vorticity, coupled with a Poisson equation for the stream function. Roache [2] treats the vorticity/stream function procedure very extensively. Algorithms to solve this set of equations by relaxation or time–marching methods have been widely used, particularly for incompressible flow. The stream function and the two components of the vorticity are the primary solution variables. The velocity components are easily obtained as derivatives of the stream function. Typically the pressure is not calculated during the course of the solution, but can be determined from the velocity field by solution of a Poisson equation. Numerous refinements have been made to improve the accuracy of the technique and extend the solution to arbitrary coordinate systems. The method has proven to be quite accurate, particularly for situations involving singularities or discontinuities in the geometry. Unfortunately, the procedure is limited to two–dimensional flow by the formal definition of the stream function.

The dual potential (or vorticity/vector potential) algorithm for incompressible, three–dimensional flow has been formulated by Aziz and Hellums [6]. This procedure requires the solution of a transport equation for the three vorticity components, a vector Poisson equation for the three components of the velocity potential and another Poisson equation for the scalar potential. As with other non–primitive variable methods, the proper treatment of boundary conditions is critical for producing a consistent numerical solution. Boundary conditions for the dual potential method are addressed by Hiraski and Hellums [7]. The algorithm has not been widely used, but Wong and Reizes [8] have recently solved incompressible three–dimensional duct flows with the procedure. The method can be extended to compressible flow. See the recent dissertation by Gegg [9] for further discussion of this method.

2. Primitive variable methods

There are a wide variety of finite–difference methods for solving the primitive variable form of the Navier–Stokes equations. Implementing a specific procedure involves many important details such as the specification of the grid structure or the order of the discretization of the equations. However, the method of coupling the velocity field solution to the pressure field solution often

outweighs all other considerations in the successful development of a procedure for modeling subsonic or incompressible flows. For the purposes of this review, algorithms are classified by the nature of the algebraic form in which the equations are solved, using the following categories:

- a. Elliptic segregated methods
- b. Elliptic direct inversion methods
- c. Elliptic time-marching methods
- d. Parabolic space-marching methods (approximate solution)
- e. Elliptic segregated space-marching methods
- f. Elliptic coupled space-marching methods

The segregated algorithms use independent equations for the velocity components and the pressure. Several strategies have been advanced for solving the coupled momentum and continuity equations for the primitive variables, including the pressure. These schemes all share the advantage that a separate procedure for imposing the continuity constraint and determining the pressure is not required. The choice of variables and the algebraic approach used to couple the equations distinguish the direct, time-marching and space-marching algorithms.

a. Elliptic segregated methods The most frequently used primitive variable algorithms for subsonic flows solve the momentum equations for the velocity components in an uncoupled (segregated) manner, holding the pressure fixed. This method of solution was originated by Harlow and Welch [10] and Welch et al. [11] in the development of the Marker-and-Cell (MAC) method. Caretto et al. [12] introduced implicit methods to solve each independent finite-difference equation more efficiently. Although different in detail, the segregated solution schemes (e.g., References 10–20) use the continuity equation indirectly in order to determine the the pressure. Typically, the pressure is evaluated by the solution of a separate Poisson equation. Iteration between the momentum equations and the pressure correction equation continues until the errors become negligible.

Since the velocity components and pressure are computed from different algorithms, rather than in a coupled manner, convergence of the iterative cycle is slowed. Furthermore, the solution of the elliptic Poisson equation consumes a large part of the total computation time for each global iteration. The early algorithms were formulated for computers with relatively small memory capacity, and the minimal use of core storage is achieved at the expense of longer execution time than competing algorithms that use more complex algebraic methods.

Numerous approaches have been tried to improve the efficiency of segregated algorithms. Van Doormal et al. [16] have evaluated improved algorithms for solving the pressure Poisson equation. Rhie [17] has employed the multigrid procedure to accelerate the convergence rate.

Most segregated techniques utilize the staggered grid arrangement to maintain stability and to efficiently link the pressure differences between nodes to corresponding velocity changes. However, the staggered grid arrangement is difficult to implement in generalized coordinates. Recently, the segregated algorithm has been extended to regular grids and generalized coordinates [17–22]. Among these investigators, Rhie and Chow [18,19] are generally credited with generalizing the pressure correction procedure.

b. Elliptic direct inversion methods Vanka et al. [23] and more recently Patankar et al. [24] have compared solutions for two-dimensional flows obtained by direct and segregated methods. For these direct methods, a large sparse matrix solver is used to invert the complete system of equations assembled from the continuity and momentum equations at every point in the flow domain. The Navier–Stokes equations are nonlinear, so multiple inversions are required to converge the linear and type-dependent coefficients. Although the computational effort per iteration is large, the total processing time is reduced since relatively few iterations are needed. Given reasonable starting values, very rapid convergence of the nonlinear terms is achieved with the Newton–Raphson method, and the procedure seems to be very stable. Direct methods would appear to be practical for two-dimensional problems, yet the large storage requirements make solution of three-dimensional problems by algorithms of this type impractical on the current generation of computers. Similar conclusions were drawn by Bender and Khosla [25], who employed a sparse matrix solver to invert the stream-function vorticity equations.

c. Elliptic time-marching methods Time-marching methods are the most widely used procedure for the solution of compressible flow problems, particularly for supersonic flows. These algorithms obtain a steady flow solution by simulating the evolution of a transient flow in time until steady-state conditions are observed. Both explicit and implicit methods have been developed to solve the time-dependent Navier–Stokes equations, which are hyperbolic in time. For supersonic flow the equations are also hyperbolic in space, however, for subsonic flow the equations are elliptic in space.

One of the most widely used explicit procedures was developed by MacCormack [26]. The maximum time step for explicit methods is limited by the CFL condition, and a prohibitive number of

steps is often required to reach a steady flow condition. Beam and Warming [27,28] and Briley and McDonald [29] were the first to develop implicit marching schemes using the alternating-direction-implicit (ADI) method for the multi-dimensional equations. The implicit formulation allows much larger time steps, and consequently steady-state convergence is generally reached much more rapidly. As an alternative to implicit methods, Chima and Johnson [30] have applied the multigrid procedure to accelerate a fast explicit solver. The recent work by Holst [31] reviews many of the current methods.

While very effective for compressible flow, there are problems with applying the time-marching method to incompressible flow. Most time-marching methods are formulated specifically for compressible flows and encounter a severe time step limitation in the incompressible limit. Briley et al. [32] have modified an implicit, compressible procedure to use pressure as one of the primary variables in place of the density. This substitution removed the primary source of the ill-conditioning of the equations, and the convergence rate to the steady-state solution was greatly improved.

Another technique that permits the use of implicit, time-marching algorithms for steady, incompressible flow is the pseudocompressibility formulation introduced by Chorin [33]. With this procedure the density is artificially varied in order to time-march the equations in a stable manner. The method is generally not time accurate as the constant density condition is usually imposed only as a steady-state solution is approached. The procedure has been generalized to nonorthogonal curvilinear coordinate systems in two and three dimensions by Kwak et al. [34] using the Beam-Warming [27] differencing strategy. With this algorithm tightly coupled sets of equations written along grid lines are efficiently solved at each time step. However, the method requires the addition of higher order smoothing terms for stability, and care must also be taken in establishing the value of the pseudocompressibility parameter in order to maintain accuracy and stability. Benocci and Ceresola [35] and Hartwich and Hsu [36] have also recently published incompressible flow solutions obtained with pseudocompressibility algorithms.

d. Parabolic space-marching methods Often the best flow simulation is produced by the simplest form of the mathematical model that is valid for the flow under consideration. Although parabolic algorithms only provide approximate solutions to the Navier-Stokes equations, these methods are included in this review because they are often capable of correctly modeling actual flow and the computational procedures employed are the predecessors of the fully elliptic space-marching methods.

For a wide variety of moderate to high Reynolds number flows, the boundary-layer equations are the simplest, and reasonably accurate, model of viscous flow. The range of applicability of the boundary-layer (or thin shear layer) model is not yet known. The equations are capable of simulating complex flow patterns, including separated regions with flow recirculation. For example, Kwon et al. [37] have successfully applied the coupled boundary-layer equations to predict the flow in ducts with symmetric sudden expansions.

For flows with streamwise curvature or large secondary flows, the boundary-layer model may not be adequate, and the parabolized Navier-Stokes (PNS) equations have been used to model these flows [38-48]. The parabolized algorithm is an efficient, once-through, procedure for obtaining an approximate flow solution. In order to parabolize the Navier-Stokes equations, the streamwise viscous terms are eliminated, the streamwise convective terms are evaluated using only upstream information, and for subsonic flow the streamwise pressure gradient is imposed as a fixed source term.

Patankar and Spalding [38], Briley [39], Ghia et al. [40] and Roberts and Forrester [41] have developed parabolic space-marching methods for subsonic internal flows. These procedures use the global conservation of mass condition to establish the magnitude of the bulk streamwise pressure gradient at every space-marching step. A key assumption regarding the pressure field is that the streamwise pressure gradient is independent of the pressure variations corresponding to secondary flows. This assumption limits the application of these methods to flows without strong transverse pressure gradients. For three-dimensional flows, the solution at each station requires the determination of the secondary flow in a plane. This is usually accomplished by a procedure similar to the segregated methods used for arbitrary two-dimensional elliptic flows. The works by Maliska [42] and Katsanis [43] are examples of the formulation of this technique in generalized coordinate systems.

The methods of Rudman and Rubin [44], Lubard and Helliwell [45] and Vigneron et al. [46] are early examples of coupled space-marching procedures for the PNS equations. Procedures of this type have been used extensively for predicting primarily supersonic, viscous flows. See Davis and Rubin [47] for a review of this class of algorithm, and Lawrence et al. [48] as an example of current techniques. These procedures differ considerably from the techniques used for subsonic flow. For the regions of supersonic flow the governing equations are hyperbolic in character. Thus, the continuity, momentum and energy equations may be efficiently solved with a coupled space-marching procedure using only upstream properties. However in order to maintain stability, the pressure

gradient in subsonic regions must use downstream information or be approximated. These methods are therefore restricted to flows with only small regions of subsonic flow, such as thin viscous boundary-layers.

e. Elliptic segregated space-marching methods The inability of the parabolic schemes to model flows with large transverse pressure gradients or extensive regions of subsonic flow, led to the development of fully elliptic procedures based on global iteration of the space-marching procedure to feedback the upstream influences produced by downstream conditions. At moderate or high Reynolds numbers the streamwise viscous terms are assumed to be negligible (PPNS assumption) and are eliminated from the momentum equations. But, the elliptic nature of the pressure field is retained. Hence, the equations are only partially parabolized. The Pratrap and Spalding [49] algorithm utilizes a segregated method to solve the momentum equations and estimate the local pressure at each streamwise station during each space-marching sweep. The methods by Moore and Moore [50] and Chilikuri and Pletcher [51] incorporate the solution of a global Poisson type equation between space-marching sweeps to correct the pressure field over the entire solution domain.

These algorithms represent a significant development compared with the parabolic, single-pass, space-marching methods. However, the lack of coupling between the momentum equations and the pressure solution causes problems in converging the calculations for flows with large pressure gradients.

f. Elliptic coupled space-marching methods Coupled space-marching algorithms for reduced forms of the Navier-Stokes equations for incompressible or subsonic flow were originated by Rubin and Reddy [52] and Israeli and Lin [53]. Brown [54] used a similar technique to study flow separation. These methods use the coupled continuity and momentum equations in a space-marching procedure to simultaneously calculate the velocity and pressure at each marching station. A staggered type grid arrangement is used to prevent decoupling of the velocity and pressure. The pressure value at any given point influences the solution at the nearest neighboring upstream point through the streamwise pressure gradient term in the momentum equations. In this manner the elliptic nature of the pressure field is propagated upstream, influencing the flow solution at all points in the domain. Repeated global iterations are required to converge the local estimates of the pressure field. Himansu and Rubin [55] have reported success with the use of a multigrid procedure to accelerate convergence.

Several methods have been proposed for implementing the coupled space-marching procedure on regular grids or in three-dimensions. The procedures by Govindan [56] and Pougare and Lakshminarayana [57] use a regular grid and have been applied to three-dimensional flows. The Beam and Warming [27] differencing scheme is utilized to solve the coupled cross-plane equations. Although the reported results with this method have generally been obtained in a parabolic mode with only a single marching sweep, the use of global iteration to converge the estimates of the pressure field is mentioned. Liu and Pletcher [58] have also developed a regular grid formulation, and proposed a systematic procedure for coupling the boundary conditions on both sides of a two-dimensional domain. Reddy and Rubin [59] have recently extended their staggered grid algorithm to three-dimensions and formulated boundary conditions for internal flow. Their procedure utilizes the modified strongly implicit method [60] to execute the coupled solution in the cross-plane.

All of the previous procedures [52-59] employ an approximate form of the Navier-Stokes equations, such as the PPNS equations. Also, only simple iteration with relaxation of the local pressure changes is used to converge the pressure solution.

During the course of the present research, solution of the two-dimensional incompressible Navier-Stokes equations in Cartesian coordinates on a staggered grid has been reported by Bentson and Vradis [61]. Also significant with respect to the current research is their use of a global pressure correction step between space-marching iterations to accelerate convergence of the pressure field. They evaluated several pressure correction strategies based on the pressure Poisson equation. To limit the processing time consumed in the pressure correction stage of the algorithm the Poisson equation was not iterated to convergence. Approximation of the pressure correction did not significantly compromise the effectiveness of the procedure and substantially reduced the effort to obtain the pressure correction. Bentson and Vradis [61] also investigated a "second stage" pressure correction procedure that is equivalent to executing a backsweep integration of the streamwise momentum equation along each different grid line. They found the procedure to be unstable, but when used intermittently, in combination with the Poisson equation technique, a significant increase in the global convergence rate was observed.

To the author's knowledge, no other solutions of the steady, compressible, Navier-Stokes equations for subsonic flow obtained with a coupled space-marching technique have been reported in the literature. Preliminary results of the present research have been reported by TenPas and Pletcher [62].

C. Scope of the Present Study

The purpose of the present research was to test the limitations of space-marching methods by the development and evaluation of a new algorithm. The specifications for the new algorithm included: 1) solution of the steady, compressible, Navier-Stokes equations in primitive variables on a regular grid in general curvilinear coordinates; 2) solution of the coupled continuity and momentum equations with pressure as a primary variable, thus avoiding the necessity of solving the pressure Poisson equation; 3) application of the second-order upwind method for the streamwise convective terms in the momentum equations; 4) a prohibition on the use of artificial dissipation terms to stabilize the method; and 5) development of a reverse-marching or "backsweep" procedure to accelerate convergence of the pressure field.

To limit the number of unknowns, and therefore minimize the computational requirements for carrying out this study, application of the method was restricted to laminar adiabatic flow. There does not appear to be any fundamental reason that would prohibit extension of the procedure to include heat transfer or turbulent flow.

This thesis describes the new formulation and presents the results of several example computations to demonstrate the capabilities of the method. The majority of the research work reported here involved the development and extensive testing of the procedure for two-dimensional flows at low Mach number. Results of the initial effort to extend the model to three-dimensions are reported. Late in the course of this study, the suitability of the primitive variable formulation for simulating purely incompressible flow was investigated. A description of the constant density version of the present procedure and example results are also included.

II. GOVERNING EQUATIONS

A mathematical model for steady, laminar, subsonic flow is presented in this chapter. First, the flow geometry is set forth. Next, the set of fundamental conservation laws that describe the motion of a homogeneous fluid are stated. The constitutive relations, equations of state and transport property formulas that close the model are given. Reductions of the general mathematical model that are valid for specific flow regimes are discussed, and the simplifications made in the model to reduce the computational requirements for this study are described. The transformation of the governing equations to the computational domain is presented in the final section of the chapter.

To reveal the important flow parameters in the mathematical model, the equations are written in dimensionless form. Use of dimensionless variables also removes the restrictions of a particular system of units and reduces computer round-off error by normalizing the variables. The procedure for obtaining the dimensionless equations is straightforward.

All dimensions are canceled from the dimensional equations by multiplying or dividing the equations by constant reference quantities, such as the mean flow velocity or the characteristic length of the flow geometry. The customary dimensionless parameters are isolated and the variables are grouped as dimensionless terms. The reference quantities and the corresponding dimensionless variables are defined in the following manner (dimensional quantities are indicated by a tilde and the reference properties are denoted by the subscript r) :

$$\begin{aligned}
 x &= \tilde{x}/\tilde{L} & y &= \tilde{y}/\tilde{L} & z &= \tilde{z}/\tilde{L} \\
 u &= \tilde{u}/\tilde{u}_r & v &= \tilde{v}/\tilde{u}_r & w &= \tilde{w}/\tilde{u}_r & p &= \tilde{p}/(\tilde{\rho}_r \tilde{u}_r^2) & T &= \tilde{T}/\tilde{T}_r \\
 \rho &= \tilde{\rho}/\tilde{\rho}_r & \mu &= \tilde{\mu}/\tilde{\mu}_r & k &= \tilde{k}/\tilde{k}_r & C_1 &= \tilde{C}_1/\tilde{C}_1 & C_2 &= \tilde{C}_2/\tilde{T}_r \\
 R &= \tilde{R}\tilde{T}_r/\tilde{u}_r^2 & \gamma &= \tilde{c}_p/\tilde{c}_v & e &= \tilde{e}/\tilde{u}_r^2 & c_v &= \tilde{c}_v\tilde{T}_r/\tilde{u}_r^2 & c_p &= \tilde{c}_p\tilde{T}_r/\tilde{u}_r^2
 \end{aligned}$$

Here, L is the flow field characteristic length; x , y and z are the Cartesian coordinates; u , v and w are the respective Cartesian velocity components; p is the static pressure; T is the static temperature; ρ is the density; μ is the dynamic viscosity; k is the thermal conductivity; C_1 and C_2 are the Sutherland constants; R is the gas constant; γ is the specific heat ratio; e is the specific internal energy; and c_v and c_p are the constant volume and constant pressure specific heats. The reference properties are

chosen to be the upstream bulk properties for internal flow cases, or the freestream properties for external flow cases. The Reynolds number, Mach number, and Prandtl number are defined as

$$\text{Re} = \bar{\rho}_r \bar{u}_r \bar{L} / \bar{\mu}_r$$

$$\text{M} = \bar{u}_r / \bar{a}_r$$

$$\text{Pr} = \bar{c}_p \bar{\mu}_r / \bar{k}_r$$

A. Flow Geometry and Coordinate Systems

The space-marching method is especially suited to problems having a principal through-flow direction. The simplest form of this type of flow domain is a rectangular duct. In general, arbitrary geometry is accommodated by the use of a curvilinear, body-fitted coordinate system with an H-type topology. Figure 2.1 depicts an arbitrary flow domain in physical coordinates and shows the corresponding uniform computational domain. The flow domain is bounded by an inlet flow surface upstream, an exit flow surface downstream, and solid or free surfaces for internal or external flow conditions on the side boundaries.

The body-fitted coordinates are oriented to identify the primary flow direction and to define each boundary. The space-marching direction (ξ direction) is roughly aligned with the flow direction. The constant ξ surfaces are approximately normal to the flow streamlines and are termed "stations". The ξ_{\min} constant surface forms the upstream boundary, and the downstream boundary is the ξ_{\max} constant surface. The transverse η and ζ directions are roughly normal to the flow; therefore constant η or constant ζ surfaces approximate streamsurfaces. The side boundaries of the flow domain define the η_{\min} , η_{\max} , ζ_{\min} , and ζ_{\max} surfaces.

In addition to aligning the space-marching direction with the principal flow direction, the use of body-fitted coordinates has two other important advantages. First, with the boundaries defined by surfaces upon which one of the coordinates is constant, the boundary conditions are more easily and more accurately applied. Secondly, the spacing of the coordinate surfaces may be systematically varied within the flow domain to concentrate grid points in regions where large gradients in the flow properties occur. Meanwhile, a uniform computational grid is maintained, allowing the use of algebraically simple and accurate finite-difference approximations of derivatives.

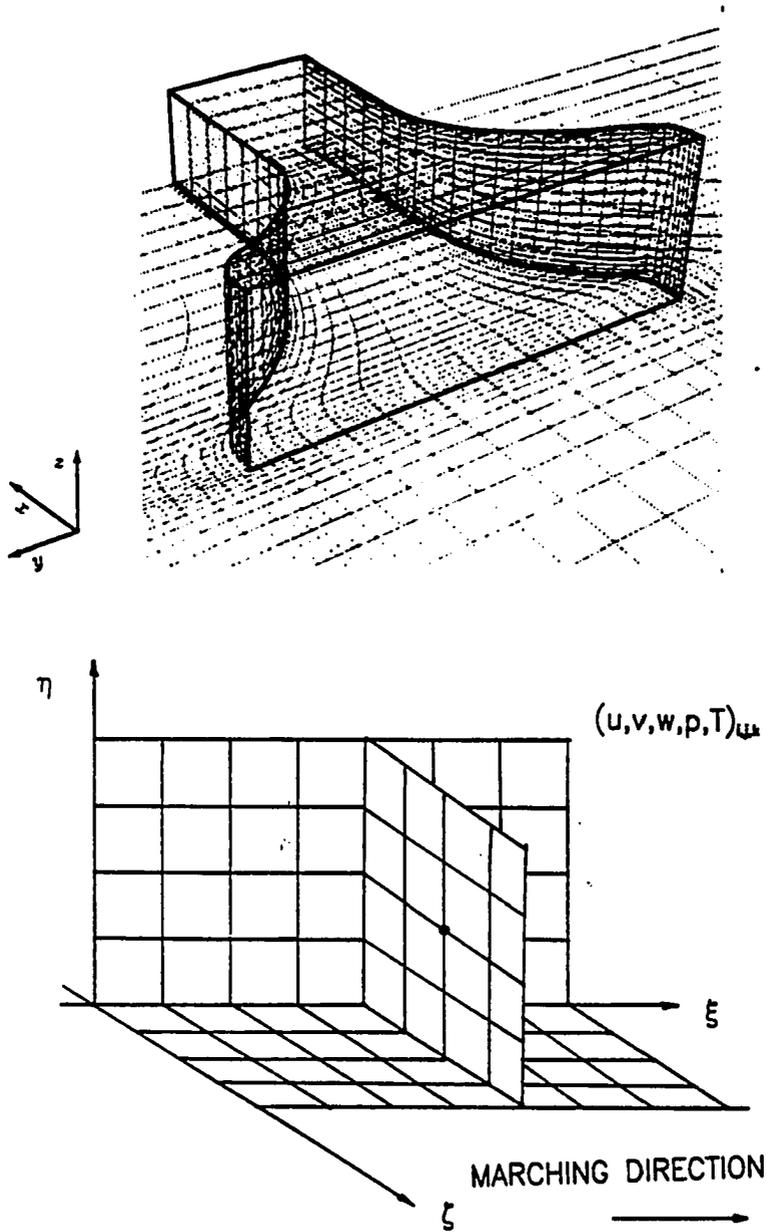


Figure 2.1. Example of body-fitted coordinates in the physical space and the computational space

The body-fitted coordinates are related to the physical coordinates through a generalized independent variable transformation of the form

$$\xi = \xi(x, y, z) \quad \eta = \eta(x, y, z) \quad \zeta = \zeta(x, y, z)$$

As shown in the following section, the transformation introduces metric terms into the equations that scale the physical and computational coordinates. Definitions of the metrics and the Jacobian of the transformation are given in Appendix A.

B. Conservation Laws

The principles of conservation of momentum, mass, and energy govern the flow of a compressible fluid. These laws are satisfied at every point within the physical space. Ultimately, the boundary conditions imposed upon the fluid determine the nature of each particular flow field.

Solution methods are classified by the form of the equations used to express the conservation laws. The form employed here is a system of partial differential equations with the Cartesian velocity components, the pressure, and the temperature as the primary dependent variables. The solution vector is defined as

$$\vec{q} = [u, v, w, p, T]^T$$

This choice of dependent variables is termed the "primitive variable" form.

Momentum is a vector quantity, so the form of the equations also depends upon the choice of the basis vectors. Here, since the Cartesian velocities are the primary unknowns, the three Cartesian component momentum equations are selected. Generalized coordinates are ultimately used as the independent variables, with the Cartesian coordinates defining the physical geometry.

1. Governing equations in physical coordinate form

For a compressible flow in the absence of body forces and internal heat sources, the three-dimensional Cartesian coordinate form of the momentum equations, the continuity equation and the energy equation are expressed in dimensionless vector form as

$$\frac{\partial \vec{U}}{\partial t} + \frac{\partial \vec{E}}{\partial x} + \frac{\partial \vec{F}}{\partial y} + \frac{\partial \vec{G}}{\partial z} = 0 \quad (2.1a)$$

The derivative with respect to time vanishes for steady flow. The vectors in the equation above are

$$\bar{U} = [\rho u, \rho v, \rho w, \rho, \rho e_t]^T$$

$$\bar{E} = \bar{E}(\bar{q}) = \begin{bmatrix} \rho u^2 + p & -\tau_{xx} \\ \rho uv & -\tau_{xy} \\ \rho uw & -\tau_{xz} \\ \rho u & \\ \rho u e_t + pu + Q_x - \Phi_x & \end{bmatrix}$$

$$\bar{F} = \bar{F}(\bar{q}) = \begin{bmatrix} \rho vu & -\tau_{yx} \\ \rho v^2 + p & -\tau_{yy} \\ \rho vw & -\tau_{yz} \\ \rho v & \\ \rho v e_t + pv + Q_y - \Phi_y & \end{bmatrix}$$

$$\bar{G} = \bar{G}(\bar{q}) = \begin{bmatrix} \rho wu & -\tau_{zx} \\ \rho wv & -\tau_{zy} \\ \rho w^2 + p & -\tau_{zz} \\ \rho w & \\ \rho w e_t + pw + Q_z - \Phi_z & \end{bmatrix}$$

Here the total specific energy of the fluid is restricted to the sum of the internal and kinetic energy defined as

$$e_t = e + (u^2 + v^2 + w^2)/2$$

The viscous dissipation terms in the energy equation expand to

$$\Phi_x = u \tau_{xx} + v \tau_{yx} + w \tau_{zx}$$

$$\Phi_y = u \tau_{xy} + v \tau_{yy} + w \tau_{zy}$$

$$\Phi_z = u \tau_{xz} + v \tau_{yz} + w \tau_{zz}$$

2. Boundary conditions

For steady subsonic flow the governing equations are elliptic. Therefore it is expected that conditions at all points on the flow boundaries influence the solution. However, for moderate and high Reynolds number flows the conditions imposed are not the same at each boundary; upstream conditions influence the flow differently than downstream conditions. In this study it is assumed that the problem is well-posed with prescribed initial conditions on the inlet surface (given [u, v, w, p, T]

profiles). The flow conditions imposed at the exit boundary are that the pressure farther downstream does not vary in the transverse direction, that streamwise diffusion is negligible, and should flow reversal occur at the final station the velocity downstream is assumed to be zero. For internal flow problems, the magnitude of the downstream pressure is unknown, and is determined during the solution procedure in order to match the specified inlet flow. For external flow problems, the downstream pressure is assumed equal to the freestream pressure.

The nature of the side boundary conditions varies from problem to problem. For example, internal flows are subject to different constraints than external flows. On solid walls the no slip and no penetration conditions on velocity are applied. However, flow may cross an external freestream boundary, but the diffusion terms are assumed to be negligible. The specific treatment of different boundary conditions is presented in greater detail in a later chapter in terms of the numerical solution.

C. Model Closure

The fundamental conservation laws contain transport terms as well as fluid properties that must be modeled to complete the mathematical formulation. The relationships given below were selected for the flow regimes of interest in this study. The solution algorithm is not believed to be restricted to these particular formulations, though use of other models would require careful evaluation and testing.

1. Constitutive relations for laminar flow

For laminar flow of a Newtonian fluid the viscous stresses are modeled as proportional to the rates of strain of a fluid element. Using the Stokes' hypothesis for the second coefficient of viscosity leads to the following expressions for the viscous stresses in terms of the rates of strain of a fluid element.

$$\begin{aligned}
 \tau_{xx} &= \frac{2}{3} \frac{\mu}{\text{Re}} \left(2 \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z} \right) & \tau_{xy} &= \tau_{yx} = \frac{\mu}{\text{Re}} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \\
 \tau_{yy} &= \frac{2}{3} \frac{\mu}{\text{Re}} \left(2 \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} \right) & \tau_{yz} &= \tau_{zy} = \frac{\mu}{\text{Re}} \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \\
 \tau_{zz} &= \frac{2}{3} \frac{\mu}{\text{Re}} \left(2 \frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right) & \tau_{zx} &= \tau_{xz} = \frac{\mu}{\text{Re}} \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right)
 \end{aligned} \tag{2.2a}$$

And the heat transfer rate, given by Fourier's law of heat conduction, is proportional to the gradient of the temperature field.

$$Q_x = -\left(\frac{1}{M}\right)^2 \frac{\mu}{(\gamma-1) \text{Re Pr}} \frac{\partial T}{\partial x}$$

$$Q_y = -\left(\frac{1}{M}\right)^2 \frac{\mu}{(\gamma-1) \text{Re Pr}} \frac{\partial T}{\partial y} \quad (2.3a)$$

$$Q_z = -\left(\frac{1}{M}\right)^2 \frac{\mu}{(\gamma-1) \text{Re Pr}} \frac{\partial T}{\partial z}$$

2. Equations of state

With the present method, incompressible or compressible fluid properties may be used. For incompressible flow, the density is constant. For the compressible formulation, the fluid is assumed to behave as a perfect gas with constant specific heats. Thus, the thermodynamic properties of the fluid are related through the following equations of state.

$$\rho = p/RT \quad (2.4)$$

$$e = c_v T \quad (2.5)$$

$$e + p/\rho = c_p T \quad (2.6)$$

The specific heats are assumed constant and determined from the gas constant and the ratio of specific heats.

$$c_v = R/(\gamma-1) \quad (2.7)$$

$$c_p = R \gamma/(\gamma-1) \quad (2.8)$$

3. Transport property relationships

Constant properties are assumed for incompressible flow. While for compressible flow, in keeping with the perfect gas model above, the transport properties are assumed to vary with temperature while the Prandtl number remains constant. The viscosity is evaluated from the Sutherland formula, and the thermal conductivity is obtained from the definition of the Prandtl number.

$$\mu = C_1 T^{\frac{3}{2}} (1 + C_2)/(T + C_2) \quad (2.9)$$

$$k = c_p \mu / \text{Pr} \quad (2.10)$$

D. Algebraic Simplification of the Governing Equations

For this study, the general mathematical model described above was simplified to diminish the cost of performing the computations. First, an algebraic form of the energy equation is presented. Then, simplified forms of the viscous terms are described. The reduction of the governing equations is believed to be valid for the specific flow regimes considered here, and does not compromise the accuracy of the solutions presented.

1. Algebraic form of the energy equation

It is not always necessary to solve a transport equation to determine the energy of the fluid. For many flows the total enthalpy of the fluid is simply constant over the entire flow domain. The conditions that produce this situation are: 1) uniform upstream total enthalpy, 2) adiabatic boundary conditions and no internal energy sources, and 3) fluid Prandtl number equal to unity. The flow conditions for the test cases reported in Chapter V approximately satisfy these conditions, therefore the total enthalpy was assumed to be constant.

For a perfect gas with constant specific heats, the constant total enthalpy condition implies constant total temperature. The algebraic energy equation for the total temperature is

$$T_o = T + (u^2 + v^2 + w^2)/(2c_p) = \text{constant} \quad (2.11a)$$

or

$$T = T_o - (u^2 + v^2 + w^2)/(2c_p) \quad (2.11b)$$

Thus, the static temperature is a simple function of the velocity, and the full energy equation may be eliminated from the system of partial differential equations, Equation 2.1a.

Rather than completely uncouple the energy equation from the set of conservation equations, the algebraic energy equation is implicitly included in the system through the density. Substitution of Equation 2.11b into the perfect gas law Equation 2.4 gives

$$\rho = p/[R(T_o - (u^2 + v^2 + w^2)/(2c_p))] \quad (2.12)$$

Equation 2.12 is substituted directly into Equation 2.1a to eliminate the density. For incompressible flow the momentum equations are independent of the energy equation.

2. Partially-parabolized Navier-Stokes approximation

For flows at moderate and high Reynolds number the effects of streamwise diffusion are negligible, and computational efficiency is improved by dropping these terms. The set of equations

incorporating this approximation is known as the partially-parabolized Navier-Stokes or PPNS equations. This option is implemented in the computer code by a switch, set by the input to the program, to by-pass calculation of the streamwise diffusion terms.

3. Constant property viscous terms

For incompressible flow, the divergence of the velocity field is equal to zero, due to the continuity constraint. This condition alone somewhat simplifies the normal stress terms, and Equation 2.2a reduces to

$$\tau_{xx} = 2 \frac{\mu}{Re} \frac{\partial u}{\partial x} , \quad \tau_{yy} = 2 \frac{\mu}{Re} \frac{\partial v}{\partial y} , \quad \tau_{zz} = 2 \frac{\mu}{Re} \frac{\partial w}{\partial z} \quad (2.2b)$$

The shear stress terms remain as given in Equation 2.2a.

Further restricting the analysis to constant viscosity greatly reduces the complexity of Equation 2.1a. Once the stresses are differentiated, the complete viscous term reduces to the product of the viscosity and the Laplacian of the velocity field [1].

For adiabatic gas flows with Mach numbers less than 0.2, the flow is generally considered to be incompressible and isothermal. Therefore, this formulation is a reasonable approximation, and since fewer derivatives must be evaluated, the computational effort is lessened. This reduction is utilized in the three-dimensional computer code. The two-dimensional code employs the variable property form of the viscous terms for both the compressible and incompressible formulations.

4. Simplified equations for three-dimensional flow

The simplified form of the general governing equations, Equation 2.1a, applicable to three-dimensional, steady, laminar, adiabatic, subsonic flow, includes the three Cartesian momentum equations and the continuity equation; the energy equation is replaced by the constant total temperature condition Equation 2.11. This yields a system of four partial differential equations as a function of the four primary unknown variables.

$$\frac{\partial \vec{E}}{\partial x}(\vec{q}) + \frac{\partial \vec{F}}{\partial y}(\vec{q}) + \frac{\partial \vec{G}}{\partial z}(\vec{q}) = 0 \quad (2.1b)$$

where

$$\vec{q} = [u, v, w, p]^T$$

The flux vectors E , F , and G are identical with those of Equation 2.1a with the exception that only the first four components appear. The density is evaluated from Equation 2.12. The constant property form of the viscous terms is used.

5. Simplified equations for two-dimensional flow

The equations for planar, two-dimensional problems are obtained from the three-dimensional equations by setting the velocity component and all gradients in one of the Cartesian directions equal to zero, and eliminating the momentum equation in that direction. A system of three partial differential equations as a function of three primary unknown variables remains.

$$\frac{\partial \vec{E}}{\partial x}(\vec{q}) + \frac{\partial \vec{F}}{\partial y}(\vec{q}) = 0 \quad (2.1c)$$

where

$$\vec{q} = [u, v, p]^T$$

The flux vectors E and F follow from Equation 2.1b by eliminating one coordinate direction. The variable property form of the viscous terms is used. The constant fluid property, and PPNS assumptions are available as alternatives to the general form of the equations.

E. Transformation of the Governing Equations

To facilitate a numerical solution for problems with arbitrary geometry, an independent variable transformation of the governing equations is performed. The change of independent variables to the computational coordinates is accomplished by applying the chain rule to each derivative. The chain rule operators are

$$\frac{\partial(\)}{\partial x} = \xi_x \frac{\partial(\)}{\partial \xi} + \eta_x \frac{\partial(\)}{\partial \eta} + \zeta_x \frac{\partial(\)}{\partial \zeta}$$

$$\frac{\partial(\)}{\partial y} = \xi_y \frac{\partial(\)}{\partial \xi} + \eta_y \frac{\partial(\)}{\partial \eta} + \zeta_y \frac{\partial(\)}{\partial \zeta}$$

$$\frac{\partial(\)}{\partial z} = \xi_z \frac{\partial(\)}{\partial \xi} + \eta_z \frac{\partial(\)}{\partial \eta} + \zeta_z \frac{\partial(\)}{\partial \zeta}$$

The transformation introduces metric terms into the equations that scale the physical and computational coordinates.

The equations may be expressed in several forms depending on the sequence in which the chain rule operations are carried out. For example, the chain-rule-conservation-law form of Equation 2.1b is

$$\begin{aligned} & \xi_x \frac{\partial \bar{E}}{\partial \xi} + \xi_y \frac{\partial \bar{F}}{\partial \xi} + \xi_z \frac{\partial \bar{G}}{\partial \xi} \\ & + \eta_x \frac{\partial \bar{E}}{\partial \eta} + \eta_y \frac{\partial \bar{F}}{\partial \eta} + \eta_z \frac{\partial \bar{G}}{\partial \eta} \\ & + \zeta_x \frac{\partial \bar{E}}{\partial \zeta} + \zeta_y \frac{\partial \bar{F}}{\partial \zeta} + \zeta_z \frac{\partial \bar{G}}{\partial \zeta} = 0 \end{aligned} \quad (2.1d)$$

Or the equations can be cast in the strong-conservation-law form as

$$\frac{\partial \bar{E}'}{\partial \xi} + \frac{\partial \bar{F}'}{\partial \eta} + \frac{\partial \bar{G}'}{\partial \zeta} = 0 \quad (2.1e)$$

where

$$\bar{E}' = (\xi_x E' + \xi_y \bar{F} + \xi_z \bar{G})/J$$

$$\bar{F}' = (\eta_x E' + \eta_y \bar{F} + \eta_z \bar{G})/J$$

$$\bar{G}' = (\zeta_x E' + \zeta_y \bar{F} + \zeta_z \bar{G})/J$$

For reasons discussed in Chapter IV, the momentum equations are solved in the chain-rule-conservation-law form, while the strong-conservation-law form is used for the continuity equation. Another detail of the transformation that should not be overlooked is to notice that the viscous stresses and heat transfer terms contain derivatives that must also be expanded with the chain rule. The transformation of the viscous terms is given in Appendix B.

1. Two-dimensional equations in computational form

In the present method, the x-momentum and y-momentum equations for two-dimensional flow (first and second components of Equation 2.1c) are cast in the chain-rule-conservation-law form defined by Equation 2.1d. The two-dimensional continuity equation is expressed in the strong-conservation-law form defined by Equation 2.1e. The three components of these equations, written in the computational coordinates are

$$\begin{aligned}
\xi_x (E_1)_\xi + \xi_y (F_1)_\xi + \eta_x (E_1)_\eta + \eta_y (F_1)_\eta &= 0 \\
\xi_x (E_2)_\xi + \xi_y (F_2)_\xi + \eta_x (E_2)_\eta + \eta_y (F_2)_\eta &= 0 \\
\left(\frac{\xi_x}{J} E_3 + \frac{\xi_y}{J} F_3 \right)_\xi + \left(\frac{\eta_x}{J} E_3 + \frac{\eta_y}{J} F_3 \right)_\eta &= 0
\end{aligned} \tag{2.1f}$$

where

$$\vec{q} = (u, v, p)^T$$

and

$$\vec{E} = \begin{bmatrix} \rho u^2 + p - \tau_{xx} \\ \rho uv - \tau_{xy} \\ \rho u \end{bmatrix} \quad \vec{F} = \begin{bmatrix} \rho uv - \tau_{yx} \\ \rho v^2 + p - \tau_{yy} \\ \rho v \end{bmatrix}$$

2. Three-dimensional equations in computational form

The three-dimensional equations are generated in a similar manner. The x-momentum, y-momentum and z-momentum equations (first three components of Equation 2.1b) are set in chain-rule-conservation-law form. The continuity equation is cast in strong-conservation-law form. The structure of the final equations is similar to the two-dimensional form given above, with the addition of the third Cartesian velocity component.

III. SOLUTION ALGORITHM

The procedure for solving the governing equations is presented in this chapter. In addition, two specific properties of the algorithm are considered. First, the stability restrictions that must be satisfied in order to advance the solution by a space-marching method are reviewed. Secondly, the global iteration strategy is examined, and several methods for accelerating the convergence of the pressure field are described.

Space-marching algorithms are the natural method for integrating parabolic equations, developing the solution in a single sweep. However, when applied to elliptic problems, a single sweep with a space-marching solver can only give a good approximation to the solution. In order for downstream conditions to properly influence the flow upstream, further iteration is necessary. The overall efficiency of the algorithm is determined by the speed of a single space-marching solution and the convergence rate of the global iteration technique. The present solution algorithm consists of two distinct procedures that are executed in turn for each global iteration:

- 1.) The primitive variables, \bar{q}^n , are calculated at each station, starting from given upstream conditions and space-marching downstream. This procedure employs an implicit, finite-difference formulation of Equation 2.1, and requires assumed values for the pressure field.
- 2.) The difference between the newly computed pressure field and the assumed pressure field is assessed. A single backsweep with a modified Poisson equation is executed to improve the estimate of the correct pressure field. No corrections are made to the computed velocity field.

These two procedures are described in greater detail in the following sections, and the numerical formulation is presented in the following chapter.

A. Approximate Space-Marching Solution

1. Coupled primitive variable formulation

The space-marching procedure is executed by numerically integrating Equation 2.1 in the primary flow direction (ξ direction) with an implicit, finite-difference method. The solution is initiated from given inlet flow conditions at the upstream boundary. At each marching station in turn, a set of coupled nonlinear equations is assembled and solved for provisional values of the primary flow

variables, \bar{q}^n . As explained below, the downstream terms in the equations are treated explicitly, using assumed initial values or values obtained from the previous global iteration. Therefore, the result of the space-marching solution is only an approximation to the converged solution, depending upon how closely the downstream terms have been approximated.

2. Stability of the space-marching procedure

For subsonic flow the system of partial differential equations (Equation 2.1) is elliptic in the spatial coordinates and is not necessarily well-posed as an initial value problem for solution by a space-marching method. If Equation 2.1 is simply integrated in the space-marching direction, using only the upstream conditions to generate the solution, it is possible for departure solutions to develop. The departure solutions may grow exponentially and therefore make the calculation meaningless. However, proper treatment of the streamwise derivative terms, to include downstream information in the solution procedure, permits a stable space-marching calculation to be executed. Schiff and Steger [63] (as well as Rubin and Reddy [52], Govindan [56], and Liu and Pletcher [58]) have used the eigenvalue analysis method to establish the stability limitations of marching procedures. This linear stability analysis of the frozen coefficient form of Equation 2.1 identifies the stability restrictions on the space-marching solver. In essence, the terms that transmit information about the condition of the downstream flow must be treated as source terms, i.e. fixed values are used in place of values determined during the marching sweep. The treatment of the type-dependent terms is described in the following paragraphs.

For subsonic flow, the downstream pressure "controls" the flow solution through the streamwise pressure gradient terms. The degree to which the streamwise pressure gradient must be restricted is a function of the local Mach number as determined by the Vigneron [46] condition. For the low Mach number flows considered in this study, the streamwise pressure gradient must be completely forward-differenced. At higher Mach numbers, as the sonic velocity is approached, it is desirable to gradually shift the differencing toward the upstream direction. This flow dependent condition reflects the characteristic domain of dependence of the pressure.

Downstream velocities may also influence the solution through the streamwise convection and diffusion terms. Where the streamwise velocity is positive, the streamwise convection terms may be determined entirely from upstream values. However, where the flow reverses, the streamwise convection terms in the momentum equations must include downstream velocities. The

type-dependency of the momentum equations is handled by the use of upwind-difference methods for the streamwise convection terms relative to the direction of the local velocity. This type-dependency does not apply to the continuity equation, which is always formulated with fluxes from the previous marching station.

Finally, downstream velocities also appear in the streamwise viscous terms. At moderate and high Reynolds numbers these terms are negligible relative to the other terms in the momentum equations. Therefore, explicit treatment of the downstream velocities in the viscous terms is stable and, except at very low Reynolds numbers, has little effect on the convergence rate of the solution.

It is emphasized that all elliptic terms are retained, and the type-dependent restrictions are applied to maintain the numerical stability of the space-marching calculation. These stability restrictions reflect the elliptic nature of the governing equations by requiring that downstream information be included in the solution of the local equations. Since the downstream values are unknown, iteration is necessary to converge the initial estimates of these terms, and this process allows downstream boundary conditions to influence the flow solution upstream.

3. Initialization of the space-marching procedure

From the stability analysis discussed above it is clear that downstream values are required in the finite-difference formulation of the governing equations. Starting the global iteration process with a stable initial space-marching sweep therefore requires estimates of these values. The following starting procedure is used in the present study. The pressure field is assumed to be uniform. Estimates of the velocity field are avoided by employing the PPNS equations, in which the streamwise viscous terms are neglected, and by using a variation of the FLARE [64] approximation of the convective terms. Instead of setting the entire convective term equal to zero in regions of reversed flow, the downstream velocity values are simply assumed to be zero. For successive global iterations the downstream velocities are taken to be the "lagged" values from the previous global iteration. Methods of correcting the estimated pressure field are described in the next section.

An alternative to this starting procedure is to utilize the computed results of a previous solution to start the calculation of a new problem. During the course of this research, difficulty was experienced in obtaining reasonable, first-pass solutions for flows containing large obstructions. For these cases, labeled as Case 4 and Case 5 in Chapter V, the fine grid computations were initialized from a solution computed on a coarse mesh and then interpolated onto the fine grid. The coarse grid solutions were started with the method described above.

Other starting procedures are certainly possible, and a very good initial estimate should reduce the number of iterations required to reach convergence. For example, the solution at one Reynolds number may be used to initialize the calculations for a different Reynolds number. Or, a potential flow solution could be used to initialize the pressure field. The simple starting procedure outlined above was employed for this study in order to clearly demonstrate the capability of the iterative space-marching algorithm with the global pressure correction procedure.

B. Pressure Correction Procedures

During the development of the present space-marching procedure, several schemes for accelerating the convergence of the pressure field have been tried and are described below. Of the various pressure correction techniques, the pressure Poisson equation method was found to be the most promising, and was used to compute the results presented in Chapter V. Considering the conditions stated in the following paragraphs, it must be noted that no velocity corrections were made and that the effectiveness of the pressure acceleration methods diminished for problems with significant upstream convection or streamwise diffusion.

1. Motivation for the pressure correction procedure

The pressure field is given special treatment in the algorithm. The reasoning underlying the pressure correction procedure is discussed in this section. In the most general case, the flow at any point may be influenced by the pressure and velocity at all of the surrounding points. With a space-marching algorithm, all of the upstream conditions needed for the solution across a given station are known and present no difficulties. Attention, therefore, is focused on the influence of the downstream terms, which are not known and must be approximated to advance the space-marching calculation.

As discussed above relative to the stability conditions, for subsonic flow the upstream influence of downstream conditions is transmitted through the streamwise pressure gradient, the streamwise viscous stresses, and the streamwise convective terms in regions of reversed flow. Of these three, for a large class of problems, the streamwise pressure gradient is the only significant downstream term in the governing equations. The streamwise viscous stresses and the strengths of any limited regions of flow recirculation are small by comparison. Under these conditions, corresponding to moderate or large Reynolds number flows ($Re > 1$) of a single primary fluid stream, the pressure is the limiting term in achieving a converged solution.

It follows that if the pressure field is known a priori, or if a method is available for determining the pressure, then a single sweep of the space-marching solver will yield the correct solution for all of the flow variables. The pressure correction step in the algorithm permits the introduction of supplemental relationships to efficiently develop the correct pressure field.

It should be noted that the space-marching algorithm is probably not well-suited to such problems where the streamwise velocity field is not clearly defined. Two examples are the flow in a driven cavity and the confluence of two opposing free jets. Both flows contain large regions of "reversed" flow, and slow convergence of the convective terms would be expected.

2. Definition of the pressure correction

In the present space-marching formulation, two pressure values are required at each point during any given global iteration: the calculated pressure, p , and the assumed pressure, \bar{p} , which provides the downstream pressure needed for stable marching. Before executing the next space-marching sweep the assumed pressure field is adjusted to improve the estimate of the assumed pressure field. The pressure correction, p' , is applied using the defining equation

$$\bar{p}^{n+1} = \bar{p}^n + p'$$

As a measure of the error of the approximate space-marched solution, the difference between the calculated pressure and the assumed pressure is defined as

$$\epsilon = p^n - \bar{p}^n$$

The true error in the assumed pressure field is the difference between the final converged pressure and the assumed pressure. This difference, δ , is defined as

$$\delta = p^\infty - \bar{p}^n$$

At convergence, δ , ϵ and p' approach zero at all points in the field.

Clearly, the ideal pressure correction is equal to δ , the true error. Unfortunately, δ cannot be easily determined. The methods described below use different assumptions to estimate the pressure correction.

3. Point relaxation method

The point relaxation method is presented as the "baseline" against which other pressure correction methods were evaluated. The point relaxation method has been used successfully by Rubin

and Reddy [52] to solve reduced forms of the Navier–Stokes equations by the iterative space–marching algorithm. The relaxation method is briefly described in the following paragraphs.

The space–marching sweep directly provides a new estimate of the pressure field. The simplest pressure correction method is to underrelax the computed change in pressure at each node before executing the next space–marching pass. The equation for the pressure correction is simply

$$p' = \omega(p^n - \bar{p}^n) = \omega \epsilon \quad (3.1)$$

Selection of the optimum relaxation factor is problem dependent.

Regardless of any relaxation factor dependence, a large number of iterations are necessary due to the fact that downstream pressure signals are only passed one station upstream for each global iteration. Thus, as many iterations as there are marching stations must elapse before the downstream boundary pressure is first felt at the inlet. During these initial iterations, a finite error must exist due to the lack of boundary information in the local solutions. After this initial transient additional iterations are needed to converge the solution to an acceptable tolerance.

4. Bulk pressure correction method

This procedure rapidly transmits a one–dimensional pressure signal upstream. Compared with the point relaxation method, the bulk pressure gradient technique was found to greatly reduce the number of iterations needed to converge the pressure distribution along straight or symmetric channels. Unfortunately, the method did not prove to be workable when the transverse pressure gradients were large in comparison with the streamwise gradients, as is the case for most external flows and for internal flow problems with large streamline curvature. Since the method does not appear to be generally applicable, no results obtained with this procedure are presented here. However, application of the procedure led to the development of the more general modified pressure Poisson equation method, so the description of the bulk pressure correction method is included. The properties of the procedure are readily apparent, and it can be shown that this procedure is equivalent to the modified pressure Poisson procedure as the diffusion parameter α takes on a large value.

For internal flow problems with small transverse pressure gradients it is useful to split the pressure into a bulk pressure, b_p , plus a transverse variation from the mean, t_p . Thus, the local pressure at any point is given as

$$P_{i,j,k} = b_{p_i} + t_{p_{i,j,k}} \quad (3.2)$$

The bulk pressure is defined as the area average pressure at each station as given by

$$bp_i = \frac{\sum_{j=1}^{j_{\max}} \sum_{k=1}^{k_{\max}} (p_{i,j,k} A_{i,j,k})}{\sum_{j=1}^{j_{\max}} \sum_{k=1}^{k_{\max}} (A_{i,j,k})} \quad (3.3)$$

The transverse pressure variation is then determined as

$$tp_{i,j,k} = p_{i,j,k} - bp_i \quad (3.4)$$

Similarly, the pressure correction is split into two components: the bulk pressure correction, bp' , and the transverse pressure correction, tp' . Thus, the complete pressure correction is

$$p'_{i,j,k} = bp'_i + tp'_{i,j,k} \quad (3.5)$$

The corrected pressures are

$$\overline{bp}^{n+1}_i = \overline{bp}^n_i + bp'_i \quad (3.6)$$

$$\overline{tp}^{n+1}_{i,j,k} = \overline{tp}^n_{i,j,k} + tp'_{i,j,k} \quad (3.7)$$

The bulk pressure correction and the transverse pressure correction are determined independently, and separate relaxation factors may be applied to each of the two components of the pressure.

The bulk pressure correction is obtained by integrating the change in the bulk streamwise pressure gradient. Since the streamwise pressure gradient is proportional to the specified mass flowrate, the bulk streamwise pressure gradient is closely approximated as part of the space-marching solution. But the bulk pressure level at any given station depends upon the pressure level at all stations farther downstream. Integration of the bulk pressure gradient upstream from the exit plane establishes a good approximation of the bulk pressure level throughout the flow domain after only a single space-marching sweep. The bulk pressure correction is therefore obtained as

$$bp'_i = \omega_b \sum_{m=i}^{m=i_{\max}} (bp_m^n - \overline{bp}_m^n) \quad (3.8a)$$

or

$$bp'_i = \omega_b (bp_i^n - \overline{bp}_i^n) + bp'_{i+1} \quad (3.8b)$$

The transverse pressure correction is obtained by relaxing the change in the pressure profile as determined by the coupled space-marching solution at each station. The transverse pressure correction is simply

$$tp'_{i,j,k} = \omega_t (tp_{i,j,k}^n - \overline{tp}_{i,j,k}^n) \quad (3.9)$$

5. Pressure Poisson equation method

The pressure gradients in the computational space can be isolated by writing the momentum equations in covariant form as

$$p_{\xi} = f_1, \quad p_{\eta} = f_2, \quad p_{\zeta} = f_3 \quad (3.10)$$

The functions on the right-hand side of Equation 3.10 contain the convective and viscous terms that depend on the velocity field. (For nonorthogonal grids, components of the pressure gradients in the other coordinate directions also appear on the right-hand side. These components are treated explicitly.) Then, a linear combination of the derivatives of the components of Equation 3.10 yields a Poisson equation for the pressure written as

$$A p_{\xi\xi} + B p_{\eta\eta} + \Gamma p_{\zeta\zeta} = S_p \quad (3.11)$$

where A, B and Γ are arbitrary constants, and

$$S_p = A \frac{\partial}{\partial \xi}(f_1) + B \frac{\partial}{\partial \eta}(f_2) + \Gamma \frac{\partial}{\partial \zeta}(f_3)$$

or, by using Equation 3.10,

$$S_p = A \frac{\partial}{\partial \xi}(p_{\xi}) + B \frac{\partial}{\partial \eta}(p_{\eta}) + \Gamma \frac{\partial}{\partial \zeta}(p_{\zeta})$$

The source term, S_p , is generated from the momentum equations using the current estimate of the velocity field. The space-marching solution of the momentum equations establishes the current velocities and provides the data needed to evaluate the source term at every node. Notice that A, B and Γ are arbitrary constants, rather than constrained to match the continuity condition.

Unfortunately, at any intermediate iteration the space-marched solution is provisional and the value of the source term can only be approximated. Therefore, a direct solution for the final pressure field is not possible. Nonetheless, the converged pressure field must satisfy Equation 3.11, which gives a sound physical basis for estimating the pressure correction.

An alternative form of Equation 3.11 in which the pressure correction explicitly appears is easily developed. The difference between the converged pressure and the assumed pressure values is defined above such that

$$p^m = \bar{p}^n + \delta$$

Assuming the "exact" pressure correction, p^* , will be obtained implies

$$p^* = \delta$$

Then expanding the linear Equation 3.11 to include p^* gives

$$A p_{\xi\xi}^* + B p_{\eta\eta}^* + \Gamma p_{\zeta\zeta}^* = S_{p^*} \quad (3.12)$$

$$S_{p^*} = A \frac{\partial}{\partial \xi}(p_\xi) + B \frac{\partial}{\partial \eta}(p_\eta) + \Gamma \frac{\partial}{\partial \zeta}(p_\zeta) - A \bar{p}_{\xi\xi} - B \bar{p}_{\eta\eta} - \Gamma \bar{p}_{\zeta\zeta}$$

Thus, Equation 3.12 could be solved to estimate the pressure correction. However, the solution of an elliptic equation at each iteration (as is done in the segregated equation algorithms) would add a substantial computational burden. To avoid the solution of the elliptic Equation 3.12, a parabolizing assumption is introduced that allows an approximate solution of Equation 3.12 to be obtained with a single "backsweep" integration from the downstream boundary to the upstream boundary.

A key consideration in making the parabolizing assumption is that in order for the solution of the pressure Poisson equation to be consistent with the converged solution of Equation 2.1, the common terms must be handled in the same manner. In particular, the streamwise pressure gradient must be forward-differenced due to the stability restrictions on the space-marching solver. As a result, the solution at any given node does not directly depend upon the pressure upstream.

Correspondingly, it is assumed that the pressure correction at a given node is independent of the errors in the pressure upstream, and that the pressure corrections upstream will rectify any errors upstream. With this assumption, a parabolized form of Equation 3.12 is obtained which can be written as

$$A p_{\xi}^* + B p_{\eta\eta}^* + \Gamma p_{\zeta\zeta}^* = S'_{p^*} \quad (3.13)$$

where

$$S'_{p^*} = A(\epsilon) + B \frac{\partial}{\partial \eta}(p_\eta) + \Gamma \frac{\partial}{\partial \zeta}(p_\zeta) - B \bar{p}_{\eta\eta} - \Gamma \bar{p}_{\zeta\zeta}$$

The development of Equation 3.13 from Equation 3.12 is presented in finite-difference form in Chapter IV.

Notice that the downstream pressure boundary condition establishes the pressure correction at the exit. For example, the pressure correction is equal to zero at a boundary where the value of the pressure is specified. Thus, Equation 3.13 is integrated in the upstream direction with an implicit,

finite-difference technique to determine the estimated pressure correction at every internal node. This backsweep procedure rapidly propagates the downstream boundary information upstream, with much less computational effort than solving an elliptic equation.

To simplify the present procedure the transverse direction "diffusion" factors for the pressure correction, B and Γ , are specified to be equal. Thus a single arbitrary parameter is defined as

$$\alpha = B/A = \Gamma/A$$

As a final step in this pressure correction procedure, a relaxation factor is applied to the estimated correction such that the "actual" pressure correction is determined as

$$p' = \omega p^* \tag{3.14}$$

And the updated pressure field becomes

$$\bar{p}^{n+1} = \bar{p}^n + \omega p^* \tag{3.15}$$

At this point it is noted that the corrected pressure field is obtained from a fixed value of the downstream pressure, and the corrected pressure at the inlet plane will, in general, not match the desired inlet pressure specified as part of the problem statement. For the purposes of this study, it was desirable to maintain a specified inlet pressure (and density), so that the flow Reynolds number could be specified as an input parameter to the computer program. Therefore, the inlet pressure was held fixed, and the pressure correction at the inlet was subtracted from the absolute magnitude of the pressure at every node in the flow domain as well as the downstream pressure. Thus, the pressure differential between the uniform inlet pressure and the uniform downstream pressure is established iteratively, during the solution procedure. This procedure is not essential. The downstream pressure may be held fixed, in which case the inlet pressure and density will fluctuate until a converged solution is reached.

IV. NUMERICAL SOLUTION

The discretization and algebraic solution of the governing equations are described in this chapter. First, the layout of the computational grid is described. Next, the finite difference forms of the momentum and continuity equations are given. The various boundary conditions are addressed, and the method of solving the system of equations is described. Finally, the numerical method for implementing the global pressure correction procedure is presented.

A. Grid Layout

In order to apply the finite-difference method, the continuous flow field is approximated by values at discrete points on a body-fitted mesh formed by lines of constant ξ , η and ζ (refer to Figure 2.1). On the regular grid, all flow properties are considered to be located at the node points. The (i,j,k) indices correspond to the respective constant (ξ,η,ζ) mesh surfaces.

The accuracy, stability and consistency of finite-difference methods depend on properties of the mesh, such as grid spacing, skewness and aspect ratio. The body-fitted mesh allows a variable physical spacing to concentrate mesh points in regions of large gradients while maintaining uniform spacing in the computational space. The generalized coordinate transformation permits a nonorthogonal mesh which allows more flexibility in grid construction than a strictly orthogonal mesh. Although highly skewed grids can cause ill-conditioned equations, a moderately skewed, nonorthogonal mesh can often give a better node distribution for arbitrary geometries.

An H-type grid topology is used, with the space-marching direction (ξ direction) roughly aligned with the primary flow direction. Grid surfaces of constant ξ are fitted approximately normal to the primary flow streamlines and are termed "stations". Thus, the ξ_{\min} constant surface (station $i=1$) coincides with the upstream boundary, and the ξ_{\max} constant (station $i=i_{\max}$) surface is located at the downstream boundary. Similarly, the η_{\min} , η_{\max} , ζ_{\min} and ζ_{\max} surfaces ($j=1$, $j=j_{\max}$, $k=1$ and $k=k_{\max}$ respectively) are fitted to the side boundaries.

There are many techniques for generating the interior mesh points. The grids used in this study are constructed with adaptations of the algebraic stretching transformations described in Anderson et al. [3] and summarized in Appendix E. To minimize errors in the numerical solution, the following conditions are imposed on the generation of the body-fitted coordinates:

1. Physical boundaries are fitted to surfaces upon which one of the computational coordinates is constant.

2. The computational coordinates are scaled to produce a uniform mesh ($\Delta\xi = \Delta\eta = \Delta\xi = 1$).
3. Grid lines intersect the boundaries in the direction normal to the boundary.
4. Grid skewness and grid aspect ratio are limited, although not formally minimized.
5. The metrics are continuous, and have continuous first derivatives.

The Cartesian coordinates of the node points do not appear directly in the transformed equations, but are needed to evaluate the metric terms. The metrics are evaluated with second-order accurate finite-difference expressions. For the two-dimensional code, the metrics are computed in a grid generation routine and stored along with the Cartesian coordinates. To reduce the memory requirements of the three-dimensional code, the metrics are recomputed at each station during each marching sweep. Definitions of the metrics and the Jacobian of the transformation are given in Appendix A, along with the finite-difference approximations.

B. Finite-Difference Equations

The basic step in the space-marching procedure is to advance the solution of the governing equations from a given station i , to the next station downstream, indexed as $i+1$. This section describes the formulation of the coupled, implicit, finite-difference equations for the unknown variables, \bar{q}^n , at station $i+1$. With the space-marching method, the newly computed profiles of \bar{q}^n at and upstream of station i appear explicitly, and assumed or lagged values of \bar{q}^{n-1} at station $i+1$, station $i+2$ and station $i+3$ downstream are required.

As described in Chapter Two, the transformed governing equations may be cast in several forms. The chain-rule-conservation-law form is used for the momentum equations, and the strong-conservation-law form is used for the continuity equation. The numerical properties of the different forms and the reasons for selecting this combination are discussed below.

Finite-difference representations in the strong-conservation-law form may be constructed such that the assembled set of equations is globally conservative. This property holds even on coarse grids, where the derivatives are not accurately approximated. The truncation errors in the equations for adjacent nodes cancel, so that the summation of all of the difference equations is identical to numerical integration over the surface of the entire flow domain. With the chain-rule-conservation-law form, the products of truncation errors and finite gradients do not cancel. Global

conservation is satisfied within reasonable accuracy, since the truncation errors vanish for any consistent difference method as the grid is refined.

Geometric considerations associated with these forms have been examined by Hindman [65] and Shamroth and Gibeling [66]. With the strong-conservation-law form, the metrics must be consistent with the particular finite-difference formulation used. Otherwise, numerical source terms arise due to truncation errors in the metrics. There are no special constraints on the metrics in the chain-rule-conservation-law form.

The flow physics of the momentum equations require several type-dependent difference expressions, which naturally vary from node to node. With the strong-conservation-law form it is difficult, if not impossible, to adjust the differencing and preserve a globally conservative set of equations. Furthermore, to correctly represent the type-dependent terms would require many different evaluations of the metrics. By using the chain-rule-conservation-law form, constraints on the metric terms are avoided, and although truncation errors are significant on coarse grids, a global momentum balance is achieved as the grid is refined. However, no type-dependent terms appear in the continuity equation. Therefore, to guarantee conservation of mass on coarse grids independent of truncation errors, the strong-conservation-law form is used for the continuity equation. The special requirements imposed on the metric terms produce an algebraic equation that is equivalent to the finite-volume continuity equation in physical coordinates.

The details of the finite-difference form of the momentum and continuity equations comprise the following sub-sections. Approximation of partial derivatives by the finite-difference method is straightforward. However, a problem related to the use of a regular grid, as opposed to the staggered grid, was anticipated. Central-differencing of first-derivative terms often uncouples the solution of the odd and even nodes. One method of controlling this problem is to add artificial dissipation terms to the equations. With the space-marching method this is not necessary for the streamwise direction, since these derivatives are naturally expressed with forward or backward-difference operators. In the development of the algorithm it was decided to use one-sided difference operators for the first-derivatives in the transverse direction as well. Splitting the three-point, second-order accurate formulas into implicit and explicit parts was elected to limit the band-width of the linearized algebraic equations. The splitting does not reduce accuracy since at convergence the forms are equivalent.

1. Momentum equations

The momentum equations in the chain-rule-conservation-law form (Equation 2.1d) are expanded as finite-differences in the computational domain about the node located at $(i+1,j,k)$. The differencing method employed for each of the terms within the flux vectors E , F and G is described below and summarized in Table 4.1. The primary values that appear in the difference equations at node $(i+1,j,k)$ are shown schematically in Figures 4.1 and 4.2., and the example difference formulas given below are all expanded about that node. For clarity the triple subscripts are replaced with the labels shown in Figure 4.1.

Table 4.1. Summary of finite-difference operators for the momentum equations

Type of Term	Finite-Difference Approximation Type	Order
Streamwise pressure gradient	Forward	1st or 2nd
Transverse pressure gradient	Forward or Backward ^a	1st or 2nd
Streamwise normal stress	Central	2nd
Transverse normal stress	Central	2nd
Streamwise shear stress	Central	2nd
Transverse shear stress	Central	2nd
Streamwise convection	Upwind	1st or 2nd
Transverse convection	Central/Hybrid-Upwind	2nd/1st

^a Direction arbitrarily depends on boundary conditions.

a. Streamwise pressure gradient terms The streamwise pressure gradient is forward-differenced. The majority of results presented here were obtained with a first-order representation. A second-order accurate expression was added to the code and successfully used for some of the cases. The downstream pressure values are the assumed values (denoted with an overbar), resulting from the global pressure correction procedure following the previous iteration. The first-order accurate, forward-difference formula at $(i+1,j,k)$ is

$$p_f = [-p_f^n + \bar{p}_f^n] / \Delta \xi \quad (4.1a)$$

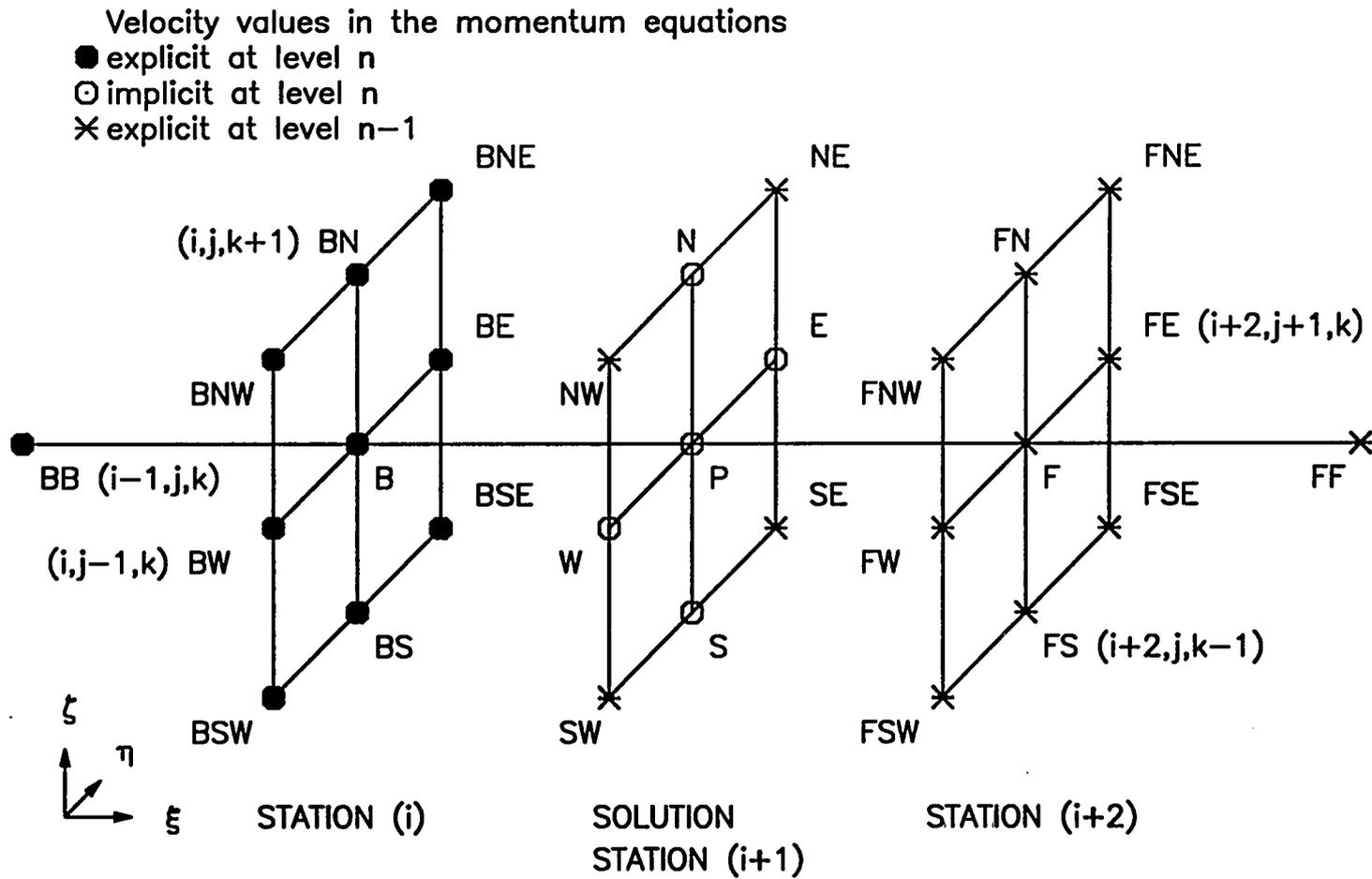


Figure 4.1. Velocity node layout for the momentum equations at $(i+1,j,k)$

Pressure values in the momentum equations

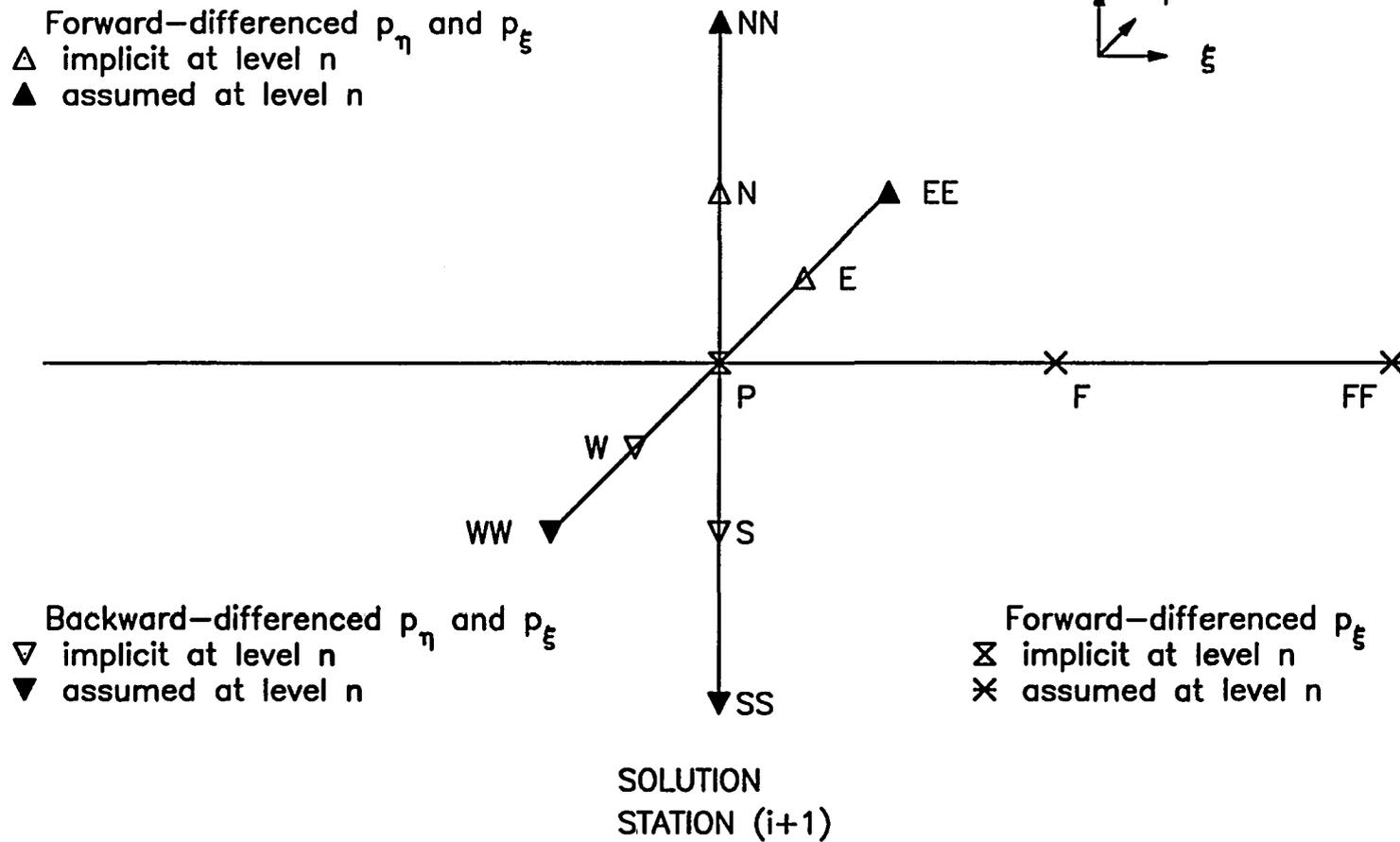


Figure 4.2. Pressure node layout for the momentum equations at $(i+1,j,k)$

And the second-order accurate, forward-difference formula is

$$p_t = [-3 p_P^n + 4 p_P^n - p_{PP}^n] / (2\Delta\xi) \quad (4.1b)$$

b. Transverse pressure gradient terms Central-differencing of the transverse pressure gradient permits the even and odd node pressures to uncouple, which can produce two independent profiles at a given station. Instead of explicitly adding dissipation terms, second-order accurate, one-sided differences are used to link the pressures at neighboring nodes. The three-point difference formulas include nodes outside of the band-width of a block-tridiagonal solver. Rather than use a pentadiagonal solver, the difference formula was split into an implicit, first-order accurate term plus an explicit, second-order correction. The second-order accurate, forward-difference formula in the η direction for the node located $(i+1,j,k)$ is

$$p_\eta = [(-2 p_P^n + 2 p_E^n) - (p_{EE}^n - 2 p_E^n + p_P^n)] / (2\Delta\eta) \quad (4.2)$$

The direction of the one-sided difference is opposite the direction chosen for the one-sided difference in the continuity equation. The choice is arbitrary in most cases, but does influence the formulation of the side boundary conditions.

c. Viscous stress terms Second-order accurate, central-differences are used to approximate the derivatives in the viscous terms. The chain-rule expansion of the viscous terms is given in Appendix B. First and second-derivatives of the velocity components appear in these terms, as well as first-derivatives of the metrics and the viscosity. The viscosity is determined from the current estimate of the temperature, which allows the viscosity and metrics to be lumped together into explicit coefficient functions.

The transverse second-derivatives and the associated first-derivatives are treated implicitly. The method may be illustrated with an example term in the η direction of the form:

$$a u_{\eta\eta} + a_\eta u_\eta \quad (4.3a)$$

where the coefficient, a , is a function of the metrics and the viscosity. The complete term can be written in implicit form as

$$(A + A') u_E^n - 2A u_P^n + (A - A') u_W^n \quad (4.3b)$$

where the implicit coefficients are evaluated by

$$A = a_P / \Delta\eta^2$$

$$A' = (a_E - a_W) / (4\Delta\eta^2)$$

Equation 4.3b is obtained by expanding the derivatives in Equation 4.3a with central-differences about the location $(i+1,j,k)$ as

$$u_{\eta\eta} = [u_E^n - 2u_F^n + u_W^n]/\Delta\eta^2 \quad (4.3c)$$

$$a_\eta = [a_E - a_W]/(2\Delta\eta) \quad (4.3d)$$

$$u_\eta = [u_E^n - u_W^n]/(2\Delta\eta) \quad (4.3e)$$

The streamwise second-derivatives and the associated first-derivatives are handled in much the same way, except known upstream values, lagged downstream values, and implicit values at the marching station are used. An example term of the form:

$$a u_{\xi\xi} + a_\xi u_\xi \quad (4.4a)$$

expanded at the node located at $(i+1,j,k)$ can be written as

$$(A + A') u_F^{n-1} - 2A u_F^n + (A - A') u_B^n \quad (4.4b)$$

Central-differences are used for the streamwise derivatives. The implicit coefficients are evaluated as

$$A = a_F/\Delta\xi^2$$

$$A' = (a_F - a_B)/(4\Delta\xi^2)$$

Example forms of the mixed partial-derivative terms are

$$a u_{\xi\eta} + a_\xi u_\eta \quad (4.5a)$$

and

$$a u_{\xi\eta} + a_\eta u_\xi \quad (4.5b)$$

The explicit central-difference form of Equation 4.5a is

$$A u_{FE}^{n-1} - A u_{FW}^{n-1} - A u_{BE}^n + A u_{BW}^n + A' u_E^{n-1} - A' u_W^{n-1} \quad (4.5c)$$

Where the implicit coefficients are evaluated by

$$A = a_F/(4\Delta\xi\Delta\eta)$$

$$A' = (a_F - a_B)/(4\Delta\xi\Delta\eta)$$

The mixed-derivatives used to expand Equation 4.5c at $(i+1,j,k)$ are

$$u_{\xi\eta} = [u_{FE}^{n-1} - u_{FW}^{n-1} - u_{BE}^n + u_{BW}^n]/(4\Delta\xi\Delta\eta) \quad (4.5d)$$

$$a_\xi = [a_F - a_B]/(2\Delta\xi) \quad (4.5e)$$

$$u_\eta = [u_E^{n-1} - u_W^{n-1}]/(2\Delta\eta) \quad (4.5f)$$

d. Streamwise convection terms The streamwise convective terms are upwinded. To minimize numerical dissipation the second-order upwind method is used, except adjacent to boundaries. In regions of reversed flow the grid direction of the differencing is reversed, and the convective flux terms at the downstream nodes are lagged. Examples of the second-order accurate difference operators used for the node located at $(i+1,j,k)$ are

For $U \geq 0$

$$(\rho u^2)_E = [3 (\rho u^2)_P - 4 (\rho u^2)_B + (\rho u^2)_{BB}] / (2\Delta\xi) \quad (4.6a)$$

For $U < 0$

$$(\rho u^2)_E = [-3 (\rho u^2)_P + 4 (\rho u^2)_{F^{-1}} - (\rho u^2)_{FF^{-1}}] / (2\Delta\xi) \quad (4.6b)$$

Where : $U = (u \xi_x + v \xi_y + w \xi_z)_P$

e. Transverse convection terms The hybrid differencing scheme is used for the transverse convective terms, formulated with a weighted average of backward and forward-differences. For mesh Reynolds numbers less than two, equal weight factors produce central-differencing. For larger mesh Reynolds numbers, possible instability is controlled by progressively weighting the differencing in the upwind direction to suppress the appearance of positive, off-diagonal coefficients. Although the formal accuracy is reduced to first-order, the type-dependency improves the physical representation of the combined convection-diffusion terms. As an example, the transverse convection terms in the η direction from the x-momentum equation are

$$\eta_x (\rho u^2)_\eta + \eta_y (\rho uv)_\eta + \eta_z (\rho uw)_\eta \quad (4.7a)$$

The metrics are evaluated at $(i+1,j,k)$. Approximating the convective derivatives about the location $(i+1,j,k)$ with the hybrid scheme and linearizing the flux terms gives the following implicit form for the u velocity component

$$f C_E u_E^u + (b-f) C_P u_P^u - b C_W u_W^u \quad (4.7b)$$

The linearized coefficients are evaluated at each node as presented in Appendix D. In the interest of clarity, the simple frozen coefficient expressions are

$$C_E = [\eta_x (\hat{\rho} \hat{u})_E + \eta_y (\hat{\rho} \hat{v})_E + \eta_z (\hat{\rho} \hat{w})_E] / \Delta\eta$$

$$C_P = [\eta_x (\hat{\rho} \hat{u})_P + \eta_y (\hat{\rho} \hat{v})_P + \eta_z (\hat{\rho} \hat{w})_P] / \Delta\eta$$

$$C_W = [\eta_x (\hat{\rho} \hat{u})_W + \eta_y (\hat{\rho} \hat{v})_W + \eta_z (\hat{\rho} \hat{w})_W] / \Delta\eta$$

The forward and backward-difference weight factors, f and b , are determined by testing the implicit convective term coefficients against the coefficients of the implicit viscous terms defined in Equation 4.3b. Notice that the viscous coefficients are non-positive, and that for central-differencing the weights, b and f , are both equal to one-half. The conditions that are satisfied are

$$\text{For } f = \frac{1}{2}, \text{ if } [(A + A') + f C_E] > 0,$$

then shift toward backward differencing and assign b and f as

$$f = -(A + A')/C_E \text{ and } b = 1 - f \quad (4.7c)$$

$$\text{For } b = \frac{1}{2}, \text{ if } [(A + A') - b C_W] > 0,$$

then shift toward forward differencing and assign b and f as

$$b = (A + A')/C_W \text{ and } f = 1 - b \quad (4.7d)$$

Else, use central differencing.

$$f = b = 1/2$$

To illustrate the upwind adjustment, the η direction example begun above is continued here. The implicit viscous terms in the η direction for the constant property case of the x -momentum equation given in Appendix B are

$$(\mu/Re)[\eta_x (\eta_x u_{\eta\eta} + \eta_{x\eta} u_\eta) + \eta_y (\eta_y u_{\eta\eta} + \eta_{y\eta} u_\eta) + \eta_z (\eta_z u_{\eta\eta} + \eta_{z\eta} u_\eta)] \quad (4.7f)$$

These terms correspond to the form of Equation 4.3a. The implicit coefficients, A and A' , are then obtained from Equation 4.3b. The viscous term coefficients in Equations 4.7c and 4.7d that determine the hybrid-difference weight factors used in example Equation 4.7b above are thus

$$A = (\mu/Re) (\eta_x^2 + \eta_y^2 + \eta_z^2) / \Delta\eta^2 \quad (4.7g)$$

$$A' = (\mu/Re) \{ \eta_x [(\eta_x)_E - (\eta_x)_W] + \eta_y [(\eta_y)_E - (\eta_y)_W] + \eta_z [(\eta_z)_E - (\eta_z)_W] \} / (4\Delta\eta^2)$$

Where the metrics are evaluated at $(i+1, j, k)$, except as indicated.

f. Metric terms Finally, it is noted that with the chain-rule-conservation-law form there are no specific constraints on the evaluation of the metrics [65]. Second-order accurate, finite-difference formulas for the metrics are applied to the metric definitions given in Appendix A.

2. Continuity equation

The discretized form of the continuity equation is derived from the strong-conservation-law form of the governing equations (Equation 2.1e). The continuity equation is expanded in finite-volume form about the mesh location $(i+1/2,j,k)$ to give

$$(E'_f - E'_b)/\Delta\xi + (F'_e - F'_w)/\Delta\eta + (G'_n - G'_s)/\Delta\zeta = 0 \quad (4.8a)$$

Here the subscripts denote the faces of the resulting control volume as shown in Figure 4.3. Substitution for the fluxes defined in Equations 2.1a and 2.2b, and canceling $\Delta\xi = \Delta\eta = \Delta\zeta$ gives

$$\left(\frac{\rho U}{J}\right)_f - \left(\frac{\rho U}{J}\right)_b + \left(\frac{\rho V}{J}\right)_e - \left(\frac{\rho V}{J}\right)_w + \left(\frac{\rho W}{J}\right)_n - \left(\frac{\rho W}{J}\right)_s = 0 \quad (4.8b)$$

where the contravariant velocities are

$$U = \xi_x u + \xi_y v + \xi_z w$$

$$V = \eta_x u + \eta_y v + \eta_z w$$

$$W = \zeta_x u + \zeta_y v + \zeta_z w$$

To ensure global and local conservation of mass in spite of truncation errors, several conditions are imposed on the set of difference equations. To satisfy global conservation the assembled control volumes must fill the physical space, and secondly, the common fluxes of neighboring cells must be equated. Thirdly, to be locally conservative, the calculated face areas must form exactly closed control volumes. The first condition is easily met by assigning a control volume to each node. The other two conditions restrict the interpolation of the fluxes and the evaluation of the metrics as discussed below. The end result of satisfying these conditions is that the transformed difference equation is algebraically equivalent to the numerical integration of the finite-volume continuity equation in physical coordinate form.

a. Cell face center-point flux interpolation The front and back face fluxes are obtained directly at the nodes P at $(i+1,j,k)$ and B at (i,j,k) in the center of these faces. However, the fluxes at the centers of the other four faces must be interpolated. Simple averaging among the four neighboring nodes leads to a central-difference expression that causes even/odd decoupling and produces saw-tooth profiles. To suppress this, the interpolation is done by a Taylor-series expansion biased to one side of the control volume center. Second-order accuracy is maintained, but some dissipation error is introduced into the equation. For example, the east face flux terms at the location $(i+1/2,j+1/2,k)$ are determined from the properties and gradients about the node P at $(i+1,j,k)$ as

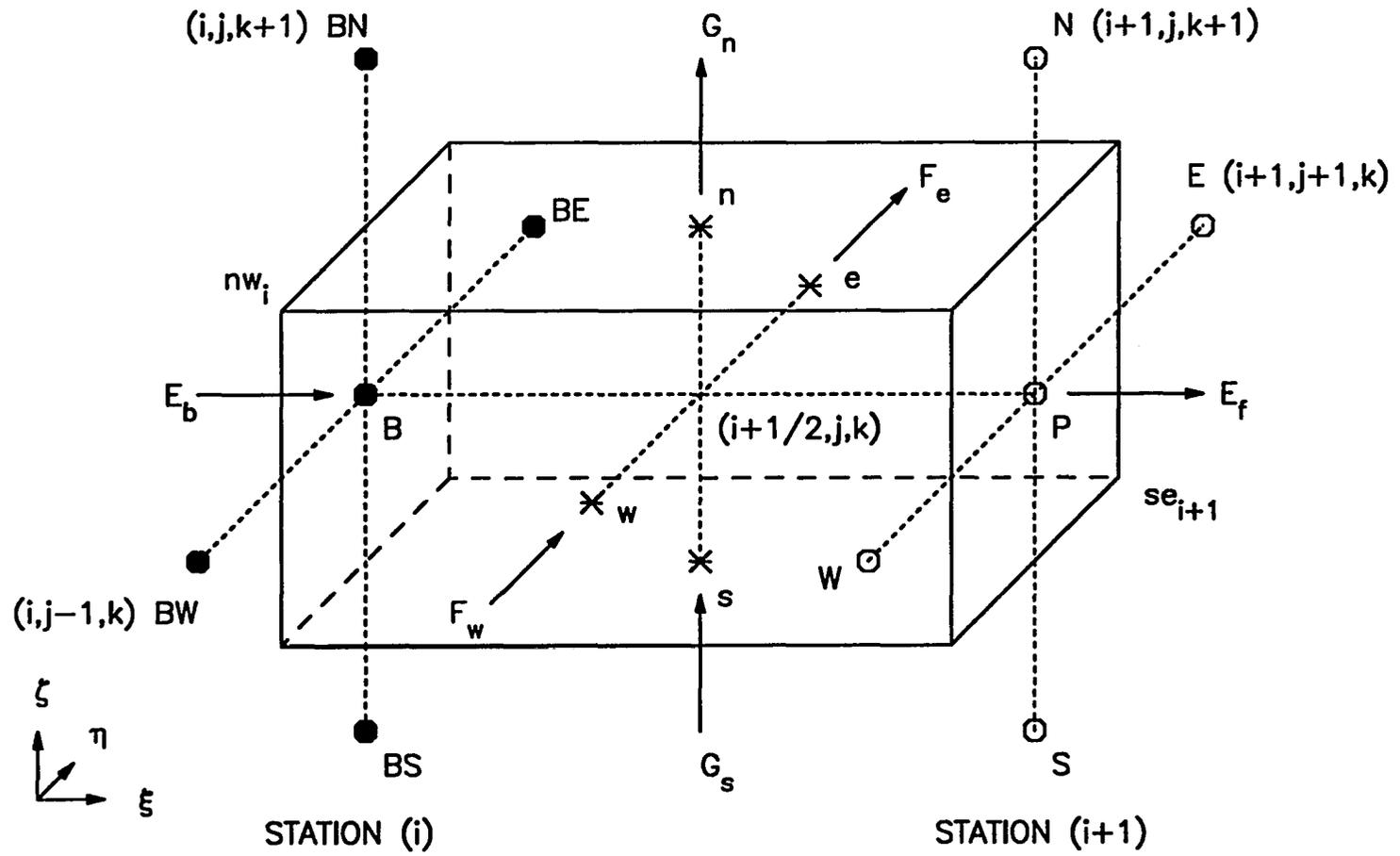


Figure 4.3. Continuity control volume centered at $(i+1/2, j, k)$

$$(\rho u)_e = (\rho u)_P + \frac{\Delta \eta}{2} \frac{\partial}{\partial \eta} (\rho u) \Big|_P - \frac{\Delta \xi}{2} \frac{\partial}{\partial \xi} (\rho u) \Big|_P \quad (4.9a)$$

$$(\rho v)_e = (\rho v)_P + \frac{\Delta \eta}{2} \frac{\partial}{\partial \eta} (\rho v) \Big|_P - \frac{\Delta \xi}{2} \frac{\partial}{\partial \xi} (\rho v) \Big|_P \quad (4.9b)$$

$$(\rho w)_e = (\rho w)_P + \frac{\Delta \eta}{2} \frac{\partial}{\partial \eta} (\rho w) \Big|_P - \frac{\Delta \xi}{2} \frac{\partial}{\partial \xi} (\rho w) \Big|_P \quad (4.9c)$$

The derivatives are approximated with first-order accurate, one-sided differences. The streamwise derivative is always backward-differenced, and the direction of the transverse derivative is opposite the direction of the transverse pressure gradient in the momentum equations. Explicit correction terms are used to maintain the block-tridiagonal matrix. The explicit terms, indicated with a caret, are lagged within the nonlinear coefficient loop. As an example, one of the east face terms is

$$\begin{aligned} (\rho v)_e &= (\rho v)_P^B + \frac{1}{2} [(\hat{\rho} \hat{v})_P - (\hat{\rho} \hat{v})_W] \\ &\quad - \frac{1}{4} [(\hat{\rho} \hat{v})_P - (\hat{\rho} \hat{v})_{BB}] \end{aligned} \quad (4.9d)$$

The west face flux at $(i+1/2, j-1/2, k)$ is then determined in a similar manner about the adjacent node at $(i+1, j-1, k)$. For example, the corresponding west face term is

$$\begin{aligned} (\rho v)_w &= (\rho v)_W^B + \frac{1}{2} [(\hat{\rho} \hat{v})_W - (\hat{\rho} \hat{v})_{WW}] \\ &\quad - \frac{1}{4} [(\hat{\rho} \hat{v})_W - (\hat{\rho} \hat{v})_{BBW}] \end{aligned} \quad (4.9e)$$

The underlying requirement is that the east face flux from the cell associated with the node $(i+1, j-1, k)$ be identical to the west face flux into the neighbor cell assigned to the node $(i+1, j, k)$. This can be verified by careful examination of the indexes in the equations above and by reference to Figure 4.1 and Figure 4.3.

b. Control volume face areas The metric terms for each face represent the control volume face areas. In order to ensure strict flux conservation the metrics must satisfy the geometric conservation law. Otherwise the control volume would not be exactly closed, and a uniform velocity field would not satisfy the discretized continuity equation. In other words, geometric truncation errors introduce local mass sources into the continuity equation. Therefore, the metrics are evaluated for each different face, using the physical coordinates of the corner points of the control volume. This

treatment of the metrics exactly fits the projected areas of the physical control volume faces. As an example, the projected area of the east face onto the z - x plane is given by

$$\begin{aligned} \frac{\eta_y}{J} \Big|_e &= x_t \Big|_e z_t \Big|_e - x_e \Big|_e z_e \Big|_e \\ &= [(x_{i+1,se} - x_{i,ne})(z_{i+1,ne} - z_{i,se}) - [(x_{i+1,ne} - x_{i,se})(z_{i+1,se} - z_{i,ne})]/2 \quad (4.10) \end{aligned}$$

where the corner point coordinates are obtained by averaging from the surrounding four nodes at the same station. For example

$$x_{i+1,ne} = x_{i+1,j+1/2,k+1/2} = (x_P + x_E + x_N + x_{NE})/4$$

$$y_{i+1,ne} = y_{i+1,j+1/2,k+1/2} = (y_P + y_E + y_N + y_{NE})/4$$

$$z_{i+1,ne} = z_{i+1,j+1/2,k+1/2} = (z_P + z_E + z_N + z_{NE})/4$$

and

$$x_{i,se} = x_{i,j+1/2,k-1/2} = (x_B + x_{BE} + x_{BS} + x_{BSE})/4$$

$$y_{i,se} = y_{i,j+1/2,k-1/2} = (y_B + y_{BE} + y_{BS} + y_{BSE})/4$$

$$z_{i,se} = z_{i,j+1/2,k-1/2} = (z_B + z_{BE} + z_{BS} + z_{BSE})/4$$

C. Numerical Boundary Conditions

The solution domain may be enclosed by five different types of boundaries. With few exceptions, boundary conditions specify the value of variables at nodes on the boundaries, eliminating the need to solve the finite-difference forms of the governing equations. Also, at interior nodes adjacent to boundaries, the finite-difference equations are restricted so that values at locations outside the domain are not needed. For example, adjacent to the inlet boundary the second-order backward-difference formula is replaced by the first-order method. The reduction in accuracy usually is not significant, since only those gradients that are negligibly small are effected. The specification of conditions at each of the boundaries and the modifications needed in the equations at nodes adjacent to these boundaries follows.

1. Upstream boundary conditions

The space-marching pass starts from the upstream boundary, where the inlet flow profiles of all the the variables are specified. These profiles must be consistent with the desired mass flowrate,

Reynolds number and Mach number, and match the assumption of constant total temperature. The bulk properties at the inlet, along with the flow field reference length, were used to nondimensionalize the variables.

Since the variables are all specified, there is no need to solve the governing equations on the upstream boundary. The equations are solved at the first station inside the boundary. These equations utilize the known conditions at the inlet. Second-order, backward-differences for streamwise derivatives that ordinarily would use data from two upstream stations are restricted to the first-order method.

It is notable that the upstream pressure influences the flow solution only through the density, since the streamwise pressure gradient term is forward differenced. As part of the pressure correction procedure the inlet pressure is matched to the interior pressure. The value of the streamwise pressure gradient at the boundary may be specified either as zero or as an unknown constant. The zero pressure gradient condition corresponds to external flow or the entrance to a channel, while the constant pressure gradient condition applies to fully developed channel flows. The constant is generally unknown and is extrapolated from the solution for the pressure at the interior stations.

2. Side boundary conditions

For internal flow problems the side boundaries are usually solid walls on which the velocity components are set equal to zero. The pressure is implicitly determined from the interior solution by setting the pressure gradient normal to the wall equal to zero. Whenever possible, symmetric side boundaries are used to limit the extent of the computational domain. In this case the equations are not altered and the required values outside the computational domain are associated with the appropriate values inside the domain.

For external flow problems the edge of the grid is located far from the body surface. The streamwise velocity component is explicitly specified equal to the freestream velocity, and the pressure is explicitly specified assuming uniform total pressure. The transverse velocity components are determined implicitly with the specification that the gradient normal to the boundary is zero. The continuity control volumes for nodes adjacent to the boundary are extended to reach the boundary.

3. Downstream boundary conditions

The governing equations are solved at the downstream station in the same manner as for interior stations, with the following modifications to the finite-difference equations.

First, streamwise gradients in velocity are assumed to be small. Therefore, the streamwise diffusion terms are neglected. The upstream convection terms are unchanged, but if reversed flow is present, convection of momentum from downstream is set equal to zero.

The forward streamwise pressure difference requires a known downstream pressure, located one step outside of the grid, that imposes the exit flow pressure field upon the interior flow solution. For external flows this downstream pressure is set equal to the freestream pressure. For internal flows the value of the downstream pressure is adjusted between iterations in order to establish a pressure change from the inlet reference pressure consistent with the specified mass flowrate. In any case, it is assumed that the downstream boundary is located sufficiently far from any disturbance that the transverse pressure gradients are negligible. Should this not be the case, an appropriate pressure profile must be specified to represent the downstream pressure variation.

4. Rearward-facing step boundary conditions

The nodes on the face of a rearward-facing step are treated like the upstream boundary nodes. No governing equations are solved on the boundary, since the velocity components are known to be zero, and the pressure does not influence the flow downstream. However, to obtain a wall pressure value, the pressure field is extrapolated from downstream. The normal momentum equation reduces to the specification of zero normal pressure gradient near the wall.

Backward-differences at the nodes located downstream of the step face are restricted to first-order differences, just as for nodes adjacent to the upstream boundary.

5. Forward-facing step boundary conditions

There is no need to solve the governing equations at nodes on the face of a forward-facing step. The velocity components are specified equal to zero, and the pressure is extrapolated from upstream assuming zero normal pressure gradient.

However, the step must influence the space-marching solution at the previous station, just upstream of the step. This is accomplished by modifying the set of equations for the nodes that face the step. The streamwise momentum equations at these nodes are deleted from the system, and replaced with the continuity equations for the control volumes located on the step face. These continuity equations "tell" the flow there is an obstruction downstream. A more detailed description of the formulation is presented in Appendix C. Also, any forward-differences at nodes adjacent to the step face are restricted to first-order expressions.

D. Solution of the System of Discretized Equations

To advance the space-marching solution, the finite-difference equations and boundary conditions for all nodes at station $i+1$ are assembled into a system of equations for the primary variables, \bar{q}^n . The implicit terms are all located at station $i+1$. Hence, three-dimensional problems require the simultaneous solution for all variables in a plane, while two-dimensional problems involve a line solver. Solution of the set is complicated by the nonlinear convection terms and, for the three-dimensional case, by the size and structure of the matrix. The steps employed to obtain an implicit solution for the primary variables are presented next.

1. Newton linearization with coupling

The Newton linearization procedure is applied to the nonlinear convective terms in the momentum and continuity equations. The nonlinear functions are analytically differentiable, allowing the terms to be expanded with the Taylor series about the nodal values from the previous global iteration. Second-order terms are discarded, leaving an expression that is linear in \bar{q}^n . Examples of the quasilinearized terms are shown below. The bracketed terms are all evaluated using coefficient values.

$$\begin{aligned} \rho u \approx & \left[\rho \left(1 + \frac{u^2}{c_p T} \right) \right] u + \left[\frac{\rho u v}{c_p T} \right] v + \left[\frac{\rho u w}{c_p T} \right] w + \left[\frac{u}{RT} \right] p \\ & - \left[\rho u \left(\frac{u^2 + v^2 + w^2}{c_p T} \right) \right] - [\rho u] \end{aligned} \quad (4.11a)$$

$$\begin{aligned} \rho u v \approx & \left[\rho v \left(1 + \frac{u^2}{c_p T} \right) \right] u + \left[\rho u \left(1 + \frac{v^2}{c_p T} \right) \right] v + \left[\frac{\rho u v w}{c_p T} \right] w + \left[\frac{u v}{RT} \right] p \\ & - \left[\rho u v \left(\frac{u^2 + v^2 + w^2}{c_p T} \right) \right] - [2\rho u v] \end{aligned} \quad (4.11b)$$

The complete derivation of the linearized expressions is given in Appendix D.

The linear system is solved using the provisional coefficients. When the solution results in a large change in the variables, the quasilinearization is performed about the new values, and the solution is repeated using the updated coefficients. Since the algorithm entails global iteration it is not necessary to converge the nonlinear coefficients to high accuracy during each marching sweep. For

the test cases presented here, iteration on the nonlinear coefficients was not typically required at most of the marching stations to advance the solution.

2. The linear system for two-dimensional flows

The two-dimensional difference equations and boundary conditions at station $i+1$ form a block-tridiagonal system of equations for \bar{q}^n . The inclusion of implicit terms in the difference operators given above was restricted to the adjacent nodes in order to preserve the tridiagonal structure. This type of linear system is easily solved by a block elimination procedure, using the routine by Chakravarthy found in Anderson et al. [3]. This general routine was modified to enhance the efficiency of the line solver specifically for the 3×3 block size of the two-dimensional equations by unrolling the short loops.

3. The linear system for three-dimensional flows

The three-dimensional difference equations and the boundary conditions at station $i+1$ form a sparse block-banded system of equations for \bar{q}^n . The inclusion of implicit terms from a five-point molecule in the $i+1$ plane results in the five-banded structure typical of plane solvers. The linearized equations were solved by the Douglas-Rachford form of the alternating-direction-implicit (ADI) algorithm (see Reference 67). This procedure involves a two-step sequence. Each step sweeps the plane with a series of line solutions oriented along one of the coordinate directions in the cross-plane. Therefore, the large sparse matrix is partitioned into numerous smaller systems with block-tridiagonal structure. These equations are solved with the same block elimination procedure used in the two-dimensional code, but modified for the 4×4 block size of the three-dimensional equations.

E. Global Pressure Correction Procedure

Several methods for correcting the assumed pressure field to accelerate convergence of the global iterations were presented in Chapter III. The methods employing simple algebraic operations have been described in sufficient detail. The finite-difference method of implementing a pressure correction based upon the pressure Poisson equation is described in this section. Also, the assumptions used to avoid the solution of an elliptic equation are presented.

1. Finite-difference form of the pressure Poisson equation

The governing equations can be manipulated to describe the pressure field by a Poisson equation, Equation 3.11. This equation is expanded to introduce the estimated value of the pressure correction, p^* . For clarity, Equation 3.12 is repeated here.

$$A p_{\xi\xi}^+ + B p_{\eta\eta}^+ + \Gamma p_{\zeta\zeta}^+ = S_{p^+} \quad (3.12)$$

where

$$S_{p^+} = A \frac{\partial}{\partial \xi}(p_{\xi}) + B \frac{\partial}{\partial \eta}(p_{\eta}) + \Gamma \frac{\partial}{\partial \zeta}(p_{\zeta}) - A \bar{p}_{\xi\xi} - B \bar{p}_{\eta\eta} - \Gamma \bar{p}_{\zeta\zeta}$$

In order for the global iteration cycle to converge, the finite-difference form of Equation 3.12 must be consistent with the finite-difference form of the governing equations. The equations can be expressed in more compact form using the second-derivative operators defined as

$$\delta_{\xi}^2(\)_{i,j,k} = [(\)_{i+1,j,k} - 2(\)_{i,j,k} + (\)_{i-1,j,k}]/\Delta\xi^2$$

$$\delta_{\eta}^2(\)_{i,j,k} = [(\)_{i,j+1,k} - 2(\)_{i,j,k} + (\)_{i,j-1,k}]/\Delta\eta^2$$

$$\delta_{\zeta}^2(\)_{i,j,k} = [(\)_{i,j,k+1} - 2(\)_{i,j,k} + (\)_{i,j,k-1}]/\Delta\zeta^2$$

The following steps are applied to obtain a finite-difference representation of Equation 3.12 about the node $(i+1,j,k)$.

First, the source term, S_{p^+} , is formulated, using the available results of the space-marching solution. The derivatives of the pressure gradients are expressed as central-differences, using estimates of the pressure gradients surrounding the node located at $(i+1,j,k)$. These terms are

$$\frac{\partial}{\partial \xi}(p_{\xi}) = [(p_{\xi})_{i+1+1/2,j,k} - (p_{\xi})_{i+1-1/2,j,k}]/\Delta\xi \quad (4.12a)$$

$$\frac{\partial}{\partial \eta}(p_{\eta}) = [(p_{\eta})_{i+1,j+1/2,k} - (p_{\eta})_{i+1,j-1/2,k}]/\Delta\eta \quad (4.12b)$$

$$\frac{\partial}{\partial \zeta}(p_{\zeta}) = [(p_{\zeta})_{i+1,j,k+1/2} - (p_{\zeta})_{i+1,j,k-1/2}]/\Delta\zeta \quad (4.12c)$$

The pressure gradients at the half-step locations are then matched to the pressure gradients used in the solution of the governing equations. It is important to recognize that the streamwise pressure gradient is forward differenced, so the pressure gradient at $(i+1+1/2,j,k)$ is the gradient used in solving the momentum equations at the node $(i+1,j,k)$. The streamwise pressure gradient uses assumed pressure values in combination with the newly computed pressure. Thus, using Equation 4.1a gives

$$(p_t)_{i+1+1/2,j,k} = [\bar{p}_{i+2,j,k}^n - p_{i+1,j,k}^n] / \Delta \xi \quad (4.13a)$$

and

$$(p_t)_{i+1-1/2,j,k} = [\bar{p}_{i+1,j,k}^n - p_{i,j,k}^n] / \Delta \xi \quad (4.13b)$$

The transverse pressure gradients in the momentum equations are expressed in a fully implicit manner, neglecting the second-order corrections, so any measure of the resulting pressure profile is sufficient. It is easiest to use simple, two-point, central-differences. Thus

$$(p_\eta)_{i+1,j+1/2,k} = [p_{i+1,j+1,k}^n - p_{i+1,j,k}^n] / \Delta \eta \quad (4.14a)$$

$$(p_\eta)_{i+1,j-1/2,k} = [p_{i+1,j,k}^n - p_{i+1,j-1,k}^n] / \Delta \eta \quad (4.14b)$$

$$(p_t)_{i+1,j,k+1/2} = [p_{i+1,j,k+1}^n - p_{i+1,j,k}^n] / \Delta \xi \quad (4.14c)$$

$$(p_t)_{i+1,j,k-1/2} = [p_{i+1,j,k}^n - p_{i+1,j,k-1}^n] / \Delta \xi \quad (4.14d)$$

Evaluating Equations 4.12b and 4.12c using Equations 4.14 yields the standard central-difference operators for the second-derivatives in the transverse directions about the location $(i+1,j,k)$.

Next, the second-derivatives of the assumed pressure field are evaluated by central-difference formulas at the locations matching those used above in Equations 4.12–4.14. The resulting expressions are the standard second-derivative operators about the node $(i+1,j,k)$ in the form:

$$\bar{p}_{\xi\xi} = [\bar{p}_{i+2,j,k} - 2\bar{p}_{i+1,j,k} + \bar{p}_{i,j,k}] / \Delta \xi^2 = \delta_\xi^2(\bar{p})_{i+1,j,k}^n \quad (4.15a)$$

$$\bar{p}_{\eta\eta} = [\bar{p}_{i+1,j+1,k} - 2\bar{p}_{i+1,j,k} + \bar{p}_{i+1,j-1,k}] / \Delta \eta^2 = \delta_\eta^2(\bar{p})_{i+1,j,k}^n \quad (4.15b)$$

$$\bar{p}_{\xi\xi} = [\bar{p}_{i+1,j,k+1} - 2\bar{p}_{i+1,j,k} + \bar{p}_{i+1,j,k-1}] / \Delta \xi^2 = \delta_\xi^2(\bar{p})_{i+1,j,k}^n \quad (4.15c)$$

And finally, the second-derivatives of the estimated pressure correction, p^* , are determined by the same operators given in Equation 4.15. The completed finite-difference form of Equation 3.12 is then

$$A \delta_\xi^2(p^*)_{i+1,j,k} + B \delta_\eta^2(p^*)_{i+1,j,k} + \Gamma \delta_\xi^2(p^*)_{i+1,j,k} = S_{p^*} \quad (4.16)$$

where

$$S_{p^*} = \frac{A}{\Delta\xi^2} (\bar{p}_{i+2}^n - p_{i+1}^n - \bar{p}_{i+1}^n + p_i^n) + B \delta_7^2(p^n)_{i+1,j,k} + \Gamma \delta_c^2(p^n)_{i+1,j,k} \\ - A \delta_c^2(\bar{p}^n)_{i+1,j,k} - B \delta_7^2(\bar{p}^n)_{i+1,j,k} - \Gamma \delta_c^2(\bar{p}^n)_{i+1,j,k}$$

The source term can be simplified in terms of the error in the pressure field, ϵ , defined in Chapter III.

This form is

$$S_{p^*} = A(\epsilon_{i,j,k} - \epsilon_{i+1,j,k})/\Delta\xi^2 + B \delta_7^2(\epsilon)_{i+1,j,k} + \Gamma \delta_c^2(\epsilon)_{i+1,j,k}$$

2. Parabolizing assumptions

Solving the elliptic equation, Equation 4.16, would require substantial computational effort. A parabolic equation is obtained by assuming that the errors in the pressure field at a given node do not directly influence the pressure correction at downstream nodes. This is not a completely arbitrary assumption, but corresponds to the finite-difference form of the governing equations and the observation that the equations at a given node do not directly depend on pressure terms upstream. The treatment of the streamwise pressure correction term is similar to the one-dimensional pressure correction procedure of Barnett and Davis [68].

To incorporate this assumption into Equation 4.16, it is assumed that the upstream pressure gradient term will be corrected to the current solution value. Using Equation 4.1a to apply this condition for the streamwise pressure gradient at the upstream node (i,j,k) gives

$$[\bar{p}_{i+1,j,k}^{n+1} - \bar{p}_{i,j,k}^{n+1}]/\Delta\xi \approx [\bar{p}_{i+1,j,k}^n - p_{i,j,k}^n]/\Delta\xi \quad (4.17a)$$

In terms of p^* and ϵ , the assumption is simply

$$[p_{i+1,j,k}^* - p_{i,j,k}^*]/\Delta\xi \approx -\epsilon_{i,j,k}/\Delta\xi \quad (4.17b)$$

Using this approximation in Equation 4.16 eliminates the pressure correction and error terms at the upstream node (i,j,k) and leaves

$$A(p_{i+2,j,k}^* - p_{i+1,j,k}^*) + B \delta_7^2(p^*)_{i+1,j,k} + \Gamma \delta_c^2(p^*)_{i+1,j,k} = S'_{p^*} \quad (4.18)$$

where

$$S'_{p^*} = -A \epsilon_{i+1,j,k}/\Delta\xi^2 + B \delta_7^2(\epsilon)_{i+1,j,k} + \Gamma \delta_c^2(\epsilon)_{i+1,j,k}$$

Equation 4.18 is parabolic and is the finite-difference form of Equation 3.13 introduced in Chapter III.

Since there is no correction to the specified pressure at the far downstream boundary, Equation 4.18 can be integrated upstream. For two-dimensional problems the simple implicit method

employed uses a tridiagonal line solver at each station. The three-dimensional method uses an ADI procedure. This single backsweep scheme rapidly transmits pressure changes in the upstream direction. The computational effort to execute the procedure is insignificant compared to the space-marching solution of the coupled governing equations. To reduce the number of arbitrary parameters in the pressure correction algorithm the B and Γ constants are taken to be equal. A single "diffusion" parameter is defined as

$$\alpha = \frac{B}{A} = \frac{\Gamma}{A} \quad (4.19)$$

Underrelaxation of the predicted pressure correction is used, so the actual pressure correction applied is

$$\bar{p}^{n+1}_{i,j,k} = \bar{p}^n_{i,j,k} + \omega p^+_{i,j,k} \quad (4.20)$$

After the pressure correction procedure is executed, a final adjustment is made to the pressure field before starting the next global iteration. The backsweep procedure corrects the pressure based on a fixed downstream value. However, to allow direct specification of the inlet mass flowrate, density, and Reynolds number as input to the computer program, it is convenient to fix the magnitude of the inlet pressure. The magnitude of the pressure correction at the inlet is therefore subtracted from the pressure throughout the flow field. This establishes the desired value for the pressure at the inlet and develops the proper pressure differential between the inlet and exit pressures.

V. RESULTS AND DISCUSSION

Computed results were obtained for six steady, incompressible, laminar flow, test cases. Although the present method is applicable to compressible subsonic flow, only low Mach number, effectively incompressible flow cases were rigorously evaluated due to the limited availability of comparable results for compressible flow. Results obtained with the incompressible model are indistinguishable from the compressible results obtained at low Mach numbers. The geometry for each case is shown in Figure 5.1. The flow conditions as well as several grid and iteration control parameters are summarized in Table 5-1.

The test cases were selected to present different aspects of elliptic flow problems. First, the entrance flow in a two-dimensional channel induces complex pressure and velocity fields as the flow develops. Next, the sudden expansion geometries produce recirculation regions. The sudden contraction geometry creates a stagnation region, and the large obstruction to the flow presents a severe test of the robustness of the space-marching procedure. A nonorthogonal grid was used to improve the resolution of the flow near the step corner. The crossflow over a cylinder presents a very general flow field, including stagnation points, separated flow, and large streamline curvature, and introduces freestream boundary conditions. This case was also computed on a nonorthogonal grid. Finally, the entrance flow in a square cross-section duct is similar to the two-dimensional entrance case, but requires three-dimensional geometry.

Example results from the computed flow simulations for these various geometries are presented in the following sections. Estimates of the computational requirements for executing the algorithm and the results of optimizing the convergence rate of the space-marching algorithm with global pressure correction are presented in the final section of the chapter.

A. Developing Flow in a Two-Dimensional Channel

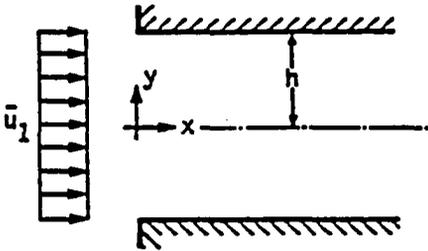
The entrance flow in a channel with an arbitrarily imposed, uniform, irrotational, inlet profile is labeled as Case 1. Solutions were calculated on the channel upper half-plane using a rectangular grid with lines packed near to the inlet plane and the wall as shown in Figure 5.2. The grid stretching procedure is described in Appendix E, and values of the stretching parameters are given in Table 13.1. The channel length was varied with the Reynolds number and was selected to ensure fully-developed flow at the exit plane. The channel half-width and the average bulk inlet velocity are the characteristic values used to establish the Reynolds number (the Reynolds number based on the

Table 5.1. Summary of test case parameters

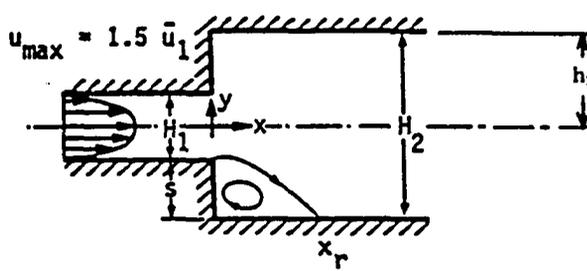
Case	Inlet Profile	Re	M	L	$\frac{x_1}{L}$	$\frac{x_2}{L}$	Grid Type	Grid Size	ϵ	α	ω	ITN	
1	uniform $u = 1$ $v = 0$	7500	0.01	h	0.0	3000	*	41x21	10^{-4}	40	0.7	7	
		75			0.0	30	*	21x11		20		10	
		75			0.0	30	*	41x21		20		17	
		75			0.0	30	*	81x41		20		43	
		10			0.0	4	*	41x21		5		60	
		0.5			0.0	2	*	41x21		5		183	
2a	parabolic u $v = 0$	280	0.00	h_2	-10.0	40	uniform	61x37	10^{-6}	80	0.5	19	
		56			-2.0	8		61x37				23	
2b	parabolic u $v = 0$	1000	0.00	h_2	-1.0	39	uniform	81x41	10^{-6}	80	0.5	28	
		320			-1.0	19		81x41				80	26
		100			-1.0	7		81x41				80	34
		50			-1.0	4		81x41				80	51
		10			-1.0	1.5		81x41				80	389
3	parabolic u $v = 0$	687	0.05	h_2	-2.0	16	uniform	46x31	10^{-6}	80	0.5	29	
		219			-2.0	8		51x31				80	27
4	parabolic u $v = 0$	400	0.00	h_1	-1.5	2.5	*	81x41	10^{-3}	200	0.05	968	
		300			-1.5	2.5		81x41				200	---
		200			-1.5	2.5		81x41				200	---
		100			-1.5	2.5		81x41				200	928
5	uniform $u = 1$ $v = 0$	40	0.05	D	-20	20	*	41x21	10^{-4}	10	0.1	263	
								81x41		80		486	
6	uniform $u = 1$ $v = 0$	100	0.01	h	0.0	40	*	41x21x21	10^{-4}	80	0.5	40	
		25			0.0	10		*		21x11x11		80	23
		5			0.0	4		*		41x11x11		80	128

* Non-uniform grid (refer to Table 13.1).

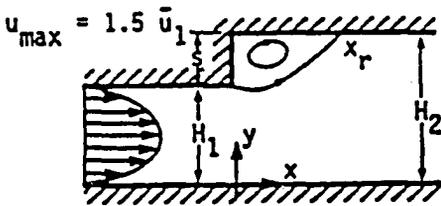
CASE 1 $Re = \frac{\bar{u}_1 h}{\nu_1}$



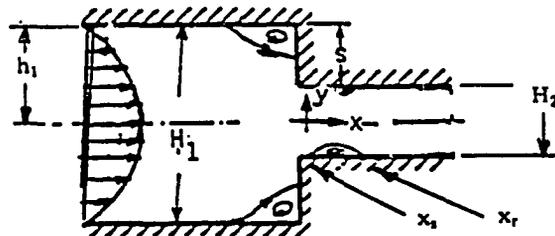
CASE 2 $Re = \frac{\bar{u}_1 h_2}{\nu_1} \frac{H_2}{H_1} = 3, 2$



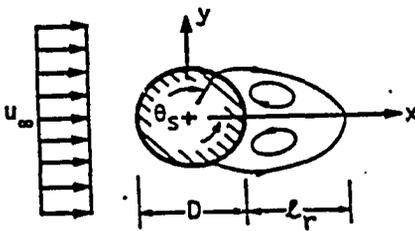
CASE 3 $Re = \frac{\bar{u}_1 H_2}{\nu_1} \frac{H_2}{H_1} = \frac{3}{2}$



CASE 4 $Re = \frac{\bar{u}_1 h_1}{\nu_1} \frac{H_2}{H_1} = \frac{1}{2}$



CASE 5 $Re = \frac{u_\infty D}{\nu_\infty}$



CASE 6 $Re = \frac{\bar{u}_1 h}{\nu_1}$

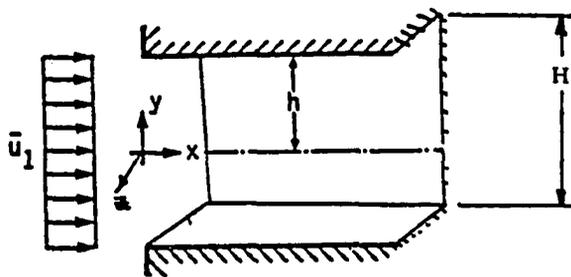


Figure 5.1. Test case flow geometries

hydraulic diameter is four times greater). Low inlet Mach numbers were used to approximate incompressible flow.

Computed results were obtained for a range of Reynolds numbers. For large Reynolds numbers the problem is effectively parabolic, but is elliptic in both the pressure and velocity at low Reynolds numbers. The predicted centerline velocity distributions for nearly incompressible flow are shown in Figure 5.3 along with the numerical results of Bodoia and Osterle [69], McDonald et al. [70] and Morihara and Cheng [71]. Good agreement was achieved for all Reynolds numbers, indicating that the relative effects of the viscous, streamwise convection and streamwise pressure gradient terms were well represented.

Computations employing the PPNS equations were performed for comparison, and the predicted centerline velocity distributions are shown in Figure 5.4. The PPNS solutions compare well with those of Chilukuri and Pletcher [51]. The observable differences between the NS and PPNS solutions confirm their observation that the streamwise viscous terms have an increasingly significant influence on the flow development for Reynolds numbers on the order of ten or less.

The effect of inlet Mach number on the centerline velocity distribution is shown in Figure 5.5. The compressible flow develops into a nearly parallel, shear flow. However, the pressure drop due to friction causes the density to decrease, and the flow never achieves a fully-developed state. In fact the solution obtained at an inlet Mach number of 0.3 was rapidly approaching the Fanno flow choking condition. Comparison of the solutions clearly demonstrates that as the Mach number approaches zero the results of the compressible formulation converge to the truly incompressible solution. No computational difficulties were encountered with the incompressible or the compressible formulations at inlet Mach numbers as low as 0.01, since pressure, instead of density, was used as a primary variable. At very low Mach numbers the method used to normalize the variables could potentially lead to a loss of accuracy. Notice that the nondimensional pressure takes on a numerically large value as the inlet Mach number approaches zero. Only pressure differences, not the magnitude of the pressure, are significant in an incompressible flow, and the computational accuracy of the pressure differences is degraded by a large number of insignificant leading digits. A simple remedy is to reformulate the dimensionless pressure.

The computed solutions were checked for grid dependency by successively increasing the number of node points and reducing the mesh spacings until an invariant solution was obtained. The predicted centerline velocity distributions at a Reynolds number of 75 for several meshes are shown in

Figure 5.6 and the predicted axial velocity profiles are shown in Figure 5.7. The velocity profiles display the "overshoots" characteristic of this flow problem and agree well with the computed results of McDonald et al. [70]. The results obtained with the 41 by 21 mesh are reasonably accurate in most respects. The largest source of error appears to be due to matching the singular boundary conditions at the inlet corner, which specify zero velocity at the wall as well as a uniform inlet velocity. The velocity at the corner node was set equal to zero, causing the inlet flow to be essentially blocked for one-half grid spacing away from the wall. Thus, for unit inlet mass flowrate and unit inlet bulk velocity, the prescribed inlet velocity is slightly greater than one, resulting in a slight alteration of the development of the centerline velocity. Specifying a non-zero "slip" velocity for the corner node is not advised since this forces an abnormally large transverse velocity at the first node inside the channel in order to satisfy continuity. Alteration of the grid structure to allow for a half-grid step at the wall alleviates this problem and is recommended specifically for the solution of abrupt entrance problems. However, in general this treatment of wall boundaries is somewhat awkward and was therefore not implemented.

Convergence histories for example calculations are shown in Figure 5.8 and the number of global iterations is listed in Table 5.1. The most rapid convergence was observed for the higher Reynolds number cases. At the largest Reynolds number the velocity field is essentially correct on the first sweep, and only a few global iterations are needed to develop the pressure field. For the low Reynolds number cases the convergence rate diminishes as the convergence limit is approached.

One possible cause of the slower convergence at low Reynolds numbers is slow convergence of the streamwise diffusion terms, that are larger in magnitude at low Reynolds numbers. These terms depend on the values of downstream velocities that are lagged from iteration to iteration, with no global correction to the velocity field to accelerate convergence. The PPNS calculations converged more rapidly than the NS calculations, and the point of departure of the convergence histories seems to correspond to the expected order of magnitude of the streamwise viscous terms. Bentson and Vradis [61] have also computed results for this case and used a similar global pressure correction procedure. They observed a similar difference in the convergence rates of NS and PPNS calculations.

A second potential cause of the slower convergence at low Reynolds numbers is the singularity at the inlet corner (see Van Dyke [72]). For this computational test case the inlet flow profile is arbitrarily specified to be uniform and irrotational. However, this profile is inconsistent with real flow.

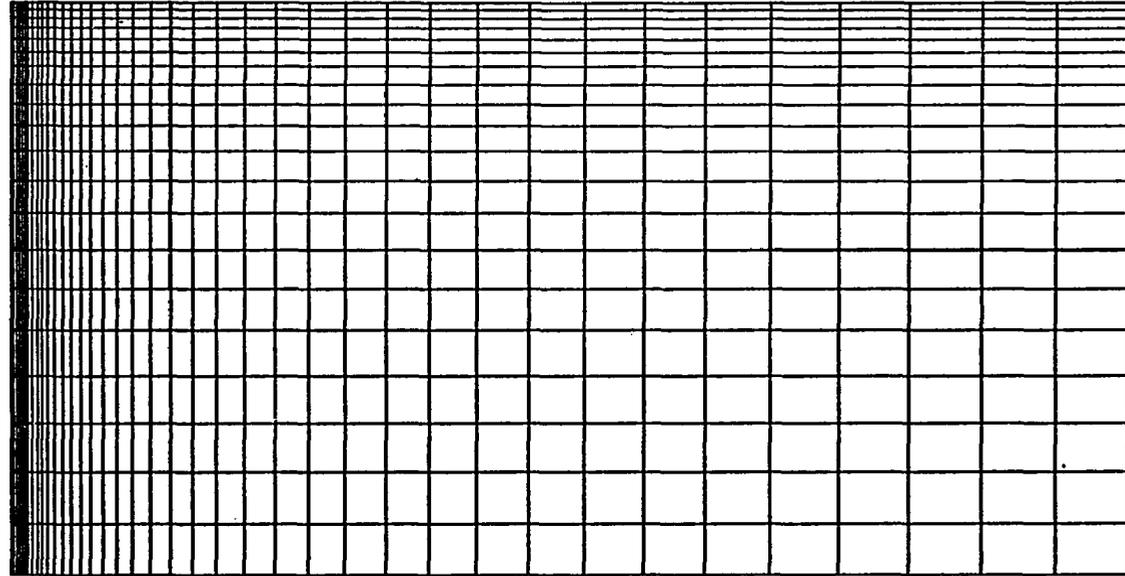


Figure 5.2. Clustered grid layout for Case 1 for $Re = 10$

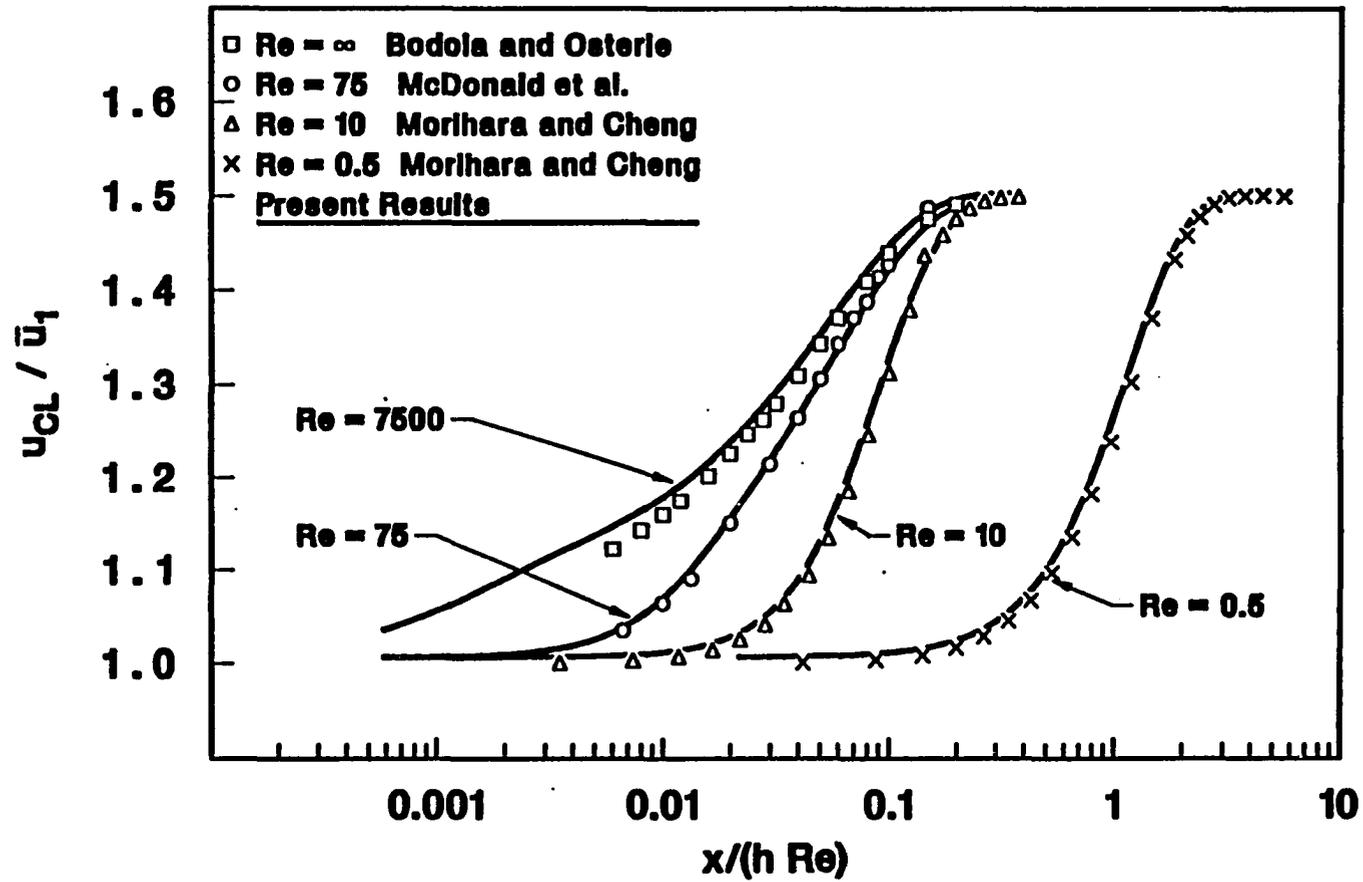


Figure 5.3. Predicted centerline velocity distribution for Case 1

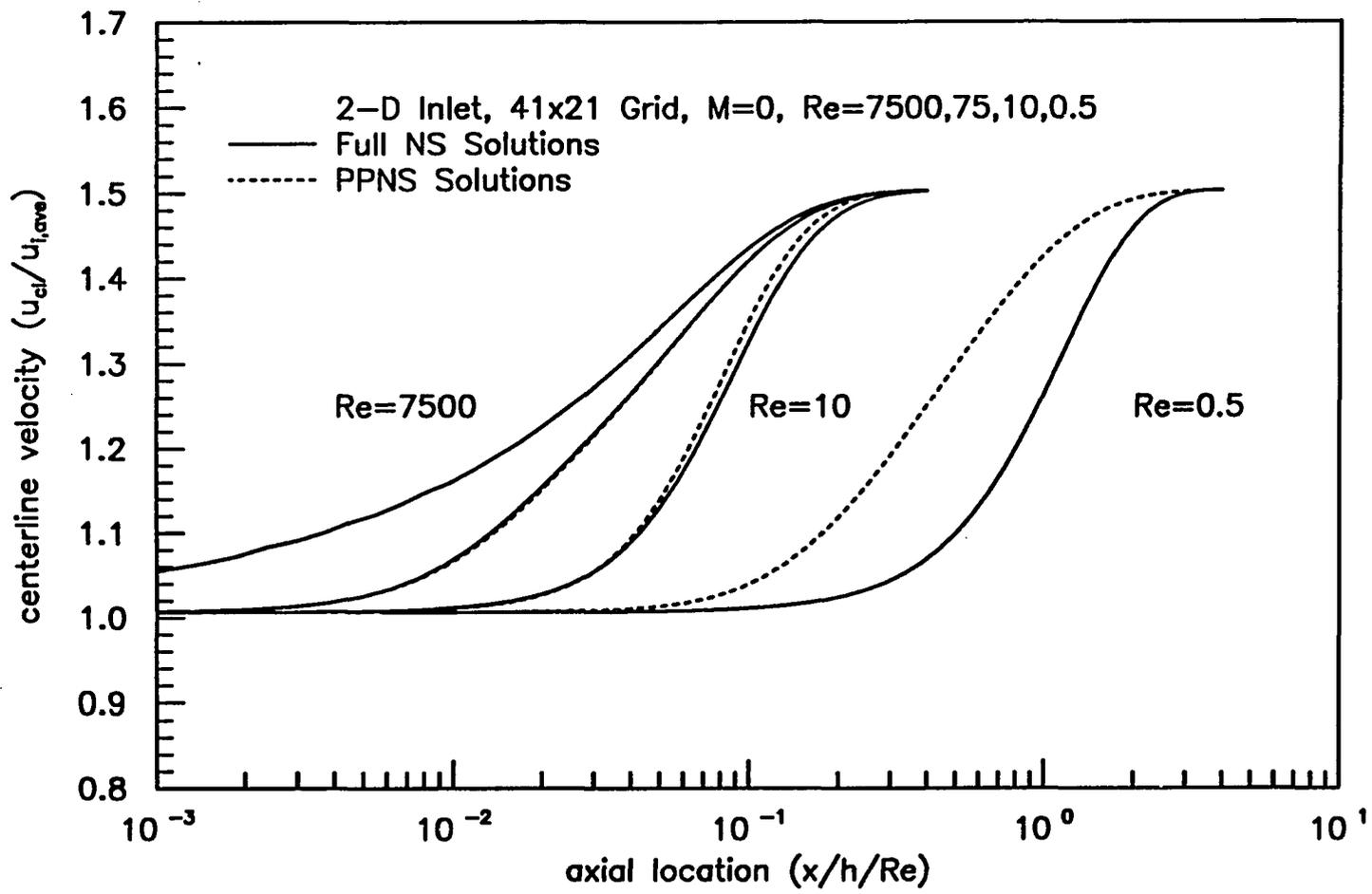


Figure 5.4. Comparison of NS and PPNS solutions for Case 1

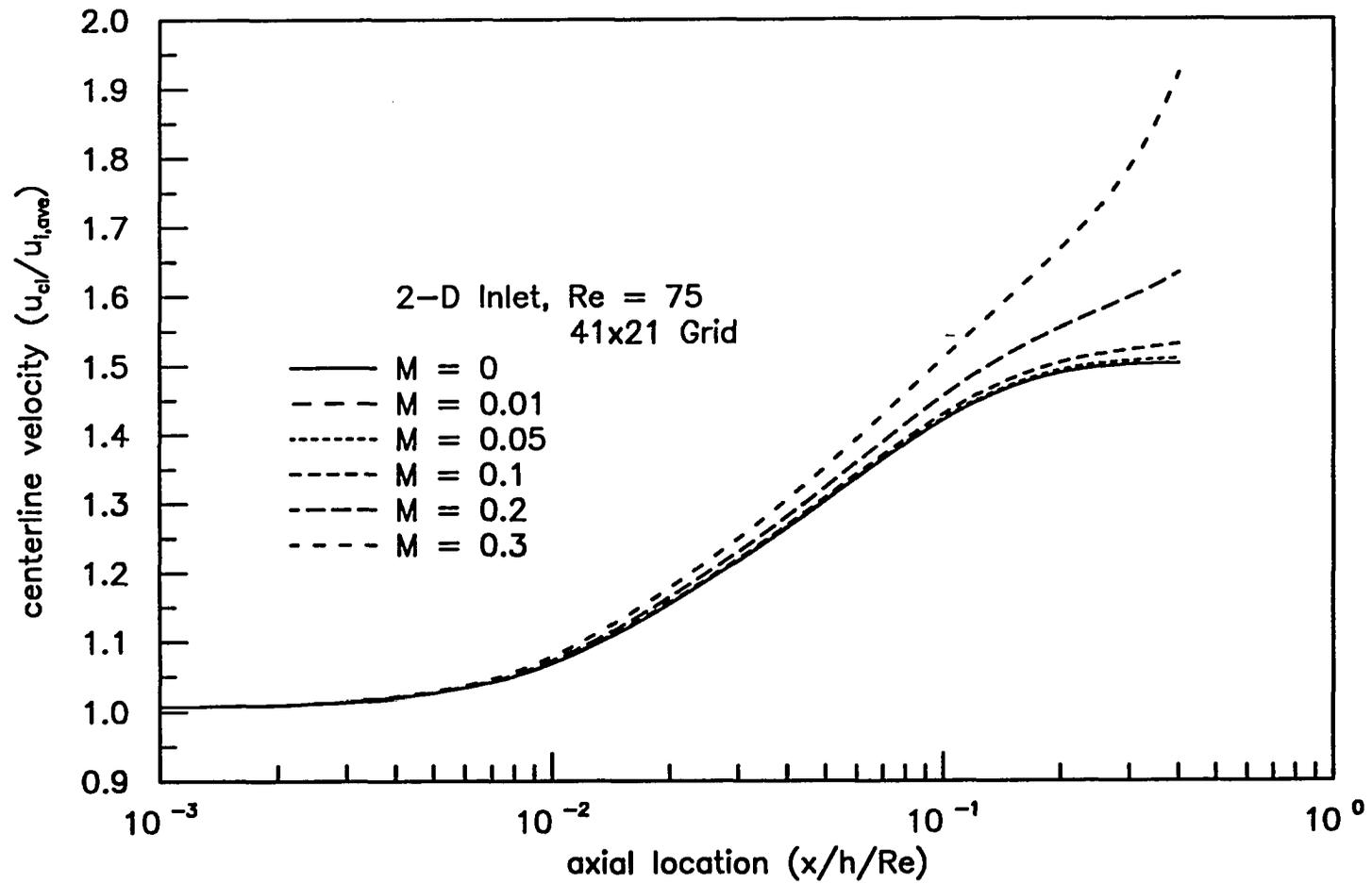


Figure 5.5. Compressible and incompressible flow predictions for Case 1

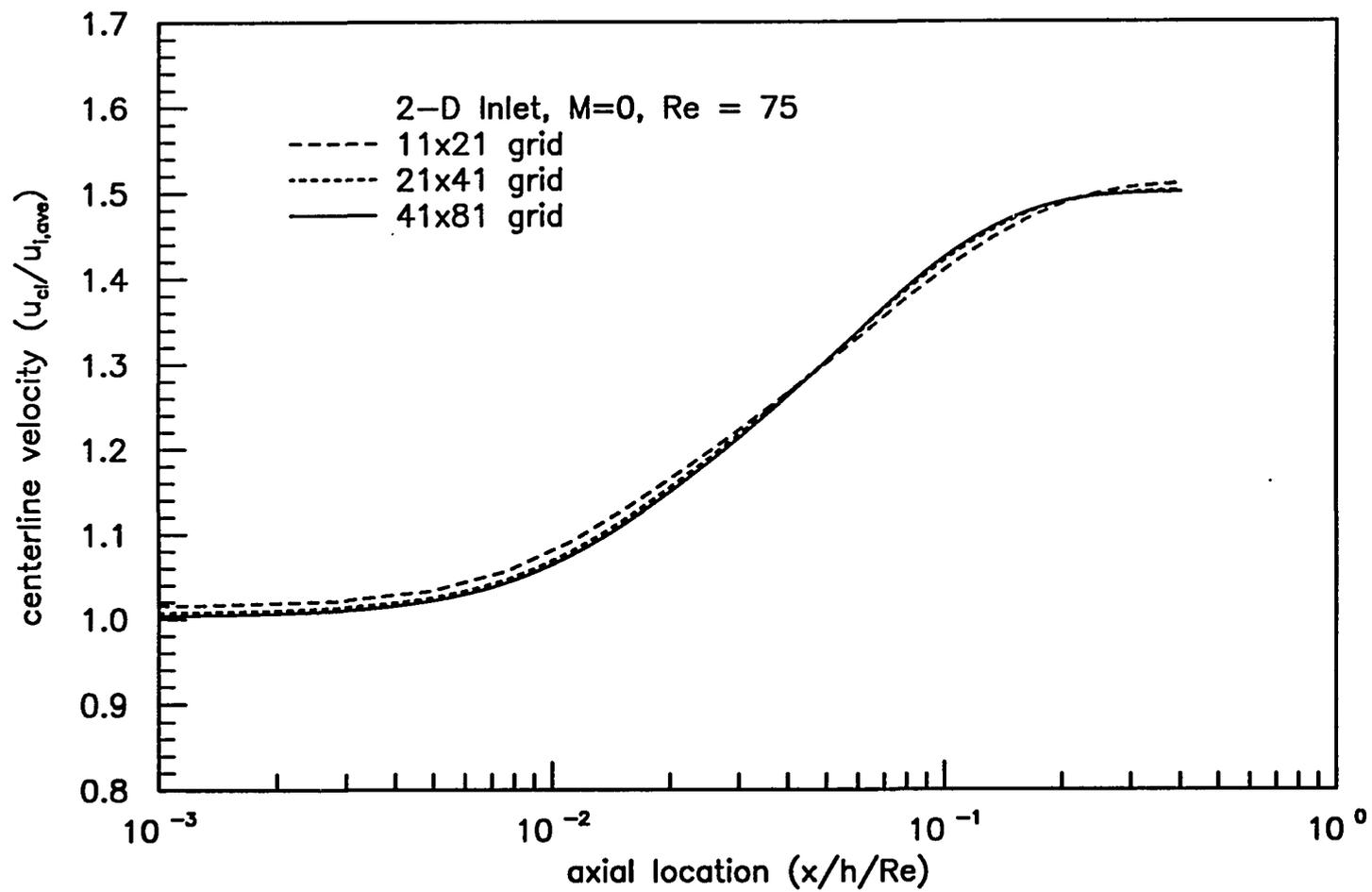


Figure 5.6. Convergence of centerline velocity distribution for Case 1

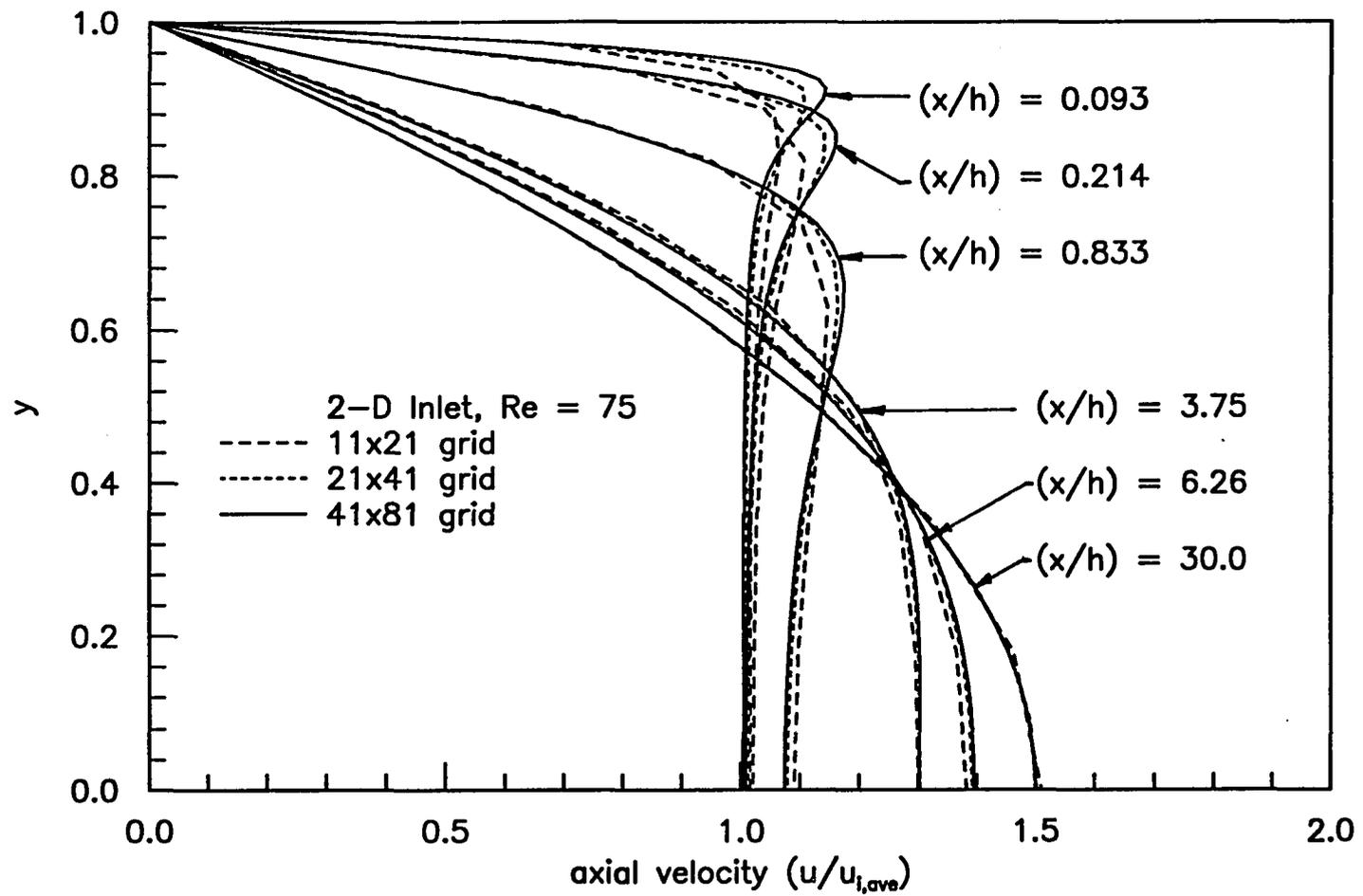


Figure 5.7. Computed axial velocity profiles for Case 1 at $Re = 75$

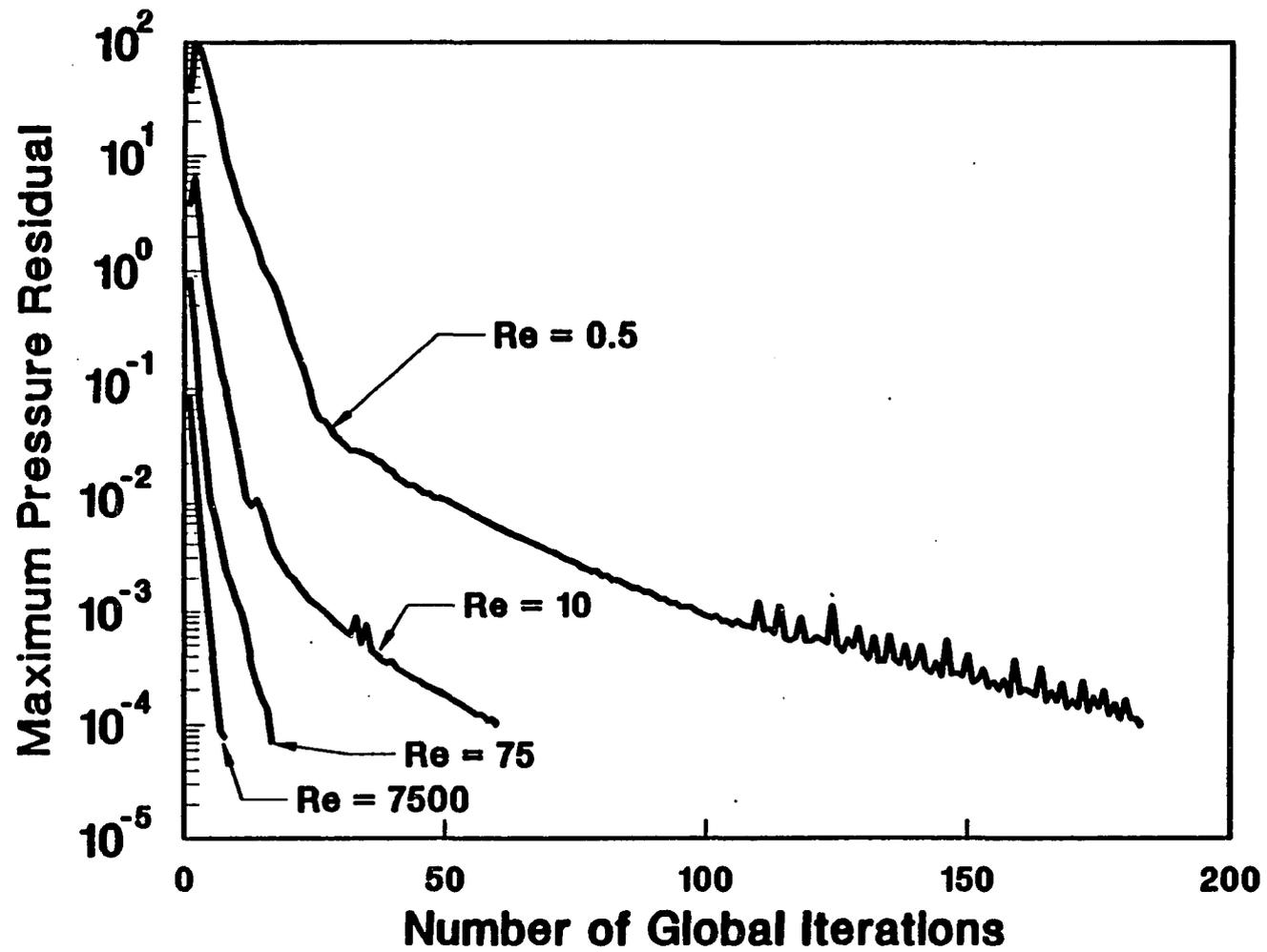


Figure 5.8. Convergence histories for Case 1 solutions

As a result, an artificially high pressure was calculated at the corner which would normally propagate upstream and alter the inlet flow. For the low Reynolds number cases the axial grid spacing is much smaller near the inlet and the computed pressure spike is larger. Resolution of this large perturbation to the pressure may have also slowed the convergence rate.

B. Channel Flow with a Symmetric Sudden Expansion

The symmetric sudden expansion flow in channels with 3/1 and 2/1 expansion ratios are respectively labeled as Case 2a and Case 2b. Incompressible results are presented here, which are not significantly different from additional results calculated at low Mach number. Computations were performed on the channel upper half-plane with a series of uniform grids in order to verify a grid independent solution. The flow conditions and grids are described in Table 5.1 and Table 5.2. Fully-developed flow was assumed at the inlet plane located upstream of the expansion. The last marching station was placed far enough downstream of the reattachment point that the pressure was constant across the channel. The flow was not assumed to be fully-developed at the exit plane, but the streamwise viscous terms were assumed to be negligible. The channel Reynolds number is evaluated using the bulk inlet velocity and the channel half-width at the exit. Alternately, a Reynolds number based on the step height and the bulk inlet velocity is frequently used in the literature.

Experimental data by Durst et al. [73] is available for Case 2a at a Reynolds number of 56 ($Re_s = 37.3$), and there are also numerical results available for comparison at this Reynolds number. Similarly, the Case 2b flow has been computed by other researchers, particularly for a Reynolds number of 100 ($Re_s = 50$). Predictions of the length of the recirculation zone downstream of the expansion are given in Table 5.2. The fine grid solutions show an eddy length of 4.05 step heights for the 3/1 contraction, and 7.03 step heights for the 2/1 expansion at the given Reynolds numbers. By comparison, for the 3/1 expansion the flow visualization experiment by Durst et al. [73] indicates an eddy length of approximately four step heights, and the Navier-Stokes solution by Osswald et al. [74] on a fine grid predicts the length to be 3.83 times the step height. For the 2/1 expansion the boundary-layer solutions of Kwon et al. [37] predict the recirculation region extends 6.9 step heights downstream of the expansion. The present predictions of the eddy length are seen to be slightly, but not significantly, higher than values obtained by other investigators.

The computed centerline velocity distributions, axial velocity profiles and wall shear stress distributions are shown in Figures 5.9 through 5.14, along with some of the comparable results

available in the literature [37,73,75,76,77,78]. Close study of the results shows that at high Reynolds numbers the flow field streamwise development scales proportionally with the Reynolds number. Although a large recirculation zone is present, for large Reynolds numbers the strength of the vortex is weak enough that the flow is essentially parabolic in character, and the present NS predictions are in good agreement with the boundary-layer solutions of Kwon et al. [37]. However, at lower Reynolds numbers the transverse velocities are large and the intensity of the recirculating eddy begins to decay due to viscous stresses. This departure from the boundary-layer behavior is expected since the eddy does not vanish in the creeping flow regime at very low Reynolds numbers. Although the quantities most frequently used for comparison are the centerline velocity distribution and the length of the recirculation zone, it has been observed by Lewis and Pletcher [76] that these values are relatively insensitive to the details of the flow within the recirculating eddy. Even though the centerline velocity distributions show little difference, the wall shear stress distributions and the velocity profiles through the eddy show clear distinctions between the high Reynolds number boundary-layer solutions and the present Navier-Stokes predictions of the flow at Reynolds numbers of the order of 100 or less.

It should be noted that the maximum velocity within the eddy is roughly one-tenth that of the inlet bulk velocity and that the eddy size is only a fraction of the channel width. Thus, a Reynolds number based on the eddy characteristics would be roughly twenty to one hundred times smaller than the characteristic channel Reynolds number, and clearly all of the viscous stress terms should play a significant role in the dynamics of these very low Reynolds number eddy flows. Figure 5.15 shows the wall shear stress distributions for Case 2b at a Reynolds number of 50 predicted by solution of the NS and PPNS equations, compared with the boundary-layer equation solution of Kwon et al. [37]. The differences between the predictions are an indication of the relative significance of the streamwise and transverse viscous terms in the momentum equations governing the dynamics of the recirculating eddy. Recent "benchmark" quality solutions of the Navier-Stokes equations with the stream function/vorticity method by Napolitano and Cinnella [78] for Case 2b at a Reynolds number of 50 are in close agreement with the present findings. Although, for their calculations a parabolic profile at the plane of the sudden expansion was specified as the inlet condition.

For this case the convergence rate was not strongly dependent on the number of nodes, and rapid convergence was obtained on all runs with the exception of the very low Reynolds number solutions. The required number of global iterations are listed in Tables 5.1 and 5.2. Figures 5.16 and 5.17 show the convergence histories obtained with different computational grids, while Figure 5.18

shows the effect of the flow Reynolds number. The decline in the convergence rate at low Reynolds number can be traced to slower convergence of the streamwise viscous terms, as Figure 5.19 shows no change in the convergence rate for the solution of the PPNS equations. Further discussion of the convergence behavior is presented in the final section of this chapter.

Table 5-2. Results of grid refinement for incompressible channel flow with a symmetric sudden expansion

N ^a	Grid Dimensions		ISTP	JSTP	Results	
	IMAX	JMAX			X _r /s ^b	ITN ^c
Case 2a, Re = 56, Re _s = 37.3						
517	31	19	7	7	4.418	21
897	41	25	9	9	4.248	20
1381	51	31	11	11	4.147	20
1969	61	37	13	13	4.089	23
2476	71	40	15	14	4.045	29
Case 2b, Re = 100, Re _s = 50						
145	17	9	3	5	7.978	20
529	33	17	5	9	7.352	18
1153	49	25	7	13	7.170	22
2017	65	33	9	17	7.082	25
3121	81	41	11	21	7.030	34

^a Number of nodes.

^b Length to reattachment/step height.

^c Number of global iterations.

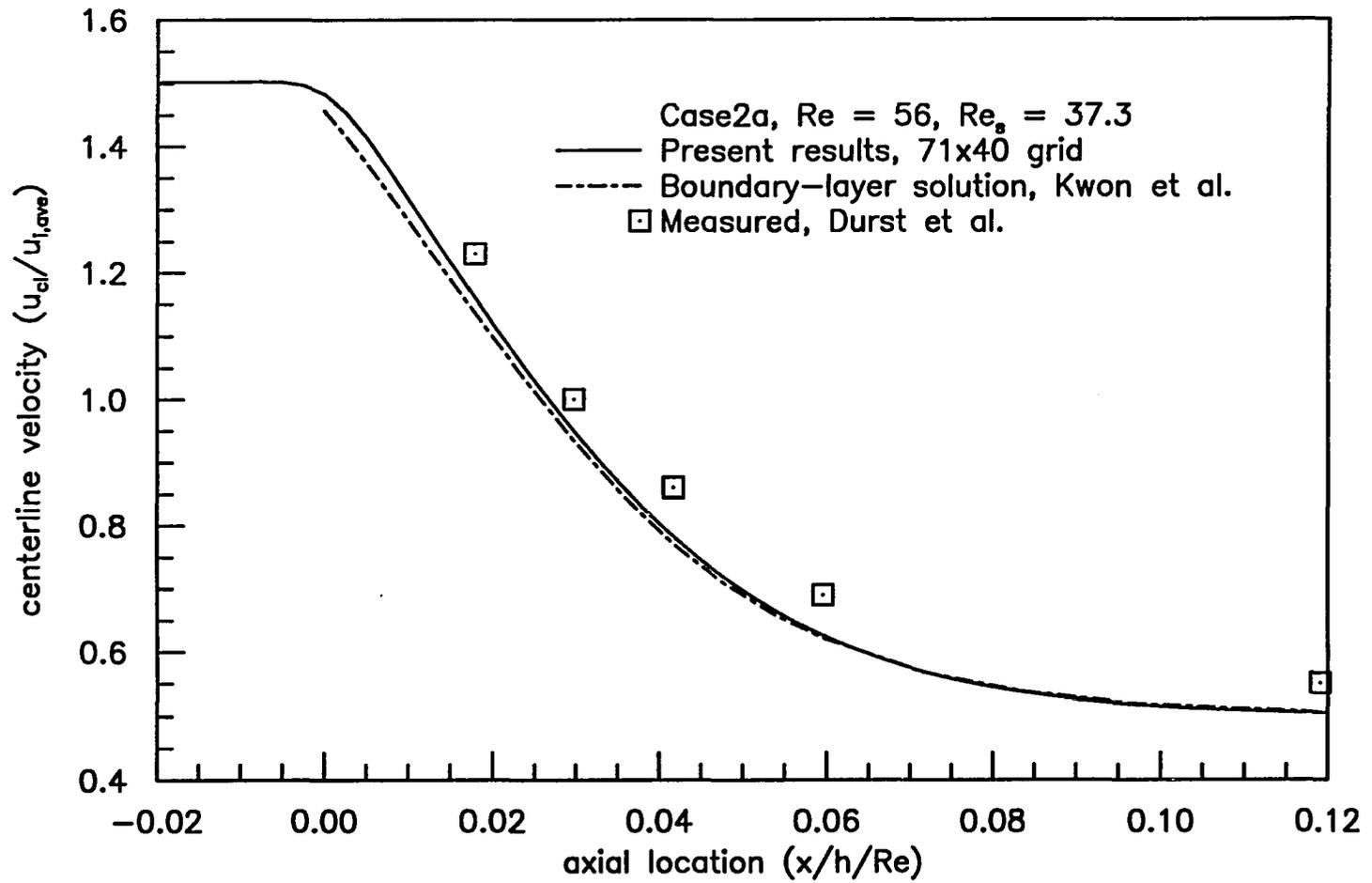


Figure 5.9. Predicted centerline velocity distribution for Case 2a

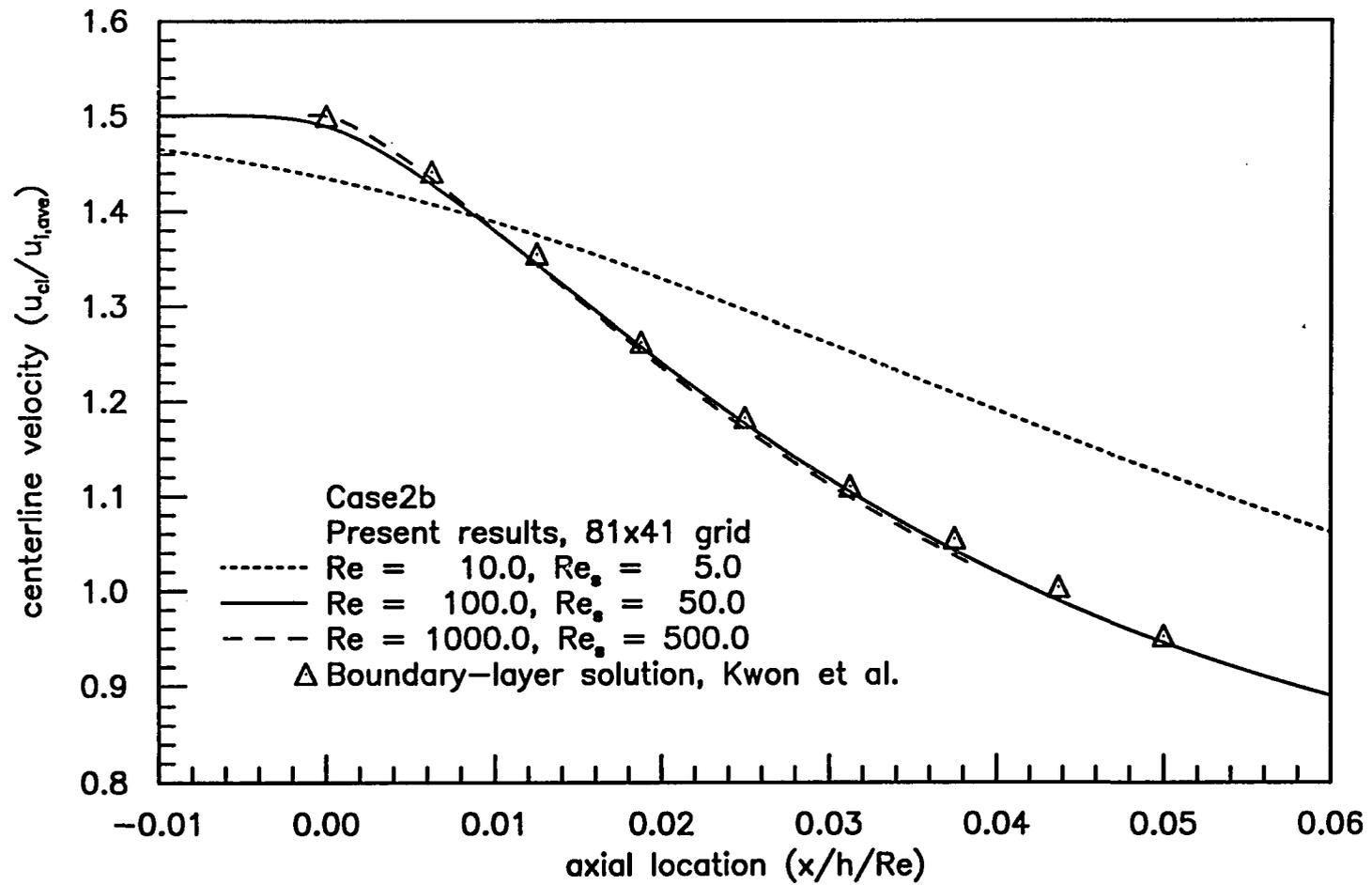


Figure 5.10. Predicted centerline velocity distribution for Case 2b

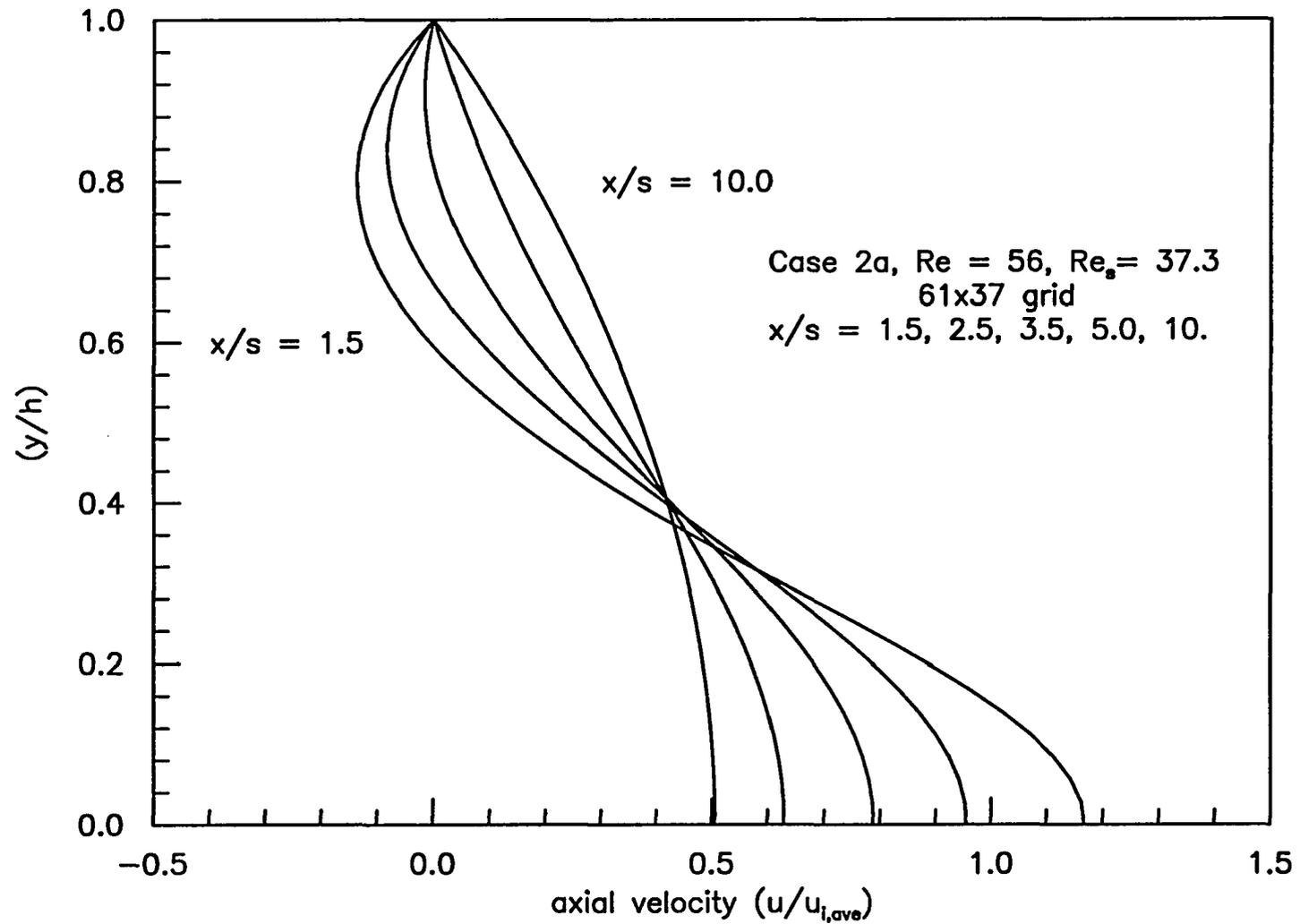


Figure 5.11. Computed axial velocity profiles for Case 2a at $Re = 56$

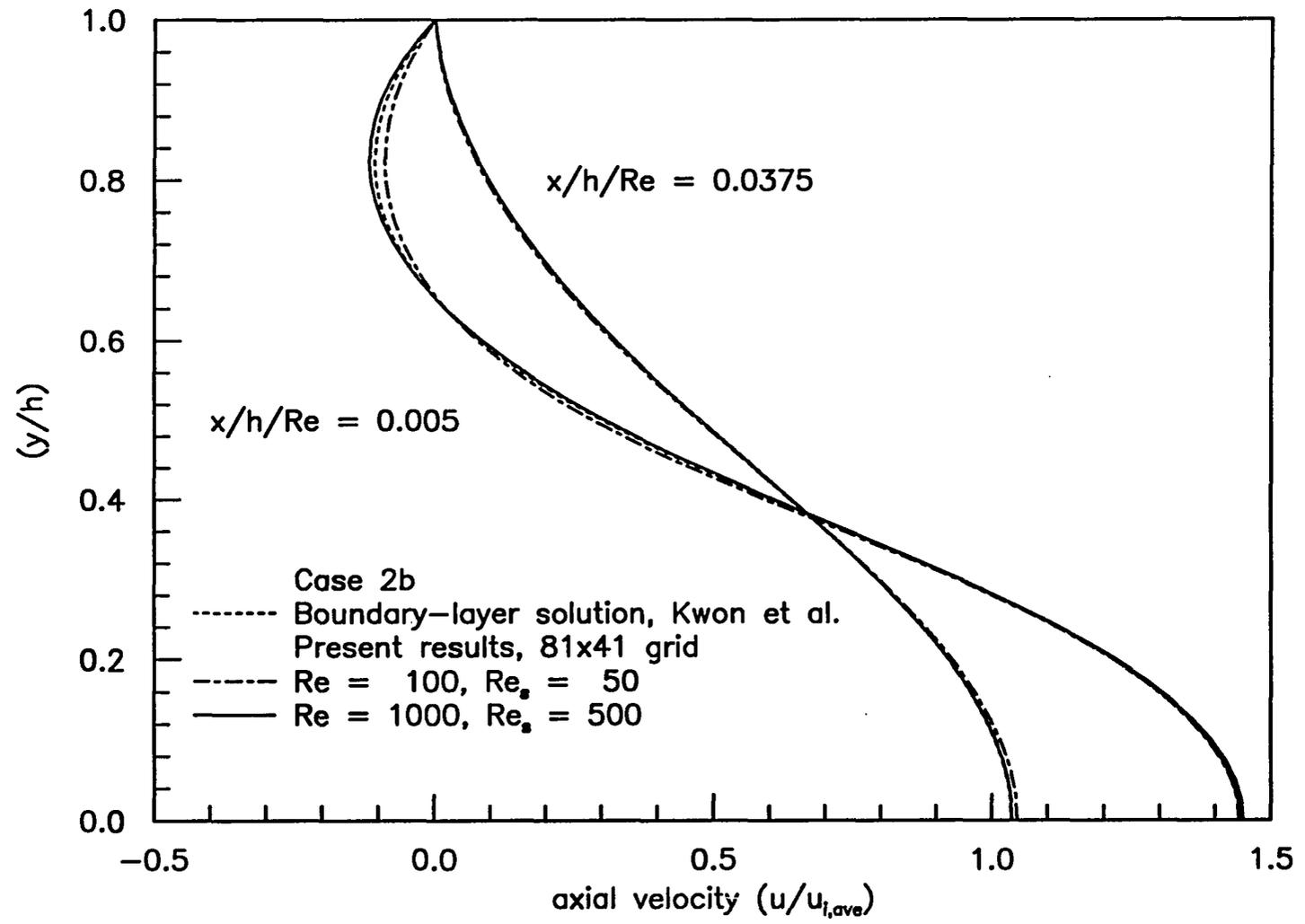


Figure 5.12. Computed axial velocity profiles for Case 2b

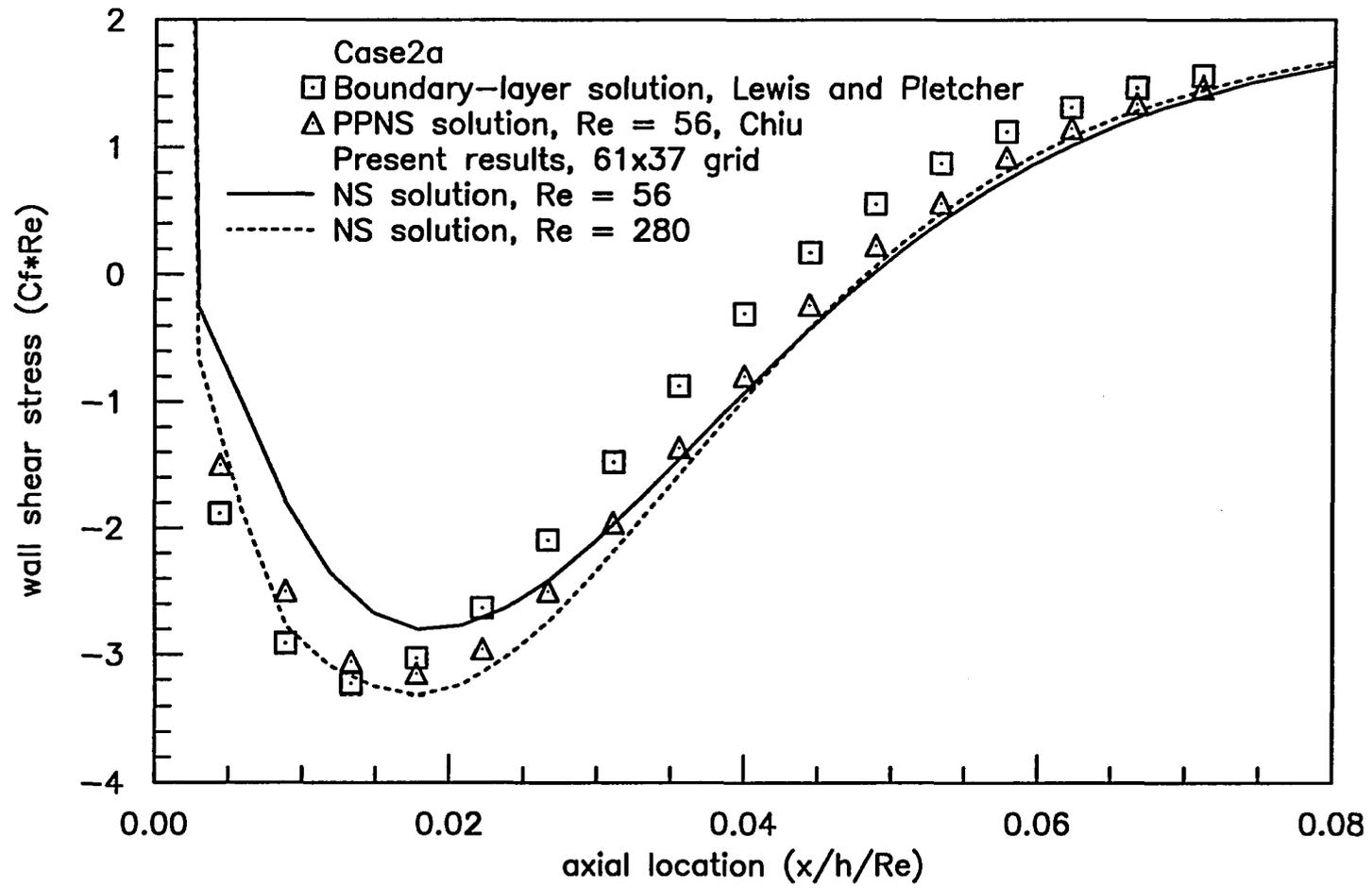


Figure 5.13. Predicted wall shear stress distribution for Case 2a

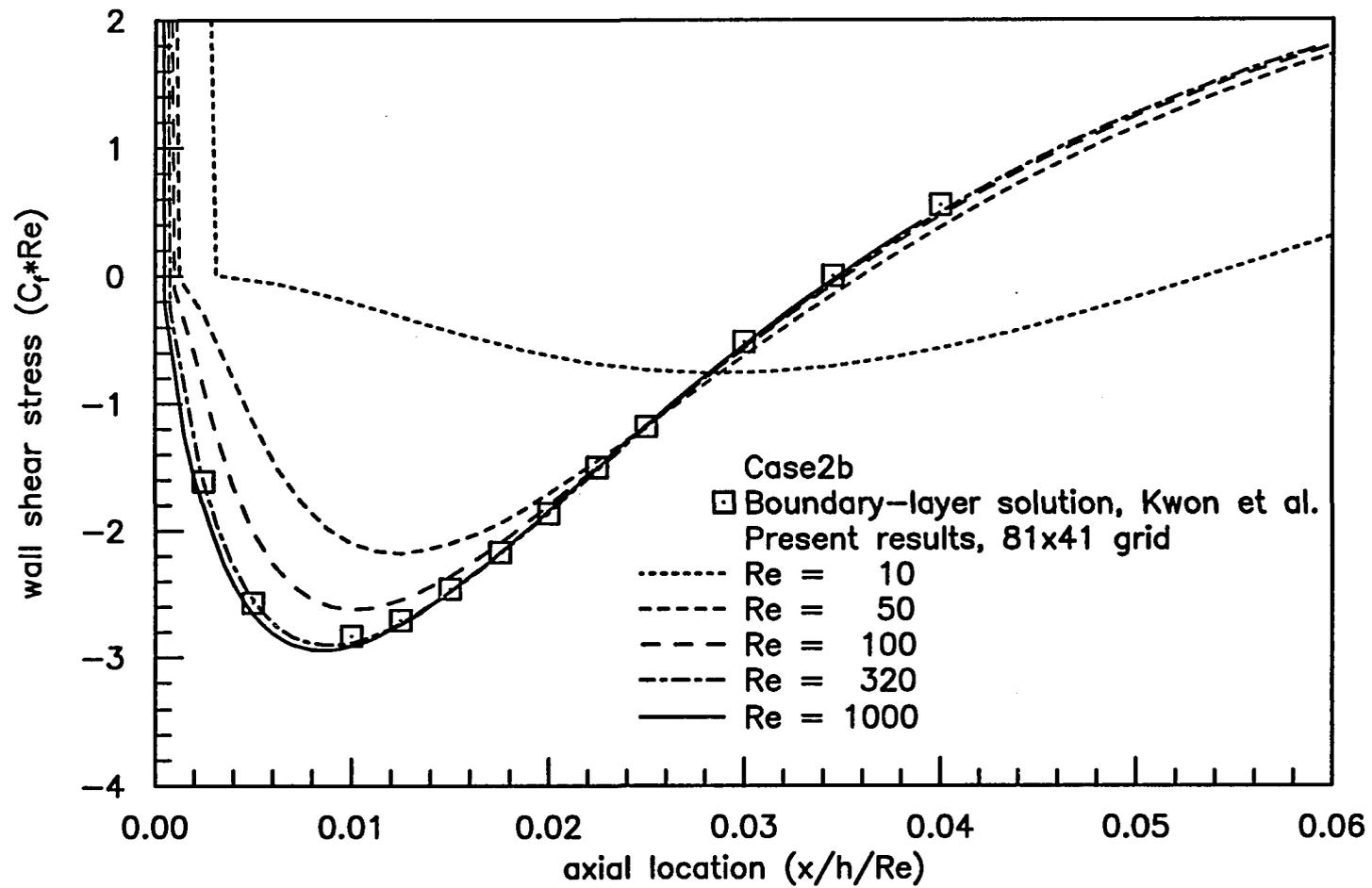


Figure 5.14. Predicted wall shear stress distribution for Case 2b

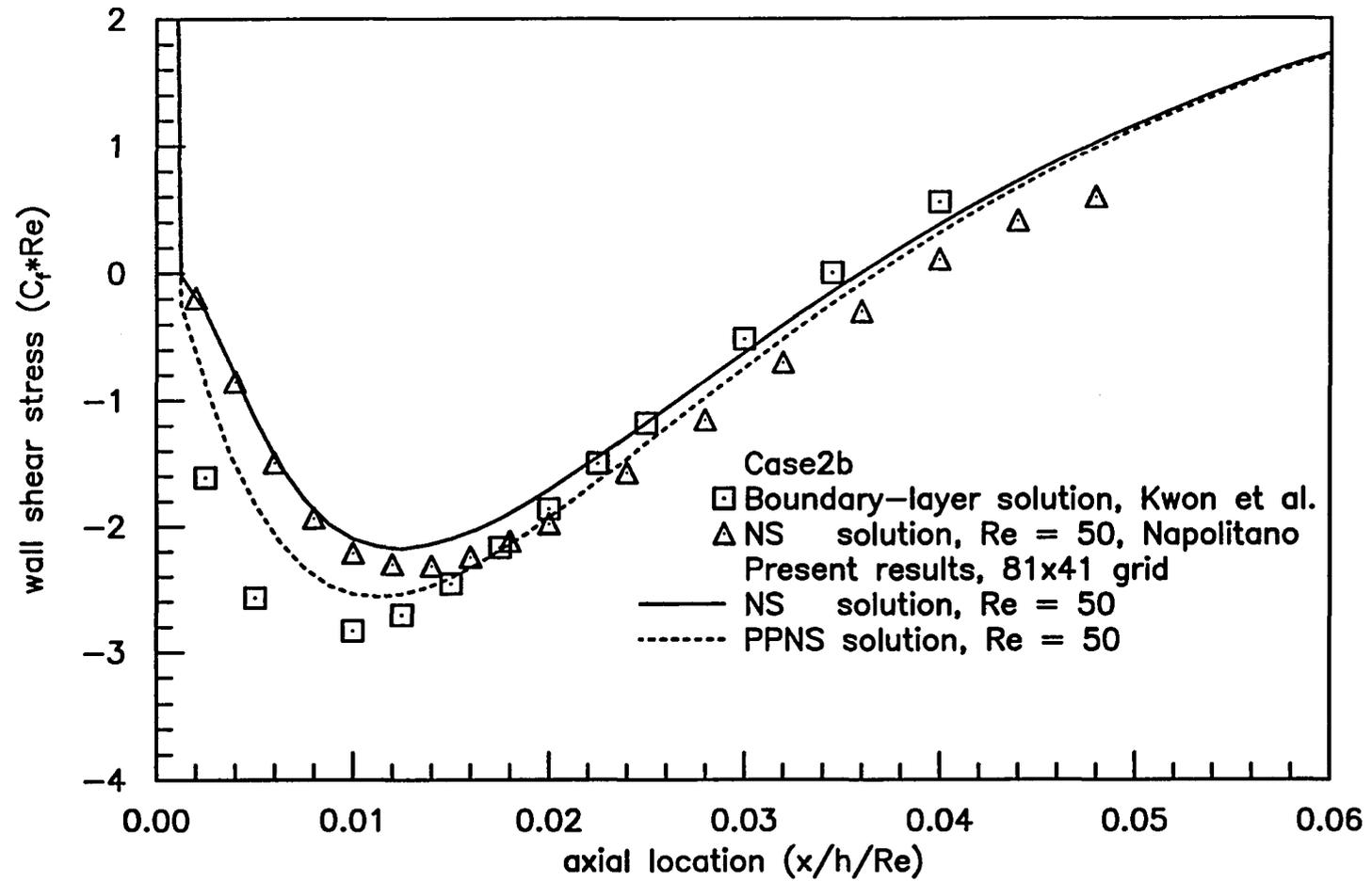


Figure 5.15. Comparison of NS and PPNS wall shear stress distributions for Case 2b

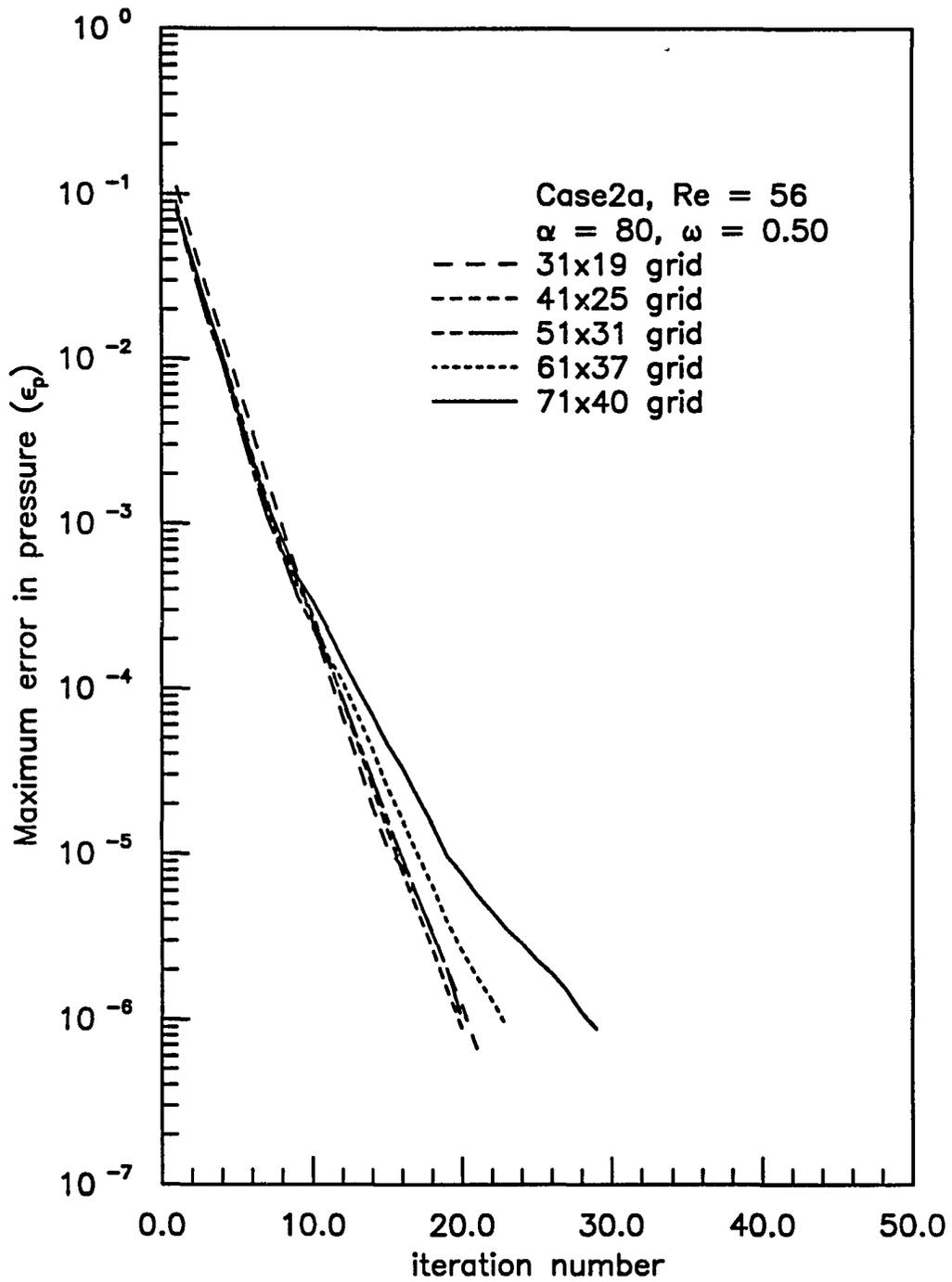


Figure 5.16. Convergence histories for Case 2a solutions at Re = 56

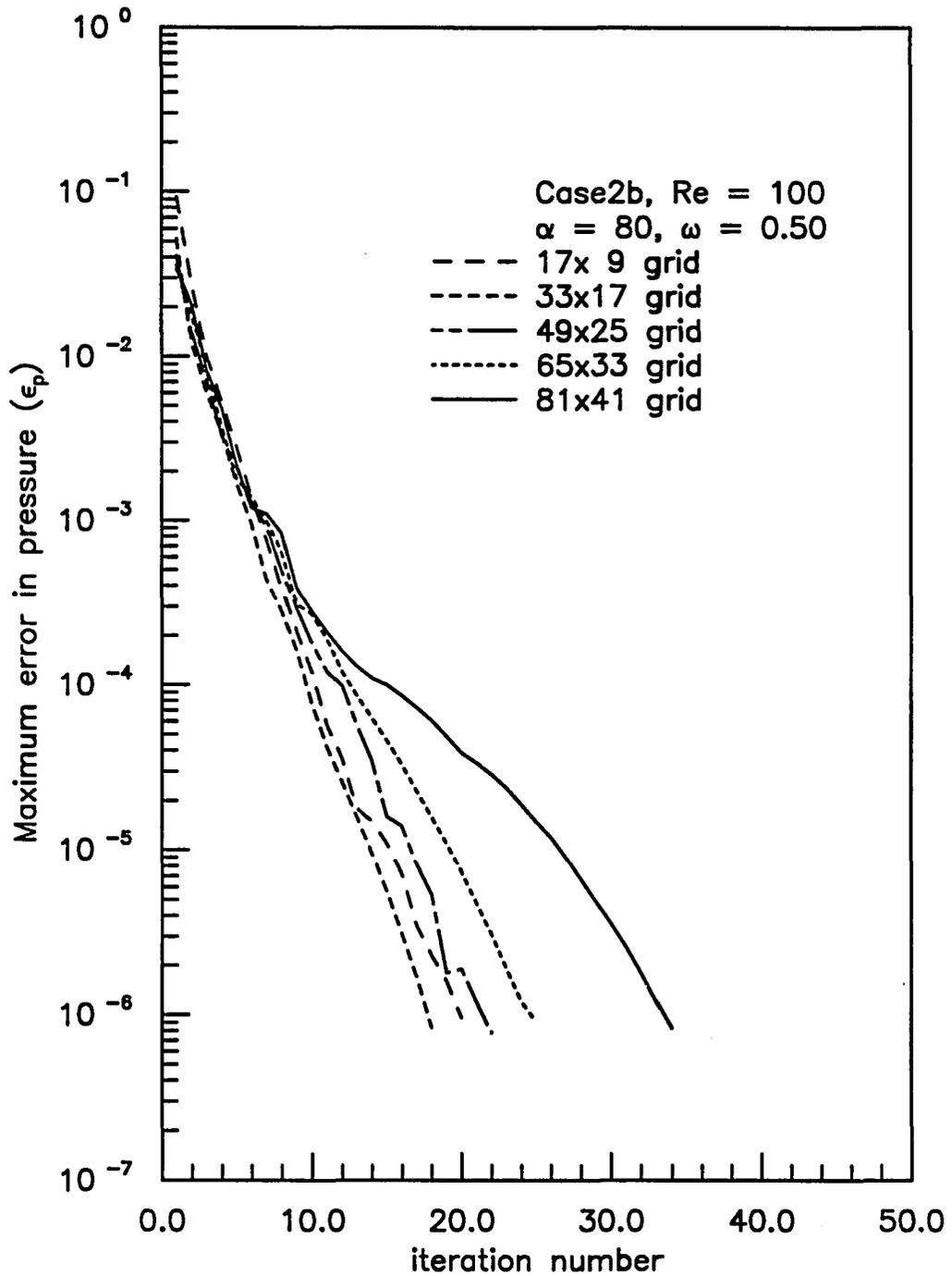


Figure 5.17. Convergence histories for Case 2b solutions at Re = 100

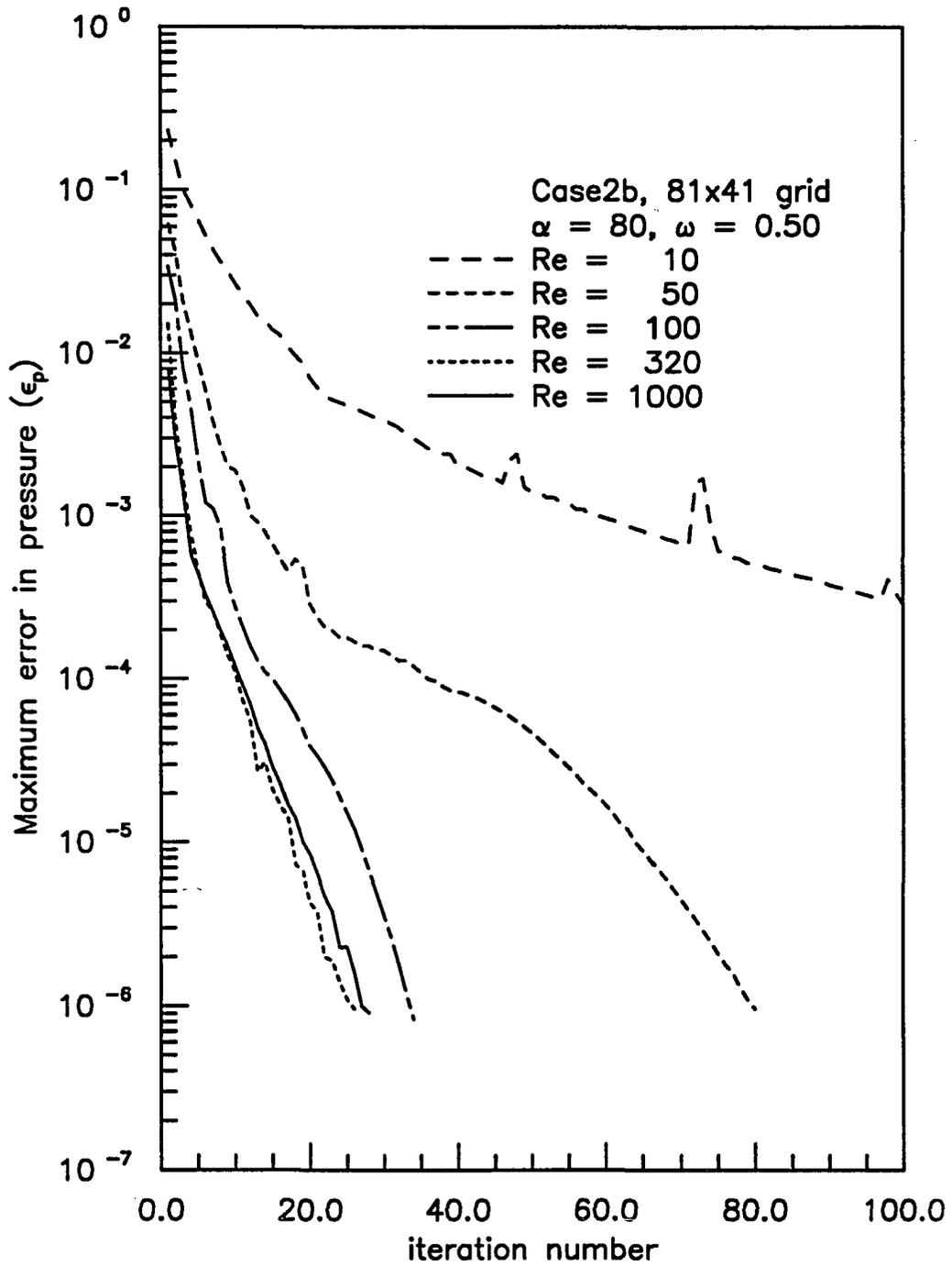


Figure 5.18. Convergence histories for Case 2b fine grid solutions

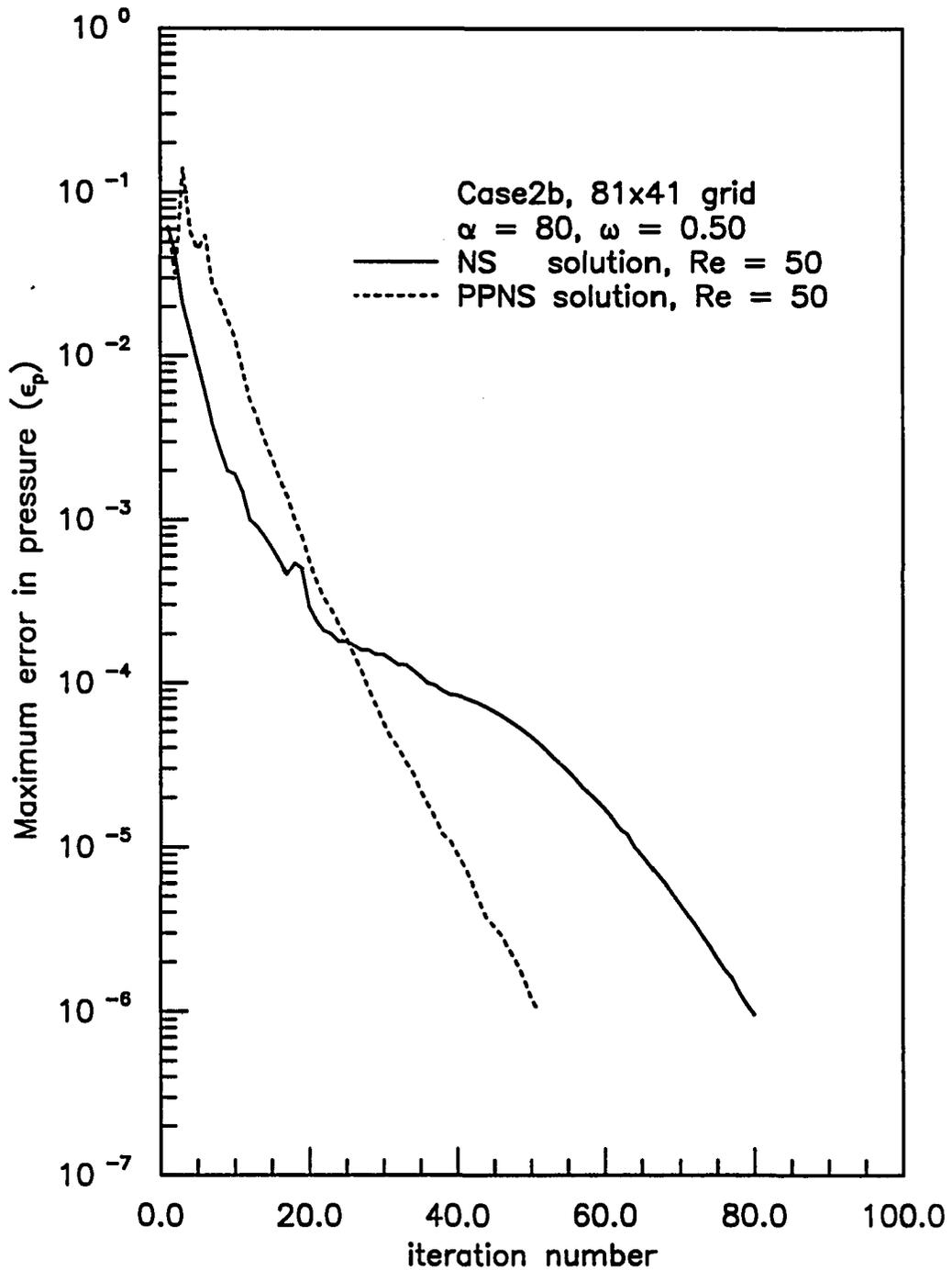


Figure 5.19. Comparison of NS and PPNS convergence histories for Case 2b

C. Channel Flow with an Asymmetric Sudden Expansion

The asymmetric $3/2$ sudden expansion flow in a channel is labeled as Case 3. As with the symmetric sudden expansion cases, the computations were performed with a series of uniform grids, fully-developed flow was assumed at the inlet plane located upstream of the expansion, and the last marching station was placed downstream of the reattachment point. The channel Reynolds number was evaluated using the bulk inlet velocity and the channel exit height. The flow conditions are given in Table 5.1 and Table 5.3 lists the grid dimensions.

Unlike the symmetric case, the asymmetric sudden expansion turns or deflects the bulk flow, which requires large transverse pressure gradients. At high Reynolds number these deflections are not damped by viscous stresses and the flow meanders, producing recirculating eddies on both sides of the channel. The boundary-layer model is therefore not suitable, although Kwon [79] has computed a boundary-layer solution. Kwon and Pletcher [80] have applied a viscous-inviscid interaction technique to the prediction of flows over rearward-facing steps where an inviscid core region is present. However, for fully-developed inlet conditions the Navier-Stokes equations or some form of the reduced or parabolized Navier-Stokes equations are needed to accurately model this flow.

Experimental measurements in a channel with expansion ratio of $3/2$ by Denham and Patrick [81] are available for Reynolds numbers of 219 through 678 ($Re_c = 73$ to 229), below the threshold for multiple separations. Calculated results for the lower and higher Reynolds numbers are presented here. The experimental inlet profiles were not fully-developed, and are therefore not directly comparable to the present predictions. Numerical solutions have been reported by Hackman et al. [82] and by Chiu [77] for both the experimentally measured inlet profile and the fully-developed inlet profile. The numerical results show good agreement with the experimental results, therefore the numerical solutions obtained with a fully-developed profile are used for comparison.

Predicted reattachment point locations are compared with the numerical results of Hackman et al. [82] in Figure 5.20. Good agreement is observed, even on very coarse grids. Preliminary studies at Reynolds numbers above 1000 indicate the presence of multiple separations. However, more work is necessary to resolve these flows. The interested reader is referred to the experimental and numerical results of Armaly et al. [83] for a $2/1$ expansion ratio channel.

The predicted wall shear stress distributions on the straight and step wall downstream of the expansion are shown in Figure 5.21 and compared with the PPNS solutions of Chiu [77]. Fairly good agreement is observed for both Reynolds numbers, and it is evident that for the asymmetric expansion

the flow field does not simply scale based on the Reynolds number. For low Reynolds numbers it is anticipated that the NS and PPNS solutions will differ, as for the symmetric expansion case above, due to the effect of streamwise viscous stresses.

Predicted wall pressure distributions and velocity profiles are shown in Figures 5.22 and 5.23. Use of the first-order forward-difference method for the streamwise pressure gradient resulted in a lag in the response of the pressure field to the sudden expansion. A downstream shift of one marching step was apparent in the pressure distributions obtained with the first-order method. The second-order forward-difference method greatly reduces this error. The predicted velocity field was largely insensitive to this error in the integration of the pressure field. The results presented here were obtained with the first-order method, with the exception of Figure 5.22.

The convergence behavior for this case is similar to that of Case 2. The convergence rate was not strongly dependent on the number of nodes, and rapid convergence was obtained on all runs as shown in Table 5.3. Optimization of the convergence behavior is presented in the final section of this chapter.

D. Channel Flow with a Symmetric Sudden Contraction

The symmetric sudden contraction flow in a channel with a 2/1 contraction ratio is labeled as Case 4. Computations were performed on the channel upper half-plane with a number of different uniform and compressed grids in an attempt to resolve the very small separation bubble located downstream of the step corner. The final grid is shown in Figure 5.24 and was constructed with the transformations given in Appendix E. The flow conditions and geometric parameters are given in Table 5.1. As with the expansion cases, fully-developed flow was assumed at the inlet plane and the last marching station was placed far enough downstream that the pressure was constant across the channel. Solutions obtained with the grid extending twice the distance upstream and downstream were used to check the influence of boundary placement on the predicted flow. The special boundary treatment required at the forward-facing step face is presented in Appendix C.

Computed results were obtained for Reynolds numbers from 100 through 400 based upon the bulk inlet velocity and the channel half-width at the inlet. The predicted centerline velocity distributions are shown in Figure 5.25 along with the experimental results of Durst et al. [84]. Predicted boundary pressure distributions are shown in Figure 5.26. The predicted wall shear stress

Table 5-3. Results of grid refinement for incompressible channel flow with an asymmetric sudden expansion

N ^a	Grid Dimensions		ISTP	JSTP	X _r /s _b	Results
	IMAX	JMAX				ITN ^c
Case 3, Re = 219, Re _s = 73						
553	31	19	7	13	5.550	33
961	41	25	9	17	5.465	31
1481	51	31	11	21	5.414	27
2113	61	37	13	25	5.386	25
2658	71	40	15	27	5.362	28
Case 3, Re = 687, Re _s = 229						
514	28	19	4	13	12.98	32
893	37	25	5	17	12.58	31
1376	46	31	6	21	12.40	29
1963	55	37	7	25	12.32	31
2469	64	40	8	27	12.28	30

^a Number of nodes.

^b Length to reattachment/step height.

^c Number of global iterations.

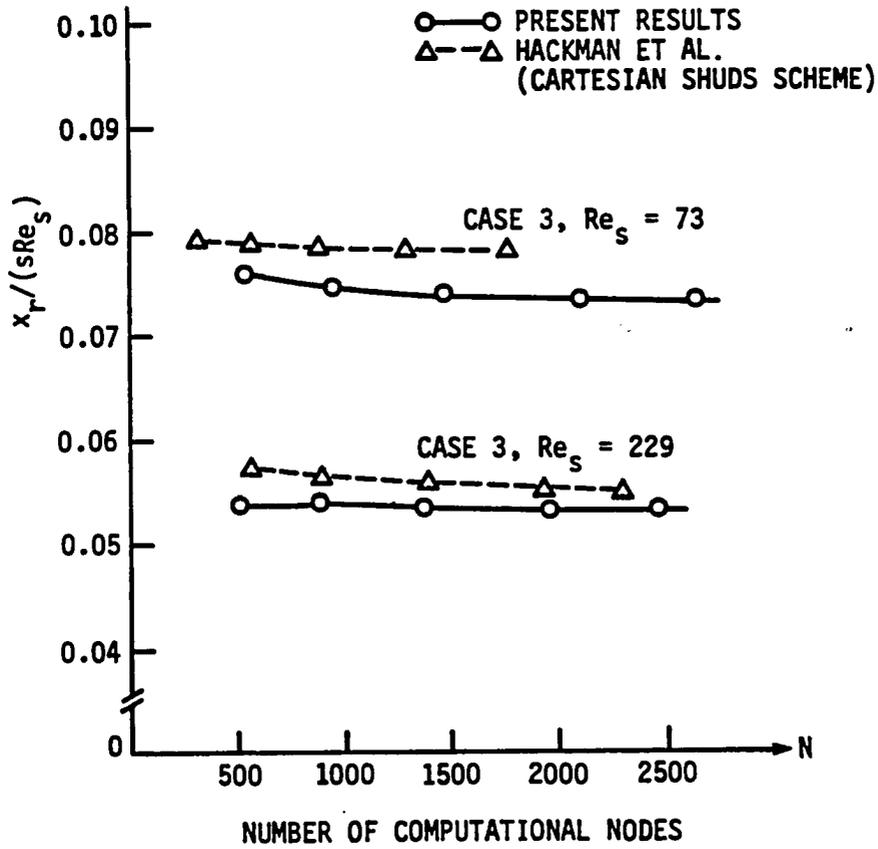


Figure 5.20. Effect of grid refinement on predicted reattachment length for Case 3

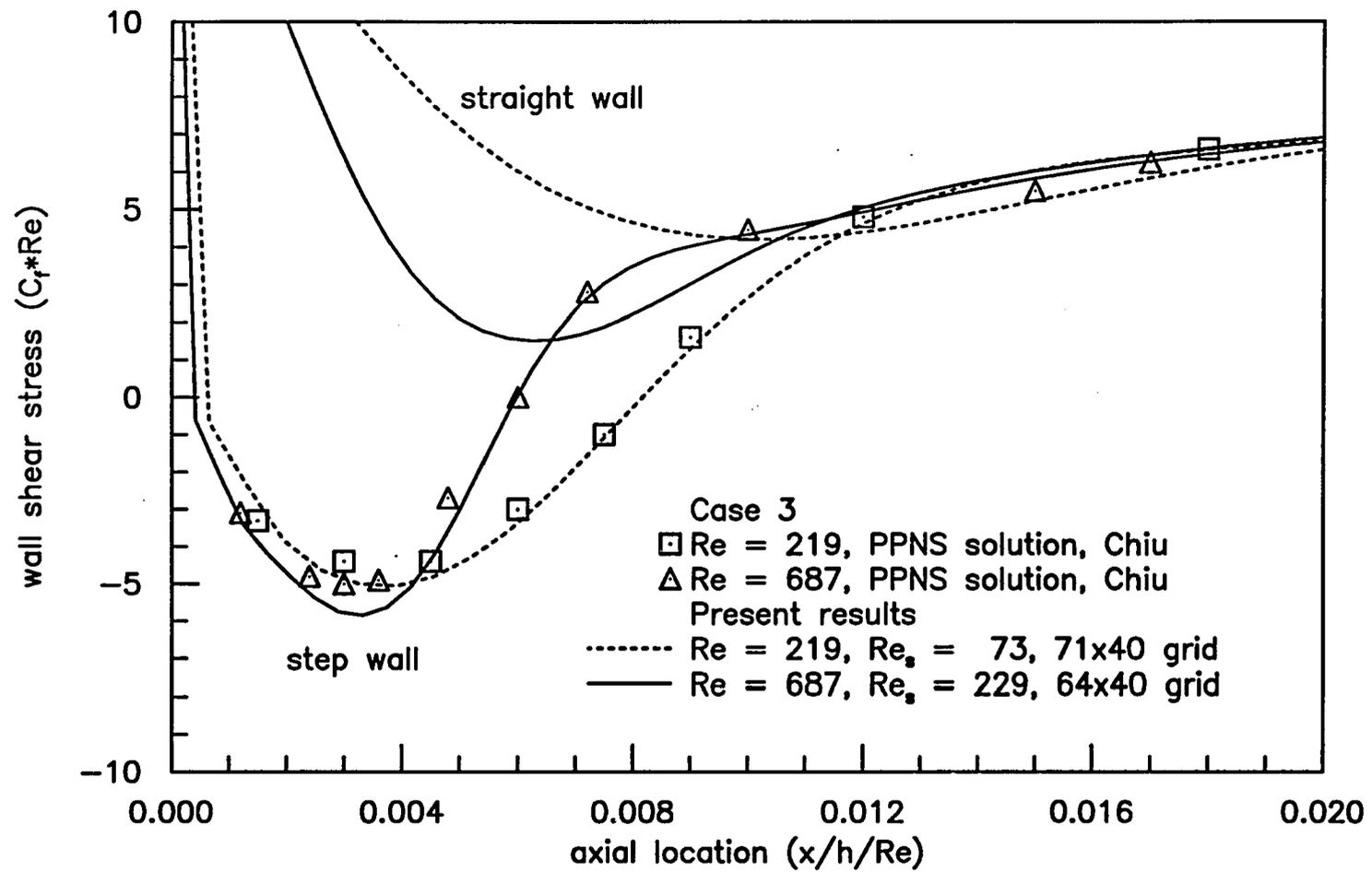


Figure 5.21. Predicted wall shear stress distribution for Case 3

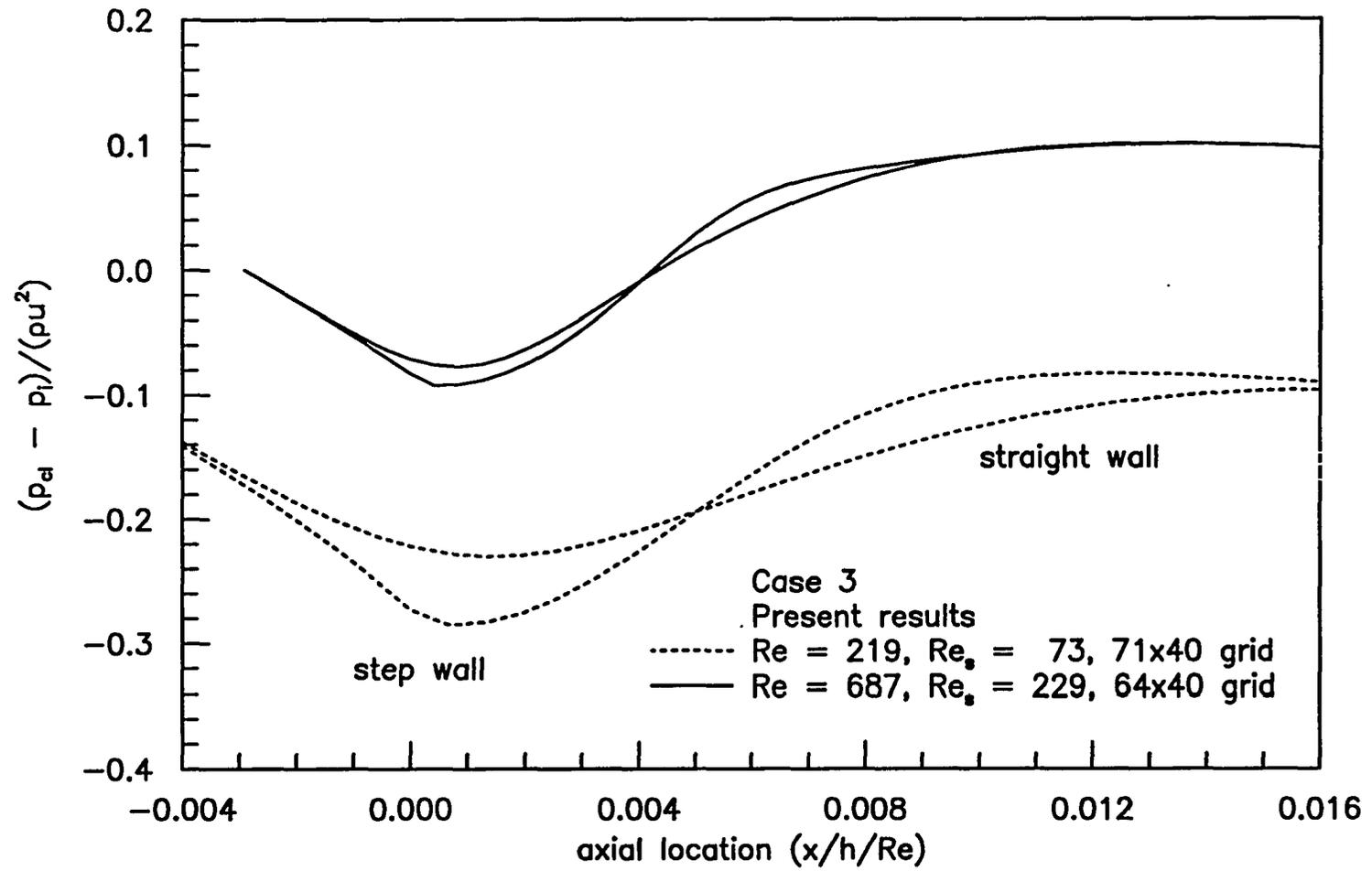


Figure 5.22. Predicted boundary pressure distribution for Case 3

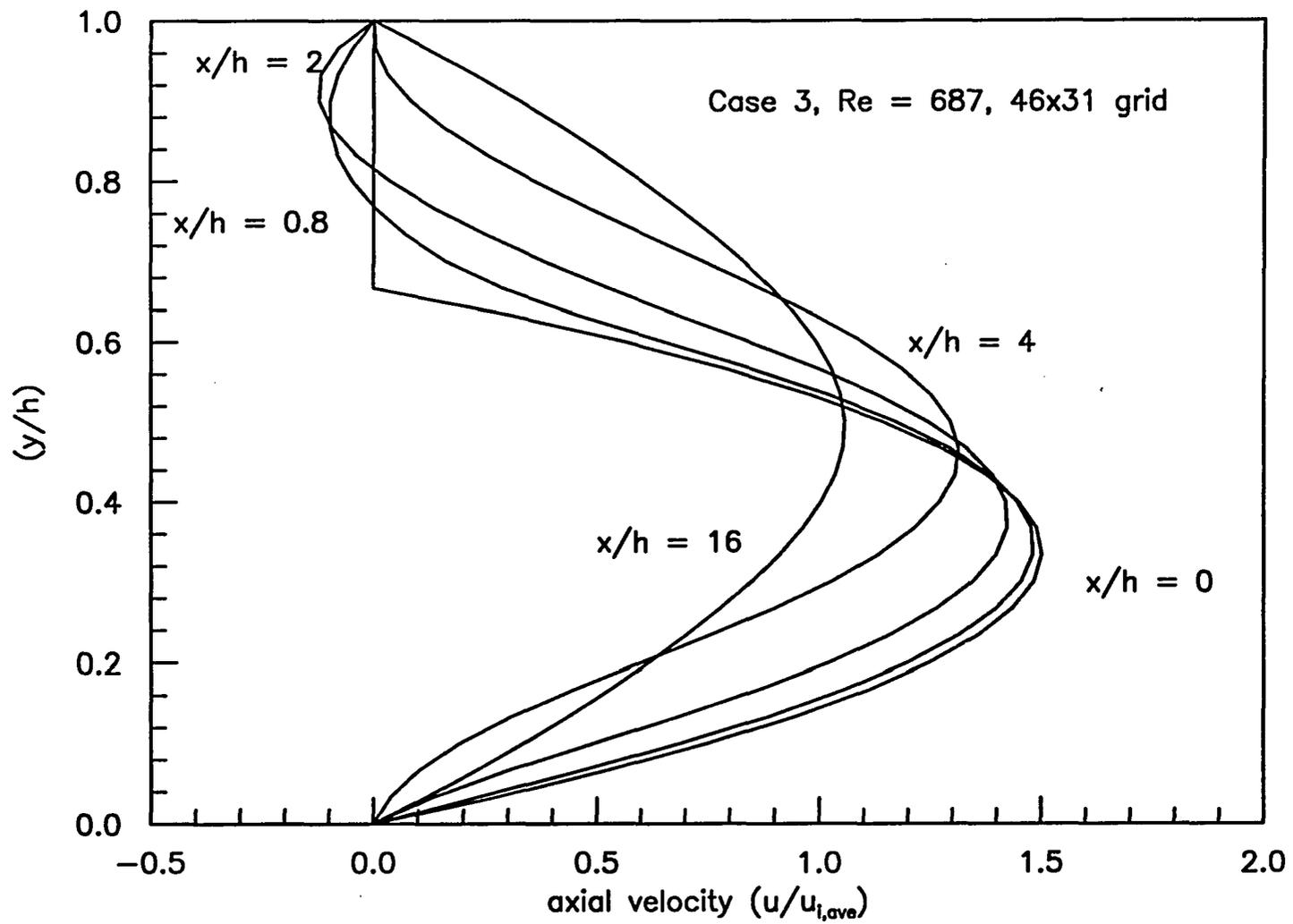


Figure 5.23. Computed axial velocity profiles for Case 3 at Re = 687

distributions upstream of the contraction are shown in Figure 5.27 and compared with the numerical results of Dennis and Smith [85] and Mei and Plotkin [86]. The predicted wall shear stress distributions downstream of the contraction are shown in Figures 5.28 along with the predictions of Mei and Plotkin [86].

The present predictions show qualitative agreement. However, the results were not found to be mesh independent, and a finer mesh is needed to accurately resolve the flow for this rather difficult test case. In particular the transverse velocities near the contraction are of the same order of magnitude as the streamwise velocity, producing large mesh Reynolds numbers on the grid used. The present method uses the hybrid differencing scheme for the transverse convective terms, and this method can produce large numerical diffusion errors for high mesh Reynolds numbers. By comparison, Dennis and Smith [85] solved the stream function and vorticity equations on a uniform grid. While the upstream eddy seems to have been well resolved, their calculations did not detect separation downstream of the corner, presumably due to the coarse grid and the artificial dissipation introduced by the first-order upwind differencing of the convective terms used in their procedure. Mei and Plotkin [86] also solved the stream function and vorticity equations, but utilized a non-uniform conformal grid with very small step sizes near the corner, and used the second-order upwind scheme for the convective terms. Mei and Plotkin were able to demonstrate a reasonable degree of mesh independence in their solutions. The very high gradients near the contraction also produced a large degree of uncertainty in the experimental data of Durst et al. [84].

The location of the separation point at the wall upstream of the step coincides with the point of zero wall shear stress. As seen in Figure 5-27, the present prediction shows significant error. The fine grid solution obtained is not yet independent of the mesh, as the grid point distribution in the upstream separation bubble was rather coarse in order to concentrate nodes near the step.

A small separation bubble was predicted downstream of the step at Reynolds numbers above 200. The separation and reattachment points downstream of the step are located by the points of zero shear stress in Figure 5-28. Here the agreement with the solutions by Mei and Plotkin is better, although the results are not mesh independent in this region either. The separation bubble is very thin. At a Reynolds number of 400 the bubble extends a distance of only 2% of the upstream channel height away from the wall.

The convergence rate for this problem was much less than achieved for the previous test cases, however the flow geometry presents a stiff challenge to the application of a space-marching

procedure. The large geometrical blockage at the forward step and the fine streamwise step size needed to resolve the flow near the step create a problem initializing the global iterations. If the space-marching procedure is initialized with a uniform pressure field, there are no signals present to alter the flow before the step is encountered. On the first marching sweep the flow solution would have to adjust to the sudden contraction in the space of one marching step, forcing extreme transverse velocities in order to pass the specified mass flow around the step. For the compressible flow formulation a choking condition can occur with no correct solution possible, and the incompressible solution produces unrealistic velocity and pressure fields. Small relaxation factors may damp the large calculated changes, but at the cost of increasing the number of iterations.

A "fairly accurate" starting assumption for the pressure field was found to be necessary to obtain a converged solution on a fine mesh. Several methods for estimating the initial pressure field were considered, such as computing the potential flow pressure distribution, or applying the pressure correction procedure with stagnation pressure boundary conditions at the step face before the first marching sweep. However, these approaches seemed to be somewhat ad hoc and would require a separate numerical procedure to obtain the starting conditions. Instead, the technique known as coarse-fine-grid-sequencing (CFGS) was utilized. The procedure entailed computing a solution on a coarse grid, and then interpolating those results to estimate the flow field to start the calculation for the next level finer grid.

The CFGS procedure was successfully used by starting with a mesh coarse enough to limit the transverse velocities calculated on the first sweep. Then for finer meshes the initial approximation of the pressure field was good enough to advance the solution. Three levels of grids were used, starting with a 21 by 11 grid, then a 41 by 21 grid, and finally an 81 by 41 grid. The convergence sequence for the CFGS procedure is shown in Figure 5.29 and the required number of global iterations on the final mesh is listed in Table 5.1. The number of iterations on the coarse grids used to start the calculation was much lower than needed on the final grid, and the additional computer time needed to obtain the coarse grid "starting" solutions accounts for less than 20% of the total computational time needed to reach the final solution.

The reasons for the slower convergence rate have not been determined. The overall convergence rate achieved was strongly dependent on the number of nodes and the grid layout. The varying aspect ratio and skewness of the nonorthogonal grid may have contributed to the reduction in the convergence rate, or physically resolving the very steep pressure gradients with a fine mesh near

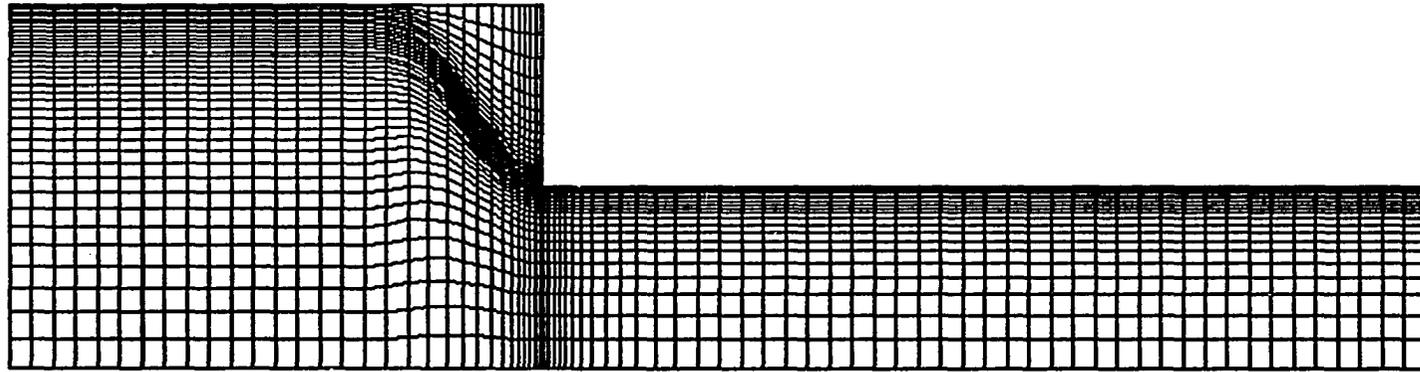


Figure 5.24. Nonorthogonal grid layout for Case 4

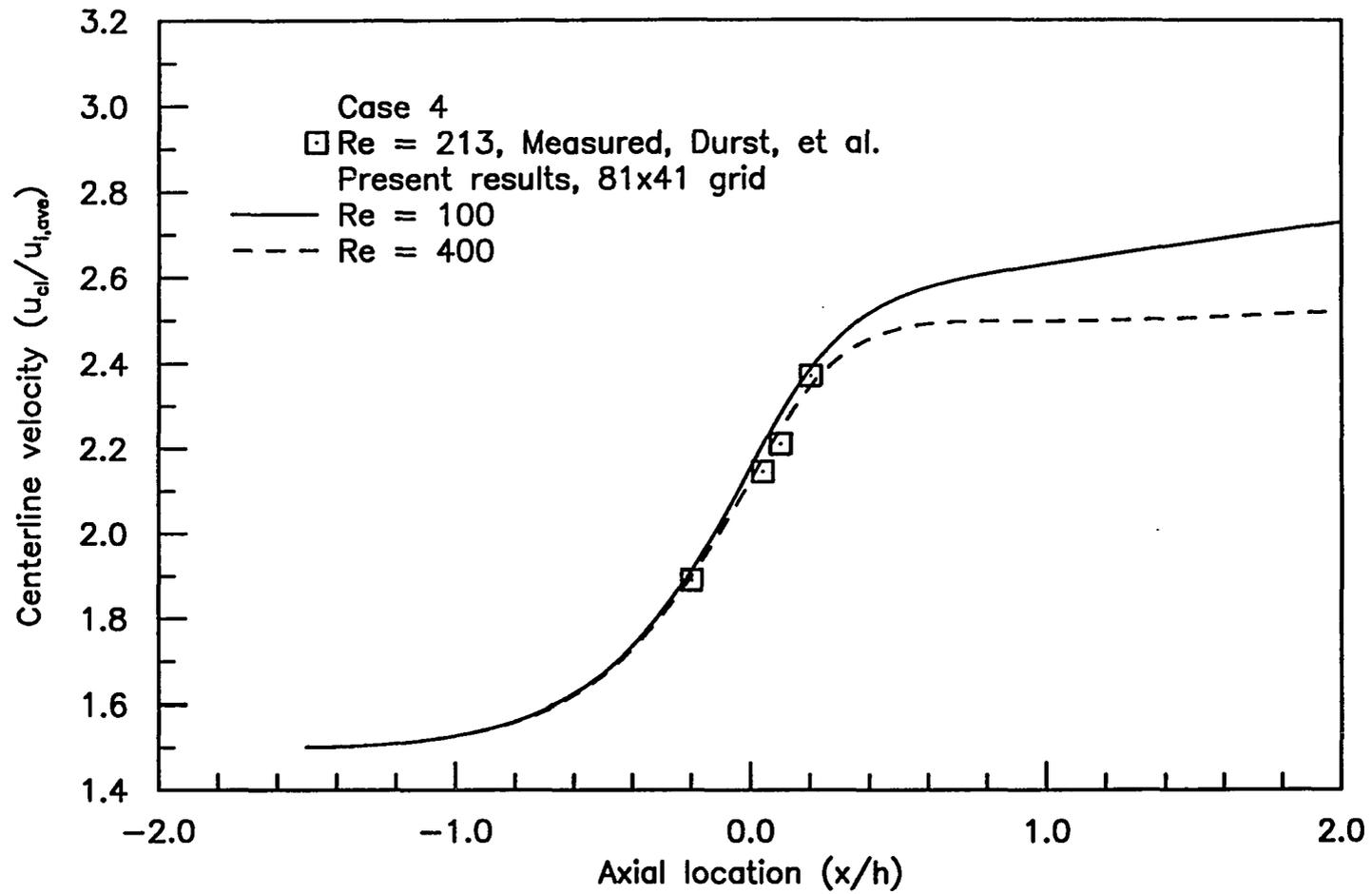


Figure 5.25. Predicted centerline velocity distribution for Case 4

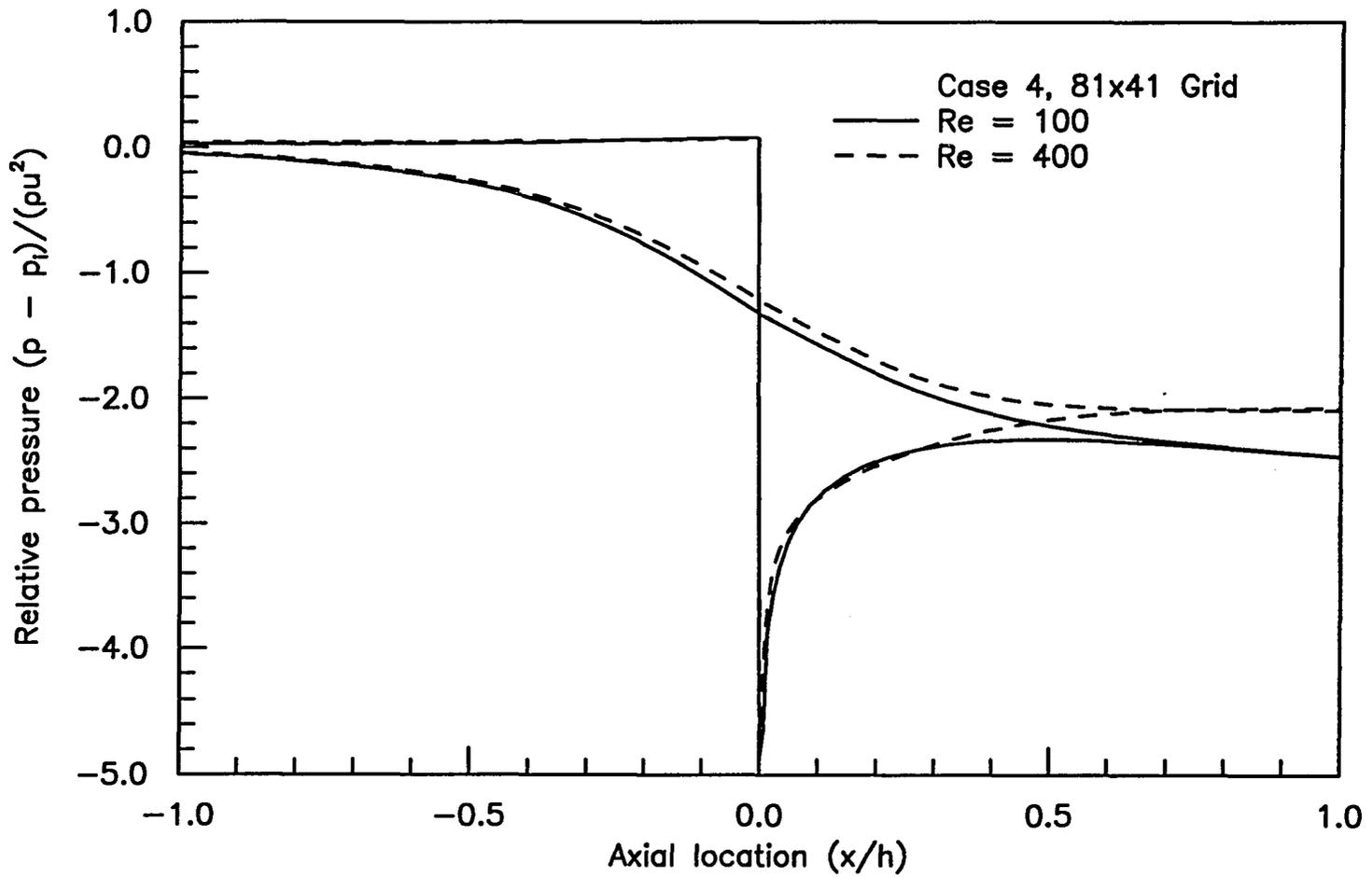


Figure 5.26. Predicted boundary pressure distribution for Case 4

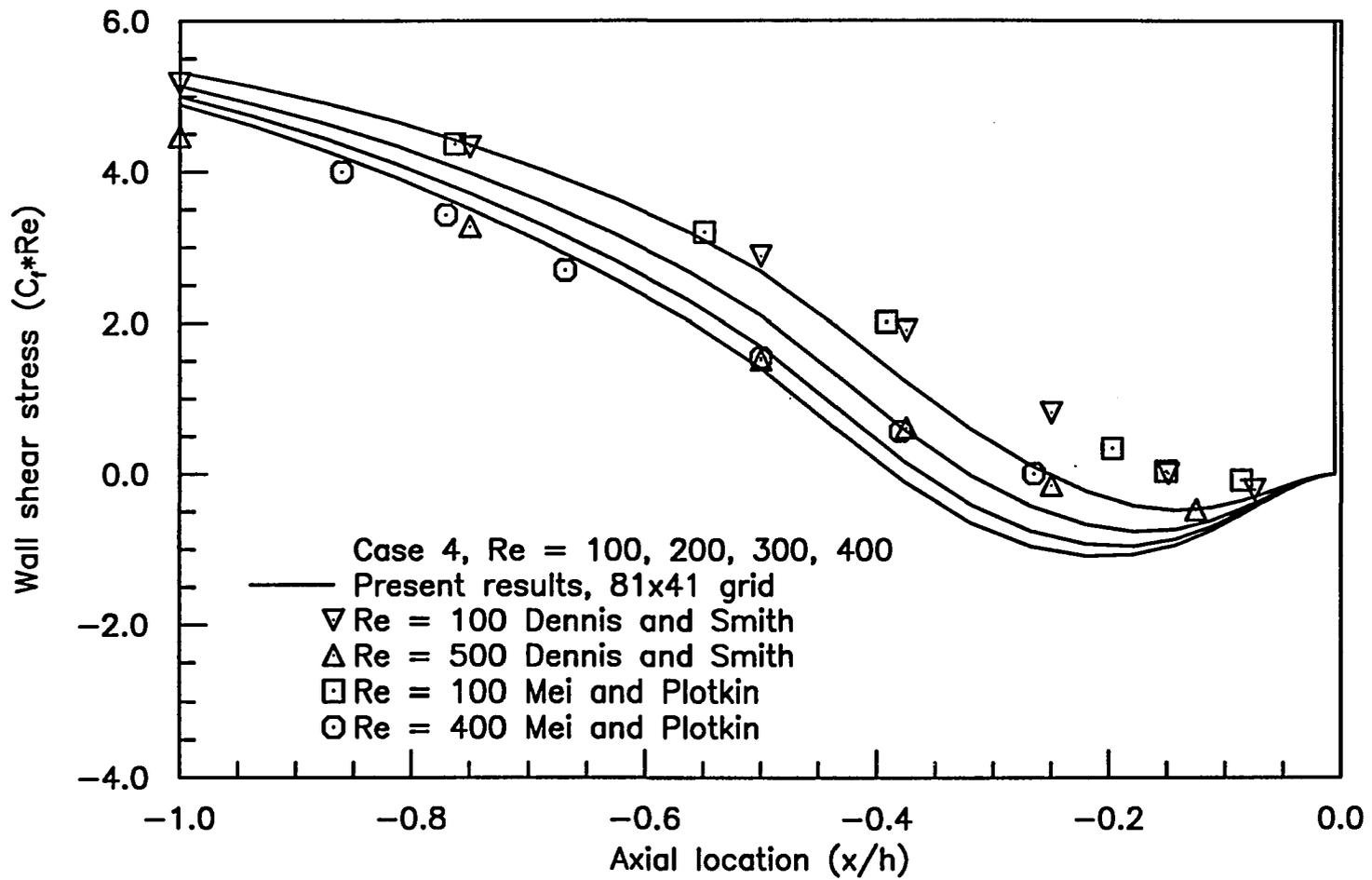


Figure 5.27. Predicted upstream wall shear stress distribution for Case 4

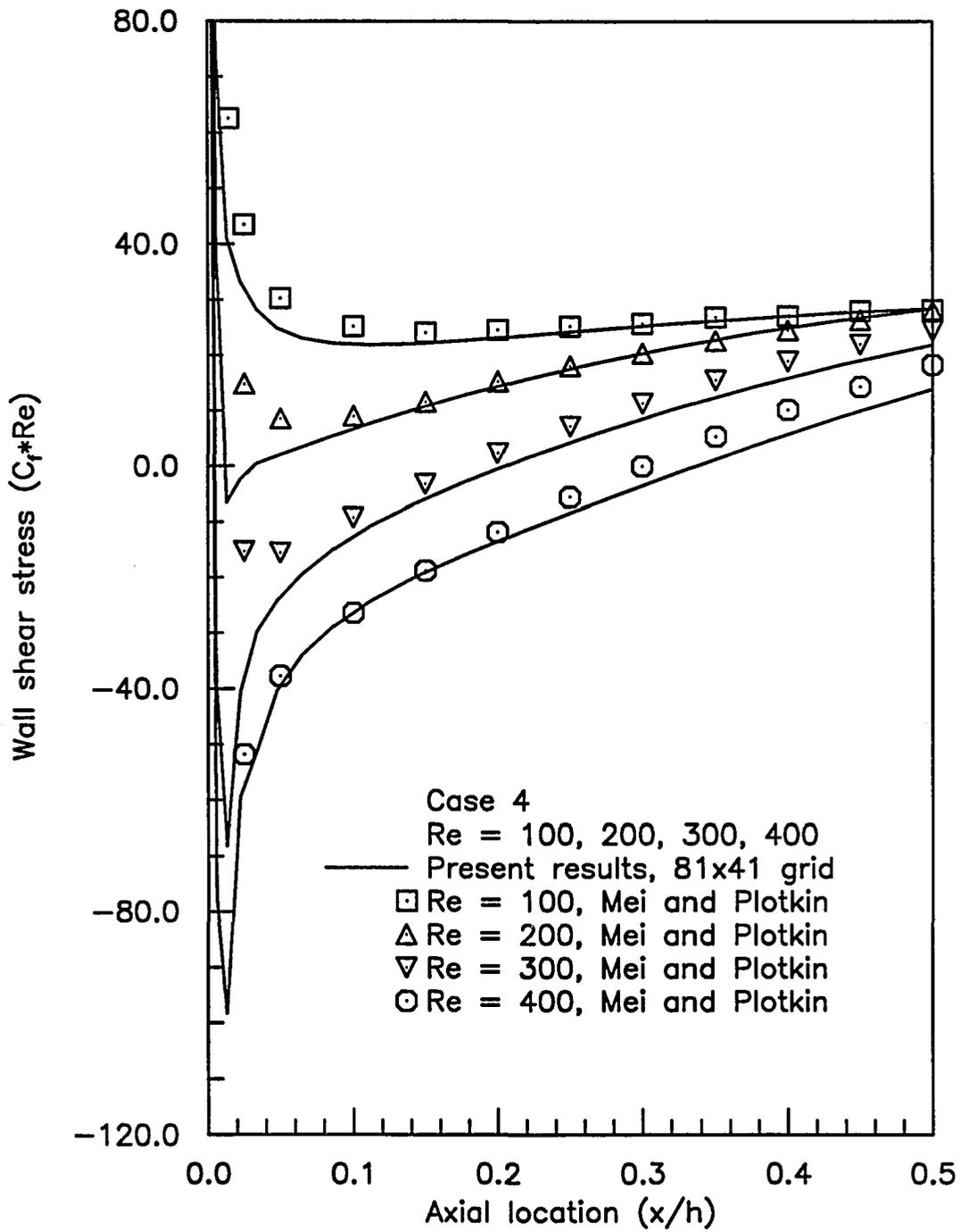


Figure 5.28. Predicted downstream wall shear stress distribution for Case 4

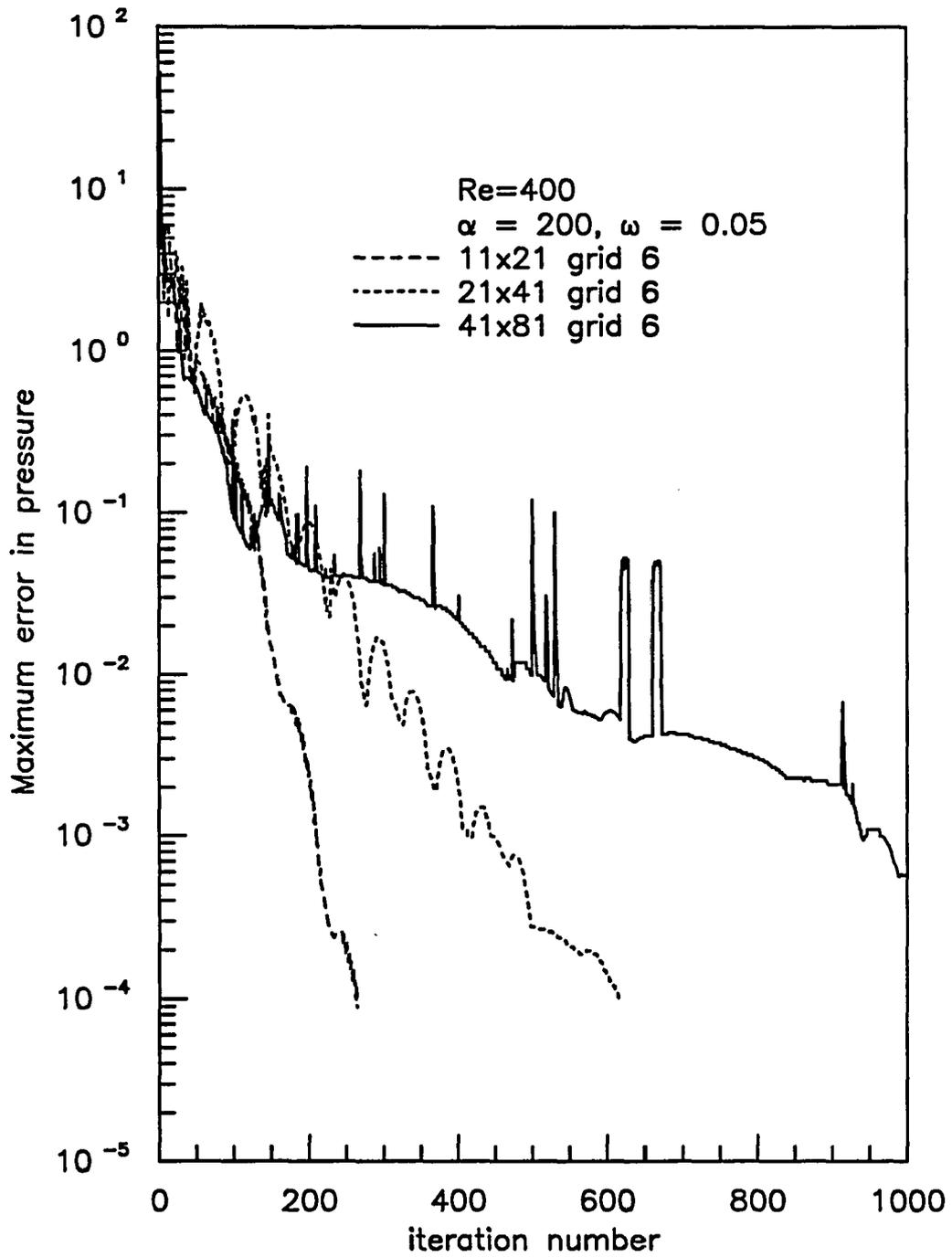


Figure 5.29. Convergence histories for Case 4 solutions at $Re = 400$

the step may simply have required additional iterations. Solutions on a uniform 81 by 41 mesh were obtained in roughly 60 iterations, but the resolution near the step with this grid was insufficient to accurately resolve the downstream separation bubble.

E. Crossflow over a Cylinder

The crossflow over a cylinder is labeled as Case 5. The flow visualization experiments by Coutanceau and Bouard [87] show that for Reynolds numbers less than 40 (based on the diameter and freestream velocity) the flow is steady with two symmetric eddies in the separated region behind the body. For Reynolds numbers above 40 the flow becomes unstable and vortices are alternately shed from either side. For Reynolds numbers less than five the flow remains attached completely around the cylinder.

The flow field was computed at a Reynolds number of 40 since other numerical solutions are available for comparison. A symmetrical flow was assumed, and the grid extended a distance of 20 diameters upstream, downstream and to the side of the center of the cylinder. The nonorthogonal grid shown in Figure 5.30 was constructed using the potential flow stream function and velocity potential function in combination with stretching functions to cluster nodes near to the body. The flow was assumed to be uniform at the upstream boundary. Along the side freestream boundaries the streamwise velocity component was set equal to the freestream value, the normal derivative of the transverse velocity was set equal to zero, and the static pressure was varied to maintain constant total pressure. At the downstream boundary the streamwise diffusion terms were assumed to be negligible and the pressure just outside the grid was assumed to have recovered to the freestream level.

The computed surface pressure distribution is compared in Figure 5.31 with the numerical results of Son and Hanratty [88] and the more recent work of Rhie [18]. The pressure distribution is reasonably well predicted over most of the cylinder, except near the leading edge. The grid does not contain enough nodes near the stagnation point to accurately resolve the steep gradients in this area. It is believed that an improved computational grid would result in more accurate prediction of the stagnation point pressure. Predictions of the separation point and the length of the trailing separation bubble are given in Table 5.4. The present predictions fall in the range between the results obtained by Kwak et al. [34] and Son and Hanratty [88].

Due to the large obstruction presented by the cylinder, it was necessary to use the CFGS procedure to obtain the fine grid solution. A very small relaxation factor was used to damp the large

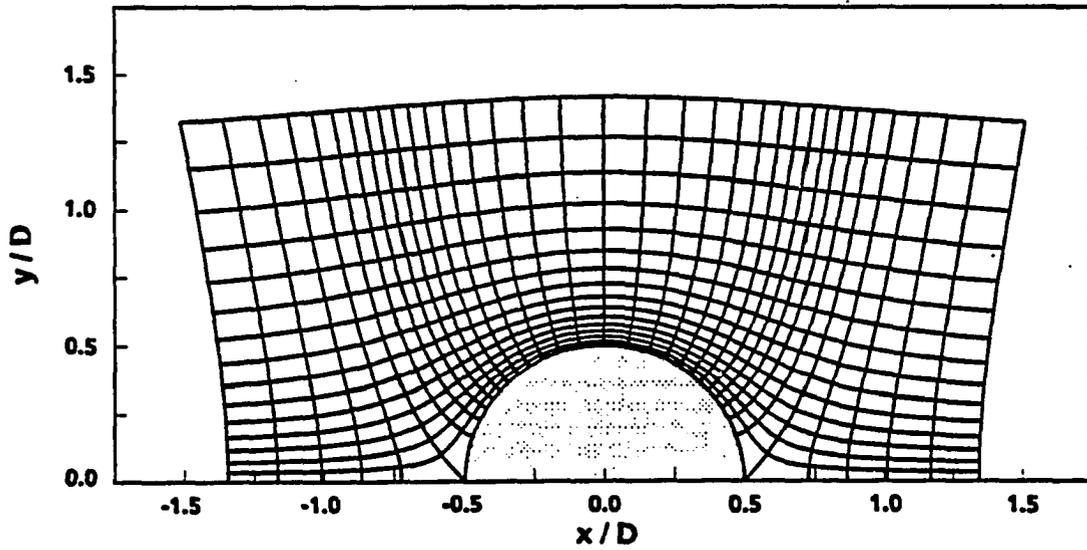
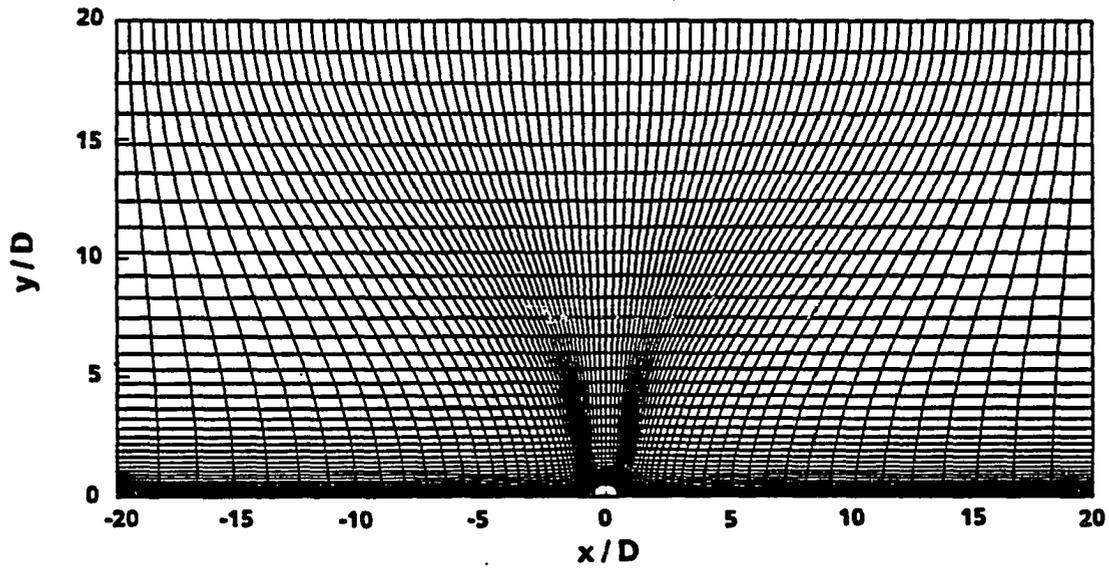


Figure 5.30. Nonorthogonal grid layout for Case 5 with detail view of the region near the cylinder

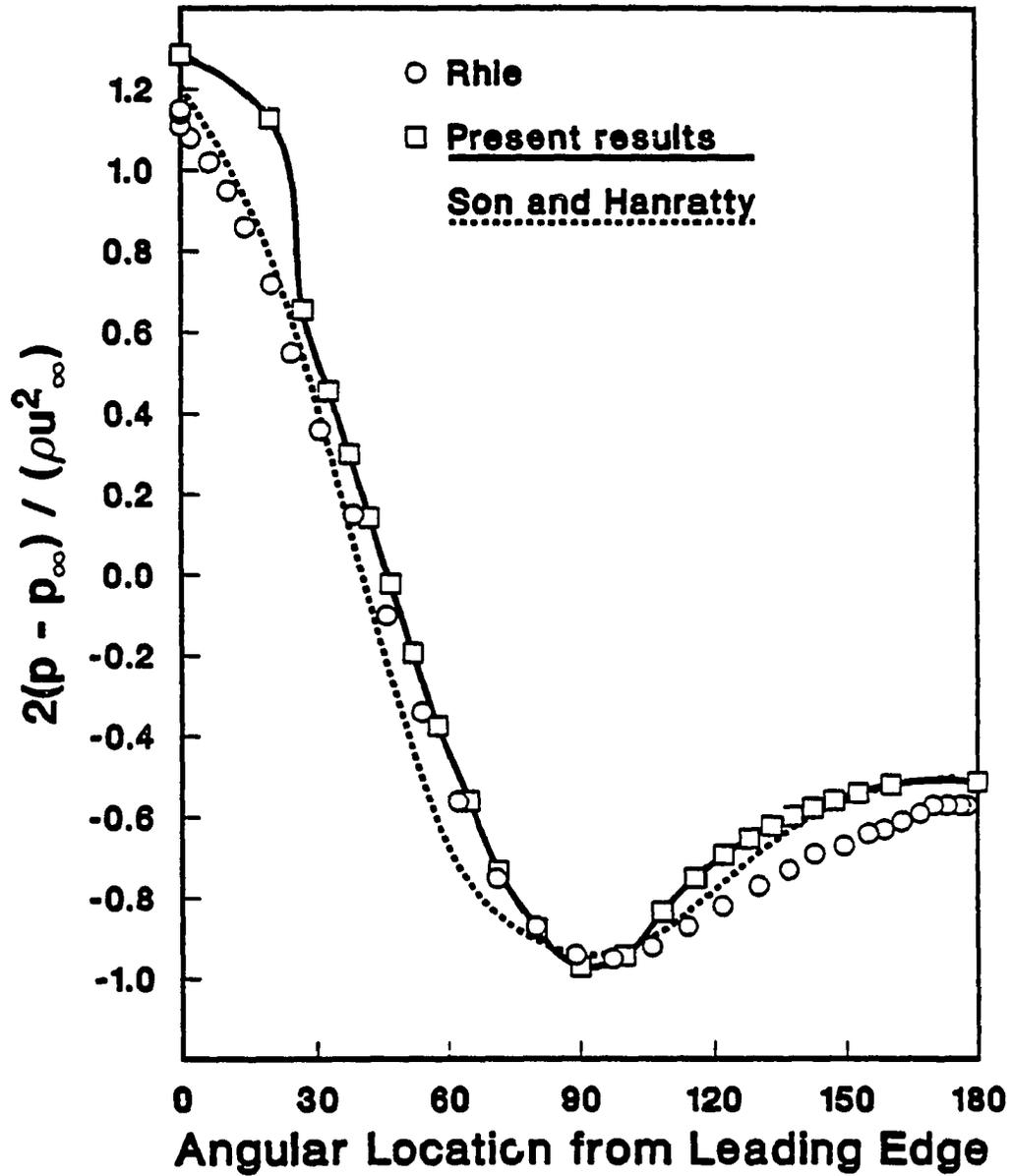


Figure 5.31. Predicted surface pressure distribution for Case 5 at $Re = 40$

pressure changes for the initial marching sweeps. After approximately twenty sweeps with the low relaxation factor, the relaxation factor was increased to the value listed in Table 5.1.

Table 5-4. Separation bubble predictions for Case 4 at $Re = 40$

	θ_s (deg)	L_r/D
Son and Hanratty	53.9	2.5
Present results	52.7	2.38
Rhie	—	2.07
Kwak et al.	52.	1.9

F. Developing Flow in a Square Cross-Section Duct

The entrance flow in a square cross-section duct with an arbitrarily imposed uniform, irrotational, inlet profile is labeled as Case 6 and was the only three-dimensional case studied. Symmetry conditions were employed such that the flow was calculated on one quadrant of the duct. The cross-sectional grids were uniform, but the marching stations were packed near to the inlet plane using the same grid transformation as for Case 1. The grid generation procedure is described in Appendix E, and values of the grid dimensions and stretching parameters are given in Table 5.1 and Table 13.1. The flow specifications are similar to those for Case 1. The channel length was varied with the Reynolds number and was selected to ensure fully-developed flow at the exit plane. The duct half-width and the average bulk inlet velocity are the characteristic values used to establish the Reynolds number. This value is one-half the Reynolds number based on the hydraulic diameter.

Computed results were obtained for a Reynolds numbers of 5, 25 and 100 ($Re_D = 10, 50$ and 200). The character of the developing flow is essentially parabolic at high Reynolds numbers, but elliptic at lower Reynolds numbers. The predicted centerline velocity distributions are shown in Figure 5.32 along with the numerical results obtained by Wong and Reizes [8] with a vector potential method. Gegg [9] has also solved this problem with the vector potential method. The present fine grid solution at a Reynolds number of 100 compares well, and the coarse grid solutions at lower Reynolds numbers indicate the proper Reynolds number trend.

Matching the uniform inlet boundary condition with a coarse grid presents a similar "blockage" problem similar to that encountered with Case 1. The inlet velocity must be specified greater than

unity to produce the desired flowrate through the restricted inlet area. As the mesh is refined this discrepancy in the boundary conditions diminishes, and the centerline velocity distribution appears to converge to a grid independent solution as shown in Figure 5.33.

Predicted velocities within the developing region at a Reynolds number of 100 are shown in Figures 5.34 through 5.37. The predicted streamwise velocity profiles as the flow develops are shown in Figure 5.34 and Figure 5.35 for every fourth marching station. Profiles are shown for the plane of symmetry bisecting the duct and for a section through the center of the quadrant that forms the computational domain. At locations near the duct entrance the velocity profiles show "overshoots" near the walls similar to those found in Case 1. The fully-developed velocity profiles at sections spaced every 5% of the channel half-width are shown in Figure 5.36. The fully-developed profiles agree closely with the analytic solution for the fully-developed profile given in White [1]. The flow at the exit plane was not "fully-developed" in the sense that the secondary velocities have not decayed to zero, as shown by the greatly magnified secondary velocity vectors in Figure 5.37. The displacement of fluid away from the walls produces counter-rotating vortex pairs in each corner, which persist for a significant distance downstream. The secondary flow direction is outward toward the duct wall along the channel bisectors, and from the corners diagonally inward toward the center. At the exit plane the predicted magnitude of the secondary velocity at the center of the quadrant is only 0.14 % of the bulk inlet velocity, so the motion does not significantly alter the axial velocity profiles.

At Reynolds numbers below 100, marching solutions could not be obtained with the present method for fine grids in the cross-plane in combination with large marching steps in the axial direction. With fine grids in the cross-plane the initial space-marching sweep solutions were unstable. This departure from a stable solution indicates either an ill-posed problem or numerical instability in the finite-difference formulation. Since the secondary flow and associated transverse pressure gradients are stronger at low Reynolds numbers, inadequate linearization or the approximate solution of the cross-plane equations by the ADI technique are believed to be the source of the errors. The present solutions were obtained by refining the step size in the marching direction, while retaining a relatively coarse mesh in the cross-plane. For those cases where a space-marching sweep was stable, the convergence rate achieved with the global pressure correction technique was quite good. The convergence histories are shown in Figure 5.38 and Figure 5.39. The convergence behavior was

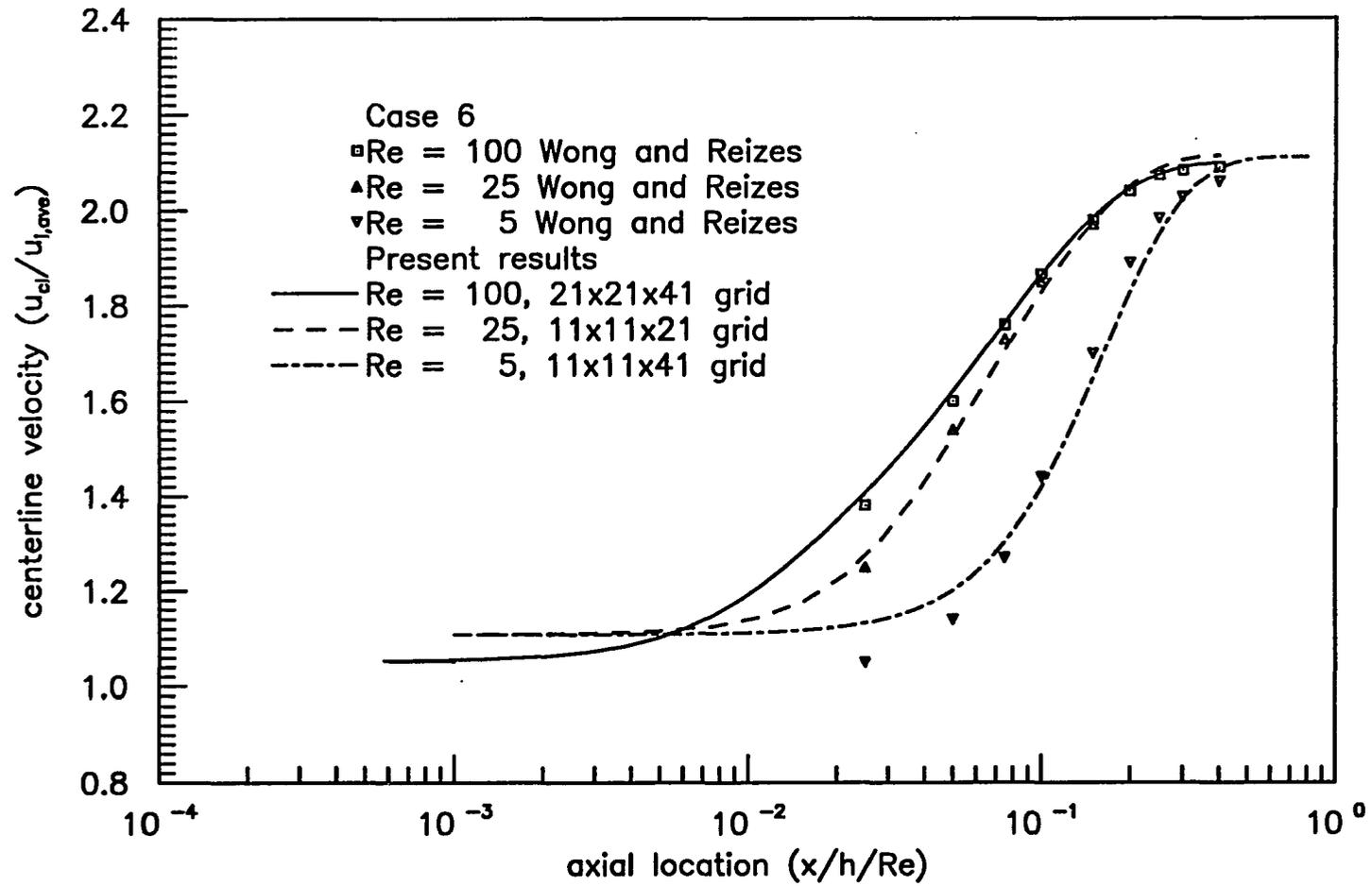


Figure 5.32. Predicted centerline velocity distribution for Case 6

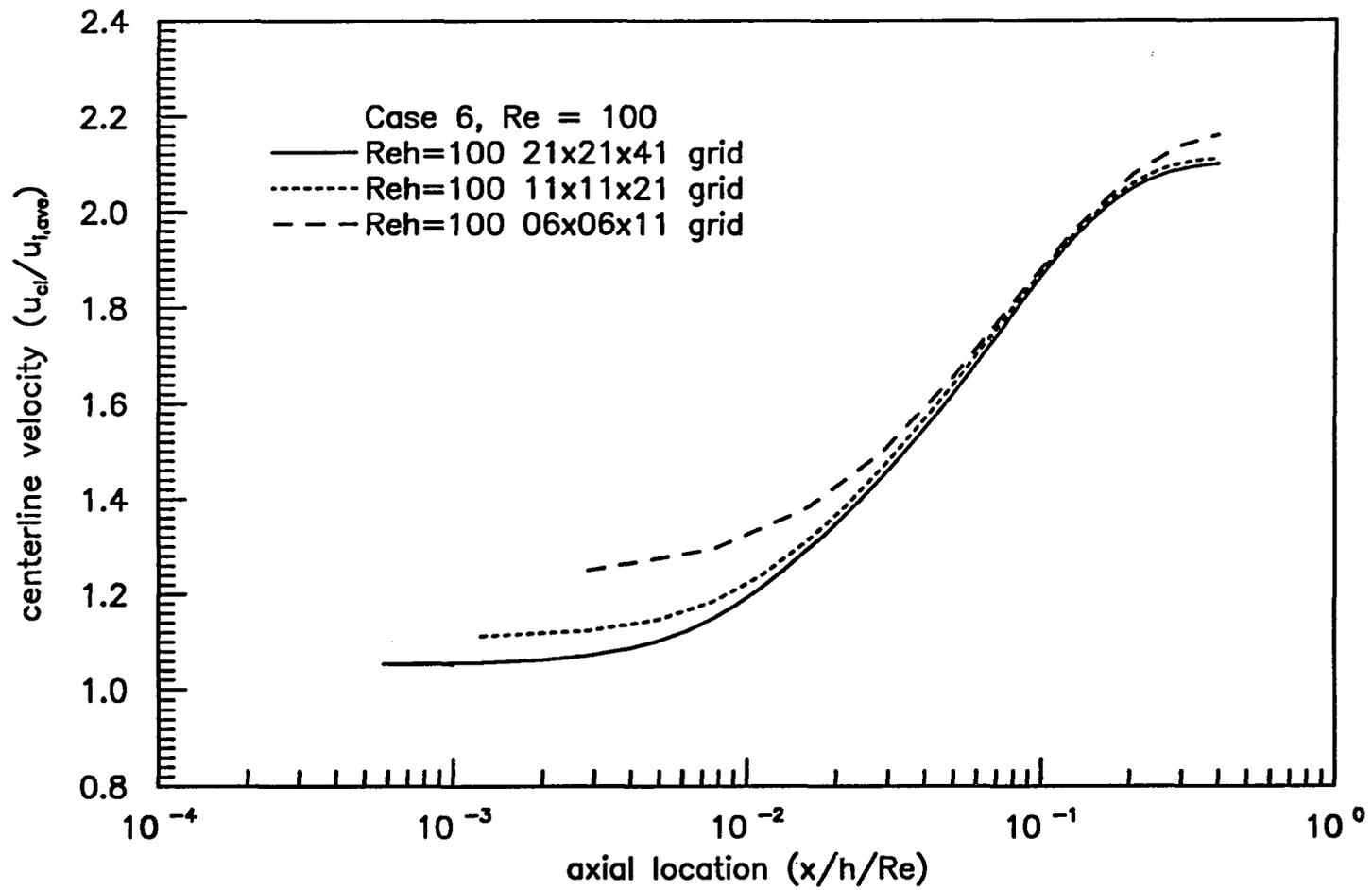


Figure 5.33. Convergence of centerline velocity distribution for Case 6

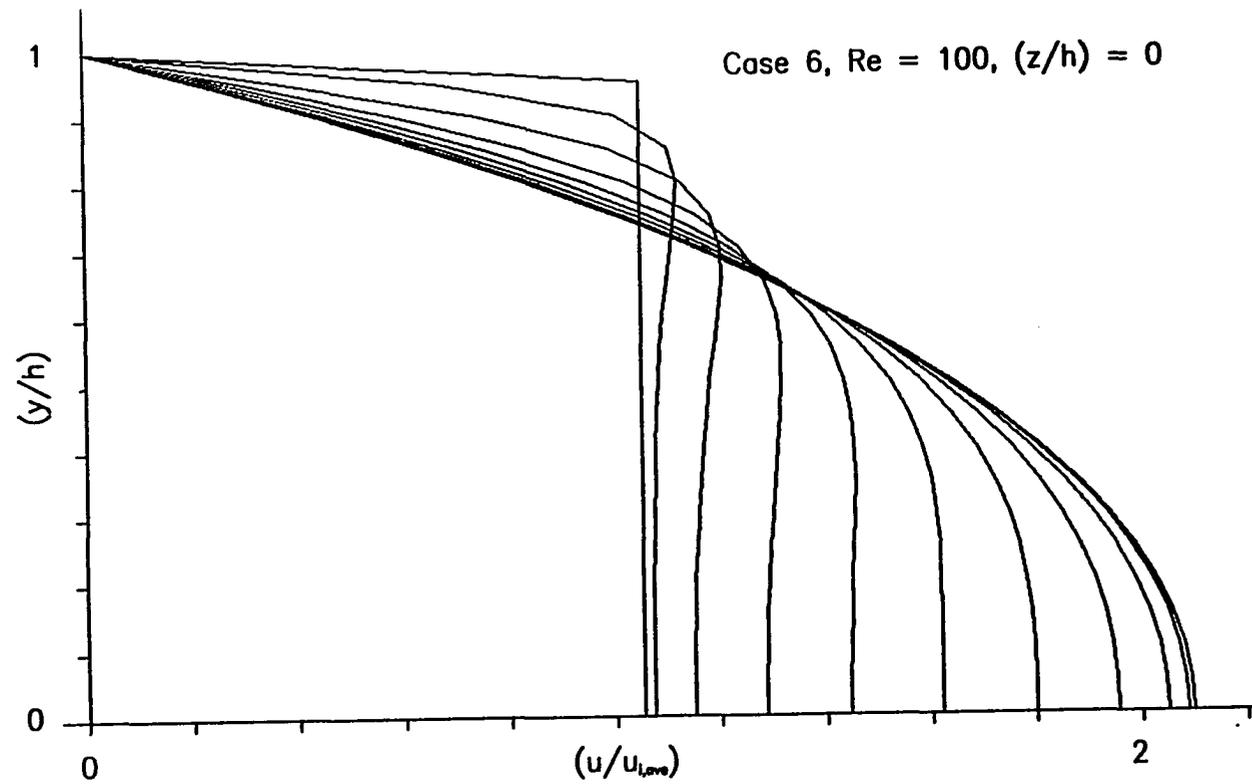


Figure 5.34. Computed axial velocity profiles at the duct mid-section for Case 6 at $Re = 100$

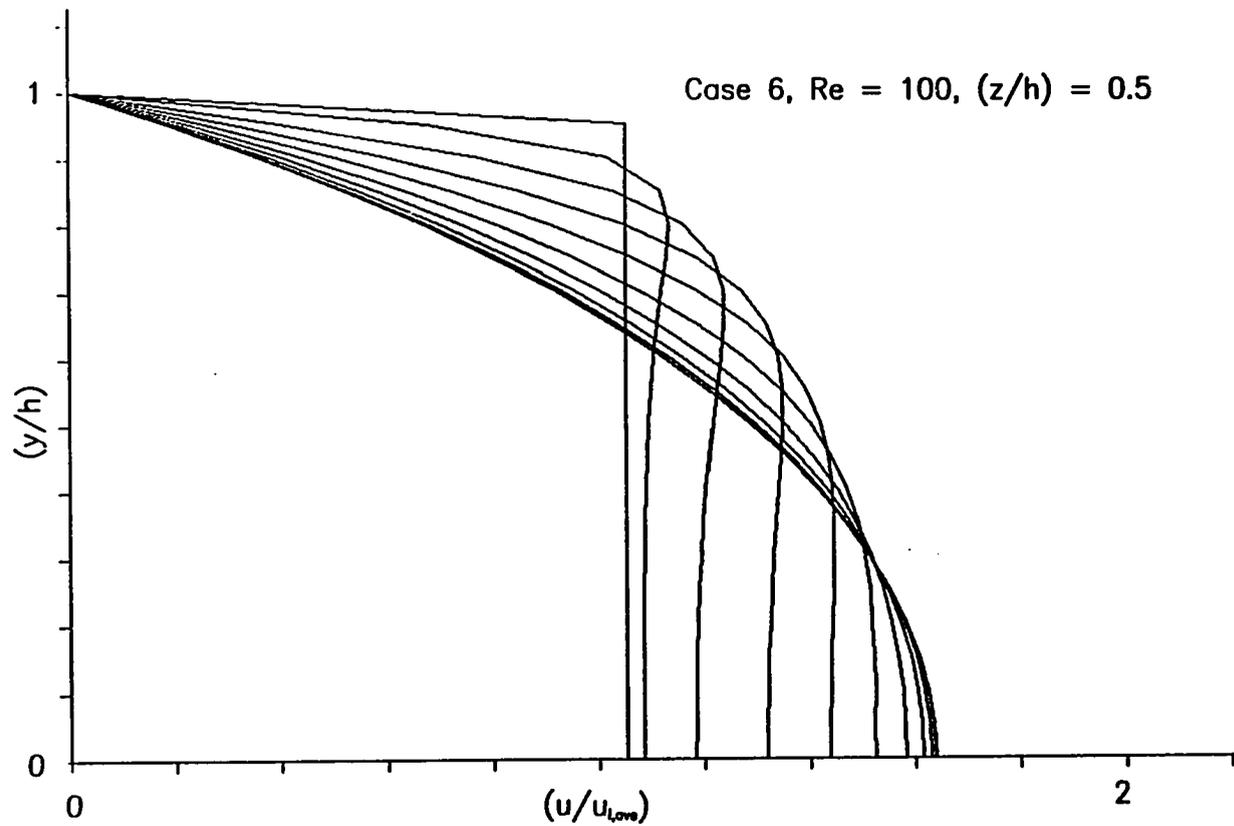


Figure 5.35. Computed axial velocity profiles at the quadrant mid-section for Case 6 at $Re = 100$

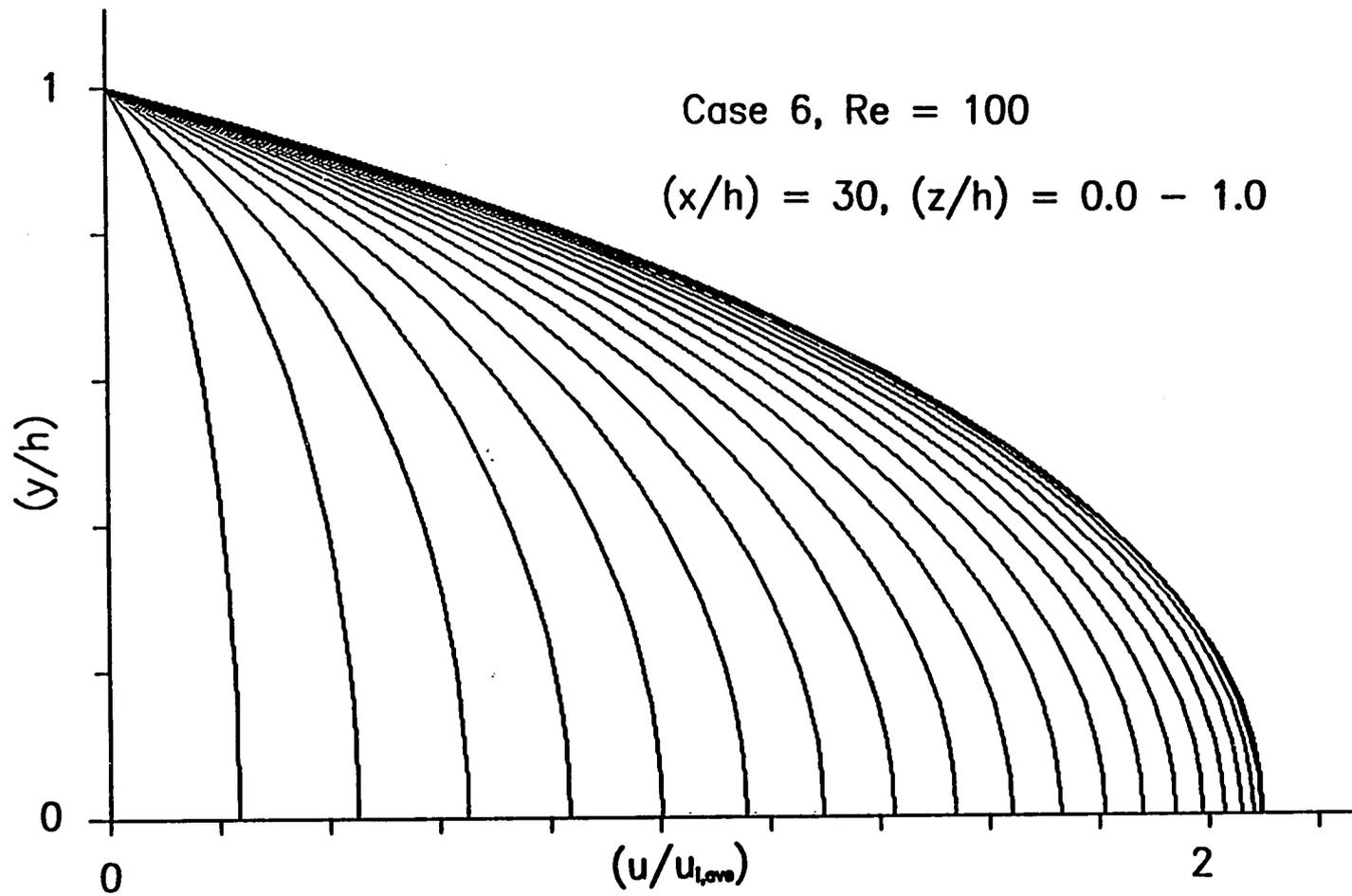


Figure 5.36. Computed fully-developed axial velocity profiles at the duct exit plane for Case 6 at $Re = 100$

Note: $v = w = -0.001$ at center of quadrant
Duct center at $y = z = 0$

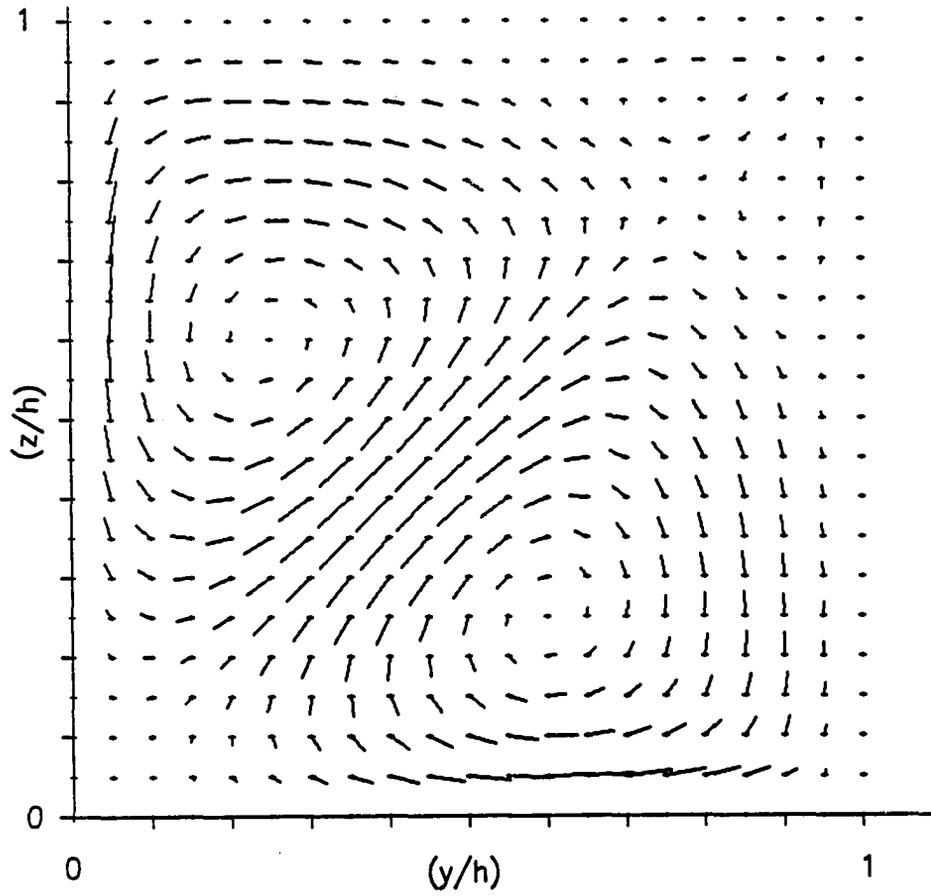


Figure 5.37. Computed secondary velocity vectors at the duct exit plane for Case 6 at $Re = 100$

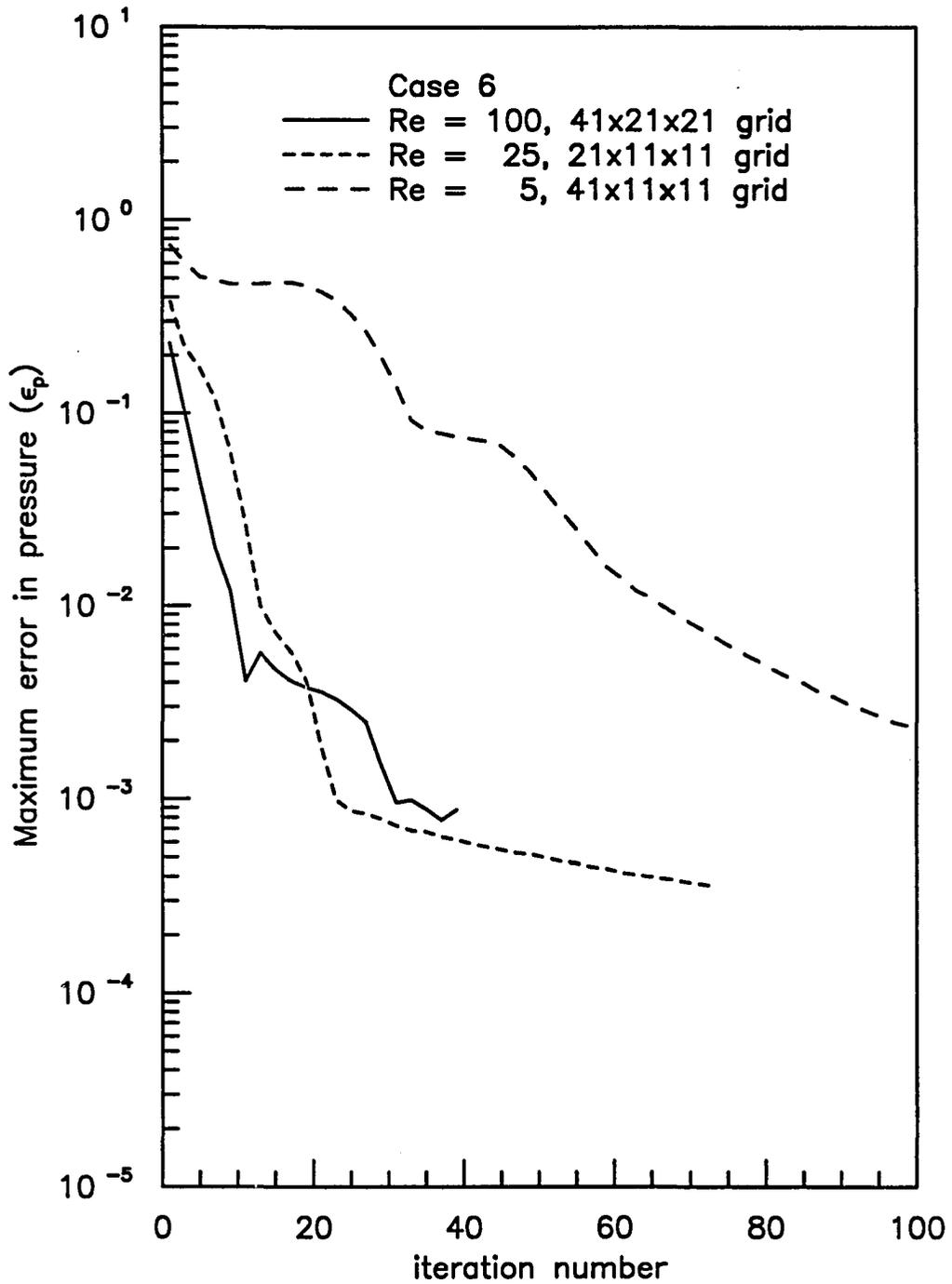


Figure 5.38. Convergence histories for Case 6 fine grid solutions

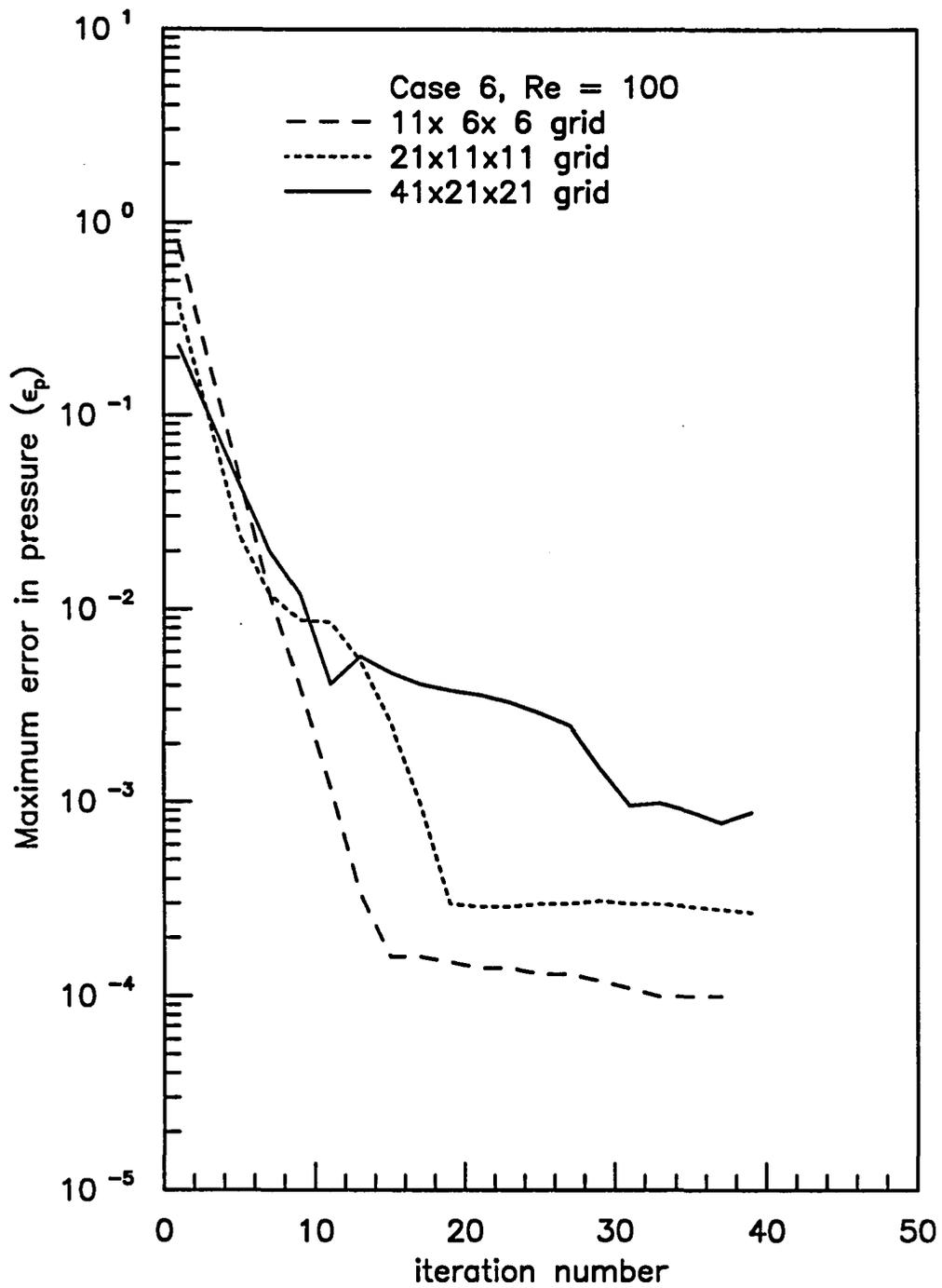


Figure 5.39. Convergence histories for Case 6 solutions at Re = 100

similar to that of Case 1, with a reduction in the convergence rate at low Reynolds numbers, and a plateau in the convergence rate at low error levels due to the residuals in the streamwise viscous terms.

G. Computational Requirements

The total computational time required to execute the coupled space-marching procedure is the time required to execute a single global marching sweep multiplied by the number of global iterations needed to reach a converged solution. These factors and the overall CPU time and memory requirements are discussed below.

1. Execution speed of the space-marching solver

The average processing speed for the present two-dimensional code is approximately 0.05 sec/node/iteration on an 8 MHz personal computer equipped with an 8087-2 coprocessor, 0.01 sec/node/iteration on an Apollo DN-4000 and 0.006 sec/node/iteration on a VAX 11/785. The respective speed for the present three-dimensional code is approximately 0.05 sec/node/iteration on an Apollo DN-4000. The nominal performance values given here are based on the conditions discussed below.

The computational execution time for a single space-marching sweep was found to depend upon the speed of the block tridiagonal matrix solver and adequate control of the tolerances for the Newton linearization loop. Because the algorithm globally sweeps the equations many times before reaching convergence, it was not necessary to converge the nonlinear coefficients during each marching sweep. For most of the problems it was only necessary to locally iterate the linearization loop during the first few marching sweeps and for the one or two stations where the local errors were greatest. Control of the tolerances on the linearization loop eliminates the unnecessary local iterations that could otherwise double or triple the execution time. The finite-difference formulation of the Navier-Stokes equations in generalized coordinates is rather complex. Still, the logic for evaluating the type dependent coefficients, linearizing the fluxes, assembling the coefficient matrix, plus execution of the global pressure correction procedure accounted for only ten to twenty percent of the total execution time. Thus, the overall speed of the coupled space-marching procedure was largely dependent upon the efficiency of the block tridiagonal solver.

To reduce execution time the general solver given in Anderson et al. [3] was recoded for the 3x3 or 4x4 block matrices used. In addition, the loops within the solver were explicitly expanded or "unrolled", rather than being expressed in general index notation. A 50% reduction in execution

time for the personal computer version of the code was achieved by making these rather simple modifications. This effect may not be as great on more sophisticated computers or with advanced compilers. However, from this example, it is apparent that reported CPU time is not a reliable measure of the computational efficiency of different numerical algorithms. That reservation aside, the processing speeds for the present FORTRAN77 codes are given above. However, the qualitative estimates discussed below are thought to be a better measure of the speed of the algorithm.

Specification of the dominant algebraic operations is a qualitative measure of the speed of the algorithm. For grids of average dimension "n" in each of the coordinate directions, the present two-dimensional algorithm requires the solution of one block tridiagonal matrix equation at each station, or roughly n matrix decompositions for each global iteration. The block size is 3x3 and one entry is included for each grid point at a given station. The present three-dimensional algorithm requires the solution of 2(n+n) block tridiagonal matrix equations for each station, or approximately $4n^2$ matrix decompositions for each global iteration. The matrices are n blocks in length with 4x4 block size. The operation count for decomposition of each matrix equation is proportional to the number of blocks times the square of the block dimension. Therefore the expected ratio of execution time per global iteration for the three-dimensional procedure compared to the two-dimensional method is 64/9, which is close to the ratio actually observed. Note that the present three-dimensional algorithm averages the solutions of two ADI steps at each station. Elimination of the averaging procedure would cut the execution time for the three-dimensional code in half. Using similar counts a three-dimensional time-marching procedure would require $3n^2$ decompositions per time step.

2. Global convergence behavior

Addition of the global pressure correction procedure to the iterative space-marching algorithm resulted in a significant reduction in the number of global iterations needed to obtain a converged solution. To demonstrate the accelerating effect, some of the Case 2a and Case 3 computations were repeated without using the global pressure correction scheme. Multiple space-marching sweeps were performed using only relaxation of the nodal pressure changes to stabilize and converge the solution. The convergence histories for coarse and fine grid solutions for Case 2a run with and without global pressure correction are shown in Figure 5.40. For the example shown, convergence was reached between four and ten times faster with the global pressure correction scheme, depending on the grid and the convergence tolerance chosen.

The convergence rate also depends upon the values of the diffusion parameter, α , and the relaxation factor, ω , used in the global pressure correction procedure. Preliminary studies to optimize the selection of these parameters have been carried out for Case 2a and Case 3. The required number of global iterations to reach convergence for various combinations of the diffusion and relaxation parameters are given in Tables 5.5 and 5.6, along with the iteration counts without global pressure correction. These results are also shown plotted in logarithmic coordinates on Figures 5.41 and 5.42. The linear relationship displayed at low values of ω for any fixed α , indicates that the pressure corrections are overdamped. The convergence rate increases as ω increases up to a point where the method becomes unstable. By examining the convergence trend as α is varied, it is observed that proper selection of α allows the use of larger relaxation factors. For the symmetric problem (Case 2a) there is little change in performance as long as the value of α is greater than a threshold value that assures stability. However, for the asymmetric problem (Case 3) the selection of too large a value of α reduces the convergence rate.

Based on limited experience with the method, the optimum values of these parameters appears to be problem dependent. Low values of α permit the pressure correction to become highly skewed from one side of the domain to the other, while high values of α produce a bulk streamwise pressure correction with little or no variation in the transverse direction. Further analysis of the procedure is needed to better understand these effects. For the cases presented above, the relaxation and diffusion parameters were held fixed as the calculations converged. The results of Bentson and Vradis [61] with a closely related pressure correction method suggest that cycling the values of the diffusion parameter during the course of the calculations may be beneficial. With the present algorithm the parameters are also constant over the entire grid. The convergence studies were limited to one type of uniform grid, and the effect of grid aspect ratio or skewness on convergence has not been evaluated. It is possible that spatial variation of the parameters, corresponding to local differences in the mesh would also be beneficial.

3. Overall CPU and memory requirements

It is difficult to make comparisons with other procedures for solving the Navier-Stokes equations due to differences in machine performance, convergence tolerances, and mesh resolution. But, a few comparisons with other published results are presented in Table 5.7. In a few instances,

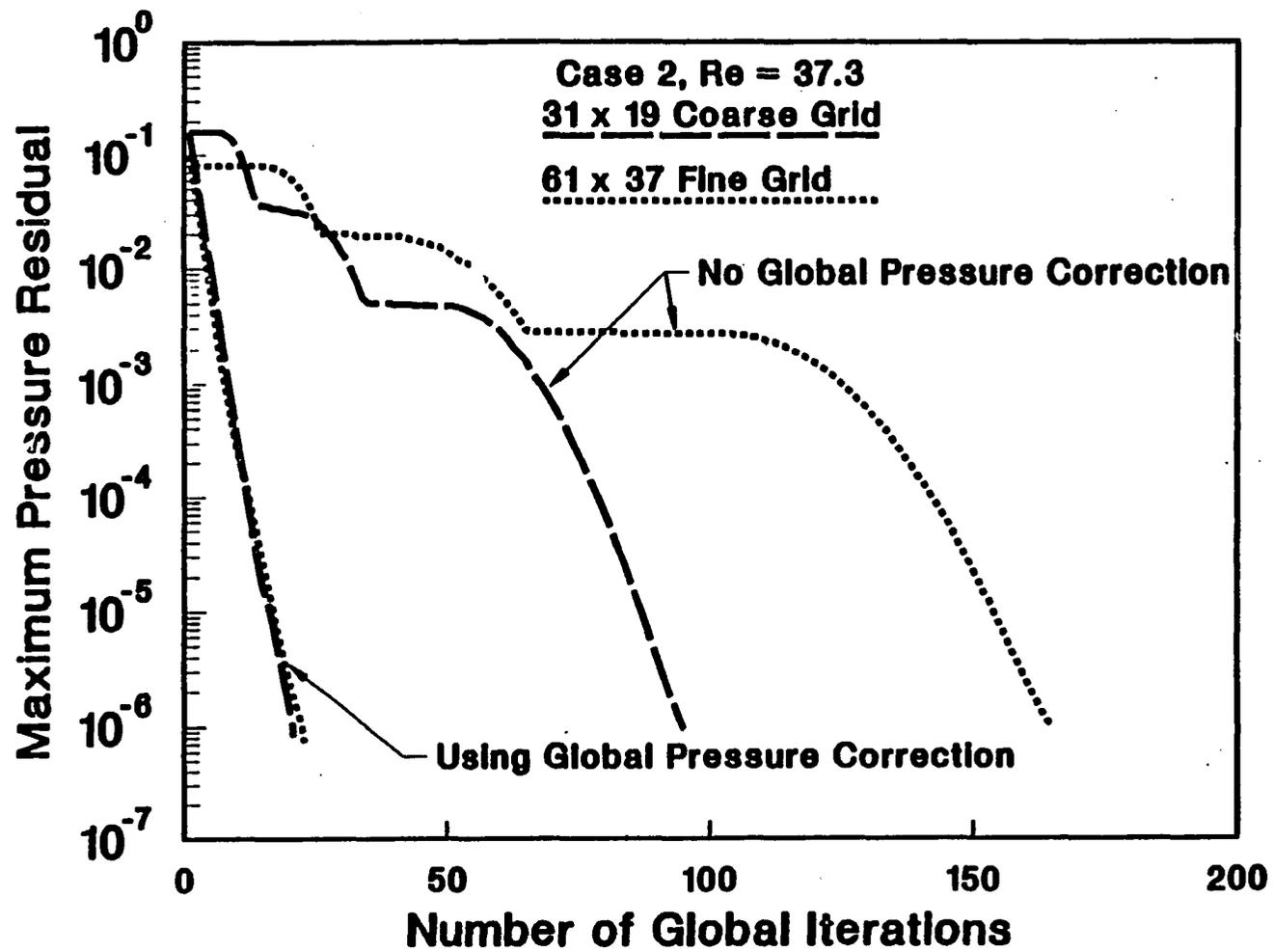


Figure 5.40. Effect of global pressure correction on convergence history for Case 2a

Table 5-5. Iterations required for Case 2a at Re = 56 and M = 0.05 on a 51x31 uniform grid

$\omega =$	0.10	0.25	0.35	0.50	0.707	0.84	0.92	1.00
$\alpha = 1$	125	59	236	*	*	*	*	*
$\alpha = 5$	116	43	48	*	*	*	*	*
$\alpha = 20$	112	42	29	22	*	*	*	*
$\alpha = 80$	113	42	30	21	17	25	38	194
$\alpha = 400$	115	43	31	22	18	25	38	157
relaxation	767	299	209	142	106	93	85	143

* Solution diverges.

Table 5-6. Iterations required for Case 3 at Re = 678 and M = 0.05 on a 46x31 uniform grid

$\omega =$	0.10	0.25	0.35	0.50	0.60	0.707	0.84
$\alpha = 10$	108	*	*	*	*	*	*
$\alpha = 20$	106	48	49	*	*	*	*
$\alpha = 40$	103	50	39	35	*	*	*
$\alpha = 80$	141	58	40	30	29	26	*
$\alpha = 120$	197	73	49	31	30	29	*
$\alpha = 200$	305	103	67	42	54	*	*
$\alpha = 400$	525	157	102	64	*	*	*
relaxation	2000+	508	231	145	145	*	*

* Solution diverges.

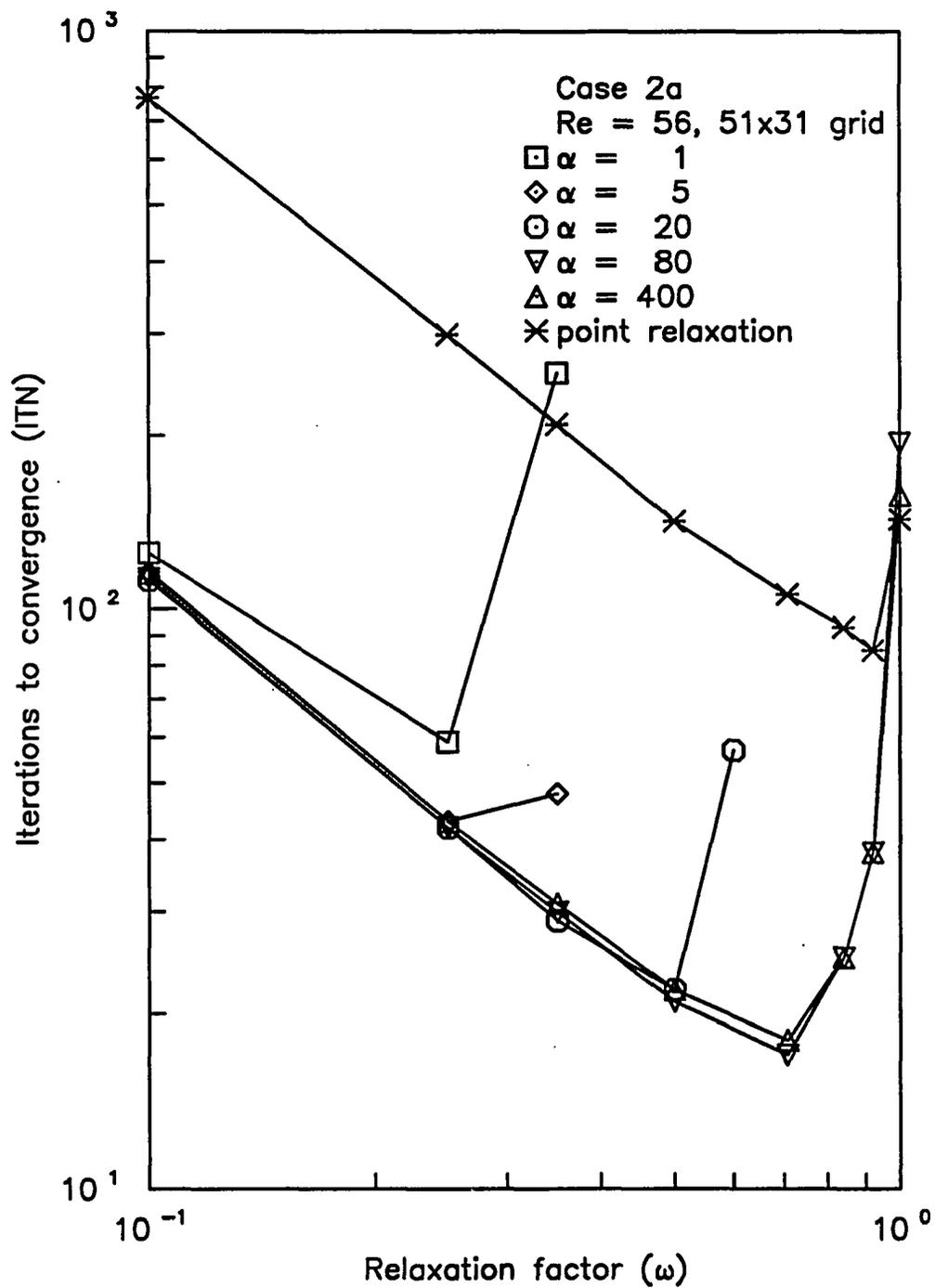


Figure 5.41. Effect of α and ω on convergence rate for Case 2a

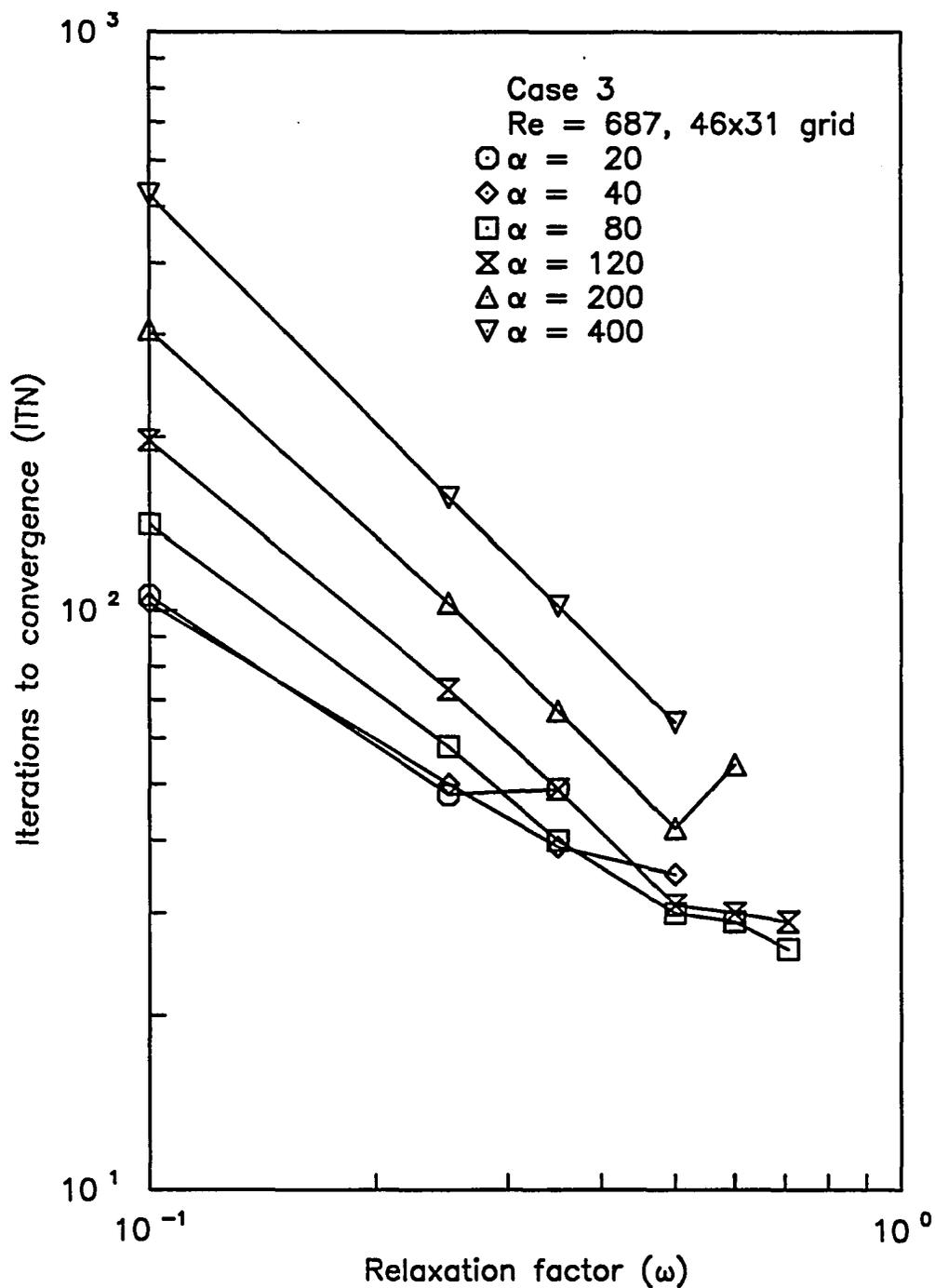


Figure 5.42. Effect of α and ω on convergence rate for Case 3

Table 5-7. Comparison of computer execution time for various methods

Method [Ref.]	Equations	grid size	global iterations	<u>CPU</u> node-iteration	CPU (sec)	Computer Model
Case 1, Re = 10						
C-S-M [58]	PPNS	uniform 50x40	100	0.01	1920	VAX 11/780
C-S-M present	PPNS	uniform 51x41	19	0.006	240	VAX
	NS	51x41	28	0.006	350	11/785
C-S-M present	PPNS	clustered 41x21	100	0.006	500	VAX
	NS	41x21	120	0.006	600	11/785
Case 1, Re = 50						
C-S-M [61]	PPNS	uniform 70x11	15	-	-	not given
	NS	70x11	40	-	-	
C-S-M present	PPNS	uniform 71x11	23	0.01	180	Apollo DN-4000
	NS	71x11	23	0.01	180	
Case 2a, Re = 56						
Seg. Eqn. [22]	NS	clustered 15x15	1000	0.006	1290	VAX 7550
C-S-M [89]	PNS	uniform 120x41	1	0.004	21	IBM 3081
C-S-M [58]	PPNS	uniform 25x30	60	0.01	400	VAX 11/780
C-S-M present	NS	uniform 31x19	21	0.006	65	VAX
		51x31	20	0.006	170	11/785
Case 5, Re = 40						
Seg. Eqn. [18]	NS	O-type 35x47	450	0.0004	324	not given
C-S-M present	NS	H-type 81x41	486	0.006	9600	VAX 11/785

additional runs have been made with the present algorithm to obtain results that are more directly comparable. Many uncontrolled factors exist so that it is not possible to make reliable judgments on the relative speed of the present method. Still, the results are encouraging. The coupled space-marching methods in general appear to show good convergence rates. Furthermore, the convergence rate is accelerated by the use of a global pressure step, as shown by the present results and the results of Bentson and Vradis [61]. The greater processing time per node for the coupled space-marching procedure of Liu and Pletcher [58] is thought to be due to additional eliminations in the matrix solver to couple the boundary conditions. Comparable data for the segregated, time-marching or dual potential algorithms were unfortunately not readily available.

The computer memory requirements for implementation of the iterative, space-marching algorithm are modest compared with general procedures for the solution of the Navier-Stokes equations. For example, the present two-dimensional, compressible, Navier-Stokes code with 81×41 array dimensions and 64-bit words (FORTRAN77 REAL*8 data type) can be executed on a personal computer with 640 kbytes of memory. Solution of the compressible Navier-Stokes equations in generalized coordinates requires full storage for each Cartesian coordinate of the grid points, each Cartesian velocity component, the temperature or density, and the pressure. A total of seven or nine arrays are needed for the two or three-dimensional algorithms respectively, as one additional pressure value was stored in the present implementation of the global pressure correction procedure. However, careful overwriting of the assumed pressure, calculated pressure, and pressure corrections would permit the use of only one pressure array. Also, for the adiabatic flows considered here, the total temperature assumption eliminates the requirement to store the temperature. Additional storage for frequently used variables such as the metrics is included in the two-dimensional code to avoid recomputing the values on each marching sweep.

For three-dimensional problems with refined grids the storage requirements can be quite large by present standards. Virtual memory systems permit convenient execution of very large problems on computers with limited core memory and available disk storage, however the use of large memory space is a consideration in the evaluation of the algorithm. The minimum memory requirements depend on the solution geometry and the degree to which the compressible Navier-Stokes equations are simplified. For example restricting the procedure to rectangular grids would eliminate the full arrays needed for the coordinate values. Incompressible flow or adiabatic flow assumptions reduce the need to store fluid properties. And most significantly, use of the PPNS equations with the FLARE

[64] approximation would not require full storage for the velocity components. Therefore it is possible to implement the algorithm for incompressible duct flows on simply constructed grids with one three-dimensional array for the pressure, plus the two-dimensional arrays needed to execute the coupled space-marching solution in the transverse plane.

Actual computational costs depend upon many variables, including execution time and memory requirements. The present results show promising execution performance, and the memory requirements for the procedure are not great. The overall computational cost of executing the present method is therefore expected to be less than or at least comparable to the cost for other current procedures.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. Concluding Remarks

In the present study, an efficient, coupled space-marching method has been developed for the numerical solution of the steady, compressible, Navier-Stokes equations in two and three dimensions. The accuracy of the finite-difference formulation has been verified for six incompressible flow test cases. The primary results of the investigation are summarized below:

1. On the development of the finite-difference form of the coupled space-marching method:

The coupled momentum equations and continuity equations are effective for directly determining the velocity components and the pressure. The Newton linearization procedure is stable and rapidly converges to the solution of the nonlinear equations. The linearization procedure is accurate for a single sweep marching method. However, since the present algorithm iterates the equations globally, local iteration to converge the coefficients of the linearized terms to high precision is not necessary.

The compressible formulation is shown to work in the incompressible limit. The solution procedure is also effective for purely incompressible fluid properties.

No dissipation terms are included in the finite-difference equations to stabilize the method. The second-order accurate approximations of the streamwise derivatives presented no apparent difficulties. However, one-sided difference expressions for the first-order derivatives in the transverse directions are needed to prevent the solution at even and odd nodes from decoupling.

Initial results with the three-dimensional procedure are encouraging. However, for fine grids in the transverse plane, the present ADI procedure for solving the coupled equations is not stable.

2. On the formulation in generalized coordinates:

The chain rule conservation law form simplifies the treatment of the metric terms at the cost of introducing small "source terms" due to truncation errors in the metric terms. This form was found to be suitable for the momentum equations, but computational tests of the continuity equation in chain rule form on nonorthogonal grids were not satisfactory. Therefore, the strong conservative law form is used for the continuity equation to strictly conserve mass.

3. On the formulation of boundary conditions:

For internal flow the specification of the normal and tangential velocity at the wall is straightforward. Ideally, a conservation of mass constraint for a control volume at the boundary

should be used to establish the pressure. With the present matrix solver, this specification of boundary conditions is practical for the upper boundary, but not the lower boundary. Therefore, the zero normal pressure gradient approximation of the momentum equations is used instead. The work by Liu and Pletcher [58] addresses this problem in more detail.

At a forward-facing step, the use of an additional continuity equation in place of the streamwise momentum equation allows continuation of the space-marching procedure through an abrupt contraction. The pressure profile is established by the transverse momentum equation.

4. On the addition of the global pressure correction step to the iterative space-marching algorithm:

For four of the six test cases, the backsweep pressure correction procedure greatly reduces the number of iterations required to obtain converged solutions. For these cases the convergence rate is not strongly dependent on the number of grid points. At low Reynolds numbers the convergence rate for solutions of the Navier-Stokes equations is less than the convergence rate for solutions of the PPNS equations. The convergence rate for solutions on highly clustered grids is lower than the convergence rate on uniform grids.

The sudden contraction and cylinder test cases require a significantly greater number of iterations to reach convergence. Comparison of the global pressure correction procedure with the simple pressure relaxation procedure has not yet been made for these cases. Initial trials of the simple relaxation method with moderate values of the relaxation factor did not converge, indicating that a low relaxation factor and a large number of iterations would be needed with the relaxation procedure.

5. On the laminar flow in the entrance of a two-dimensional channel:

For Reynolds numbers below 10, both the downstream velocity and pressure influence the flow upstream through the streamwise viscous stress and pressure gradient terms. The PPNS model is therefore not valid. For Reynolds numbers greater than 10, the upstream influence is only significant in terms of the pressure field.

6. On the laminar flow in channels with sudden expansions:

The reattachment length is well predicted, even on coarse grids and at high Reynolds numbers. This indicates that the second-order upwind model for the convective terms produces little numerical diffusion. Comparable calculations with the first-order upwind model significantly underpredict the reattachment length.

For moderate to large Reynolds numbers, the main core flow is reasonably well predicted by

the boundary-layer or PPNS equations. For Reynolds numbers on the order of 100, the wall shear stress distributions show that the parabolized models significantly overpredict the magnitude of the wall shear stress in the recirculation region. Asymmetric expansions also produce large transverse pressure gradients which may be difficult to predict with parabolized methods. For these conditions, the present Navier-Stokes procedure is effective over a broad range of Reynolds numbers.

7. On the laminar flow in channels with sudden contractions:

Large transverse velocities and extremely high pressure gradients are present in the vicinity of the step. A very thin separation region is predicted downstream of the step for Reynolds numbers greater than 200. The present results are not mesh independent but the trends in the predicted results are in accordance with the predictions of Mei and Plotkin [86]. The separation region upstream of the step is not well predicted with the present mesh.

8. On the laminar flow over a cylinder in crossflow:

The present predictions of the separation point and eddy length are in good agreement with the available experimental and numerical results. The surface pressure distribution is also correctly predicted, except in the vicinity of the leading edge stagnation point where the present grid is relatively coarse.

9. On the laminar flow in the entrance to a square duct:

The present predictions of the centerline velocity distribution are in good agreement with other numerical results. Local velocity "overshoots" are predicted near the inlet plane that are analogous to the predicted behavior for two-dimensional entrance flow. The computed transverse velocities show a weak secondary flow.

Based upon the results summarized above the following principal conclusions regarding the objectives of this research are drawn:

1. The present results, as well as the work of the previous investigators cited, confirm the ability of the multi-pass, coupled space-marching algorithm to capture the global influences that characterize elliptic flows. Use of the Navier-Stokes equation model extends the range of application of the procedure to the low Reynolds number region where parabolized models are inadequate. The finite-difference formulation expressed in generalized coordinates on a regular grid is computationally efficient and accurate.

2. The coupled procedure for the primitive variable form of the continuity and momentum equations allows a direct solution for the pressure, eliminating the need to solve a Poisson type

equation. The present choice of variables and application of Newton linearization are effective for solving the compressible equations in the incompressible limit. The procedure is also suitable for incompressible flow.

3. The second-order upwind model for the streamwise convective terms is accurate for the high Reynolds number laminar flows studied.

4. The finite-difference representation of first-order partial differential equations, such as the continuity equation, with central differences on a regular grid is likely to result in the decoupling of the solution values at even and odd nodes. Dissipation terms of one sort or another are needed to stabilize the solution. With space-marching algorithms, the streamwise terms are conveniently upwinded, but the transverse derivatives still present difficulties. In the present method, one-sided differences are successfully used; however, these terms introduce asymmetries into the formulation that complicate the procedures for imposing boundary conditions.

5. The new global pressure correction scheme greatly reduces the number of global marching sweeps needed to obtain a converged solution for an important class of problems. Compared with simple relaxation of the pressure change, the new procedure reduces the number of iterations by a factor of four to ten. However, for very low Reynolds number flows this convergence acceleration method has limited benefits, and acceleration techniques for the velocity field are also needed.

In general, the results of the present research show that the space-marching algorithm is ideally suited to simulating flows with a primary streamwise flow, such as internal flow in ducts with continuous changes in cross-section. Obstructions to the flow that produce extremely high streamwise curvature and severe pressure gradients create challenges to the application of the space-marching algorithm. Nonetheless, the present results demonstrate that the method is not fundamentally limited to approximately parabolic problems. Since the storage requirements and computational effort needed to execute the algorithm are moderate, application of the coupled space-marching algorithm to the simulation of complex flow in two or three dimensions is practical on the current generation engineering workstations.

B. Recommendations for Future Work

Most of the original objectives of this research have been achieved, but further study is needed in several areas to address unresolved problems. First, other algebraic procedures for solving the cross-plane equations for three-dimensional flows should be evaluated. The modified strongly

implicit procedure [60] used by Reddy and Rubin [59] or direct inversion methods are recommended. Secondly, use of one-sided difference methods to approximate the transverse derivatives lead to difficulties in the formulation of the boundary conditions. Furthermore, these expressions are an uncontrolled source of dissipation errors. Improvements to the finite-difference approximation of these terms should be considered. Thirdly, further study and optimization of the new global pressure correction procedure is recommended. Continued development of the procedure should examine methods for varying the parameters in the pressure correction procedure from iteration to iteration, or from point to point to account for non-uniform grids.

Several restrictions and simplifications were made at the very outset of this research. Since the results obtained to date are encouraging, it seems reasonable to re-evaluate the limitations, and attempt to generalize the procedure. There does not appear to be any fundamental reason that would prohibit straightforward extension of the procedure to include heat transfer or turbulent flow. However, the method of incorporating the additional equations in the algorithm deserves study. Methods with independent equations should be compared to fully-coupled solvers. Finally, generalization of the procedure to encompass subsonic, transonic and supersonic flow presents greater difficulties. The characteristic domains of dependence must be taken into account, and the current global pressure correction procedure is based on a linear operator. It is doubtful that the present method would be stable near sonic conditions without extensive modification.

Practical procedures for solving the Navier-Stokes are just now becoming widely available. There are many complex viscous flow phenomena involving flow separation and reattachment that are not well understood. The present procedure is an efficient simulation tool. It is therefore recommended that computational experiments of complex flows be undertaken to add to the basic understanding of viscous flow.

VII. REFERENCES

1. White, F. M. Viscous Fluid Flow. New York: McGraw-Hill Book Co., 1974.
2. Roache, P. J. Computational Fluid Dynamics. Albuquerque: Hermosa Publishers, 1972.
3. Anderson, D. A., Tannehill, J. C., and Pletcher, R. H. Computational Fluid Mechanics and Heat Transfer. New York: Hemisphere Publishing Corp., 1984.
4. Patankar, S. V. Numerical Heat Transfer and Fluid Flow. New York: Hemisphere Publishing Corp., 1980.
5. Minkowycz, W. J., Sparrow, E. M., Schneider, G. E., and Pletcher, R. H. Handbook of Numerical Heat Transfer. New York: John Wiley & Sons, Inc., 1988.
6. Aziz, K., and Hellums, J. D. "Numerical Solution of the Three-Dimensional Navier-Stokes Equations of Motion for Laminar Natural Convection." Physics of Fluids 10 (1967): 314-325.
7. Hiraski, G. J., and Hellums, J. D. "Boundary Conditions on the Vector and Scalar Potentials in Viscous Three-Dimensional Hydrodynamics." Quarterly of Applied Mathematics 28 (1970): 293-296.
8. Wong, A. J., and Reizes, J. A. "An Effective Vorticity-Vector Potential Formulation for Numerical Solution of Three-Dimensional Duct Flow Problems." J. Computational Physics 55 (1984): 98-114.
9. Gegg, S. G. "A Dual Potential Formulation of the Navier-Stokes Equations." Ph.D. dissertation, Iowa State University, 1989.
10. Harlow, F. H., and Welch, J. E. "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface." Physics of Fluids 8 (1965): 2182-2189.
11. Welch, J. E., Harlow, F. H., Shannon, J. P., and Daly, B. J. "The MAC Method." Los Alamos Scientific Laboratory Report LA-3425, Los Alamos, New Mexico, 1966.
12. Caretto, L. S., Gosman, A. D., Patankar, S. V., and Spalding, D. B. "Two Calculation Procedures for Steady, Three-Dimensional Flows with Recirculation." Third International Conference on Numerical Methods in Fluid Mechanics, pp. 60-68. New York: Springer-Verlag, 1972.
13. Ghia, K. N., Hankey Jr., W. L., and Hodge, J. K. "Use of Primitive Variables in the Solution of Incompressible Navier-Stokes Equations." AIAA Journal 17 (1979): 298-301.
14. Ghia, U., Ghia, K. N., Rubin, S. G., and Khosla, P. K. "Study of Incompressible Flow Separation Using Primitive Variables." Computers & Fluids 9 (1981): 123-142.
15. Patankar, S. V. "A Calculation Procedure for Two-Dimensional Elliptic Situations." Numerical Heat Transfer 4 (1981): 409-426.
16. Van Doormaal, J. P., Turan, A., and Raithby, G. D. "Evaluation of New Techniques for the Calculation of Internal Recirculating Flows." AIAA Paper AIAA-87-0059, 1987.

17. Rhie, C. M. "A Pressure Based Navier-Stokes Solver Using the Multigrid Method." AIAA Journal 27 (1989): 1017-1018.
18. Rhie, C. M. "A Numerical Study of the Flow Past an Isolated Airfoil with Separation." Ph.D. dissertation, University of Illinois, 1981.
19. Rhie, C. M., and Chow, W. L. "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation." AIAA Journal 21 (1983): 1525-1532.
20. Reggio, M., and Camerero, R. "Numerical Solution Procedure for Viscous Incompressible Flows." Numerical Heat Transfer 10 (1986): 131-146.
21. Karki, K. C., and Patankar, S. V. "Pressure Based Calculation Procedure for All Speeds in Arbitrary Configurations." AIAA Journal 27 (1989): 1167-1174.
22. Hobson, G., and Lakshminarayana, B. "Fully Elliptic Flow Calculations on a Regular Grid by a New Pressure Substitution Method." AIAA Paper AIAA-90-0239, 1990.
23. Vanka, S. P., and Leaf, G. K. "Fully Coupled Solution of Pressure-Linked Fluid Flow Equations." Argonne National Laboratory Report ANL-83-73, 1983.
24. Patankar, S. V., Karki, K. C., and Mongia, H. C. "Development and Evaluation of Improved Numerical Schemes for Recirculating Flows." AIAA Paper AIAA-87-0061, 1987.
25. Bender, E. E., and Khosla, P. K. "Solution of the Two-Dimensional Navier-Stokes Equations using Sparse Matrix Solvers." AIAA Paper AIAA-87-0603, 1987.
26. MacCormack, R. W. "The Effect of Viscosity in Hypervelocity Cratering." AIAA Paper AIAA-69-345, 1969.
27. Beam, R. M., and Warming, R. K. "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation-Law Form." J. Computational Physics 22 (1976): 87-110.
28. Beam, R. M., and Warming, R. K. "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations." AIAA Journal 16 (1978): 393-401.
29. Briley, W. R., and McDonald, H. "On the Structure and Use of Linearized Block Implicit Schemes." J. Computational Physics 34 (1980): 54-73.
30. Chima, R. V., and Johnson, G. M. "Efficient Solution of the Euler and Navier-Stokes Equations with a Vectorized Multi-Grid Algorithm." AIAA Paper AIAA-83-1893, 1983.
31. Holst, T. L. "Numerical Solution of the Navier-Stokes Equations about Three-Dimensional Configurations." In Supercomputing in Aerospace, pp. 281-298, NASA CP 2454, 1987.
32. Briley, W. R., McDonald, H., and Shamroth, S. J. "A Low Mach Number Euler Formulation and Application to Time-Iterative LBI Schemes." AIAA Journal 21 (1983) 1467-1469.
33. Chorin, A. J. "A Numerical Method for Solving Incompressible Viscous Flow Problems." J. Computational Physics 2 (1967): 12-26.
34. Kwak, D., Chang, J. L. C., Shanks, S. P., and Chakravarthy, S. R. "An Incompressible Navier-Stokes Flow Solver in Three-Dimensional Curvilinear Coordinate System Using Primitive Variables." AIAA Journal 24 (1986): 390-396.

35. Benocci, C., and Ceresola, N. "Solution of the Incompressible Navier-Stokes Equations with the Approximate Factorization Technique." VKI Technical Memorandum 39, 1985.
36. Hartwich, P. M., and Hsu, C.-H. "High Resolution Upwind Schemes for the Three-Dimensional, Incompressible Navier-Stokes Equations." AIAA Paper AIAA-87-0547, 1987.
37. Kwon, O. K., Pletcher, R. H., and Lewis, J. P. "Prediction of Sudden Expansion Flows Using the Boundary-Layer Equations." J. Fluids Engineering 106 (1984): 285-291.
38. Patankar, S. V., and Spalding, D. B. "A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows." Int. J. Heat Mass Transfer 15 (1972): 1787-1806.
39. Briley, W. R. "Numerical Method For Predicting Three-Dimensional Steady Viscous Flow in Ducts." J. Computational Physics 14 (1974): 8-28.
40. Ghia, U., Ghia, K. N., and Stauder, C. J. "Three Dimensional Laminar Incompressible Flow in Straight and Polar Ducts." Computers and Fluids 5, (1977): 205-218.
41. Roberts, D. W., and Forrester, C. K. "Parabolic Procedure for Flows in Ducts with Arbitrary Cross-Sections." AIAA Journal 17 (1979): 33-40.
42. Maliska, C. R. "A Solution Method for Three-Dimensional Parabolic Fluid Flow Problems in Nonorthogonal Coordinates." Ph.D. dissertation, University of Waterloo, 1981.
43. Katsanis, T. "Calculation of Three-Dimensional, Viscous Flow Through Turbomachinery Blade Passages by Parabolic Marching." NASA TM 86984, 1985.
44. Rudman, S., and Rubin, S. G. "Hypersonic Viscous Flow over Slender Bodies with Sharp Leading Edges." AIAA Journal 6 (1968): 1883-1889.
45. Lubard, S. C., and Helliwell, W. S. "Calculation of the Flow on a Cone at High Angle of Attack." AIAA Journal 12 (1974): 965-974.
46. Vigneron, Y. C., Rakich, J. V., and Tannehill, J. C. "Calculation of Supersonic Viscous Flow over Delta Wings with Sharp Subsonic Leading Edges." NASA TM 78500, 1978.
47. Davis, R. T., and Rubin, S. G. "Non-Navier-Stokes Viscous Flow Computations." Computers and Fluids 8 (1980): 101-131.
48. Lawrence, S. L., Chausee, D. S., and Tannehill, J. C. "Application of an Upwind Algorithm to the Parabolized Navier-Stokes Equations." AIAA Journal 27 (1989): 1175-1187.
49. Prattrap, V. S., and Spalding, D. B. "Fluid Flow and Heat Transfer in Three-Dimensional Duct Flows." Int. J. Heat Mass Transfer 19 (1976): 1183-1188.
50. Moore, J., and Moore, J. G. "A Calculation Procedure for Three-Dimensional Viscous, Compressible Duct Flow, Parts I and II." J. Fluids Engineering 101 (1979): 415-423.
51. Chilukuri, R., and Pletcher, R. H. "Numerical Solution to the Partially Parabolized Navier-Stokes Equations for Developing Flow in a Channel." Numerical Heat Transfer 3 (1980): 169-188.

52. Rubin, S. G., and Reddy, D. R. "Analysis of Global Pressure Relaxation for Flows with Strong Interaction and Separation." Computers and Fluids 11 (1983): 281-306.
53. Israeli, M., and Lin, T. "Iterative Numerical Solutions and Boundary Conditions for the Parabolized Navier-Stokes Equations." Computers and Fluids 13 (1985): 397-410.
54. Brown, J. L. "Parabolized Navier-Stokes Solutions of Separation and Trailing Edge Flows." NASA TM 84378, 1983.
55. Himansu, A., and Rubin, S. G. "Multigrid Acceleration of a Relaxation Procedure for the RNS Equations." AIAA Paper AIAA-87-1145, 1987.
56. Govindan, T. R., "A Space-Marching Method for the Navier-Stokes Equations for Internal Flows." Ph.D. dissertation, Pennsylvania State University, 1983.
57. Pougare, M., and Lakshminarayana, B. "A Space-Marching Method for Incompressible Navier-Stokes Equations." AIAA Paper AIAA-85-0170, 1985.
58. Liu, X., and Pletcher, R. H. "A Coupled Marching Procedure for the Partially Parabolized Navier-Stokes Equations." Numerical Heat Transfer 10 (1986): 539-556.
59. Reddy, D. R., and Rubin, S. G. "Consistent Boundary Conditions for Reduced Navier-Stokes (RNS) Scheme Applied to Three-Dimensional Internal Viscous Flow." NASA CR 180874, 1988.
60. Schneider, G. E., and Zedan, M. "A Modified Strongly Implicit Procedure for the Numerical Solution of Field Problems." Numerical Heat Transfer 4 (1981): 1-19.
61. Bentson, J., and Vradis, G. "A Two-Stage Pressure Correction Technique for the Incompressible Navier-Stokes Equations." AIAA Paper AIAA-87-0545, 1987.
62. TenPas, P. W., and Pletcher, R. H. "Solution of the Navier-Stokes Equations for Subsonic Flows using a Coupled Space-Marching Method." AIAA Paper AIAA-87-1173, 1987.
63. Schiff, L. B., and Steger, J. L. "Numerical Simulation of Steady Supersonic Viscous Flow." AIAA Journal 12 (1980): 1421-1430.
64. Reyhner, T. A., and Flugge-Lotz, I. "The Interaction of a Shock Wave with a Laminar Boundary Layer." Int. J. Nonlinear Mechanics 3 (1968): 173-199.
65. Hindman, R. G. "Generalized Coordinate Forms of Governing Fluid Equations and Associated Geometrically Induced Errors." AIAA Journal 20 (1982): 1359-1367.
66. Shamroth, S. J., and Gibelg, H. J. "A Compressible Solution for the Navier-Stokes Equations for Turbulent Flow About an Airfoil." NASA CR 3183, 1979.
67. Mitchell, A. R., and Griffiths, D. F. The Finite-Difference Method in Partial Differential Equations. New York: John Wiley & Sons, 1980.
68. Barnett, M., and Davis, R.T. "A Procedure for the Calculation of Supersonic Flows with Strong Viscous-Inviscid Interaction." AIAA Paper AIAA-85-0166, 1985.

69. Bodoia, J. R., and Osterle, J. F. "Finite Difference Analysis of Plane Poiseuille and Couette Flow Development." Applied Scientific Research 10, Section A (1961): 265-276.
70. McDonald, J. W., Denny, V. E., and Mills, A. F. "Numerical Solutions of the Navier-Stokes Equations in Inlet Regions." J. Applied Mechanics 39, Series E (1972): 873-878.
71. Morihara, H., and Cheng, R. T. "Numerical Solution of the Viscous Flow in the Entrance Region of Parallel Plates." J. Computational Physics 11 (1973): 550-572.
72. Van Dyke, M. "Entry Flow in a Channel." J. Fluid Mechanics 44 (1970): 813-823.
73. Durst, F., Melling, A., and Whitelaw, J. H. "Low Reynolds Number Flow over a Plane Symmetric Sudden Expansion." J. Fluid Mechanics 64 (1974): 111-128.
74. Osswald, G. A., Ghia, K. N., and Ghia, U. "Unsteady Navier-Stokes Simulation of Internal Separated Flows over Plane and Axisymmetric Sudden Expansions." AIAA Paper AIAA-84-1584, 1984.
75. Kwon, O. K., Pletcher, R. H., and Lewis, J. P. "Prediction of the Incompressible Flow over a Rearward Facing Step." Technical Report No. HTL-26, CFD-4, ISU-ERI-Ames-82019, Engineering Research Institute, Iowa State University, 1981.
76. Lewis, J. P., and Pletcher, R. H. "Limitations of the Boundary-Layer Equations for Predicting Laminar Symmetric Sudden Expansion Flows." J. Fluids Engineering 108 (1986): 208-213.
77. Chiu, I.-T. "Prediction of Laminar Flows over a Rearward-Facing Step Using the Partially Parabolized Navier-Stokes Equations." M.S. Thesis, Iowa State University, 1984.
78. Napolitano, M., and Cinnella, P. "A Numerical Study of Planar and Axially-Symmetric Sudden Flows." Computers and Fluids 17 (1989): 185-193.
79. Kwon, O. K. "Prediction of the Incompressible Flow over a Rearward-Facing Step." Ph.D. dissertation, Iowa State University, 1981.
80. Kwon, O. K., and Pletcher, R. H. "A Viscous-Inviscid Interaction Procedure - Part 1: Method for Calculating Two-Dimensional Incompressible Separating Flows." J. Fluids Engineering 108 (1986): 64-71.
81. Denham, M. K., and Patrick, M. A. "Laminar Flow over a Downstream-Facing Step in a Two-Dimensional Channel." Trans. Institute of Chemical Engineers 52 (1974): 361-367.
82. Hackman, L. P., Raithby, G. D., and Strong, A. B. "Numerical Prediction of Flow over Backward-Facing Steps." Int. J. Numerical Methods in Fluids 4 (1984): 711-724.
83. Armaly, B. F., Durst, F., Pereira, J. C. F., and Schonung, B. "Experimental and Theoretical Investigation of Backward-Facing Step Flow." J. Fluid Mechanics 127 (1983): 473-496.
84. Durst, F., Schierholz, W. F., and Wunderlich, A. M. "Experimental and Numerical Investigations of Plane Duct Flows with Sudden Contraction." J. Fluids Engineering 109 (1987): 376-383.
85. Dennis, S. C. R., and Smith, F. T. "Steady Flow Through a Channel with a Symmetrical Constriction in the Form of a Step." Proceedings of the Royal Society of London series A 372 (1980): 393-414.

86. Mei, R. W., and Plotkin, A. "Navier-Stokes Solutions for Some Laminar Incompressible Flows with Separation in Forward Facing Step Geometries." AIAA Paper AIAA-86-0110, 1986.
87. Coutanceau, M., and Bouard, R. "Experimental Determination of the Main Features of Viscous Flow in the Wake of a Cylinder in Uniform Translation." J. Fluid Mechanics 79 (1977): 231-256.
88. Son, J. S., and Hanratty, T. J. "Numerical Solution for the Flow Around a Circular Cylinder at Reynolds Number of 40, 200, and 500." J. Fluid Mechanics 35 (1969): 369-386.
89. Kirtley, K. R., and Lakshminarayana, B. "Computation of Internal Incompressible Flows Using a Space-Marching Technique." AIAA Paper AIAA-85-1624, 1985.

VIII. ACKNOWLEDGMENTS

The author wishes to express his abiding gratitude to Dr. Richard H. Pletcher, his thesis advisor and committee chairman, for his endless stream of ideas and unflagging enthusiasm for the subject. His continual support and unwavering encouragement during the development and completion of this thesis is deeply appreciated.

The author is indebted to his doctoral committee, Dr. William J. Cook, Dr. David K. Holger, Dr. Patrick Kavanaugh, Dr. Joseph M. Prusa, and Dr. John C. Tannehill, for their guidance and cooperation during the course of the study.

The financial support from the National Science Foundation for sponsoring the research under grant CBT-8211713 is gratefully acknowledged.

The author would like to thank the staff of the Department of Mechanical Engineering of the University of Kansas for their assistance in the production of this report.

The author is grateful to his parents and sisters for fostering his interest in graduate study and the inspiration to undertake this endeavor.

Finally, the author wishes to thank his wife, Deborah, and daughter, Heidi, for their endless patience, sympathetic understanding and unconditional love, which sustained him throughout these difficult yet rewarding years.

IX. APPENDIX A: DEFINITION OF THE TRANSFORMATION METRICS

To conveniently form finite-difference expressions for non-uniform grids, the Cartesian coordinate equations are transformed from the physical space to the computational coordinate space. The scaling parameters introduced into the equations as a result of the transformation are called metrics. As a consequence of requiring a one-to-one correspondence between the physical space and the computational space the metrics of the transformation are defined as

$$\begin{aligned} \frac{\xi_x}{J} &= (y_\eta z_\zeta - y_\zeta z_\eta) & \frac{\xi_y}{J} &= -(x_\eta z_\zeta - x_\zeta z_\eta) & \frac{\xi_z}{J} &= (x_\eta y_\zeta - x_\zeta y_\eta) \\ \frac{\eta_x}{J} &= -(y_\xi z_\zeta - y_\zeta z_\xi) & \frac{\eta_y}{J} &= (x_\xi z_\zeta - x_\zeta z_\xi) & \frac{\eta_z}{J} &= -(x_\xi y_\zeta - x_\zeta y_\xi) \\ \frac{\zeta_x}{J} &= (y_\xi z_\eta - y_\eta z_\xi) & \frac{\zeta_y}{J} &= -(x_\xi z_\eta - x_\eta z_\xi) & \frac{\zeta_z}{J} &= (x_\xi y_\eta - x_\eta y_\xi) \end{aligned}$$

The Jacobian is given by

$$J = [x_\xi (y_\eta z_\zeta - y_\zeta z_\eta) - x_\eta (y_\xi z_\zeta - y_\zeta z_\xi) + x_\zeta (y_\xi z_\eta - y_\eta z_\xi)]^{-1}$$

For algebraically constructed grids the metrics can be determined by analytic differentiation. However, a closed-form, functional relationship for the grid coordinates typically does not exist. Therefore, with the present method the metrics are evaluated by finite-difference approximations of the partial derivatives given above. The procedure has two steps:

1. The partial derivatives of the Cartesian coordinates with respect to the computational coordinates are calculated using central differences. At boundaries, appropriate symmetry conditions or one-sided differences are used where needed.
2. The metric terms and the Jacobian are evaluated using the partial derivatives obtained in step 1.

Notice that the terms that appear most frequently in the chain-rule-conservation-law form of the momentum equation are those given above. Also notice that many authors define the Jacobian as the reciprocal of the value given above.

In the two-dimensional code, the four metric terms and the Jacobian are calculated once during the grid generation process and stored. To improve the accuracy of the numerical evaluation of the metric terms on highly stretched grids, additional grid points are initially generated at the

half-step positions between node points. These values, at plus or minus one half-step about the node point, are used to evaluate the metrics by a central-difference approximation of the first-derivatives.

To reduce the memory requirements for executing the three-dimensional computer code, the metrics are evaluated at each station during each marching sweep. Since metric terms upstream and downstream of the solution station are needed for the approximation of the viscous terms, thirty arrays are used to temporarily store the nine metric terms and the Jacobian for three cross-sectional stations. The first derivatives are evaluated numerically with central differences using the Cartesian coordinates of the neighboring nodes.

X. APPENDIX B: TRANSFORMATION OF THE VISCOUS STRESSES

Derivatives of the viscous stresses appear in the momentum equations. Also, the stress terms themselves are comprised of derivatives. Therefore, the chain rule must be applied repeatedly to express the viscous stresses in terms of the generalized coordinates. The general three-dimensional equations are quite lengthy, so the general two-dimensional form is presented for clarity. The constant property three-dimensional terms is developed at the end of this section.

As an example the viscous terms in the two-dimensional x-momentum equation are

$$-\left[\frac{\partial}{\partial x} (\tau_{xx}) + \frac{\partial}{\partial y} (\tau_{xy}) \right]$$

Transformation of these terms to the chain-rule-conservation-law form in the generalized coordinates gives

$$-\left[\xi_x \frac{\partial}{\partial \xi} (\tau_{xx}) + \eta_x \frac{\partial}{\partial \eta} (\tau_{xx}) + \xi_y \frac{\partial}{\partial \xi} (\tau_{xy}) + \eta_y \frac{\partial}{\partial \eta} (\tau_{xy}) \right]$$

Before the partial derivatives can be approximated by finite-differences, the viscous stresses must be evaluated. For laminar flow of a Newtonian fluid, the viscous stresses in dimensionless form are given by Equation 2.2a. In two dimensions the stresses reduce to

$$\begin{aligned} \tau_{xx} &= \frac{2}{3} \frac{\mu}{\text{Re}} \left(2 \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right) \\ \tau_{yy} &= \frac{2}{3} \frac{\mu}{\text{Re}} \left(2 \frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} \right) \\ \tau_{xy} &= \tau_{yx} = \frac{\mu}{\text{Re}} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \end{aligned}$$

The chain rule is applied to the partial derivatives of the velocity to obtain the generalized coordinate form. As an example the x-direction normal stress term is expressed as

$$\tau_{xx} = (2/3) \mu [2 (\xi_x u_\xi + \eta_x u_\eta) + (\xi_y v_\xi + \eta_y v_\eta)] / \text{Re}$$

Each of the viscous stress terms is transformed in a similar manner. The derivatives of the transformed stress terms are then expanded with the chain rule. In the general case the viscosity and the metric terms are variables. In the present procedure the viscosity and metric terms are lumped

together and differentiated as one term. The completed transformation of the viscous terms for the two-dimensional x-momentum equation is

$$\begin{aligned} \frac{\partial}{\partial \xi} (\tau_{xx}) &= (4/3) [\mu_{\xi x}^{\xi} u_{\xi\xi} + u_{\xi} (\mu_{\xi x}^{\xi})_{\xi} + \mu \eta_x u_{\eta\xi} + u_{\eta} (\mu \eta_x)_{\xi}] / \text{Re} \\ &\quad - (2/3) [\mu_{\xi y}^{\xi} v_{\xi\xi} + v_{\xi} (\mu_{\xi y}^{\xi})_{\xi} + \mu \eta_y v_{\eta\xi} + v_{\eta} (\mu \eta_y)_{\xi}] / \text{Re} \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial \eta} (\tau_{xx}) &= (4/3) [\mu_{\xi x}^{\eta} u_{\xi\eta} + u_{\xi} (\mu_{\xi x}^{\eta})_{\eta} + \mu \eta_x u_{\eta\eta} + u_{\eta} (\mu \eta_x)_{\eta}] / \text{Re} \\ &\quad - (2/3) [\mu_{\xi y}^{\eta} v_{\xi\eta} + v_{\xi} (\mu_{\xi y}^{\eta})_{\eta} + \mu \eta_y v_{\eta\eta} + v_{\eta} (\mu \eta_y)_{\eta}] / \text{Re} \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial \xi} (\tau_{xy}) &= [\mu_{\xi y}^{\xi} u_{\xi\xi} + u_{\xi} (\mu_{\xi y}^{\xi})_{\xi} + \mu \eta_y u_{\eta\xi} + u_{\eta} (\mu \eta_y)_{\xi}] / \text{Re} \\ &\quad + [\mu_{\xi x}^{\xi} v_{\xi\xi} + v_{\xi} (\mu_{\xi x}^{\xi})_{\xi} + \mu \eta_x v_{\eta\xi} + v_{\eta} (\mu \eta_x)_{\xi}] / \text{Re} \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial \eta} (\tau_{xy}) &= [\mu_{\xi y}^{\eta} u_{\xi\eta} + u_{\xi} (\mu_{\xi y}^{\eta})_{\eta} + \mu \eta_y u_{\eta\eta} + u_{\eta} (\mu \eta_y)_{\eta}] / \text{Re} \\ &\quad + [\mu_{\xi x}^{\eta} v_{\xi\eta} + v_{\xi} (\mu_{\xi x}^{\eta})_{\eta} + \mu \eta_x v_{\eta\eta} + v_{\eta} (\mu \eta_x)_{\eta}] / \text{Re} \end{aligned}$$

The terms for the two-dimensional y-momentum equations are derived in a similar manner.

The complete expansion of the viscous terms in three dimensions follows the two-dimensional procedure given above. For each momentum equation, the expanded derivatives are collected in terms of the partial derivatives of the velocity components. The derivatives of the metrics and viscosity are evaluated to determine the coefficients for the nine partial-derivatives (3 second-order, 3 mixed and 3 first-order) of each of the three velocity components. For the variable property equations there are 27 terms in each equation, or total of 81 composite terms.

In the present three-dimensional computer code, the constant property form of viscous term is used. This greatly reduces the complexity of the formulation. The constant property form of the viscous term in the x-momentum equation is

$$-(\mu/\text{Re}) \nabla^2 u = -(\mu/\text{Re}) (u_{xx} + u_{yy} + u_{zz})$$

The chain rule is applied twice in succession to transform the second-derivatives. For example

$$\begin{aligned}
u_{xx} &= \frac{\partial}{\partial x} (\xi_x u_\xi + \eta_x u_\eta + \zeta_x u_\zeta) \\
&= \xi_x [\xi_x u_{\xi\xi} + u_\xi (\xi_x)_\xi + \eta_x u_{\eta\xi} + u_\eta (\eta_x)_\xi + \zeta_x u_{\zeta\xi} + u_\zeta (\zeta_x)_\xi] \\
&\quad + \eta_x [\xi_x u_{\xi\eta} + u_\xi (\xi_x)_\eta + \eta_x u_{\eta\eta} + u_\eta (\eta_x)_\eta + \zeta_x u_{\zeta\eta} + u_\zeta (\zeta_x)_\eta] \\
&\quad + \zeta_x [\xi_x u_{\xi\zeta} + u_\xi (\xi_x)_\zeta + \eta_x u_{\eta\zeta} + u_\eta (\eta_x)_\zeta + \zeta_x u_{\zeta\zeta} + u_\zeta (\zeta_x)_\zeta]
\end{aligned}$$

The other second-derivatives are expanded in the same manner. The common derivatives are collected so that the x-momentum equation term contains just the nine partial derivatives of the x-component of velocity. An identical procedure is followed for the y-momentum and z-momentum equations. The coefficients of the velocity derivatives include derivatives of the metrics. The burden of determining the coefficients is lightened somewhat, as the coefficients of corresponding terms are identical for all three momentum equations.

XI. APPENDIX C: BOUNDARY CONDITIONS AT A FORWARD-FACING STEP

In general, the primitive variables at the boundary nodes are determined from either predetermined specifications, which are part of the problem statement, or by the solution of equations for the proper boundary conditions. The different types of boundary conditions are discussed in Chapter IV. The details of the formulation and coupled space-marching solution of the boundary equations at a forward-facing step are presented below.

At a typical internal marching station, downstream values are included in the momentum equations. However, since the continuity control volumes are shifted upstream, no downstream terms appear in the continuity equations. For the special station located just upstream of a forward-facing step, the set of coupled equations must be modified to recognize the presence of the blockage downstream. Otherwise, at this station (labeled ISTP-1) the solution will indicate a net flow across the surface facing the step. At the next station downstream (labeled ISTP) the velocity components on the step face are specified equal to zero. Thus, the solution at station ISTP-1 would be inconsistent with the boundary values at station ISTP, and the global conservation of mass constraint would not be met.

Therefore, a continuity constraint for the computational cells located on the forward-facing step face, downstream of station ISTP-1, must be included in the set of coupled equations solved at station ISTP-1. This is done by replacing the x-momentum equation at all nodes facing the step with the continuity equation for the cells at the step face. The additional continuity equation includes implicit terms for the velocities at station ISTP-1, without introducing any unknown values at station ISTP. The streamwise faces of the additional control volumes are located at station ISTP-1 and station ISTP. The streamwise flux terms at station ISTP-1 are identical to the corresponding terms for the control volumes upstream of station ISTP-1, and the streamwise flux terms at the step face are specified equal to zero. Linear velocity profiles are assumed to approximate the flow through the transverse faces.

The coupled equations for each node facing the step along station ISTP-1 are thus, 1) the additional continuity equation for the downstream cell at the step face, 2) the y-momentum equation and 3) the continuity equation for the upstream cell located between stations ISTP-2 and ISTP-1. The equations for the nodes at station ISTP-1 that do not directly face the step are not altered, and the side-wall boundary equations at station ISTP-1 are not changed. The formulation for the internal nodes at station ISTP is also unchanged. However, a mass source equal to the net transverse flow

from the cells along the step face is added to the continuity equation boundary conditions for the node at station ISTP nearest the corner of the forward-facing step.

Finally, the velocities along the step face are specified equal to zero, and the pressure is extrapolated from the values at the upstream stations by applying a boundary condition of zero pressure gradient normal to the surface. The pressure at the step corner has two values, one representing the pressure on the forward face, the other for the pressure on the face downstream of the corner. The downstream value obtained from the coupled solution at station ISTP is retained.

XII. APPENDIX D: LINEARIZATION OF THE CONVECTIVE FLUXES

The finite-difference equations include the nodal values of the convective fluxes. These terms are nonlinear algebraic functions of the primitive variables. The numerical solution of the nonlinear finite-difference equations is accomplished by iterative solution of the linearized equations with the local iteration process continuing until the coefficients are converged. The present algorithm uses Newton linearization (or quasilinearization) to determine the coefficients. The procedure converges very rapidly, and an acceptable result is usually obtained with only one or two local iterations.

Notably, since the space-marching procedure requires global iteration, it is not necessary to converge the coefficients at each station during every marching sweep. Local iteration during the space-marching sweep is only necessary during the initial marching sweeps. The changes in the primitive variables and the coefficients of the linearization simultaneously approach zero as the solution converges globally. The overall computer time to solve the nonlinear equations is thus not significantly greater than globally iterating to solve a system of linear equations.

The development of the quasilinear expressions from the nonlinear functions is straightforward. In general a nonlinear function $f(u,v,w,p)$ may be approximated by a Taylor series expansion about the value at a known state point. For example

$$f(u, v, w, p) \approx f(\hat{u}, \hat{v}, \hat{w}, \hat{p}) + \Delta f \quad (12.1)$$

where

$$\Delta f = f_u \Delta u + f_v \Delta v + f_w \Delta w + f_p \Delta p$$

$$\Delta u = u - \hat{u}, \Delta v = v - \hat{v}, \Delta w = w - \hat{w}, \Delta p = p - \hat{p}$$

The caret designates the known state point values and the partial derivatives are all evaluated at that point. Equation 12.1 is a linear function of the primary variables, as shown more clearly by rewriting the equation as

$$f(u, v, w, p) \approx f_u u + f_v v + f_w w + f_p p + C \quad (12.2)$$

where

$$C = f(\hat{u}, \hat{v}, \hat{w}, \hat{p}) - f_u \hat{u} - f_v \hat{v} - f_w \hat{w} - f_p \hat{p}$$

The nonlinear convective terms are all analytically differentiable functions and are linearized in the form of Equation 12.2. The expressions for the coefficients for adiabatic, compressible flow of a perfect gas are given in Table 12.1, and Table 12.2 lists the coefficients for incompressible flow.

Table 12.1. Quasilinear coefficients for the adiabatic, compressible flow of a perfect gas

Flux (f)	f_u	f_v	f_w	f_p
qu	$\left[\hat{\rho} \left(1 + \frac{\bar{u}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho} \left(0 + \frac{\bar{u}\bar{v}}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho} \left(0 + \frac{\bar{u}\bar{w}}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{u}}{R\bar{T}} \right]$
qv	$\left[\hat{\rho} \left(0 + \frac{\bar{u}\bar{v}}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho} \left(1 + \frac{\bar{v}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho} \left(0 + \frac{\bar{v}\bar{w}}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{v}}{R\bar{T}} \right]$
qw	$\left[\hat{\rho} \left(0 + \frac{\bar{w}\bar{u}}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho} \left(0 + \frac{\bar{w}\bar{v}}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho} \left(1 + \frac{\bar{w}^2}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{w}}{R\bar{T}} \right]$
qu^2	$\left[\hat{\rho}\bar{u} \left(2 + \frac{\bar{u}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{v} \left(0 + \frac{\bar{u}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{w} \left(0 + \frac{\bar{u}^2}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{u}^2}{R\bar{T}} \right]$
qv^2	$\left[\hat{\rho}\bar{u} \left(0 + \frac{\bar{v}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{v} \left(2 + \frac{\bar{v}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{w} \left(0 + \frac{\bar{v}^2}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{v}^2}{R\bar{T}} \right]$
qw^2	$\left[\hat{\rho}\bar{u} \left(0 + \frac{\bar{w}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{v} \left(0 + \frac{\bar{w}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{w} \left(2 + \frac{\bar{w}^2}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{w}^2}{R\bar{T}} \right]$
quv	$\left[\hat{\rho}\bar{v} \left(1 + \frac{\bar{u}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{u} \left(1 + \frac{\bar{v}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{w} \left(0 + \frac{\bar{u}\bar{v}}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{u}\bar{v}}{R\bar{T}} \right]$
qvw	$\left[\hat{\rho}\bar{u} \left(0 + \frac{\bar{v}\bar{w}}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{w} \left(1 + \frac{\bar{v}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{v} \left(1 + \frac{\bar{w}^2}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{v}\bar{w}}{R\bar{T}} \right]$
quw	$\left[\hat{\rho}\bar{w} \left(1 + \frac{\bar{u}^2}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{v} \left(0 + \frac{\bar{w}\bar{u}}{c_p \bar{T}} \right) \right]$	$\left[\hat{\rho}\bar{u} \left(1 + \frac{\bar{w}^2}{c_p \bar{T}} \right) \right]$	$\left[\frac{\bar{w}\bar{u}}{R\bar{T}} \right]$

Table 12.2. Quasilinear coefficients for the flow of an incompressible fluid

Flux (f)	f_u	f_v	f_w	f_p
ρu	ρ	0	0	0
ρv	0	ρ	0	0
ρw	0	0	ρ	0
ρu^2	$2u$	0	0	0
ρv^2	0	$2v$	0	0
ρw^2	0	0	$2w$	0
ρuv	v	u	0	0
ρvw	0	w	v	0
ρwu	w	0	u	0

XIII. APPENDIX E: GENERATION OF CLUSTERED GRIDS

The non-uniform, clustered grids were generated with the stretching transformations given in Anderson et al. [3]. The type of grid transformation and the values of the stretching parameters for the grids generated for the test case geometries are listed in Table 13.1. Use of stretching transformations allows a high degree of local mesh refinement near a specified concentration point. The refinement ratio listed in Table 13.1 is defined as the step size on a uniform grid divided by the minimum step size for the clustered grid. For example, at the duct entrance of Case 6 the initial streamwise step size is 17 times smaller than the step size for a uniform grid with the same number of nodes. Two different stretching transformations are used for the test cases presented in this study. The methods are described below, retaining the labeling convention given in Anderson et al.

Transformation 1 is a monotonic stretching transformation used to cluster node points near to one boundary. In the form given below clustering will occur near $x = 0$. The stretching parameter, β , must be greater than one, and values of β near to unity produce greater stretching.

With $\xi = 0$ at $x = 0$ and $\xi = 1$ at $x = L$

$$x = L \left\{ \frac{(\beta + 1) - (\beta - 1)[[(\beta + 1)/(\beta - 1)]^{1-\xi} - \xi]}{[(\beta + 1)/(\beta - 1)]^{1-\xi} + 1} \right\} \quad (13.1)$$

Transformation 2 is a generalization of Transformation 1, which permits clustering at both boundaries. This transformation was not used for the test cases presented here.

Transformation 3 clusters nodes near an interior point, such as the step corner in the sudden contraction geometry. Note that the method does not necessarily position a node point at the point of grid concentration. To do so requires a node value with $\eta = B$. Large values of the parameter τ produce greater concentration of nodes near the location $y = C$.

With $y = 0$ at $\eta = 0$ and $y = h$ at $\eta = 1$

$$y = C \left\{ 1 + \frac{\sinh[\tau(\eta - B)]}{\sinh(\tau B)} \right\} \quad (13.2)$$

where

$$B = \frac{1}{2\tau} \ln \left[\frac{1 + (e^\tau - 1)(C/h)}{1 + (e^{-\tau} - 1)(C/h)} \right]$$

The grid for sudden contraction, Case 4, contains four different regions and is shown in Figure 5.24. Transformation 1 is used to cluster nodes axially about the sudden contraction at $x = 0$. The grid generated downstream of the contraction is simply reflected to negative values of x , upstream of the contraction. The y coordinate distribution is established with Transformation 2. The y coordinate cluster locations and degree of stretching upstream of the contraction differ from the values used after the contraction. The cluster coordinate, C , and the stretching factor, τ , are varied with a cubic function in the marching coordinate, ξ , to smoothly blend the different grid regions.

The grid for the cylinder in crossflow, Case 5, is shown in Figure 5.30, and was constructed by first applying the stretching transformations to a rectangular grid. These points are assigned values of the potential function and stream function corresponding to the position on the rectangular grid. The (x,y) coordinates are then determined by inverting the stream and potential functions. Transformation 1 is used to cluster the stream function values near the line of symmetry at $y = 0$. Cubic functions are used to distribute the potential function coordinates, in order to obtain a uniform point distribution at the freestream boundary, while clustering points near the cylinder.

Table 13.1. Summary of clustered grid parameters

Case	Streamwise Transformation				Transverse Transformation			
	Type	cluster location	stretching parameter	stretching ratio	Type	cluster location	stretching parameter	stretching ratio
1 (Re = 7500)	1	x = 0	$\beta = 1.00200$	66	1	y = 1	$\beta = 1.07095$	4.0
1 (Re < 7500) (41x21 grid)	1	x = 0	$\beta = 1.010365$	17	1	y = 1	$\beta = 1.07095$	4.0
2 and 3	0	uniform grid		1	0	uniform grid		1
4 (4 regions)								
-1.5 < x < -0.5	0	uniform grid		1	3	y = 1.0 $\tau = 3.6$		4.1
-0.5 < x < 0.0	1	x = 0	$\beta = 1.044134$	1	3	cubic variation of C and τ		
0.0 < x < 0.5	1	x = 0	$\beta = 1.044134$	5.4	3	y = 0.5 $\tau = 6.9$		9.1
0.5 < x < 2.5	0	uniform grid		1	3	y = 0.5 $\tau = 6.9$		9.1
5 (see Figure 5.31)		- cubic variation of Φ along $\Psi = 0$ - uniform variation of Φ along $\Psi = 20$			1	$\Psi = 0$	$\beta = 1.01365$	17
6 (41x21x21 grid)	1	x = 0	$\beta = 1.010365$	17	0	uniform grid		1

