LARGE EDDY SIMULATION OF INCOMPRESSIBLE TURBULENT CHANNEL FLOW

by

P. Moin,
W. C. Reynolds,
and
J. H. Ferziger

Prepared from work done under Grant
NASA-NgR-05-020-622

Report No. TF-12

Thermosciences Division
Department of Mechanical Engineering
Stanford University
Stanford, California

May 1978

LARGE EDDY SIMULATION OF INCOMPRESSIBLE TURBULENT CHANNEL FLOW

by

P. Moin, W. C. Reynolds, and J. H. Ferziger

Prepared from work done under Grant
NASA-NgR-05-020-622

Technical Report Number TF-12

Thermosciences Division
Department of Mechanical Engineering
Stanford University
Stanford, California

May 1978

Acknowledgments

The authors gratefully acknowledge many useful contributions of Dr. N. Mansour. We would like to express our gratitude also to Drs. T. Coakley, J. Kim, U. Mehta, J. Stager, R. Rogalo, M. Rubesin, and Messrs. A. Cain and W. Feiereisen for fruitful discussions and contributions to this work.

Thanks are due to Dr. A. Leonard for his critical reading of the manuscript and numerous useful discussions throughout this work.

This work was supported by NASA-Ames Research Center under Grant NASA-NgR-05-020-622.

The authors acknowledge the excellent job of typing done by Mrs. Ann Ibaraki and Mrs. Ruth Korb.

LARGE EDDY SIMULATION OF INCOMPRESSIBLE TURBULENT CHANNEL FLOW

Abstract

The three-dimensional, time-dependent primitive equations of motion have been numerically integrated for the case of turbulent channel flow. For this purpose, a partially implicit numerical method has been developed. An important feature of this scheme is that the equation of continuity is solved directly. The residual field motions were simulated through an eddy viscosity model, whereas the large-scale field was obtained directly from the solution of the governing equations. 16 uniform grid points were used in each of the streamwise and spanwise directions, and 65 grid points with non-uniform spacings in the direction normal to the walls. An important portion of the initial velocity field was obtained from the solution of the linearized Navier-Stokes equations. The pseudospectral method was used for numerical differentiation in the horizontal directions, and second-order finite-difference schemes were used in the direction normal to the walls.

It has been shown that the Large Eddy Simulation technique is capable of reproducing some of the important features of wall-bounded turbulent flows. The overall agreement of the computed mean velocity profile and turbulence statistics with experimental data is satisfactory. The resolvable portions of the root-mean square wall pressure fluctuations, pressure velocity-gradient correlations, and velocity pressure-gradient correlations are documented.

Table of Contents

List of Tables

List of Figures

Figure		Page

Nomenclature

A		Amplitude of Orr Sommerfeld waves.
a		Grid transformation constant.
a_n		Chebyshev expansion coefficient of a flow variable.
B		Boundary of the physical domain.
b_n		Chebyshev expansion coefficient of the first derivative of a flow variable.
c		Complex speed of an Orr-Sommerfeld wave. Also a constant.
C_i	$\equiv$	$1 + \delta_{i2}$.
C_1, C_2, C_3		Constants.
C_s		Smagorinsky's constant.
D		Dissipation.
d_i	$\equiv$	$1 - \delta_{i2}$.
e_i		Unit vector in the i-direction.
f		A flow variable.
$\overline{f}$		Filtered component of f.
f'		Subgrid scale (SGS) component of f.
$G(\underline{x}-\underline{x}')$		Filter function.
$\hat{G}(k)$		Fourier transform of the filter function.
h_i		Mesh size in the i-direction.
h_i^+	$\equiv$	$h_i u_\tau/\nu$.
k		Wave number $\equiv \sqrt{k_1^2 + k_2^2}$ or $\equiv \sqrt{\alpha^2 + \beta^2}$
k_i		Wave number in the i-direction.
ℓ		SGS length scale.
ℓ'		Prandtl's mixing length.
L_x		Length of the computational box in the x-direction.
L_z		Length of the computational box in the z-direction.

$\underset{\sim}{n}$		Unit vector normal to the wall.
N		Number of mesh points in the y-direction.
N_x		Number of mesh points in the x-direction.
N_z		Number of mesh points in the z-direction.
N_j		Number of mesh points in the j-direction.
p		Pressure.
$\overline{p}$		Filtered pressure.
$\tilde{p}$	$\equiv$	$\overline{p}/\rho + R_{kk}/3.$
$\overline{P}$	$\equiv$	$\tilde{p} + \frac{1}{2}(\overline{u}_j\overline{u}_j)$
$\hat{p}$		Fourier transform of pressure.
$\mathscr{P}$		SGS energy production.
P_D		Pressure solution using Dirichlet boundary condition.
P_j		Fourier transform of $\overline{P}$ at y_j.
P_N		Pressure solution using Neumann boundary condition.
q		r.m.s. velocity.
$\hat{Q}$		Fourier transform of the right-hand side of the Poisson equation for pressure.
Re		Reynolds number based on channel half-width and the centerline velocity.
Re_m		Reynolds number based on channel half-width and mean velocity, U_m.
Re_τ		Reynolds number based on channel half-width and shear velocity.
R_{ij}	$\equiv$	$\overline{u_i'u_j'} + \overline{u_j'\overline{u}_i} + \overline{u_i'\overline{u}_j}$
$R_{ii}(r,0,0)$		Experimental two-point velocity correlation function $\equiv$ $\langle u_i(x,y,z)\, u_i(x+r,y,z) \rangle$
$R_{ii}(0,0,r)$	$\equiv$	$\langle u_i(x,y,z)\, u_i(x,y,z+r) \rangle$
r		Separation distance in the two-point correlation function.
$\underline{r}$		Vector in r direction.
$\overline{S}_{ij}$	$\equiv$	$\frac{1}{2}\left(\frac{\partial}{\partial x_j}\overline{u}_i + \frac{\partial}{\partial x_i}\overline{u}_j\right)$, strain rate tensor.

t	Dimensionless time.
u	Streamwise velocity.
$\overline{u}$	Filtered streamwise velocity.
u'	Subgrid scale component of u.
$\hat{u}$	Fourier transform of u; also $\equiv (\alpha\hat{u}_1 + \beta\hat{u}_3)/k$.
$\left(\overline{u'^2}\right)^{1/2}$	r.m.s. streamwise velocity fluctuation.
u_i	Velocity in the i-direction.
$\overline{u}_i$	Filtered component of u_i.
u'_i	SGS component of u_i.
$\underset{\sim}{u}$	Velocity vector.
u_τ	Shear velocity $\equiv \sqrt{\frac{\tau_w}{\rho}}$
$\tilde{u}_i(x,y,z)$	Solution of the linearized Navier-Stokes equations.
$\hat{u}_i(y)$	Eigenfunctions of the linearized Navier-Stokes equations.
U	Mean velocity profile.
U_∞	Freestream velocity.
U_o	Centerline velocity.
U_m	Mean profile average velocity.
v	Velocity in the vertical direction.
$\left(\overline{v'^2}\right)^{1/2}$	r.m.s. vertical velocity fluctuation.
$\overline{v}$	Filtered component of v.
v'	SGS component of v.
$\hat{v}$	Fourier transform of v; also the solution of the Orr-Summerfeld equation.
w	Velocity in the spanwise direction.
$\left(\overline{w'^2}\right)^{1/2}$	r.m.s. spanwise velocity fluctuation.
$\overline{w}$	Filtered component of w.
w'	SGS component of w.

x, x'		Streamwise coordinate
x_i		Coordinate in the i-direction.
$\underline{x}, \underline{x}'$		Coordinate vector.
X_1		Twice the vanishing distance, r, of $R_{11}(r,0,0)$.
X_3		Twice the vanishing distance, r, of $R_{11}(0,0,r)$.
y		Coordinate in the direction perpendicular to the walls.
y_c		y coordinate of the first computational grid point away from the wall at which the planar average of inner and outer layer models are closest to each other.
y_j		j^{th} mesh point in the vertical direction.
y_w		Distance to the nearest wall.
y^+	$\equiv$	$y_w u_\tau/\nu$.
z		Spanwise coordinate.

Greek Letters

α		Wave number in the x-direction of the solution of the linearized Navier-Stokes equation.
β		Wave number in the z-direction of the solution of the linearized Navier-Stokes equation.
$\beta_i(y)$	$\equiv$	$-(2/\Delta t)/(\frac{1}{Re_\tau} + C_i\tilde{\nu}_T(y))$
Δ		Filter width.
Δx		Average dimensionless distance between the structures in x-direction.
Δz		Average dimensionless distance between the structures in z-direction.
$\Delta\xi$		Grid spacing in the transformed (uniform mesh) space.
Δ_i		Filter width in the i-direction $(= 2h_i)$.
κ		von Karman constant $(\simeq .4)$.
λ		Mean streak spacing.
λ_i		Mean spacing of the turbulent structures in the i-direction.

λ_i^+	$\equiv$	$\lambda_i u_\tau/\nu$.
λ^+	$\equiv$	$\lambda u_\tau/\nu$.
ρ		Density.
ξ_j		j^{th} mesh point in the vertical direction of the transformed (uniform mesh) space.
$\underset{\sim}{\tau}$		Unit vector tangent to the solid boundary $\equiv \underline{e}_1 + \underline{e}_2$.
τ_{ij}	$\equiv$	$R_{ij} - R_{kk}\delta_{ij}/3$.
τ_w		Average wall shear stress $(\equiv \mu \frac{\partial U}{\partial y})$.
Δt		Dimensionless time step.
μ		Molecular viscosity.
ν		Kinematic viscosity.
ν_T		Eddy viscosity.
$\tilde{\nu}_T$	$\equiv$	$< \nu_T >$.
ν_T'	$\equiv$	$\nu_T - \tilde{\nu}_T$.
ν_T''	$\equiv$	$\nu_T - \max_{x_1,x_2,x_3} (\nu_T)$
$\underset{\sim}{\omega}$		Vorticity vector.
ω		Complex frequency $(\equiv - \alpha c)$
δ		Channel half-width.
δ_{ij}	$\equiv$	$\begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$
< >		Horizontal average (xz plane); also horizontal average and running time average; in the case of experimental two-point correlation function, indicates time average.

Superscripts

n	Time step.

Chapter I

INTRODUCTION

1.1 Historical Background

It has been known for some time that any turbulent flow contains structures ("eddies") in a wide range of spatial as well as temporal scales. It is also generally recognized that large eddies differ markedly from one flow type to another (e.g., jets vs. boundary layers), while the small eddies are quite similar in all flows.

Unfortunately, in the numerical simulation of (high Reynolds number) turbulent flows, we find that due to computer limitations one cannot resolve all the scales. It is this deficiency which provides the primary inducement for the utilization of the large eddy simulation (L.E.S.) approach.

The foundation on which this approach relies concerns the contrast between large and small eddy modeling. More specifically, one finds that large eddies cannot and should not be modeled, whereas with small eddies successful modeling is possible.

The large eddy simulation method is initiated by the introduction of a procedure which separates the small and large scale structures. The large scale structures will then be computed explicitly, while the small scales are necessarily modeled.

The problem of decay of homogeneous isotropic turbulence has been the subject of extensive study at Stanford University (Kwak et al. (1975), Shaanan et al. (1975), Mansour et al. (1977), Ferziger et al. (1977)). These studies have shown that with the use of algebraic models and a relatively small number of mesh points (16 x 16 x 16 or 32 x 32 x 32), homogeneous turbulent flows can be simulated reasonably well.

The first application of the method to problems of engineering interest was made by Deardorff (1970) who treated the channel flow problem. In his pioneering work, Deardorff showed that a three dimensional numerical simulation of turbulence is feasible. He was able to predict some of the features of turbulent channel flow with a fair amount of success. However, as will be clear in the next section, neither Deardorff nor the followup work of Schumman (1973) treated the most important part of the flow, namely the region very near the wall. It is in this region that virtually all of the turbulent energy production occurs. By introducing artificial boundary conditions, they, in effect, modeled the turbulence production mechanism in the wall region.

Finally, we note that, concurrent with the present work, Mansour et al. (1978) simulated a time developing turbulent mixing layer. They showed that essentially all the features of a turbulent mixing layer can be reproduced using the L.E.S. approach.

1.2 Experimental Background

Many early studies of the structure of turbulence consisted of measurments of the root-mean square and spectra of the turbulent velocity fluctuations. Among the measurements that were primarily concerned with turbulent boundary layers were those of Townsend (1951), Klebanoff (1954), Willmarth and Wooldridge (1963), and for flow near the wall (in a pipe) Laufer (1954).

Willmarth made a single, unpublished attempt, in 1960, to bring together the then existing results of turbulence-intensity profiles of the boundary layer on a single plot (see Willmarth, 1975). The curves of $\sqrt{u'^2}/u_\tau$, $\sqrt{v'^2}/u_\tau$, and $\sqrt{w'^2}/u_\tau$, as a function of y_w/δ (or $y^+ = y_w u_\tau/\nu$) did not agree very well (not within 50%). Here, y_w is the distance to the wall, u_τ is the shear velocity, and δ is the boundary layer thickness. Part of the lack of agreement was attributed to freestream disturbances or differences in the methods used to trip the boundary layers. However, in spite of the differences between various measurements of turbulence intensity, it is definitely established that within a turbulent boundary layer, $\sqrt{u'^2}/U_\infty > \sqrt{w'^2}/U_\infty > \sqrt{v'^2}/U_\infty$. These differences between the root-mean-square velocity fluctuations become larger as one approaches the wall. Furthermore, the profiles $\sqrt{u'^2}$ and $\sqrt{w'^2}$ have pronounced local maxima very near the wall.

From the measured distributions of turbulence kinetic energy, turbulence shear stress, and dissipation, it is possible to obtain a turbulence energy balance. Townsend (1951) and Laufer (1954) (among others) made such a balance in a boundary layer and pipe flow respectively. From these data, it can be seen that the production and dissipation terms are nearly equal but opposite to each other, and so are the terms representing diffusion by turbulence of kinetic energy and of pressure energy. Furthermore, it may be noted that the turbulence kinetic energy, its production and its dissipation, all show sharp maxima in the buffer region ($y+ \simeq 10$) near the wall. On the basis of energy measurements, Townsend (1956) proposed a two-layer model for the energy transformation process. According to this model, the whole layer is arbitrarily divided into two parts: (i) an inner layer which is nearly in energy equilibrium but within which most of the turbulence production takes place, and (ii) an outer layer whose Reynolds stresses retard the mean flow but whose principal source of turbulent energy comes from the inner layer.

The level of turbulent intensity in the outer two-thirds of the flow is maintained by transport of energy from the inner region since the production of energy in the outer region is too small to balance the viscous dissipation and transport losses. Townsend concluded that the interaction between the inner and outer layers of the flow may be considered as two distinct processes: (i) the transfer of mean-flow energy from the outer region to the inner layer at a rate controlled by the gradient of Reynolds stresses in the outer layer, and (ii) the transport of turbulent energy from the inner layer to the outer layer.

To gain insight into the mechanics of turbulence production a thorough study of the structure of the inner layer was required. Runstadler et al. (1959), (1963) advanced a model for the inner layer based on visual observations using dye and hydrogen bubbles. Their studies revealed new features of turbulent boundary layers. In particular, they demonstrated that the wall layer is not two dimensional and steady; rather it consists of relatively coherent structures of low and high speed streaks alternating in the spanwise direction over the entire wall. The non-dimensional <u>mean</u> spacing between the low speed streaks was shown to have a universal correlation for fully turbulent layers based on wall layer parameters; this is given by the relation

$$\bar{\lambda}^{+} = \frac{\lambda u_\tau}{\nu} \simeq 100$$

The streak pattern is not stationary in space. It migrates and displays strong intermittent motion. These intermittent motions involve primarily the movement of low speed streaks away from the wall. When the streak has reached a point corresponding to $y+ \leq 8\text{-}12$, it begins to oscillate. The oscillation grows in amplitude and it is followed by breakup. The region where most of the low speed streak breakups are observed to occur, i.e., the inner edge of the buffer zone, is the region where a sharp peak is seen to occur in the production curve (Klebanoff 1954). Kline et al. (1967) and Clark and Markland (1970) observed U shaped vortices occasionally in the inner region. In the studies of Clark and Markland, an average spanwise spacing of these U shaped vortices of $\lambda_3^+ \simeq 100$ and streamwise spacing of λ_1^+ of 440 was found.

Kim et al. (1971) studied bursts using motion pictures of the trajectories of hydrogen bubbles. From their analysis, they concluded that in the region $0 < y^+ < \overline{100}$ essentially all the turbulence production occurs during bursting. They also observed that during gradual liftup of low-speed streaks from the sub-layer, unstable (inflectional) instantaneous velocity profiles were formed. One of the important findings of Kim et al. was that, while the bursting process indeed contributes to the turbulent energy, its main effect is to provide turbulence with u' and v' in proper phase to give large positive Reynolds stresses as required for the increase in production.

The findings of Kline and his colleagues were largely confirmed and supplemented by the visual studies of Corino and Brodkey (1969). One of their observations was that, after formation of low speed streak a much larger high speed bulk of fluid came into view and by "interaction" began to accelerate the low speed fluid. The entering high speed fluid carried away the slow moving fluid remaining from the ejection process; this they called the "sweep" event.

The above experimental investigations of the structure of turbulent boundary layers are by no means the only ones reported. The number of publications on the subject is already very large. Among these is the work of Narahari, Rao, Narasimha, and Badri Narayanan (1971), where the frequency of occurrence of bursts was studied. Their investigation showed that the mean bursting frequency scaled with the outer rather than inner flow variables. This was also reported by Kim et al. (1971). The recent experimental investigation of Blackwelder and Kaplan (1976) studied the near wall structure of the turbulent boundary layer using hot-wire rakes and conditional sampling techniques. Among their findings was that, the normal velocity is directed outwards in the regions of strong streamwise-momentum deficit (with respect to the mean velocity), and inwards in the regions of streamwise-momentum excess. This was also reported by Grass (1971). For further details and description of other works on the structure of turbulent boundary layers the reader is referred to the review articles of Willmarth (1975) and Laufer (1975). An entire meeting was recently devoted to review of the state of knowledge in this area (Abbott 1978).

1.3 Motivation and Objectives

The present study is one in a systematic program investigating large eddy simulation of turbulence. In order to extend the available technology of the L.E.S. approach to wall-bound flows, we chose to study incompressible turbulent channel flow. Due to the simplicity of its geometry and some experimental advantages, channel flow has been a particularly attractive reference flow for both theoretical and experimental investigations. As a result, there is a considerable amount of experimental as well as theoretical findings available for a detailed evaluation of the large eddy simulation technique. In addition, this flow possesses important features of the flows of practical interest. This, in turn, allows the evaluation of the L.E.S. approach from a practical point of view.

The specific objectives of this work may be stated as follows:

a) To develop a numerical method for long time integration of the three-dimensional governing equations for the large scale field in a turbulent channel flow;

b) To carry out numerical solution of these equations using a simple subgrid scale model;

c) To evaluate the performance of the Large Eddy Simulation technique in reproducing some of the laboratory observations and measurements described above, and to compute quantities such as pressure velocity gradient correlations that cannot be measured.

1.4 Summary

The contributions of the present work include:

a) Demonstration of the inherent numerical problems associated with the explicit numerical solution of the dynamical equations of motion in primitive form.

b) Derivation of consistency conditions for the initial velocity field such that the Neumann and Dirichlet problems for the pressure have the same solution.

c) Development of a new semi-implicit numerical scheme for the solution of dynamical equations in primitive form.

d) Development and use of a new subgrid model in the wall region of the turbulent flow.

e) Development and use of a solution of the Orr-Sommerfeld equation for a three-dimensional disturbance as an important part of the initial velocity field.

f) Demonstration that the Large Eddy Simulation technique is capable of reproducing many of the important features of the turbulent boundary layer.

Chapter II

MATHEMATICAL FOUNDATIONS

2.1 Definition of Filtered and Residual Fields

In the large eddy simulation approach, the first and most fundamental step is defining the large-scale field. To accomplish this task, each author has adopted a slightly different approach, but they can be treated within a single conceptual framework as shown by Leonard (1974). If f is some flow variable, we decompose it as follows:

$$f = \bar{f} + f' \tag{2.1}$$

where $\bar{f}$ is the large-scale component and f' is the residual field. Leonard defined the large scale field as:

$$\bar{f}(\underline{x}) = \int G(\underline{x}-\underline{x}')\, f(\underline{x}')\, d\underline{x}' \tag{2.2}$$

where $G(\underline{x}-\underline{x}')$ is a filter function with a characteristic length Δ, and the integral is extended over the whole flow field. It is to be noted that the above form of G (a function of $(\underline{x}-\underline{x}')$) is most suited for filtering in the directions in which the flow is homogeneous. In other words, we point out that the filter function need be neither isotropic nor homogeneous and there are many flows (or directions in a given flow) in which neither of these properties are desirable. In the present work we use the Gaussian filter,

$$G(\underline{x}-\underline{x}') = \prod_{i=1}^{n} \left(\frac{6}{\pi\Delta_i}\right)^{\frac{1}{2}} \exp\left[-6(x_i-x_i')^2/\Delta_i^2\right] \tag{2.3}$$

where $\Delta_i = 2h_i$, h_i is the mesh size in the i-direction, and $n = 1$, 2, or 3, is the number of dimensions in which the flow is homogeneous. Thus in the simulation of the decay of homogeneous isotropic turbulence, $n = 3$, while in the simulation of turbulent channel flow, we have used

$n = 2$. A convenient property of a homogeneous filter, $G(\underline{x}-\underline{x}')$, is its commutivity with partial differentiation operators; using integration by parts one can show (Kwak et al. (1975)):

$$\overline{\frac{\partial f}{\partial x_i}} = \frac{\partial \overline{f}}{\partial x_i} \tag{2.4}$$

Due to variation of the physical length scale of turbulence in the direction in which the flow is homogeneous, one should not use homogeneous filters in that direction. This is particularly true in turbulent boundary layers. Instead, one should use a filter with variable width $\Delta(\underline{r})$, where $\underline{r}$ is the direction in which the flow is inhomogeneous. On the other hand, using a filter with variable width causes some mathematical difficulties; in particular (2.4) will no longer hold. In Appendix A, we explore filters with nonuniform width in some detail.

Finally, we note that, in the numerical simulation of turbulent channel flow, we filter only in the directions in which the flow is homogeneous, (streamwise and spanwise directions) i.e., we do not formally filter in the direction perpendicular to the walls. The justification for this choice is twofold:

a) We are using a second order finite difference scheme to approximate partial derivatives in the inhomogeneous direction, and finite difference shcemes in general have inherent filtering effect.

b) The Leonard term is fairly well represented by the truncation error of the second order central differencing scheme. (See Shaanan (1975)).

The main disadvantage of this choice is that we do not have a formal closed mathematical expression relating the filtered to the unfiltered field.

2.2 Dynamical Equations in Primitive Form

Now let us derive the primitive dynamical equations for the large-scale flow field. Starting with the incompressible Navier-Stokes equations,

$$\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j} u_i u_j = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j}$$

we can apply the operation (2.2) to get the dynamical equations of large scale field,

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j} \overline{\overline{u}_i \overline{u}_j} = -\frac{1}{\rho}\frac{\partial \tilde{P}}{\partial x_i} - \frac{\partial}{\partial x_j}\tau_{ij} + \nu \frac{\partial^2 \overline{u}_i}{\partial x_j \partial x_j} \tag{2.5}$$

where we have decomposed u_i as in (2.1) and:

$$\tau_{ij} = R_{ij} - R_{kk}\,\delta_{ij}/3$$

$$\tilde{P} = \overline{P}/\rho + R_{kk}/3$$

$$R_{ij} = \overline{u_i' u_j'} + \overline{u_j' \overline{u}_i} + \overline{u_i' \overline{u}_j}$$

The τ_{ij} represents the (negative) subgrid scale stresses and must be modeled. We can write (2.5) in the following equivalent form:

$$\frac{\partial \overline{u}_i}{\partial t} + \overline{\overline{u}_j\left(\frac{\partial \overline{u}_i}{\partial x_j} - \frac{\partial \overline{u}_j}{\partial x_i}\right)} = -\frac{\partial \overline{P}}{\partial x_i} - \frac{\partial}{\partial x_j}\tau_{ij} + \nu \frac{\partial^2 \overline{u}_i}{\partial x_j \partial x_j} \tag{2.6}$$

where

$$\overline{P} = \tilde{P} + \frac{1}{2}\,(\overline{\overline{u}_j \overline{u}_j})$$

The rationale for using this form of the equation will be explained in Section 3.5.

In order to calculate the second term on the left-hand side of (2.6), we use (2.2) to write:

$$\overline{\bar{u}_j\left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i}\right)} = \int_{-\infty}^{+\infty} G(\underline{x}-\underline{x}')\, \bar{u}_j\left(\frac{\partial \bar{u}_i}{\partial x'_j} - \frac{\partial \bar{u}_j}{\partial x'_i}\right) d\underline{x}'$$

Note that, here, the filtering and the corresponding integration is performed only in the directions in which the flow is homogeneous. Let us Fourier transform the above equation (in the homogeneous directions) to get:

$$\widehat{\overline{\bar{u}_j\left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i}\right)}} = \hat{G}\widehat{\left[\bar{u}_j\left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i}\right)\right]} \tag{2.7}$$

where $\hat{}$ denotes a Fourier-transformed quantity; a $\hat{}$ over a bracket means the transform of the bracketed quantity. Thus, given a velocity field, $\bar{u}_i$, one can compute the term in the brackets on the right-hand side of the above equation, Fourier-transform it, multiply it by $\hat{G}$, and invert the transform to obtain the desired term.

2.3 Residual Stress Model

An eddy viscosity model is used for τ_{ij} :

$$\tau_{ij} = -2\nu_T \bar{S}_{ij} \tag{2.8}$$

where

$$\bar{S}_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right)$$

is the strain rate tensor and ν_T is an eddy viscosity associated with the residual field motions. In the remainder of this section, we present the models used for ν_T . Throughout, we assume that the subgrid scale production and dissipation of turbulent kinetic energy are equal.

Production of the subgrid scale turbulent kinetic energy is given by:

$$\mathcal{P} = 2\nu_T \overline{S}_{ij} \overline{S}_{ij} \tag{2.9}$$

Inclusion of the experimental observation that, remote from the wall, dissipation is controlled only by the largest subgrid-scale eddy parameters such that $D = D(q^2,\ell)$, coupled with dimensional analysis, produce the result first found by Kolmogorov in 1942 that $D \propto q^3/\ell$. Here, q and ℓ are the characteristic velocity and length scale of subgrid scale eddies respectively. Using Prandtl's assumption for eddy viscosity, $\nu_T = C_1 q\ell$, and equating the subgrid production and dissipation, we get:

$$2C_1 q\ell \, \overline{S}_{ij} \overline{S}_{ij} = q^3/\ell \tag{2.10}$$

From (2.10), we readily obtain:

$$q = C_3 \ell \sqrt{2\overline{S}_{ij}\overline{S}_{ij}}$$

Again, using Prandtl's assumption, we get:

$$\nu_T = (C_s \ell)^2 \sqrt{2\overline{S}_{ij}\overline{S}_{ij}} \tag{2.11}$$

This is Smagorinsky's (1963) model, and is to be used in the regions away from the solid boundaries.

On the other hand, very near the wall, the size of the eddies is inhibited, and the eddies are of such a size that viscosity can be a dissipative agent for the largest eddies. In fact, at the wall, the eddy viscosity as well as its gradients should vanish. Under such conditions viscosity is a factor and $D = D(\nu,q^2,\ell)$. Application of dimensional analysis to this condition produces the result that $D \propto (\nu q^2/\ell^2) f(q\ell/\nu)$. Moreover, <u>at the wall</u> the subgrid scale dissipation is given by:

$$D = \nu\left[\overline{\left(\frac{\partial u'}{\partial y}\right)^2} + \overline{\left(\frac{\partial w'}{\partial y}\right)^2}\right] \propto \frac{\nu q^2}{\ell^2}$$

Thus, in the vicinity of the wall, we assume that $D \propto \nu q^2/\ell^2$. Equating subgrid scale production and dissipation, we obtain for the inner region of the boundary layer:

$$\nu_T = (C_2 \ell^4/\nu)\ (2\overline{S}_{ij}\overline{S}_{ij}) \tag{2.12}$$

where C_2 is a constant.

In order to determine the value of C_2, we assume that C_s, Smagorinsky's constant, is known from some other calculation e.g., simulation of the decay of isotropic turbulence. Strictly speaking, there is no rigorous justification that the constant obtained from the simulation of a totally homogeneous flow is applicable in the simulation of a wall-bounded turbulence with mean shear. Furthermore, in order to determine the value of C_2, several known characteristics of turbulent boundary layers will be applied. Among these characteristics is that, in the logarithmic section of the layer, the slope of the mean velocity profile in the semilogarithmic wall coordinates is $1/\kappa$, where κ is the von Karman constant. Hence, in what follows, we give only a rough estimate of the value of C_2, which will be used throughout our simulation of turbulent channel flow.

At the edge of the logarithmic section of the boundary layer, (say $y^+ = 27$), we demand that the inner and outer layer models have the same planar mean value. If we nondimensionalize all the velocities by the shear velocity, u_τ, and the lengths by the channel half width, δ, we have in the logarithmic region:

$$\sqrt{2\overline{S}_{ij}\overline{S}_{ij}} \simeq \frac{\partial U}{\partial y} \simeq \frac{1}{\kappa y_w} \tag{2.13}$$

where y_w is the distance to the lower wall (the lower wall is located at $y = -1$ and the upper wall at $y = +1$). Note that here, we have assumed that the mean velocity gradient is much larger than all the other velocity gradients. Equating the two models at $y^+ = 27$, we obtain:

$$C_2 = \frac{C_s^2}{\kappa y+} = \frac{C_s^2}{27\kappa} \tag{2.14}$$

where we have assumed that $\ell = \kappa y_w$. Thus, the actual model used for the eddy viscosity at each time step in the calculation is:

$$\nu_T = \begin{cases} C_2 \; Re_\tau \; \ell^4 (2\overline{S}_{ij}\overline{S}_{ij}) & y \le y_c \\ \\ (C_s \ell)^2 \sqrt{2\overline{S}_{ij}\overline{S}_{ij}} & y > y_c \end{cases} \tag{2.15}$$

Here y_c is the coordinate of the first computational grid point away from the wall at which the planar average of the two models are closest to each other. It is to be noted that, y_c can vary in time and in general it does. The same relation as (2.15) is used in the upper half of channel $(0 \le y \le 1)$. Finally, we turn our attention to the specification of ℓ .

Due to the no-slip boundary condition, ℓ must vanish at the walls. Furthermore, due to lack of spatial resolution in the homogeneous directions (see Section 3.1), and with no further reasoning, we have used the following expression for ℓ in the simulation of turbulent channel flow:

$$\ell = \left[\min\begin{pmatrix} \Delta_1 \\ \ell' \end{pmatrix} \cdot \min\begin{pmatrix} \Delta_3 \\ \ell' \end{pmatrix} \cdot \min\begin{pmatrix} h_2 \\ \ell' \end{pmatrix} \right]^{1/3} \tag{2.16}$$

where ℓ' is the Prandtl's mixing length:

$$\ell' = \begin{cases} 0.1 & y_w > .1/\kappa \\ \kappa y_w & y_w \le .1/\kappa \end{cases}$$

Δ_1 and Δ_3 are the nondimensionalized filter widths in streamwise and spanwise directions respectively, and h_2 is the local grid size in the vertical direction. Two remarks are in order. First, due to the particular grid sizes chosen (see Section 3.1), we have the following global inequalities:

$$h_2(y) < .1$$
$$\Delta_1 > \Delta_3 > .1$$

(Note that all the lengths are nondimensionalized with respect to channel half width δ). Second, we should mention that the expression (2.16) for ℓ is strictly speaking, based on ad hoc foundations and more work in this area is strongly recommended (see Chapter V). This expression was chosen initially on a trial basis; nevertheless, we did not find any alteration of it necessary. Thus, we emphasize that in obtaining the computational results presented here, no fine adjustments of either C_s or ℓ were made. In spite of this, the numerical results (see Chapter IV) are satisfactory. It is believed however, that an optimum choice for C_s and ℓ would somewhat improve the quantitative results.

2.4 Governing Equations for the Large Scale Field

In the numerical simulation of turbulent channel flow, all the variables are nondimensionalized by turbulent shear velocity, u_τ , and the channel half width, δ . In this case, we solve the following equations numerically:

$$\frac{\partial \bar{u}_i}{\partial t} + \overline{\bar{u}_j \left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i} \right)} = - \frac{\partial \bar{P}}{\partial x_i} + \delta_{i1} + \frac{\partial}{\partial x_j} (2\nu_T \bar{S}_{ij}) + \frac{1}{Re_\tau} \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} \tag{2.19}$$

and

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0 \qquad (2.20)$$

where Re_τ is the Reynolds number based on shear velocity, u_τ , and channel half width, δ . Note that the second term on the right-hand side of equation (2.19) is the mean pressure gradient imposed on the flow.

Chapter III

NUMERICAL METHODS

3.1 Grid Selection

For a given number of grid points, N , one has to choose the grid size(s) based on the physical properties of the problem at hand. In the simulation of the decay of homogeneous isotropic turbulence, for example, it is desirable to select the grid size, h , such that the filtered field contains as much of the turbulence energy as possible (Kwak et al., 1975). On the other hand, the length of the side(s) of the computational box in the direction(s) in which periodic boundary conditions are used should be long enough to include the important large eddies (Ferziger et al., 1977).

In the grid size selection process for the numerical simulation of turbulent channel flow, one has to consider the average spanwise and streamwise spacing of the turbulent structures in the vicinity of the wall (see Section 1.3) as well as the integral scales of turbulence. In addition, quantities such as the thickness of the viscous sublayer should be taken into consideration. With this in mind we proceed to specify our grid system:

In the vertical direction $(-1 \leq y \leq 1)$, a nonuniform grid spacing is used. The following transformation gives the location of grid points in the vertical direction (Mehta, 1977).

$$y_j = \frac{1}{a} \tanh\left[\xi_j \tanh^{-1}(a)\right] \tag{3.1}$$

where

$$\xi_j = -1 + 2(j-2)/(N-3) \qquad j=1,2,\ldots,N$$

and N is the total number of grid points in the y direction. Here, a is the adjustable parameter of the transformation $(0 < a < 1)$; a large value of a distributes more points near the boundary. In our computation we have used $a = .98346$, and $N = 65$. Table 3.1 shows the distribution of the grid points in the vertical direction with the corresponding values of $y+ = y_w u_\tau/\nu$. Note that in reference to the vertical direction, index (or subscript) 1 and N refer to grid points just outside the lower and upper walls respectively.

For the grid selection in the streamwise, x, and spanwise, z, directions, one needs to consider the experimentally measured two point correlation functions

$$R_{ii}(r,0,0) = \langle u_i(x,y,z)\, u_i(x+r,y,z) \rangle$$

and

$$R_{ii}(0,0,r) = \langle u_i(x,y,z)\, u_i(x,y,z+r) \rangle$$

Here $\langle\ \rangle$ denotes the average over an ensemble of experiments. The use of periodic boundary conditions in a given direction can be justified if the length of the side of the computational box in that direction is at least twice the distance r, at which the appropriate R_{ii} vanishes.

Experimental data of Comte-Bellot (1963), indicates that

$$X_1 = 6.4\delta$$

and

$$X_3 = 3.2\delta$$

where X_1 and X_3 are twice the distance, r, beyond which $R_{11}(r,0,0)$ and $R_{11}(0,0,r)$ respectively, are negligible. Here δ is the channel half width.

For a complete simulation of the important large scale field, one has to select the number of grid points in the streamwise, x, and spanwise z, directions with careful consideration to laboratory observations. We assume that L_x and L_z, the lengths of the computational box in the streamwise and spanwise directions, are fixed in accordance with the above considerations. As was mentioned in Chapter I,

Table 3.1

GRID DISTRIBUTION IN THE VERTICAL, y , DIRECTION

n	y	$y_w = \|1+y\|$	y^+ *
1	-1.002	.002	
2	-1.000	.000	0.000
3	- .997219	.00278	1.78
4	- .993983	.00602	3.85
5	- .99022	.00978	6.26
6	- .985847	.01415	9.06
7	- .980767	.01923	12.31
8	- .974871	.02513	16.09
9	- .968035	.03197	20.47
10	- .960117	.03988	25.53
11	- .950956	.04904	31.40
12	- .940372	.05963	38.18
13	- .928164	.07184	45.99
14	- .914109	.08589	54.99
15	- .898	.102	65.33
16	- .879	.121	77.47
17	- .858	.142	90.91
18	- .834	.166	106.28
19	- .807	.193	123.57
20	- .776	.224	143.42
21	- .741	.259	165.82
22	- .702	.298	190.79
23	- .659	.341	218.32
24	- .611	.389	249.06
25	- .559	.441	282.35
26	- .502	.498	318.84
27	- .440	.560	358.54
28	- .374	.626	400.80
29	- .304	.696	445.61
30	- .231	.769	492.35
31	- .156	.844	540.37
32	- .078	.922	590.31
33	.0	1.000	640.25

*For Re_τ = 640.25.

experimental data indicate that the average (spanwise) streak spacing corresponds approximately to $\lambda_3^+ \simeq 100$ and the average streamwise spacing of the U shaped vortices corresponds to $\lambda_1^+ \simeq 440$. Therefore, for the channel flow under consideration (see Chapter IV), the average dimensionless distance between the spanwise and streamwise structures are:

$$\frac{\Delta z}{\delta} = 100/Re_\tau = 0.156$$

and

$$\frac{\Delta x}{\delta} = 440/Re_\tau = 0.687$$

respectively. Here Re_τ is the Reynolds number based on shear velocity, u_τ and channel half width, δ and is 640 in our simulation.

Using the above values of X_1 and X_3 , and assuming that, at least four grid points are needed to resolve one wavelength (structure), we arrive at the following requirements for the number of grid points in x and z directions:

$$N_x = 37$$

$$N_z = 82$$

It is emphasized that the above values for N_x and N_z are based on ensemble averaged spacing of the structures. Hence for an adequate simulation of the important large scales, the following values for N_x and N_z are recommended (with due consideration to the capability of present computers):

$$N_x = 32$$

$$N_z = 128$$

In the present numerical simulation of turbulent channel flow, we have chosen the following values for the nondimensionalized streamwise and spanwise computational box lengths:

$$L_x = 2\pi$$

$$L_z = \frac{4}{3}\pi$$

The value of $L_z = \frac{4}{3}\pi$ is somewhat bigger than the above value for X_3/δ. This choice was made with due consideration to stability and resolution requirements of linear hydrodynamic stability theory (see Section 4.3). In addition, due to computer cost and storage limitations, we have used 16 grid points with uniform spacing, in each of the streamwise and spanwise directions. Therefore, the actual grid spacing used in these directions corresponds to $h_1^+ = 251$ and $h_3^+ = 168$ respectively. Hence, it is clear that we have inadequate resolution, particularly in the spanwise direction.

3.2 Numerical Differentiation

In the vertical direction, central differencing is employed with variable grid spacing $y_{j+1} = y_j + h_{j+1}$ where $h_j = y_j - y_{j-1}$ and $j = 1,2,\ldots,N$ (see Section 3.1). The partial derivatives for this case are the following expressions with the first truncation error term included:

$$\left(\frac{\partial f}{\partial y}\right)_j = \frac{f_{j+1}-f_{j-1}}{h_{j+1}+h_j} - \frac{1}{2}(h_{j+1}-h_j)\left(\frac{\partial^2 f}{\partial y^2}\right)_j + 0(h_j^2) \qquad (3.2)$$

$$\left(\frac{\partial^2 f}{\partial y^2}\right)_j = 2\left[\frac{f_{j-1}}{h_j(h_j+h_{j+1})} - \frac{f_j}{h_j\, h_{j+1}} + \frac{f_{j+1}}{h_{j+1}(h_j+h_{j+1})}\right]$$

$$- \frac{h_{j+1}-h_j}{3}\left(\frac{\partial^3 f}{\partial y^3}\right)_j + 0(h_j^2) \qquad (3.3)$$

Note that the second term of the right-hand side of Eq. (3.2) and (3.3) is the "extra error" introduced by the use of a nonuniform grid. In general, however, this term is very small if the grid size varies slowly (Blottner, 1974) (this is the case with 3.1). It can be easily shown (Blottner 1974) that a variable grid scheme is equivalent to a coordinate stretching method if a relation of the form of Eq. (3.1) is used to specify both the grid spacing in the variable grid method and the relationship between the coordinates for the stretching method. In both cases the derivatives are second order accurate in terms of $\Delta\xi$, i.e.,

$$\left(\frac{\partial f}{\partial y}\right)_j = \frac{f_{j+1} - f_{j-1}}{h_{j+1} + h_j} + 0(\Delta\xi^2) \tag{3.4}$$

and

$$\left(\frac{\partial^2 f}{\partial y^2}\right)_j = 2\left[\frac{f_{j-1}}{h_j(h_j+h_{j+1})} - \frac{f_j}{h_j\ h_{j+1}} + \frac{f_{j+1}}{h_{j+1}(h_j+h_{j+1})}\right] + 0(\Delta\xi^2) \tag{3.5}$$

In the streamwise and spanwise directions the pseudo-spectral method is used for the calculation of partial derivatives $\frac{\partial}{\partial x}$, $\frac{\partial}{\partial z}$, $\frac{\partial^2}{\partial z^2}$, etc. For a given number of grid points, the maximum accuracy is achieved by using this method (see Moin et al., 1978, for a discussion of the accuracy of numerical differentiation operators in terms of modified wave number concept). For periodic boundary conditions, which are of interest in x and z directions, we can represent a flow variable such as $\overline{u}$ by a discrete Fourier expansion

$$\overline{u}(x_1,x_2,x_3) = \sum_{n_1}\sum_{n_3} \hat{\overline{u}}(k_1,x_2,k_3)\ e^{i(k_1x_1+k_3x_3)} \tag{3.6}$$

where

$$k_i = \frac{2\pi}{(Nh)_j}\, n_i = \text{wave number in the } x_j \text{ direction}$$

$$N_j = \text{number of mesh points in the } j \text{ direction}$$

$$n_i = -\frac{N_i}{2}, \ldots, 0, 1, \ldots, \frac{N_i}{2} - 1$$

$$h_j = \text{mesh size in the } x_j \text{ direction.}$$

The sum extends over all n_1 and n_3. Suppose we wish to compute $\partial\bar{u}/\partial x_1$; we may regard (3.6) as an interpolation formula, treating x_1 as a continuous variable, and differentiate to obtain

$$\frac{\partial\bar{u}}{\partial x_1} = \sum_{n_1}\sum_{n_3} \hat{\bar{u}}(k_1,x_2,k_3) ik_1 e^{i(k_1x_1+k_3x_3)} \tag{3.7}$$

Multiplying both sides of (3.7) by $\exp(-ik_1'x_1 - ik_3'x_3)$, summing over all x_1 and x_3 , and using orthogonality, we get:

$$\widehat{\frac{\partial\bar{u}}{\partial x_1}} = ik_1 \hat{\bar{u}}(k_1,x_2,k_3) \tag{3.8}$$

Thus, in order to compute $\partial\bar{u}/\partial x_1$, we simply have to Fourier transform $\bar{u}$ in the x_1-direction, multiplying it by ik_1 , and take the inverse transform of the result; this is called the "pseudo-spectral" approach (Orszag (1972), Fox and Orszag (1973)). The use of pseudo-spectral method in x and z directions, partially addresses the grid resolution problem in these two directions.

For a limited number of problems with nonperiodic boundary conditions we can use some other set of orthogonal functions rather than $\{e^{ikx}\}$ (see Orszag, 1971). For completeness and for later use in this report, we conclude this section by describing the numerical differentiation using Chebyshev polynomials.

We can express a variable such as $f(y)$ by a discrete Chebyshev expansion

$$f(y) = \sum_{n=0}^{N}{}'' a_n T_n(y) \tag{3.9}$$

where $T_n(y)$ is the nth order Chebyshev polynomial of the first kind, and double prime denotes that the first and last terms are taken with factor ½ . Similarly, we can express the derivative of f , which is a polynomial of degree $N-1$, in terms of $T_n(y)$. We then write

$$\frac{\partial f}{\partial y} = \sum_{n=0}^{N-1}{}' \; b_n \, T_n(y) \tag{3.10}$$

and seek to compute the coefficients b_n in terms of a_n . It can be easily shown (see Fox and Parker, 1968) that the coefficients b_n are given by the following recurrence relations:

$$b_{n-1} - b_{n+1} = 2n \, a_n \qquad n=1,2,\ldots,N-2$$

$$b_{N-2} = 2(N-1)a_{N-1}$$

$$b_{N-1} = N \, a_{N-1} \tag{3.11}$$

Finally, we note that

$$T_n(\cos\theta) = \cos n\theta \tag{3.12}$$

Thus, the transformation $(y = \cos\theta)$ which is roughly adequate for boundary layer coordinate stretching, renders the evaluation of the Chebyshev expansion coefficients, a_n , particularly simple with the use of FFT routines.

3.3 Fundamental Numerical Problem

In this section we describe an inherent numerical problem associated with the fully explicit solution of the dynamical equations in primitive form in a bounded domain. Consider the momentum equations

$$\frac{\partial u_i}{\partial t} = -\frac{\partial p}{\partial x_i} + H_i \tag{3.13}$$

where H_i contains the viscous and convective terms. In the fully explicit (time advancing) numerical solution of (3.13) one normally specifies an arbitrary initial solenoidal velocity field satisfying the no-slip condition. Then, one proceeds to solve the appropriate Poisson equation for pressure obtained from the application of the divergence operator to the momentum equations to ensure that $\nabla \cdot \underset{\sim}{u} = 0$. The resulting pressure is then used together with the computed H_i in (3.13) to advance u_i in time. The Neumann boundary condition,

$$\frac{\partial p}{\partial \underset{\sim}{n}} = \nu \underset{\sim}{n} \cdot \nabla^2 \underset{\sim}{u} \qquad (3.14)$$

is normally used in conjunction with the Poisson equation for pressure. Here $\underset{\sim}{n}$ is a unit vector normal to the solid boundary. This condition is obtained from the normal momentum equation evaluated at the solid boundary.

With regard to the boundary treatment, one has two choices:

a) Enforce the no-slip condition, and time advance the velocity field via (3.13) only in the interior domain (not at the boundaries);

b) Time advance the velocity field throughout (interior domain as well as boundaries).

If one chooses (a); for the tangential momentum equations to be satisfied at the boundaries, the initial field would have to be such that the p it generates satisfies the Dirichelet condition

$$\frac{\partial p}{\partial \underset{\sim}{\tau}} = \nu \underset{\sim}{\tau} \cdot \nabla^2 \underset{\sim}{u} \qquad (3.15)$$

($\underset{\sim}{\tau}$ is a unit vector tangent to the solid boundary). The momentum equations in the directions tangential to the solid boundary will not necessarily be satisfied if the only constraints on the initial field are that it be solenoidal and satisfy the no-slip condition. Since the tangential momentum equations are not in general satisfied at the solid boundary, the Poisson equation will not be satisfied there either, and hence we conclude that in case (a) the continuity equation will not be satisfied at the boundary, $\left(\frac{\partial}{\partial \underset{\sim}{n}} (\underset{\sim}{n} \cdot \underset{\sim}{u}) \neq 0 \right)$. This can cause serious numerical instability.

On the other hand, if one chooses case (b), continuity will be satisfied everywhere, but the no-slip condition may not be satisfied, and this is unacceptable.

It should be noted that, if one uses the Dirichlet condition (3.15) as the pressure boundary condition then the Neumann condition (3.14) will not necessarily be satisfied and hence similar problems will arise in either approach (a) or (b).

In Appendix B we formally demonstrate the numerical problems addressed in this section. In addition, in Section 3.6 it will be shown that the numerical problems discussed here can be avoided if one uses three-point finite differences to approximate partial derivatives in the direction normal to the boundaries.

3.4 Consistency Conditions for the Initial Velocity Field

In this section, we present a set of consistency conditions* for the initial velocity field of the channel flow such that the Neumann and Dirichlet problems for the pressure have the same solution, i.e., we solve the problem addressed in Section 3.3.

Fourier transforming the Poisson equation in the streamwise and spanwise directions, we get:

$$\frac{d^2\hat{P}}{dy^2} - k^2\hat{P} = \hat{Q} \qquad (3.16)$$

---

*The consistency condition requirements conflict with the proven existence and uniqueness theorems for the Navier-Stokes equations. Therefore, we emphasize that the problems addressed in the previous section are purely numerical and mathematically there is no difficulty. Saffman (P. G. Saffman, 1978, private communication) points out that the fact that the Neumann problem does not satisfy the Dirichlet condition appears in the nonanalyticity of $\nabla^2 \underset{\sim}{u}$ on the boundary at $t = 0$, which can be interpreted physically as an initial vortex sheet diffusing from the boundary.

where $k^2 = k_1^2 + k_3^2$, and k_1 and k_3 are the wave numbers in streamwise and spanwise directions respectively. Here,

$$\hat{Q}(k_1,y,k_3) = \widehat{\left[\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_j} u_i u_j\right]} .$$

For $k^2 \neq 0$, the general solution of (3.16) is:

$$\hat{P} = \phi(y) + c_1 \sinh ky + c_2 \cosh ky \tag{3.17}$$

where:

$$\phi(y) = \left[\int_{-1}^{y} \frac{\hat{Q}\cosh k\eta}{k} d\eta\right] \sinh ky - \left[\int_{-1}^{y} \frac{\hat{Q}\sinh k\eta}{k} d\eta\right] \cosh ky$$

and, c_1 and c_2 are constants. Thus, for the Dirichlet and Neumann problems, we can determine c_1 and c_2 separately to get P_D and P_N which are the solutions of Dirichlet and Neumann problems respectively. Note that for the Dirichlet problem to have a solution, we must have

$$\left.\frac{\partial^2 P}{\partial x \partial z}\right|_{y=\pm 1} = \left.\frac{\partial^2 P}{\partial z \partial x}\right|_{y=\pm 1}$$

The above condition is equivalent to $\underset{\sim}{n} \cdot \nabla^2 \underset{\sim}{\omega} = 0$ on the boundaries $(y = \pm 1)$, or

$$\frac{\partial}{\partial z}\left[\left.\frac{\partial^2 u}{\partial y^2}\right|_{y=\pm 1}\right] = \frac{\partial}{\partial x}\left[\left.\frac{\partial^2 w}{\partial y^2}\right|_{y=\pm 1}\right] \tag{3.18}$$

or

$$ik_3 H_1(\pm 1) = ik_1 H_3(\pm 1)$$

where

$$H_1 = \nu \frac{\partial^2 \hat{u}}{\partial y^2} \quad \text{and} \quad H_3 = \nu \frac{\partial^2 \hat{w}}{\partial y^2}$$

and $\underset{\sim}{\omega}$ is the vorticity vector.

Equating P_D and P_N (after some algebra) we arrive at the following constraints for the initial velocity field:

$$\frac{H_3(1) - H_3(-1)}{ik_3} - \phi(1) = \frac{\tanh k}{k}\left[H_2(1) + H_2(-1) - \phi'(1)\right] \tag{3.19}$$

$$\frac{H_3(1) + H_3(-1)}{ik_3} - \phi(1) = \frac{\coth k}{k}\left[H_2(1) - H_2(-1) - \phi'(1)\right] \tag{3.20}$$

Therefore, for a successful, fully explicit numerical simulation, the initial velocity field must satisfy the following conditions:

- it must be solenoidal,
- it must satisfy the no-slip condition, and
- it must satisfy (3.18), (3.19), and (3.20).

Note that for $k_3 = 0$ and $k_1 \neq 0$, one can use (3.19) and (3.20) with the subscript 3 replaced by 1.

3.5 Conservation Properties

As was pointed out by Phillips (1959), numerical integration of the finite-difference analog of the Navier-Stokes equations may introduce nonlinear instabilities if proper care is not taken. Differencing the transport terms in the form of (2.5) will automatically conserve momentum in an inviscid flow. However, in general, the computation becomes unstable and the kinetic energy increases. This can happen in spite of the dissipative nature of τ_{ij} and the viscous terms. The nonlinear instability arises because the momentum conservative form does not necessarily guarantee energy conservation (in the absence of dissipation), and the effect of truncation errors on the energy is not negligible.

Moin et al. (1978) have shown that writing the dynamical equations in the form of (2.6) results in vorticity, momentum, and energy conservation for a large class of differencing schemes. Therefore, in all the calculations reported here, we use the dynamical equations in the form shown by Eqn. (2.6).

3.6 Explicit Time Advancing

By introducing one plane of grid points just outside of each boundary, one is able to obtain some degree of freedom. With proper use of this freedom, one can avoid the problem discussed in Section 3.3 (case a). The reader should be cautioned that here we are strictly referring to the explicit numerical solutions in which three point finite differences are used for the numerical differentiation. (However, the latter statement does not apply, for example, to the cases in which Chebyshev polynomials are used in a finite series expansion to represent a flow variable and its derivatives in the normal direction (see Sec. 3.2).) In practice, one can determine the normal velocity at the exterior point such that the continuity equation evaluated at the wall,

$$\left.\frac{\partial \bar{v}}{\partial y}\right|_{y=\pm 1} = 0 \tag{3.21}$$

is identically enforced. This velocity, in turn, is used in obtaining the Neumann boundary condition for pressure. For the proper choice of the numerical ∇^2 operator for the Poisson equation, the reader is referred to Moin et al. (1978).

For explicit time advancement, a second-order Adams-Bashforth method was used. It has been shown by Lilly (1965) that this method is weakly unstable, but the total spurious computational production of kinetic energy is small. The Adams-Bashforth formula for $\bar{u}_i$ at time step $n+1$ is

$$\bar{u}_i^{n+1} = \bar{u}_i^n + \Delta t \left(\frac{3}{2}\mathcal{L}_i^n - \frac{1}{2}\mathcal{L}_i^{n-1}\right) + O(\Delta t^3) \tag{3.22}$$

where

$$\mathcal{L}_i = -\overline{\bar{u}_j \left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i}\right)} - \frac{\partial \bar{P}}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} + \delta_{i1} + \frac{1}{Re_\tau}\frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j}$$

Using the above method, we have successfully integrated the governing equations for the numerical simulation of turbulent channel flow (not reported here). However, due to the presence of a very fine mesh near the boundaries, one is forced to use extremely small time steps. This stringent requirement is caused by the well-known numerical stability criterion of the diffusion equation.

3.7 A Semi-Implicit Numerical Scheme

As was mentioned in the previous section, due to the presence of diffusion terms in the governing equations, the time-step requirement of a fully explicit method becomes severe. To circumvent this difficulty, we have devised a semi-implicit algorithm. All the results reported here were obtained using this method. Thus, in what follows, we outline a method which treats part of the diffusion terms and pressure in the dynamical equations implicitly, and the remaining terms explicitly. The equation of continuity is solved directly.

Let us start with Eqn. (2.19), written in the following form:

$$\frac{\partial \bar{u}_i}{\partial t} = H_i - \frac{\partial \bar{P}}{\partial x_i} + \left(C_i \tilde{\nu}_T + \frac{1}{Re_\tau}\right) \frac{\partial^2 \bar{u}_i}{\partial x_2^2} \qquad \text{(no summation)} \tag{3.23}$$

where

$$H_i = -\overline{\bar{u}_j \left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i}\right)} + \frac{1}{Re_\tau}\left(\frac{\partial^2 \bar{u}_i}{\partial x_1^2} + \frac{\partial^2 \bar{u}_i}{\partial x_3^2}\right)$$

$$+ \frac{\partial}{\partial x_1}\left[\nu_T\left(\frac{\partial \bar{u}_i}{\partial x_1} + \frac{\partial \bar{u}_1}{\partial x_i}\right)\right] + \frac{\partial}{\partial x_3}\left[\nu_T\left(\frac{\partial \bar{u}_i}{\partial x_3} + \frac{\partial \bar{u}_3}{\partial x_i}\right)\right]$$

$$+ \left(\frac{\partial}{\partial x_2}\nu_T\right)\left(\frac{\partial \bar{u}_i}{\partial x_2} + \frac{\partial \bar{u}_2}{\partial x_i}\right) + \nu_T' \frac{\partial}{\partial x_2}\left(\frac{\partial \bar{u}_i}{\partial x_2} + \frac{\partial \bar{u}_2}{\partial x_i}\right) + \delta_{i1} + Q_i$$

$$Q_i = \tilde{\nu}_T \frac{\partial}{\partial x_2}\left(\frac{\partial \bar{u}_2}{\partial x_i}\right) d_i \qquad \text{(no summation)}$$

$$C_i = 1 + \delta_{i2}$$

$$d_i = 1 - \delta_{i2}$$

$$\tilde{\nu}_T = \langle \nu_T(x_1,x_2,x_3) \rangle_{x_1,x_3}$$

$\langle \ \rangle_{x_1,x_3}$ indicates the average of bracketed quantity in x_1-x_3 plane,

$$\nu_T' = \nu_T - \tilde{\nu}_T$$

The rationale for this decomposition of ν_T will be explained later in this section. For time advancing, we are going to use the Adams-Bashforth method (see Sect. 3.6) on H_i, and the Crank-Nicolson method (Richtmyer and Morton, 1967) on $\partial\overline{P}/\partial x_i$ and $\partial^2\overline{u}_i/\partial x_2^2$, in the right-hand side of Eqn. (3.23). For convenience, we evaluated $\tilde{\nu}_T$ at time step n. Thus, we have:

$$\overline{u}_i^{n+1} = \overline{u}_i^n + \Delta t\left(\frac{3}{2} H_i^n - \frac{1}{2} H_i^{n-1}\right) - \frac{\Delta t}{2}\left(\frac{\partial\overline{P}^{n+1}}{\partial x_i} + \frac{\partial\overline{P}^n}{\partial x_i}\right)$$

$$+ \left(\frac{1}{Re_\tau} + C_i\tilde{\nu}_T^n\right)\frac{\Delta t}{2}\left(\frac{\partial^2\overline{u}_i^{n+1}}{\partial x_2^2} + \frac{\partial^2\overline{u}_i^n}{\partial x_2^2}\right) \qquad (3.24)$$

(no summation)

Let

$$\beta_i(x_2) = -\frac{2/\Delta t}{\left(\frac{1}{Re_\tau} + C_i\tilde{\nu}_T^n(x_2)\right)} \ ;$$

rearrangement of Eqn. (3.24) yields:

$$\frac{\partial^2\overline{u}_i^{n+1}}{\partial x_2^2} + \beta_i\overline{u}_i^{n+1} + \beta_i\frac{\Delta t}{2}\frac{\partial\overline{P}^{n+1}}{\partial x_i} = \beta_i\overline{u}_i^n + \beta_i\Delta t\left(\frac{3}{2} H_i^n - \frac{1}{2} H_i^{n-1}\right)$$

$$- \beta_i\frac{\Delta t}{2}\frac{\partial\overline{P}^n}{\partial x_i} - \frac{\partial^2\overline{u}_i^n}{\partial x_2^2} \qquad (3.25)$$

(no summation)

Finally, we write the continuity equation at time step $n + 1$:

$$\frac{\partial\overline{u}_i^{n+1}}{\partial x_i} = 0 \qquad (3.26)$$

Now let us Fourier transform Eqns. (3.25) and (3.26) in x_1 and x_3 directions. This transformation converts the set of partial differential equations (3.25) and (3.26) to a set of ordinary equations for every pair of Fourier modes k_1, k_3 with x_2 as the independent variable. Note that the dependent variables have superscript $n+1$. In the remainder of this section all the dependent variables are to be interpreted as two-dimensional Fourier transformed quantities. Fourier transforming equations (3.25) and (3.26) results in the following set of ordinary differential equations for the dependent variables:

$$\frac{\partial^2 u_1^{n+1}}{\partial x_2^2} + \beta_1 u_1^{n+1} + ik_1\beta_1 \frac{\Delta t}{2} P^{n+1} = \beta_1 u_1^n + \beta_1 \frac{\Delta t}{2}\left(3H_1^n - H_1^{n-1}\right) - ik_1\beta_1 \frac{\Delta t}{2} P^n - \frac{\partial^2 u_1^n}{\partial x_2^2} \tag{3.27a}$$

$$\frac{\partial^2 u_2^{n+1}}{\partial x_2^2} + \beta_2 u_2^{n+1} + \beta_2 \frac{\Delta t}{2} \frac{\partial P^{n+1}}{\partial x_2} = \beta_2 u_2^n + \beta_2 \frac{\Delta t}{2}\left(3H_2^n - H_2^{n-1}\right) - \beta_2 \frac{\Delta t}{2} \frac{\partial P^n}{\partial x_2} - \frac{\partial^2 u_2^n}{\partial x_2^2} \tag{3.27b}$$

$$\frac{\partial^2 u_3^{n+1}}{\partial x_2^2} + \beta_3 u_3^{n+1} + ik_3\beta_3 \frac{\Delta t}{2} P^{n+1} = \beta_3 u_3^n + \beta_3 \frac{\Delta t}{2}\left(3H_3^n - H_3^{n-1}\right) - ik_3\beta_3 \frac{\Delta t}{2} P^n - \frac{\partial^2 u_3^n}{\partial x_2^2} \tag{3.27c}$$

$$ik_1 u_1^{n+1} + \frac{\partial u_2^{n+1}}{\partial x_2} + ik_3 u_3^{n+1} = 0 \tag{3.27d}$$

Thus, for every pair of k_1 and k_3 we have four coupled linear ordinary differential equations with $u_1^{n+1}(k_1,x_2,k_3)$, $u_2^{n+1}(k_1,x_2,k_3)$, $u_3^{n+1}(k_1,x_2,k_3)$, and $P^{n+1}(k_1,x_2,k_3)$ as unknowns. Note that, with no further complications, one can treat more terms in Eqn. (2.19) (e.g., $\frac{1}{Re_\tau}\frac{\partial^2}{\partial x^2}$, $\frac{1}{Re_\tau}\frac{\partial^2}{\partial z^2}$, etc.) implicitly.

Finally, it should be mentioned that, in order to avoid evaluating complicated convolution sums, we have decomposed ν_T, to its planar average, $\tilde{\nu}_T(y)$ and "fluctuating" component $\nu'_T(x_1,x_2,x_3)$. We have used explicit time advancing for $\nu'_T\ (\partial^2 u_i/\partial x_2^2)$, whereas $\tilde{\nu}_T(\partial^2 u_i/\partial x_2^2)$ is advanced by a partial implicit scheme. This decomposition of ν_T may not be an optimum one from the standpoint of numerical stability and accuracy. Other choices are possible. For example, one can decompose ν_T as follows:

$$\nu_T(x_1,x_2,x_3) = \max_{x_1,x_2,x_3} (\nu_T) + \nu''_T(x_1,x_2,x_3)$$

Although we did not incorporate any other decomposition than the one used here, relatively simple numerical experiments with the diffusion equation may result in a better decomposition for ν_T.

3.8 Finite-Difference Formulation and Boundary Conditions

In order to solve Eqns. (3.27) numerically, we use the finite difference operators (3.2) and (3.3) to approximate $\partial/\partial x_2$ and $\partial^2/\partial x_2^2$. Having done this, we shall have a set of linear algebraic equations for the Fourier transform of the dependent variables. This system of algebraic equations is of block tri-diagonal form and can be solved very efficiently. However, in order to close the system we must provide a set of boundary conditions, i.e., we have to specify the values of u_1, u_2, u_3, and P at the solid boundaries.

Implementation of velocity boundary conditions poses no problem; we simply set the value of the velocity vector at zero on the walls. In order to obtain the pressure boundary conditions, we note that evaluation of Eqn. (3.27b) at the solid boundaries yields:

$$\left[\frac{\partial^2 u_2^{n+1}}{\partial x_2^2} + \beta_2 \frac{\Delta t}{2} \frac{\partial P^{n+1}}{\partial x_2}\right]_{x_2=\pm 1} = -\left[\beta_2 \frac{\Delta t}{2} \frac{\partial P^n}{\partial x_2} + \frac{\partial^2 u_2^n}{\partial x_2^2}\right]_{x_2=\pm 1}$$

Consider the following Neumann boundary condition for pressure:

$$\left.\frac{\partial P}{\partial x_2}\right|_{x_2=\pm 1} = \frac{1}{Re_{\tau'}} \left.\frac{\partial^2 u_2}{\partial x_2^2}\right|_{x_2=\pm 1} \qquad (3.28)$$

Equation (3.28) was obtained from the Fourier transform of Eqn. (2.19, i = 2), and evaluated at the solid boundaries. It is clear that this equation is consistent with the numerical analog of that equation (3.27b) evaluated at the walls. Note that

$$\beta_2 \left.\frac{\Delta t}{2}\right|_{x_2=\pm 1} = - Re_\tau \quad .$$

Thus, we formally use Eqn. (3.28) as the pressure boundary condition. However, for closure the finite-difference equations require the value of pressure at the boundaries, not its normal derivative. For this we use the following difference relation in conjunction with the difference analog of Eqn. (3.28):

$$\frac{1}{2}\left(\left.\frac{\partial P}{\partial x_2}\right|_{j=2} + \left.\frac{\partial P}{\partial x_2}\right|_{j=3}\right) = \left.\frac{P_{j+1} - P_j}{h_{j+1}}\right|_{j=2} + O(h_2^2) \qquad (3.29)$$

where $h_j = x_{2_j} - x_{2_{j-1}}$. $j = 2$ indicates the grid point on the lower wall.

Substituting the finite-difference analog of Eqn. (3.28) into the left-hand side of Eqn. (3.29) and using the finite-difference form of the continuity equation at the wall, we obtain:

$$P_2 = \left[\frac{2P_3}{h_3} - \frac{P_4}{(h_3 + h_4)} - \frac{2u_{2_3}}{Re_\tau h_2 h_3}\right] \Bigg/ \left(\frac{2}{h_3} - \frac{1}{h_4 + h_3}\right) \qquad (3.30)$$

An analogous relation is used for the value of the pressure at the upper wall $(j = N - 1)$. Note that the pressure is still indeterminate by a constant, as it should be due to the use of Neumann boundary conditions; i.e., we are not using Dirichlet boundary conditions.

In the case $k_1 = k_3 = 0$, a special solution technique must be undertaken. First observe that in this special case Eqns. (3.27a) and (3.27c) are independent of each other and Eqns. (3.27b) and (3.27d). Furthermore, the former two equations are of simple tridiagonal form and can be solved directly to yield $u_1^{n+1}(0,x_2,0)$ and $u_3^{n+1}(0,x_2,0)$. Second, the continuity equation together with the boundary conditions for u_2 yield

$$u_2(0,x_2,0) = 0 \tag{3.31}$$

Since pressure is indeterminate by a constant, let

$$P(0,x_2,0)\Big|_{x_2=-1} = 0 \tag{3.32}$$

Using Eqns. (3.30), (3.31), and (3.32) in conjunction with the finite difference analog of Eqn. (3.27b) allows one to solve for $P^{n+1}(0,x_{2_j},0)$, $j = 3,4,\ldots,N+1$.

Before concluding this section, we emphasize that, in obtaining the pressure boundary conditions, we used a momentum equation evaluated at the boundary. We were able to do this because the finite difference equations are generally enforced inside the spatial domain and not on its boundaries. Consequently, we did not use a redundant equation. Consider for a moment a hypothetical case in which we have the means to integrate the governing equations of motion analytically. In this case, the equations of motion are and should be valid at the boundaries as well as inside the domain (we do not have any singularity at the boundaries). So, in this case, use of momentum equations for the pressure boundary conditions will not provide any new information. The roots of this apparent dilemma lie in the basic physics of fluid mechanics. The fact is that physics does not provide a priori boundary conditions for pressure.

A manifestation of this dilemma will appear if, for example, Chebyshev polynomials are used in a finite series expansion to represent a flow variable in the y direction (see Section 3.2). However, since the equation of continuity is solved directly, it appears that the numerical problems which were addressed in Section 3.2 will not cause any difficulty if one uses Chebyshev polynomials in conjunction with the semi-implicit scheme developed here.

3.9 Computational Details

The numerical solution of the equations described here (see also the next chapter) were carried out on the CDC 7600 computer at NASA-Ames Research Center. The dimensionless time step, during most of the calculations, was set at $\Delta t = 0.001$. Throughout the computations reported here, the values of the following quantities,

$$c_1(t) = \mathrm{Max}\left\{\Delta t\left[\left|\frac{\bar{u}}{h_1}\right| + \left|\frac{\bar{v}}{h_2(y)}\right| + \left|\frac{\bar{w}}{h_3}\right|\right]\right\}$$

and

$$c_2(t) = \mathrm{Max}\left\{\Delta t\ \frac{\left|\nu_T - \langle \nu_T \rangle_{x_1,x_3}\right|}{h_2^2(y)}\right\}$$

did not exceed 0.3 and 0.08, respectively. In addition, the numerical stability was checked by a 200-step numerical integration in which the value of $\Delta t = 0.0005$ was used. The computer-generated results of this run agreed (within two significant figures) with the corresponding numerical integration in which the value of $\Delta t = 0.001$ was used. Comparison was made at the same total time of integration.

The computer time per time step was approximately 20 seconds (CPU time). However, the present computer program is not an optimum one, and we believe that at least a 25% savings in computer time can be achieved by some modifications of this program.

Finally, it should be noted that, in the present computation, approximately 80% of the small-core memory and only 50% of the available large-core memory of the CDC 7600 was used. Therefore, a computation with twice as many grid points as the present one is possible using the available core memory of the CDC 7600.

Chapter IV

INCOMPRESSIBLE TURBULENT CHANNEL FLOW

4.1 Physical Parameters

In order to solve Eqns. (2.19), we need to specify Re_τ, Reynolds number based on channel half-width δ and shear velocity u_τ. In the present numerical simulation of turbulent channel flow, $Re_\tau = 640.25$ was used. In their experimental investigation of the mechanics of organized waves, Hussain and Reynolds (1975) considered a channel flow with the same Reynolds number. The mean flow parameters of their experiment are listed below.

$$Re = 13800$$

$$\frac{u_\tau}{U_o} = 0.0464$$

$$\frac{U_m}{U_o} = 0.881$$

$$U_o = 21.9 \quad \text{ft/sec (6.67 m/sec)}$$

where Re is the Reynolds number based on channel half-width, δ, and the centerline velocity, U_o; U_m is the mean profile average velocity, and u_τ is the shear velocity.

4.2 Initial Condition

A number of initial velocity fields were explored. With the simple sub-grid scale model used, it is important that the initial turbulence field be able to continually extract energy from the mean flow in order that a statistically steady solution develop. For this purpose, we employed the governing equations of small disturbances used in hydrodynamic stability theory (other choices are possible) to obtain a velocity field with negative Reynolds stress.

The equations for a small wave disturbence $\tilde{u}_i$ on a parallel mean flow $U(y)$ are (Lin, 1955, Eqn. (1.3.9)):

$$\tilde{u}_i = \hat{u}_i(y)\, e^{i(\alpha x + \beta z - \alpha c t)} + \text{conj} \tag{4.1a}$$

$$i\alpha\hat{u}_1 + i\beta\hat{u}_3 + D\hat{u}_2 = 0 \tag{4.1b}$$

$$i\omega u_1 + U i\alpha\hat{u}_1 + D\overline{U} \cdot \hat{u}_2 = -\, i\alpha\hat{P} + \frac{1}{Re}(D^2 - k^2)\,\hat{u}_1 \tag{4.1c}$$

$$i\omega\hat{u}_2 + U i\alpha\hat{u}_2 = -\, D\hat{P} + \frac{1}{Re}(D^2 - k^2)\,\hat{u}_2 \tag{4.1d}$$

$$i\omega\hat{u}_3 + U i\alpha\hat{u}_3 = -\, i\beta\hat{P} + \frac{1}{Re}(D^2 - k^2)\,\hat{u}_3 \tag{4.1e}$$

Here $\omega = -\alpha c$ is the (complex) frequency, and $D = d/dy$.

The Squire transformation (Lin, Eqn. (3.1)),

$$k^2 = \alpha^2 + \beta^2 \tag{4.2a}$$

$$\hat{v} = \hat{u}_2 \tag{4.2b}$$

$$\alpha\hat{u}_1 + \beta\hat{u}_3 = k\hat{u} \tag{4.2c}$$

permits reduction to a single fourth-order equation for $\hat{v}$, the Orr-Sommerfield equation (Lin, Eqn. (1.3.15)):

$$(D^2 - k^2)^2\hat{v} = i\alpha Re\left\{(U - c)\,(D^2 - k^2)\,\hat{v} - D^2\overline{u} \cdot \hat{v}\right\} \tag{4.3}$$

For a given set of α, Re, β, and $U(y)$, (4.3) is solved numerically using the algorithm of Lee and Reynolds (1967).

After final calculation of $\hat{v}$, $\hat{u}$ is calculated from (4.1b), and $\hat{P}$ is calculated from (4.1c) and (4.1e). The results are then used to solve for $\hat{u}_1$, via (4.1c). Solution of (4.1c) is carried out numerically using a second-order algorithm. Starting at the centerline of the channel, two solutions, each satisfying the centerline boundary conditions (here we are primarily concerned with symmetric $\hat{u}_2$ and anti-symmetric $\hat{u}_1$ and $\hat{u}_3$) are constructed using the Kaplan filtering technique to maintain linear independence. These two solutions are then combined to satisfy the wall boundary conditions. The eigenvalues are automatically adjusted until an eigensolution is obtained.

For the Reynolds number under consideration ($Re_\tau = 640.25$) and with proper choice of α, β, and $U(y)$, one can obtain a set of $\tilde{u}_1$, $\tilde{u}_2$, and $\tilde{u}_3$ such that the corresponding Reynolds stress has the same sign as $-D\bar{u}$. This corresponds to an unstable disturbance from the view of hydrodynamic stability theory. The resulting three-dimensional disturbance extracts energy from the mean flow in a continuous fashion. In the present study we have used $\alpha = 1.0$, $\beta = 1.5$, and the mean velocity profile:

$$U(y) = 10(1 + \cos \pi y)$$

for the generation of initial disturbances.

This profile was chosen with due consideration to the proper representation of the resulting disturbances on the grid system in the normal direction. In addition, note that the above mean velocity profile has inflection points (at $y = \pm \frac{1}{2}$) which produces Kelvin-Helmholtz type instability.

In order to avoid a net momentum in the spanwise direction, one can add two oblique waves with the same amplitude that are traveling in the directions which are at angles of ϕ and $-\phi$ with the streamwise, x, direction. Combining two oblique waves in this fashion yields a set of streamwise vortices (roll cells). Thus, the following velocity field was used as the major part of the initial disturbance (initial large eddies):

$$\tilde{u}_1(x,y,z) = A[\hat{u}_1(y) \cos \beta z \, e^{i\alpha x} + \text{conj}]$$

$$\tilde{u}_2(x,y,z) = A[\hat{u}_2(y) \cos \beta z \, e^{i\alpha x} + \text{conj}]$$

$$\tilde{u}_3(x,y,z) = A[\hat{u}_3(y) \sin \beta z \, e^{i\alpha x} + \text{conj}]$$

Here, A is a constant, $\alpha = 1.0$, $\beta = 1.5$, and $\hat{u}_i(y)$ $(i = 1,2,3)$ are the eigensolutions of the linearized equations. In order to allow the development of all the waves that can be resolved on the grid system, a solenoidal velocity field with random phase was added to the above velocity field. Furthermore, to ensure the initial dominance of the $\tilde{u}_i$ field, the amplitude of random field was about 10% of the maximum aplitude of $\tilde{u}_i$. Finally, in order to avoid a very long time numerical integration, the measured mean velocity profile of Hussain and Reynolds (1975) was used as the <u>initial</u> mean velocity.

4.3 Preliminary Numerical Experiments

In the following three sections we shall present and discuss various calculated quantities pertinent to turbulent channel flow. The results will consist of running time averaged mean velocity profile and turbulence statistics, horizontally (xz plane) averaged turbulent quantities, and some instantaneous velocity profiles. However, first, it is instructive to discuss some of our initial numerical experiments (failures).

In our first integration attempt, we observed that the absolute value of the horizontally averaged Reynolds stress, $< \overline{u}\overline{v} >$, decreased continuously in time. This vanishing trend occurred in spite of the fact that the Reynolds stress profile was below the expected value. The total time of integration was approximately 1 nondimensional unit, and the value of eddy viscosity constant, C_s, was specified to be 0.2 (see Moin et al., 1978). It is interesting to note that the profiles of $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$ were generally increasing, and the corresponding profiles of $< \overline{v}^2 >^{1/2}$ were decreasing <u>slightly</u>. In other words, the correlation between $(\overline{u} - < \overline{u} >)$ and $\overline{v}$, and not the respective intensities, had a rapid vanishing trend. At this point it was determined that the effective Reynolds number (taking the eddy viscosity into account) was probably too small for a small amplitude disturbance to grow. With this in mind, and noting that the production of Reynolds stress is directly proportional to $< \overline{v}^2 >$, the existing turbulent velocities were multiplied by a factor of two (and the Reynolds stress was amplified by a factor of four). Note that no changes were made to the final mean velocity profile, $< \overline{u} >$. In fact, at this time $< \overline{u} >$ was deviated considerably from its original profile.

Using the resulting velocity field as a new initial condition (in what follows, we shall call this velocity field "field A"), we carried out two parallel computations, one with $C_s = 0.44$ and the other with $C_s = 0.2$. In the former case, the Reynolds stress profile grew continuously for a nondimensional time, t, of 0.3. However, during a further integration period $(t = 0.7)$, it decayed drastically to a vanishing level. Thus, it was concluded that the value of 0.44 for the subgrid scale model constant is too large, causing turbulent motions to damp out.

The results to be presented in the following sections were obtained using the value of 0.2 for C_s. This value is probably not the omptimum one (more likely the optimum value is between 0.2 and 0.3); however, in the absence of a more rigorous subgrid scale model formulation, further adjustments of C_s seem to be unjustified.

4.4 A Time History of the Horizontally Averaged Turbulent Quantities

As was pointed out in the previous section, we use the velocity field A as the new initial condition. Fig. 4.1 shows the horizontally averaged resolvable shear stress $<\overline{uv}>$ of this field. For purposes of discussion, we concentrate on the lower half of the channel in this section. Furthermore, due to the relationship between the materials to be discussed herein and the bursting process in a turbulent boundary layer, virtually all of our discussion will be concerned with the region near the (lower) wall.

Figure 4.2 shows the $<\overline{uv}>$ profile at the non-dimensional time*, $t = 0.45$. It can be seen that the resolvable shear stress profile has increased considerably. In particular, near the wall it has increased significantly beyond the expected equilibrium (time-averaged) value. Figs. 4.3, 4.4, and 4.5 show the profiles of the same quantity $(<\overline{uv}>)$ at three later times $(t = .65, .85, 1.05,$ respectively). It is clear that, especially in the region $-.95 < y < -.7$, a dynamic process exists which nearly repeats itself in time. If we carry out the integration still further, we see the same behavior (almost cyclic) in the $<\overline{uv}>$ profile.

*One nondimensional time unit corresponds approximately to the time in which a particle moving with centerline velocity travels 22 channel half-widths.

Figs. 4.6 and 4.7 show the vertical distribution of $< \overline{uv} >$ obtained at two later times corresponding to $t = 1.425$ and $t = 2.025$, respectively.

Since the production of the resolvable turbulent kinetic energy is directly proportional to $< \overline{uv} >$, it should be interesting to study the effect of the cyclic behavior of $< \overline{uv} >$ on $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$. Figs. 4.8, 4.9, and 4.10 show the profiles of $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$ in the vicinity of the wall $(y^+ < 128)$. They correspond to the $< \overline{uv} >$ profiles presented in Figs. 4.5, 4.6, and 4.7, respectively. Examination of these figures shows clearly the effect of production on the $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$ profile. It can be seen that, during the times at which $< \overline{uv} >$ has a relatively high value, the corresponding $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$ profile possesses a pronounced local maximum. It is interesting to note that, during the quiescent (low $< \overline{uv} >$) periods, the turbulence energy level is still quite large. In fact, a close examination of Figs. 4.9 and 4.10 reveals that, during these times, the energy that gave rise to the local maxima is distributed throughout the $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$ profile. This results in a wide maximum (in contrast to a sharp local one) in $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$.

During their investigation of the "bursting" process in a turbulent boundary layer, Kim et al. (1968) showed that, while the bursting process indeed contributes to the turbulent energy, its main effect is to provide turbulence with u' and v' in proper phase to give the large turbulence stress required for an increase in production. This is precisely what is observed here. To clarify this point, consider, for example, Figs. 4.6 and 4.7. If we focus out attention on the vicinity of $y^+ \simeq 64$ $(y \simeq -.90)$, we see that the value of $< \overline{uv} >$ in Fig. 4.6 is about twice the corresponding value in Fig. 4.7. On the other hand, the corresponding value of $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$ in Fig. 4.9 is only 6% higher than the one in Fig. 4.10. And the corresponding values of $< \overline{v}^2 >^{1/2}$ (Fig. 4.11) and $< \overline{w}^2 >^{1/2}$ (Fig. 4.12) show no significant change during this period. This is expected, since the governing equations of $< \overline{v}^2 >^{1/2}$ and $< \overline{w}^2 >^{1/2}$ do not contain direct production terms. These quantities can only be fed by the inter-component transfer mechanism, which is generally a slow process.

We conclude this section by considering, once again, our initial numerical experiment (see Section 4.3). Recall that, during the first integration attempt, $< \overline{uv} >$ had a rapid vanishing trend while the individual components $< (\overline{u} - < \overline{u} >)^2 >^{1/2}$ and $< \overline{v}^2 >^{1/2}$ did not (the latter had a slight decreasing trend). With this and the discussion of the present section in mind, one can see the importance of the phase relationship between $(\overline{u} - < \overline{u} >)$ and $\overline{v}$. Indeed, the correlation between $(\overline{u} - < \overline{u} >)$ and $\overline{v}$ is the essential factor for the maintenance of turbulence. We believe (on the basis of a cursory scan) that the increase in $< \overline{uv} >$ is also highly localized in space.

It should be noted that, in a computation with a large number of mesh points in the horizontal planes, the transitory behavior of $< \overline{uv} >$ described in this section, will not occur. In this case, the horizontal averaging is approximately equivalent to long-time averaging; and in order to study the relationship of the bursting process to the turbulence stress, one should study the time history of the $(\overline{u} - < \overline{u} >)\ \overline{v}$ profile at one (x,z) location. Such a study, in turn, would yield the mean bursting frequency.

4.5 Detailed Flow Structures

In this section we examine some of the detailed flow patterns. Particular attention will be given to instantaneous velocity profiles. Fig. 4.13 shows typical instantaneous streamwise velocity profiles, $\overline{u}$. These profiels are obtained at the same location $(x = 0,\ z = 13\ h_3)$, but at two different times $(t = 1.625,\ t = 1.825)$. For comparison, the mean velocity profile is also included. Fig. 4.14 shows the corresponding normal velocity profiles, obtained at the same location and times. Examination of these figures reveals that the profile with a momentum defect (with respect to the mean) corresponds to a case in which fluid is being ejected from the wall $(\overline{v} > 0)$, while the profile with excess momentum corresponds to a case where the flow is toward the wall $(\overline{v} < 0)$. In addition, both pairs $((\overline{u} - < \overline{u} >) > 0,\ \overline{v} < 0)$ and $((\overline{u} - < \overline{u} >) < 0,\ \overline{v} > 0)$ have positive contributions to the resolvable Reynolds stress and, hence, they contribute to the production of turbulence.

The velocity profiles presented here are in good qualitative agreement with the flow visualization data of Kim et al. (1968) and Grass (1971). In their study of the bursting process in a turbulent boundary layer, Kim et al. observed that during the gradual lift-up of low speed streaks from the sublayer, inflectional instantaneous velocity profiles were formed. In fact, the appearance of the inflectional profile was used as one of their criteria for the detection of the bursts.

Using the terminology of Grass, the $\bar{u}$ profile with momentum defect corresponds to the ejection phase of the bursting process while the profile with excess momentum corresponds to the inrush phase (sweep). In the lower left-hand corner of Fig. 4.13, we have included the instantaneous velocity profiles from the measurements of Grass (1971) in a flow over a smooth flat plate. In Figs. 4.15 and 4.16, the same quantities as in Figs. 4.13 and 4.14 are plotted, but they are obtained at a different location and at different times $(x = 10\ h_1,\ \ z = 10\ h_3,\ \ t = 1.05,\ 1.275)$. The same behavior (qualitatively) as in Figs. 4.13 and 4.14 are displayed by Figs. 4.15 and 4.16. Fig. 4.17 shows the instantaneous streamwise velocity profiles obtained at time $t = 2.025$, but at two different (x,z) locations. This figure, together with Figs. 4.13 and 4.15, clearly demonstrate the highly three-dimensional and unsteady nature of this flow.

The reader is cautioned against establishing a direct relationship between the times, t, at which the instantaneous profiles are presented here, and the corresponding times at which $< \bar{u}\bar{v} >$ assumes a relatively high or low value (see the previous section). Recall that in this section instantaneous velocity profiles were presented at one (x,z) location, while in the previous section we were concerned with the planar averages of $< \bar{u}\bar{v} >$. At most we can say that, during the times at which $< \bar{u}\bar{v} >$ has a relatively high value near the wall, there are more locations where the relationship between the $\bar{u}$ and $\bar{v}$ profiles are the same as those shown in Figs. 4.13 and 4.14 $(((\bar{u} - < \bar{u} >) > 0,\ \ \bar{v} < 0)$ or $((\bar{u} - < \bar{u} >) < 0,\ \ v > 0))$. This is in contrast to the times at which $< \bar{u}\bar{v} >$ has a relatively low value.

At this point, let us consider the spanwise instantaneous velocity profiles. Figs. 4.18 and 4.19 show a typical spanwise variation of the streamwise velocity $\bar{u}$ in the vicinity of the lower wall (second grid point away from the wall, $y^+ = 3.85$) at eight consecutive streamwise locations. The profiles presented here are obtained at time $t = 1.05$. These figures demonstrate distinct regions of high-speed fluid located adjacent to the low-speed ones. In addition, these profiles clearly show the long streamwise extent of the high- and low-speed streaks. In their visual studies, Runstadler et al. (1959, 1963) (see Section 1.2) demonstrated that the viscous sublayer consists of relatively coherent structures of low- and high-speed streaks alternating in the spanwise direction over the entire wall. It appears, therefore, that at least there is a qualitative agreement between the calculated results and the laboratory observations. Figs. 4.20 and 4.21 show the spanwise profiles of $\bar{u}$ at the same locations as in Figs. 4.18 and 4.19, but at time $t = 1.425$. Once again, these profiles show the coherent structures of alternating low-and high-speed streaks. Note that the profiles shown in Figs. 4.20 and 4.21 are generally different in magnitude and details of structures from those presented in Figs. 4.18 and 4.19 (see, for example, the profiles at ($x = 0$, and $x = 4\ h_1$). Fig. 4.22 shows typical spanwise variation of $\bar{v}$ and $\bar{w}$, obtained at $y^+ = 3.85$, $t = 1.05$, and $x = 4\ h_1$. The rapid spanwise variations of $\bar{v}$ and $\bar{w}$ clearly show the lack of grid resolution in the z direction (see the following discussion). Nevertheless, these profiles demonstrate, once again, that the viscous sublayer is the region of high flow activity, and it is three-dimensional. In addition, the spanwise variations of $\bar{v}$ indicate the distinct presence of secondary longitudinal vortices in the wall region.

Before concluding our present discussion of the spanwise velocity profiles, it is appropriate to make a comment about the grid resolution. Examination of the spanwise velocity profiles, in particular $\bar{v}$ and $\bar{w}$, seems to show that a better resolution in the z direction is required (see Section 3.1 and also note that our streak spacings are far larger than experimental observations). In other words, more grid points in the spanwise direction are necessary to represent the relatively rapid variations of the velocities (streaks) properly. This is necessary in spite of the fact that the pseudo-spectral method is used for numerical differentiation in the z direction. However, since the eddies away from the boundaries are larger than the ones near the walls (see Fig. 4.23), it is probably sufficient to have more grid points just in the vicinity of the walls. This requires a non-rectangular grid system (conical), which is generally accompanied by computational difficulties. Finally, Fig. 4.24 shows typical streamwise variations of $\bar{v}$ and $\bar{w}$ which are obtained at $t = 1.05$, $y^+ = 3.85$, and $z = 8h_3$. Note that, in spite of the fact that these profiles are obtained at the same plane as those in Fig. 4.22, the streamwise grid resolution seems to be adequate. However, it appears that the streamwise extent of the computational box, L_x, is too small.

4.6 Running Time Average of Mean Velocity Profile and Turbulent Statistics

In this section, we shall present the calculated mean velocity profile and turbulence quantities, averaged over horizontal planes and in time. The total averaging time is about one dimensionless time unit, which is much smaller than corresponding time intervals commonly used in laboratory measurements. However, the horizontal averaging should somewhat improve the overall statistical sample. In addition, note that, during the time interval used for the averaging $(1.05 < t < 2.025)$, the resolvable shear stress profile $< \overline{u}\,\overline{v} >$ traversed (roughly) one cycle (see Sect. 4.4).

Vertical profiles of the resolvable mean Reynolds stress, $< \overline{u}\,\overline{v} >$ (unless otherwise stated in this section, $< \; >$ indicates horizontal as well as time averaging), and the total Reynolds stress

$$< \overline{u}\,\overline{v} > + < - \nu_T \left(\frac{\partial \overline{u}}{\partial y} + \frac{\partial \overline{v}}{\partial x} \right) >$$

are shown in Figs. 4.25 and 4.26a. These profiles indicate that an approximately steady mean velocity is obtained. In other words, the average Reynolds stress profile has nearly attained the equilibrium shape which balances the downstream mean pressure gradient in the regions away from the walls. In the vicinity of the walls, the viscous stresses are significant, and they, together with the total Reynolds stress, balance the mean pressure gradient. Moreover, it should be noted that the subgrid scale cobtribution to the total Reynolds stress is significant only in the vicinity of the walls (see Figs. 4.25, 4.26a, and 4.26b).

Figure 4.27 shows the profile of $<\bar{u}>$, the mean velocity, averaged over both halves of the channel. The latter averaging was performed in order to improve the overall statistical sample*. The calculated mean velocity profile shows a distinct logarithmic region. In addition, the agreement with experimental data is satisfactory.

Figures 4.28, 4.29, and 4.30 show the profiles of the resolvable and total turbulent intensities averaged over both halves of the channel**. The contribution of the subgrid scale motions to the turbulent intensities is obtained from Eqn. (2.8) and from

$$\frac{1}{3} < \overline{u_i' u_i'} > \; = \; < \frac{2}{3} \nu_T^2 / (C\ell)^2 > \tag{4.4}$$

$$C = .094$$

(see Moin et al., 1978, or Lilly, 1967).

It should be noted, however, that due to the presence of a relatively coarse grid and the high degree of anisotropy in the channel flow, the validity of Eqn. (4.4) is questionable, especially in the vicinity of the walls. For comparison, we have also included some of the available experimental data in Figs. 4.28, 4.29, and 4.30. Examination of these figures reveals that, aside from a relatively high value of $< \bar{v}^2 >^{1/2}$ in the vicinity of the channel centerline, the qualitative behavior and the relative magnitudes of the turbulent intensity profiles are in accord with the experimental measurements. The quantitative agreement of calculated turbulent intensities with experimental measurements is good for $< (u - < u >)^2 >^{1/2}$ and $< w^2 >^{1/2}$ and fair for $< v^2 >^{1/2}$.

One may note that the subgrid scale contribution to the total streamwise and spanwise turbulent intensities is relatively small. However, Fig. 4.30 shows that, especially in the vicinity of the walls, a large fraction of the vertical turbulent intensity component $< v^2 >^{1/2}$ lies

*The maximum deviation of the calculated mean velocity profile in each half of the channel from the one presented in Fig. 4.27 is less than 5%.

**The maximum deviation of the calculated turbulent intensities in each half of the channel from the ones shown in Figs. 4.28, 4.29, and 4.30 is less than 12%.

in the subgrid scale motions. The deficiency in the contribution of the resolvable motions to $< \overline{v}^2 >^{1/2}$ suggests that a subgrid scale model which extracts less energy from $< \overline{v}^2 >^{1/2}$ might be required. This, in turn, may necessitate the use of transport equations for the subgrid scale Reynolds stresses (Deardorff, 1973).

For many problems in fluid mechanics, a knowledge of pressure fluctuations is desired. For instance, the generation of noise by turbulence is related to the distribution of pressure fluctuations. In addition, information about the structure of turbulence in the vicinity of the wall may be gained from the knowledge of pressure fluctuations at the wall. Unfortunately, due to experimental difficulties, direct measurements of pressure fluctuations within a turbulent flow are not possible. However, from experimental measurements and theoretical considerations, a number of investigators have obtained values for the root-mean-square wall pressure fluctuations in a turbulent boundary layer (see Willmarth and Wooldridge, 1962, and Lilley, 1960).

In our computer runs, we neglected to calculate the running time average of the RMS wall pressure fluctuations. However, we had stored the pressure and velocity fields at several dimensionless times. Table 4.1 shows a time history of root-mean square value of the <u>resolvable</u> wall pressure fluctuations, $< \overline{p}^2 >^{1/2}/\tau_w$. Here, < > indicates the average of the bracketed quantity over all the grid points on a wall.

Table 4.1

RMS Value of Wall Pressure Fluctuations

Dimensionless Time, t	$< \overline{p}^2 >^{1/2}/\tau_w$ Lower Wall (y = -1)	$< \overline{p}^2 >^{1/2}/\tau_w$ Upper Wall (y = +1)
1.05	2.04	2.01
1.275	1.78	2.81
1.425	1.87	2.50
1.625	2.13	2.00
1.825	2.00	1.95
2.025	1.72	2.01

The average value of the entries in this table (an approximation for the running time average), $< \overline{p}^2 >^{1/2}/\tau_w = 2.07$, is in accordance with experimental measurements (see Willmarth and Wooldridge, 1962, for the data from several measurements) and theoretical estimates (Lilley, 1960).

A quantity of particular interest to turbulence modelers is the pressure work term, $- < \frac{\partial}{\partial y} Pv >$, which appears in the governing equation for the turbulent kinetic energy (Hinze, 1975). This term is sometimes neglected, partly because it cannot be measured and partly because pressure tends to be poorly correlated with velocities, except near the wall (Townsend, 1956, and Tennekes and Lumley, 1972). Fig. 4.31 shows the profile of the resolvable pressure work term, $- < \frac{\partial}{\partial y} \widetilde{\overline{Pv}} >$. It can be seen that in the regions away from the wall $(y > -.8)$, $- < \frac{\partial}{\partial y} \widetilde{\overline{Pv}} >$ is much smaller than its corresponding values in the vicinity of the wall. In addition, the general shape of $- < \frac{\partial}{\partial y} \widetilde{\overline{pv}} >$ is in accordance with the estimates of Laufer (1954) and Townsend (1956). These estimates were obtained from the turbulent energy balance in a pipe flow (see Chapter I).

The average resolvable pressure velocity-gradient correlations (pressure-strain terms), $< \tilde{P} \frac{\partial \overline{u}}{\partial x} >$, $< \tilde{P} \frac{\partial \overline{v}}{\partial y} >$, and $< \tilde{P} \frac{\partial \overline{w}}{\partial z} >$ are shown in Fig. 4.32. These terms govern the exchange of energy between the three components of resolvable turbulent kinetic energy. Note that since the sum of the above pressure velocity-gradient correlations is zero, these terms only transfer energy from one component to another, without changing the total energy. Moreover, the negative sign for $< \tilde{P}(\partial \overline{u}_\ell/\partial x_\ell >$ (no summation) indicates transfer of energy from $< (\overline{u}_\ell - < \overline{u}_\ell >)^2 >^{1/2}$ to other components (loss), whereas a positive sign denotes energy gain. The profiles of $< \tilde{P} \frac{\partial \overline{u}}{\partial x} >$ and $< \tilde{P} \frac{\partial \overline{w}}{\partial z} >$ show that throughout the channel the averaged streamwise component of resolvable turbulence intensity transfers energy to the other components, while the spanwise component receives energy. It is interesting to note that in the vicinity of the wall there is a large transfer of energy from the vertical component of turbulence intensity to the spanwise component. This is consistent with the deficiency of the resolvable portion of the $< v^2 >^{1/2}$ profile in the region close to the wall shown in Fig. 4.28.

In order to gain better insight into the flow of energy caused by the fluctuating pressure gradients, one might consider the governing equations for each component of the resolvable turbulence energy. In these equations, the only terms where pressure appears explicitly are: $- < \overline{u} \frac{\partial \tilde{P}}{\partial x} >$, $- < \overline{v} \frac{\partial \tilde{P}}{\partial y} >$, and $- < \overline{w} \frac{\partial \tilde{P}}{\partial z} >$ for x, y, and z components of turbulence energy, respectively. Note that

$$- < \overline{u} \frac{\partial \tilde{P}}{\partial x} > \; = \; < \tilde{P} \frac{\partial \overline{u}}{\partial x} >$$

and

$$- < \overline{w} \frac{\partial \tilde{P}}{\partial z} > \; = \; < \tilde{P} \frac{\partial \overline{w}}{\partial z} >$$

but

$$- < \overline{v} \frac{\partial \tilde{P}}{\partial y} > \; \neq \; < \tilde{P} \frac{\partial \overline{v}}{\partial y} >$$

The average resolvable velocity pressure-gradient correlations are shown in Fig. 4.33. Examination of the $< \overline{v} \frac{\partial \tilde{P}}{\partial y} >$ profile reveals that, aside from some energy loss in the region $-.95 < y < -.83$, the vertical component of the resolvable turbulent energy receives energy via $- < \overline{v} \frac{\partial \tilde{P}}{\partial y} >$. Thus, $- < \overline{v} \frac{\partial \tilde{P}}{\partial y} >$ is primarily the source of energy for $< \overline{v}^2 >^{1/2}$.

Chapter V

CONCLUSIONS AND RECOMMENDATIONS

In this work, we have numerically integrated the three-dimensional, time-dependent primitive equations of motion for the case of turbulent channel flow. To accomplish this task, a new, partially implicit algorithm and a new subgrid-scale model for the inner region of the boundary layer were developed. An important feature of this partial implicit scheme is that the equation of continuity is solved directly. This, in turn, allows one to abandon the use of the Poisson equation for pressure. In addition, the stringent requirement on the time step caused by the numerical stability criterion for the diffusion equation is largely eased.

The present computation has shown that many of the important features of wall-bounded turbulent flows can be reproduced using the Large Eddy Simulation approach. The overall agreement of the computed mean velocity and turbulence statistics with experimental data is satisfactory.

In the present formulation of the subgrid scale model, the specification of the SGS length scale is not based on a well-definted foundation. There are several choices available for this quantity which warrant systematic study in this area. It would be desirable, for example, to incorporate a Reynolds number dependence in the function defining the SGS length scale. This function, in turn, should allow for the vanishing of the subgrid scale model in a laminar flow. The profiles of total turbulent intensities indicate that, with the present grid resolution, a subgrid scale model which allows anisotropy of SGS energy components is desirable. This modification of the subgrid scale model may not be necessary, if better grid resolution could be utilized. Nevertheless, the performance of the subgrid scale model used here is encouraging.

In the light of our discussions about the grid resolution, a simulation with $32 \times 65 \times 128$ mesh in x, y, and z direction, respectively, is strongly recommended. We believe that such a calculation will considerably improve the results obtained here and will provide the means for an objective evaluation as well as improvement of the subgrid scale model. It should be noted that this computation can presently be performed on the ILLIAC IV computer. In addition, the use of a computer graphic system in conjunction with this simulation is highly desirable. This would provide the means for an efficient and a relatively convenient study of the detailed structures in the flow. Such a study, in turn, can considerably increase our knowledge of the structure and the mechanics of turbulent boundary layers.

Based on the experience gained in our initial numerical experiment (Section 4.3), the following recommendations are made for the numerical simulation of laminar-turbulent flow transition:

- Using an eddy viscosity model, the numerical simulation of transition from laminar to fully turbulent flow may be possible, provided that <u>finite</u> amplitude disturbances are added to the laminar flow.
- However, if one wants to study the time evolution of <u>small</u> disturbances, the eddy viscosity model should be used only after breakdown. Prior to breakdown, the use of any subgrid scale model may not be necessary.

In extending the method to other flows, an important numerical problem which must be resolved is the handling of inflow-outflow boundary conditions. In addition, an efficient numerical method should be devised which can be used in calculating flows that are inhomogeneous in more than one direction. Fully developed turbulent flow in a straight duct with a rectangular cross section is an example of such a flow. In simulating this flow, one can use periodic boundary conditions in the streamwise direction.

An important problem to study would be the case of turbulent flow over a smooth, flat plate. This flow is homogeneous only in one direction. Moreover, its numerical simulation involves the handling of inflow-outflow boundary conditions. In addition, a suitable coordinate transformation should be used to map the infinite physical domain to a finite computational box. It is believed that the numerical simulation of this flow is an essential step towards the utilization of the Large Eddy Simulation approach in problems of engineering interest.

It will be some time before the Large Eddy Simulation technique can be used in calculating flows of practical interest. However, in the interim, much information on the structure of turbulence can be obtained by applying the method to simple but basic flows. The information, in turn, can be used in developing turbulence models in a simpler method for complex flows. A knowledge of the pressure-velocity gradient correlations, for example, is of considerable value to the turbulence modelers. As was shown in this study, using the Large Eddy Simulation approach one can compute their large-scale components. Moreover, with the Large Eddy Simulation technique one can simultaneously obtain detailed quantitative information about the large-scale structures of the flow at thousands of spatial locations (grid points) throughout the flow field. This information cannot be gained from laboratory measurements. On the other hand, in the laboratory, one is capable of obtaining a long time history of the flow at relatively few spatial locations with minor expense. With the present computers, this latter information about the flow can be gained only at high cost. Thus, at present, combined efforts of measurements and Large Eddy Simulation of turbulence seems to be an attractive approach to a better understanding of turbulent flows.

The Large Eddy Simulation of turbulence is just beginning to emerge from its infancy, but it has already demonstrated a great potential in supplementing laboratory measurements.

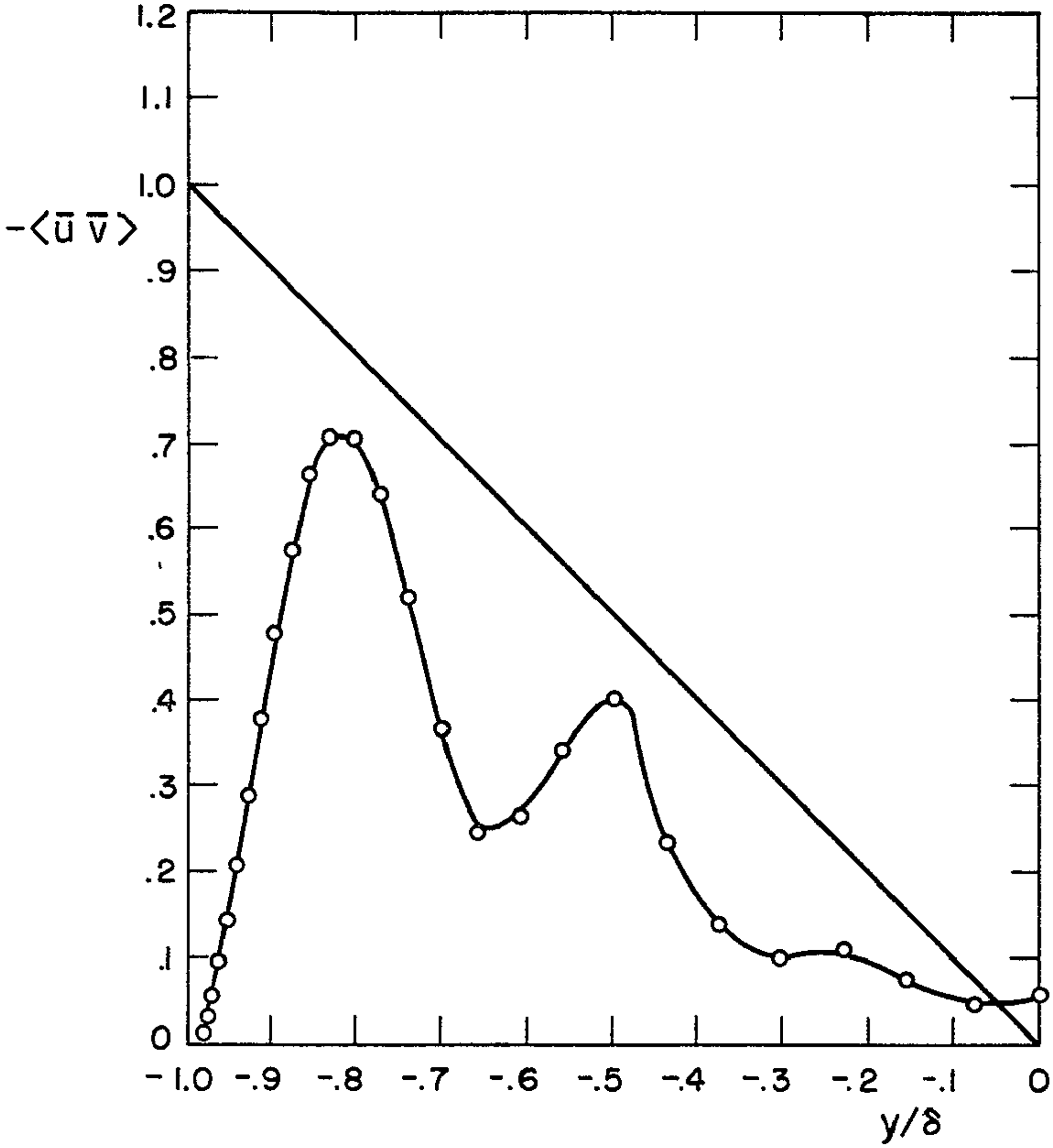

Fig. 4.1. The resolvable portion of turbulence stress in the lower half of the channel at $t = 0$.

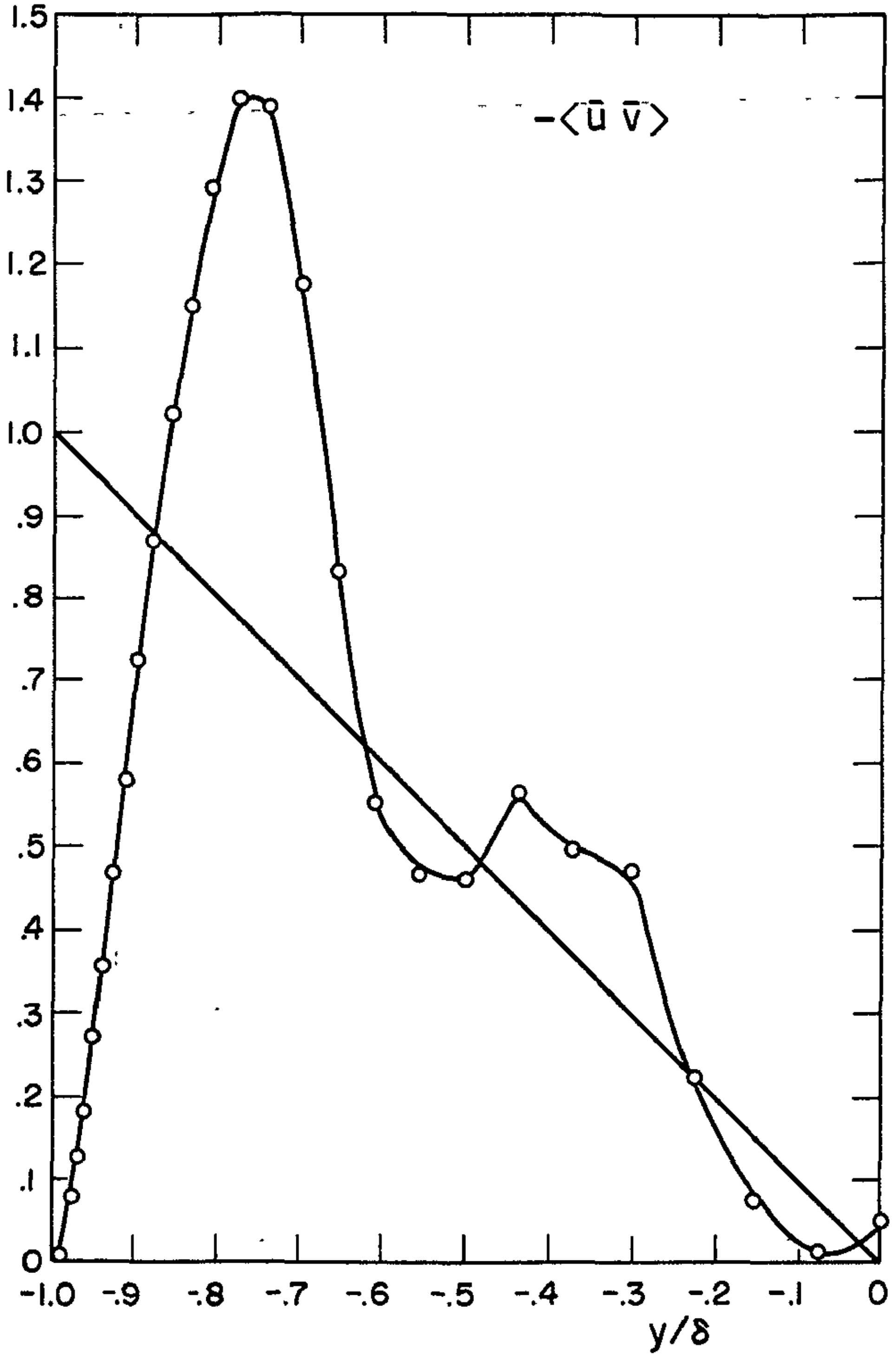

Fig. 4.2. The resolvable portion of turbulence stress in the lower half of the channel at $t = 0.45$.

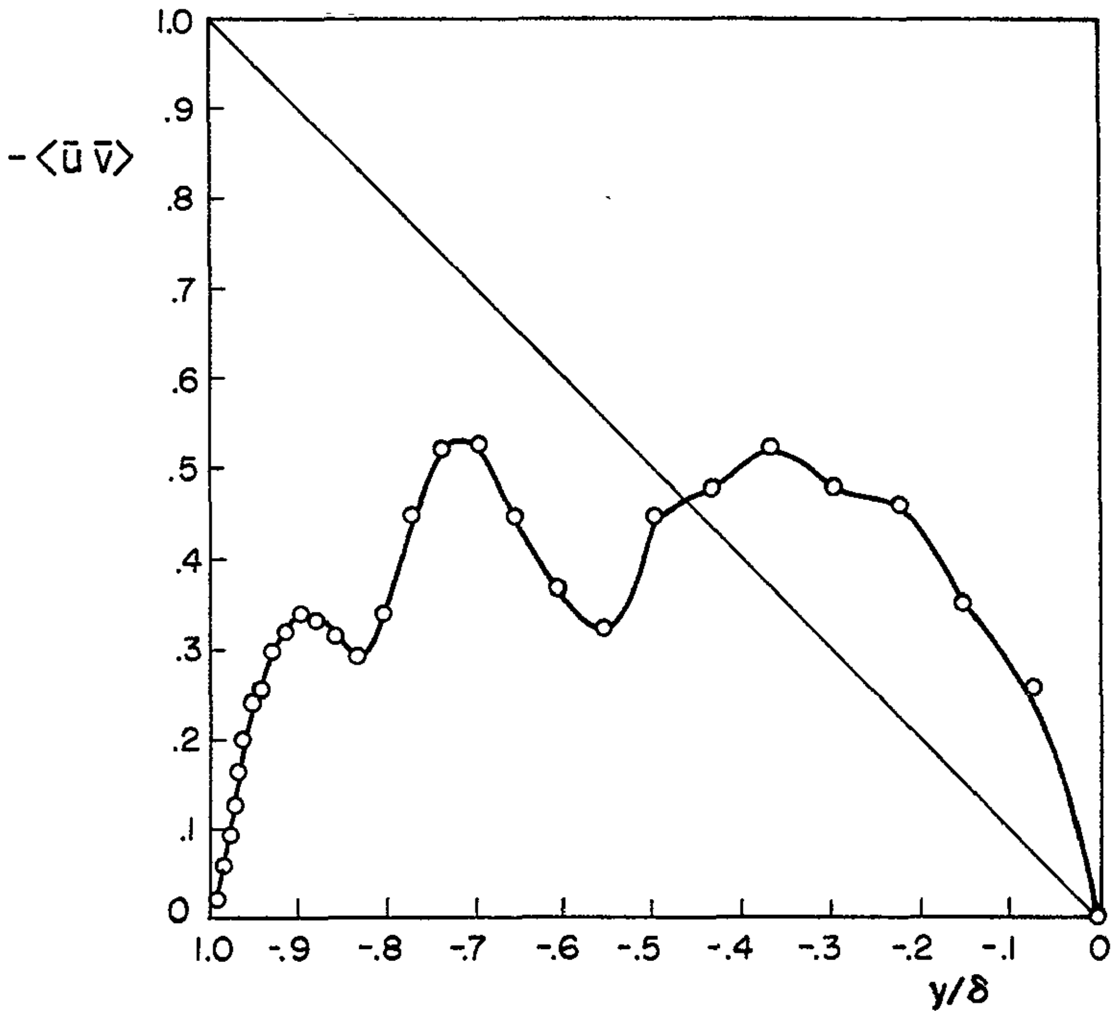

Fig. 4.3. The resolvable portion of turbulence stress in the lower half of the channel at $t = 0.65$.

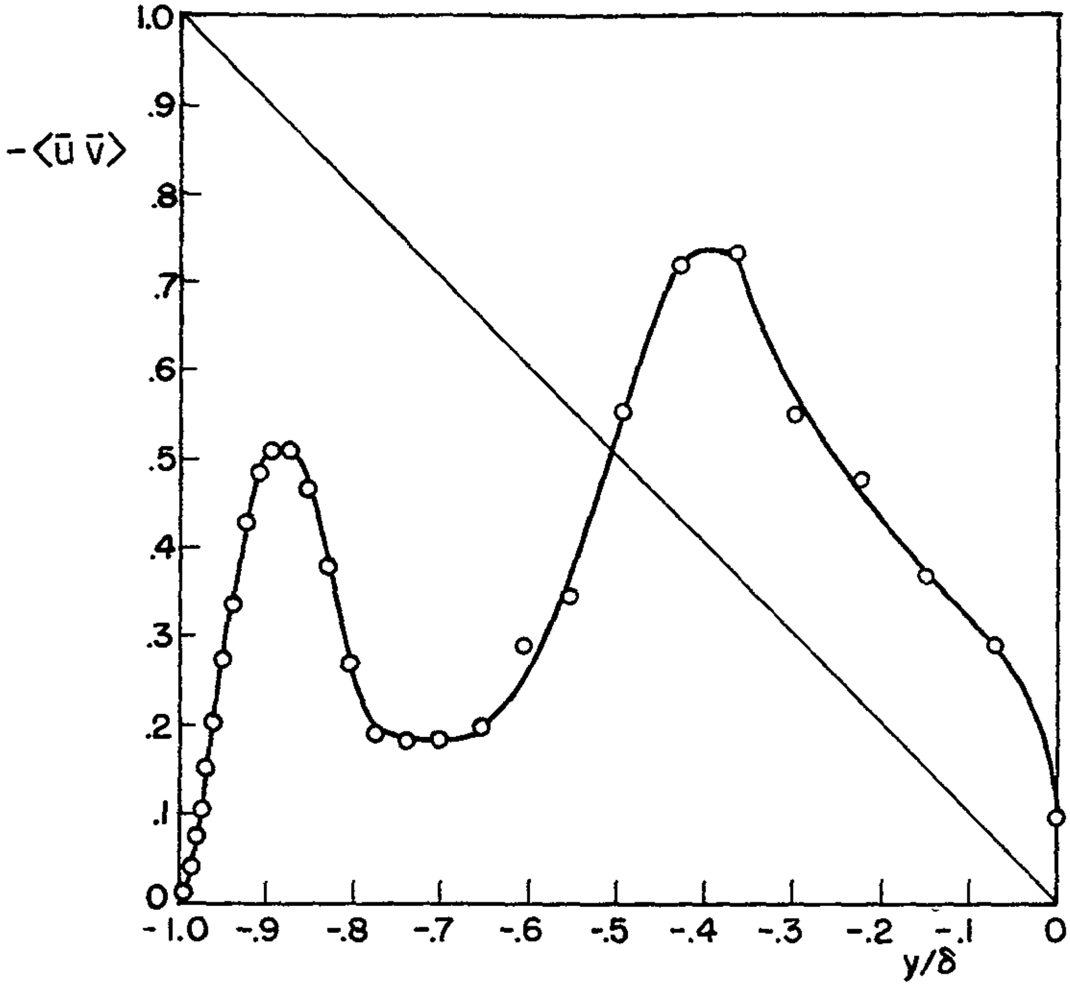

Fig. 4.4. The resolvable portion of turbulence stress in the lower half of the channel at $t = 0.85$.

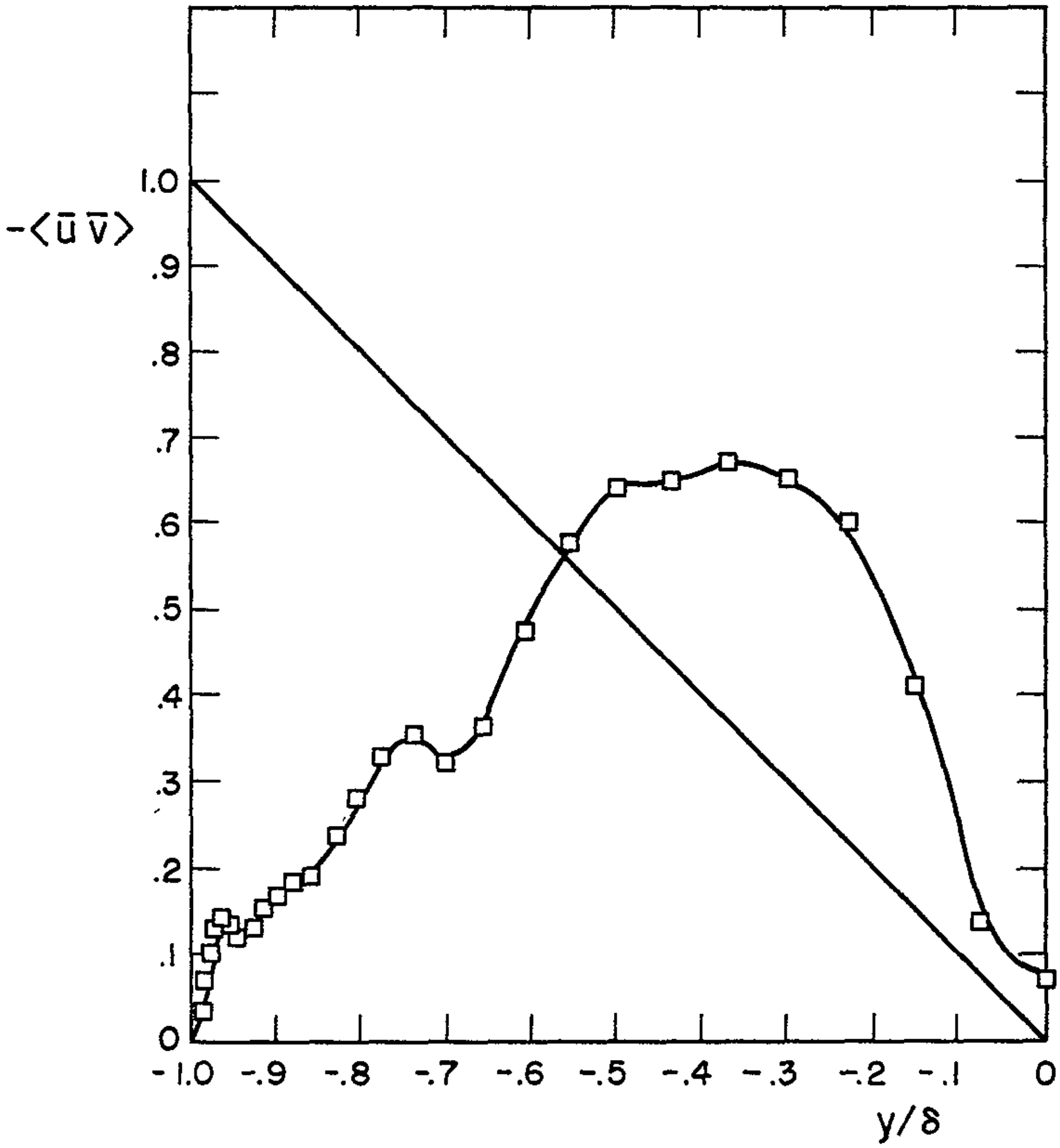

Fig. 4.5. The resolvable portion of turbulence stress in the lower half of the channel at $t = 1.05$.

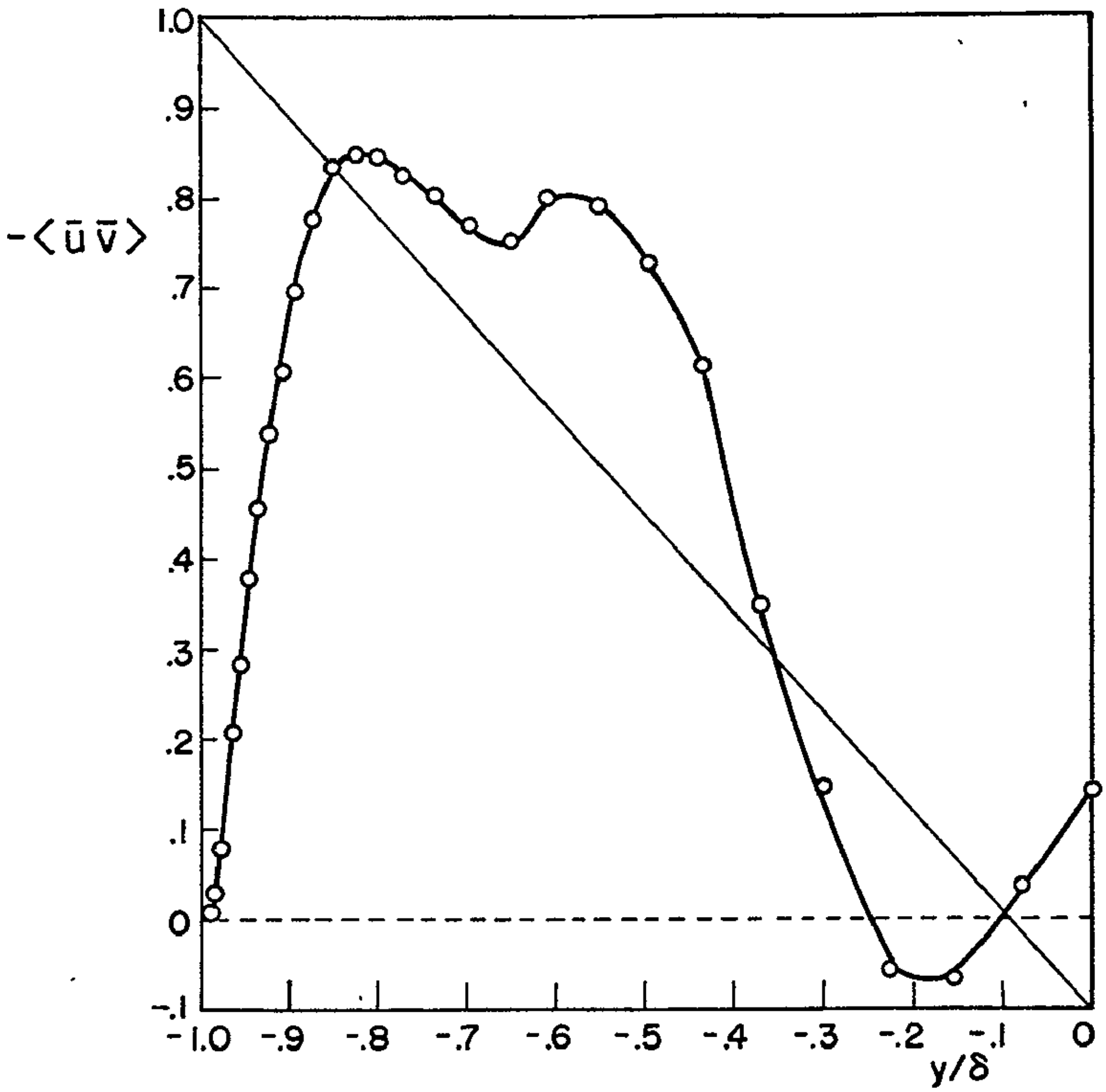

Fig. 4.6. The resolvable portion of turbulence stress in the lower half of the channel at $t = 1.425$.

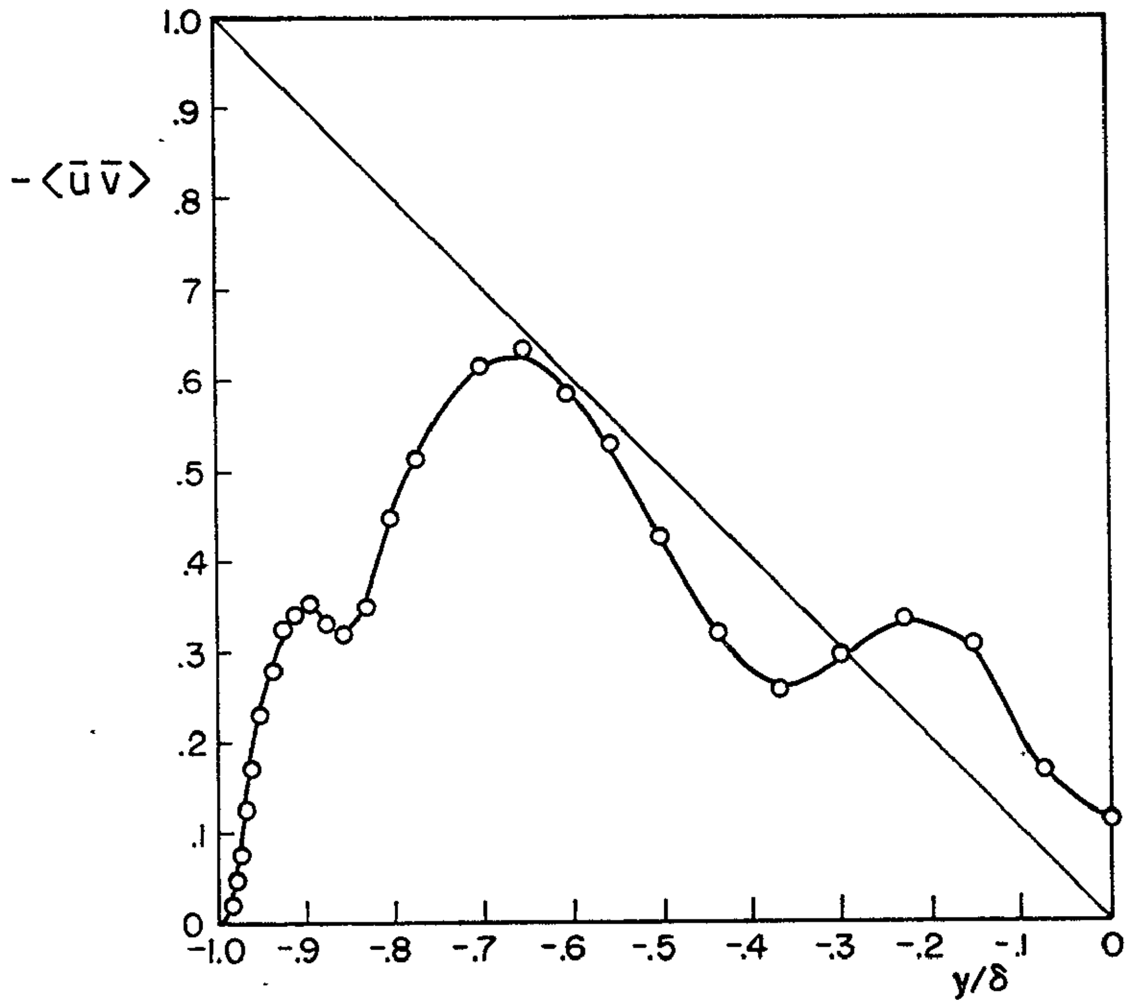

Fig. 4.7. The resolvable portion of turbulence stress in the lower half of the channel at $t = 2.025$.

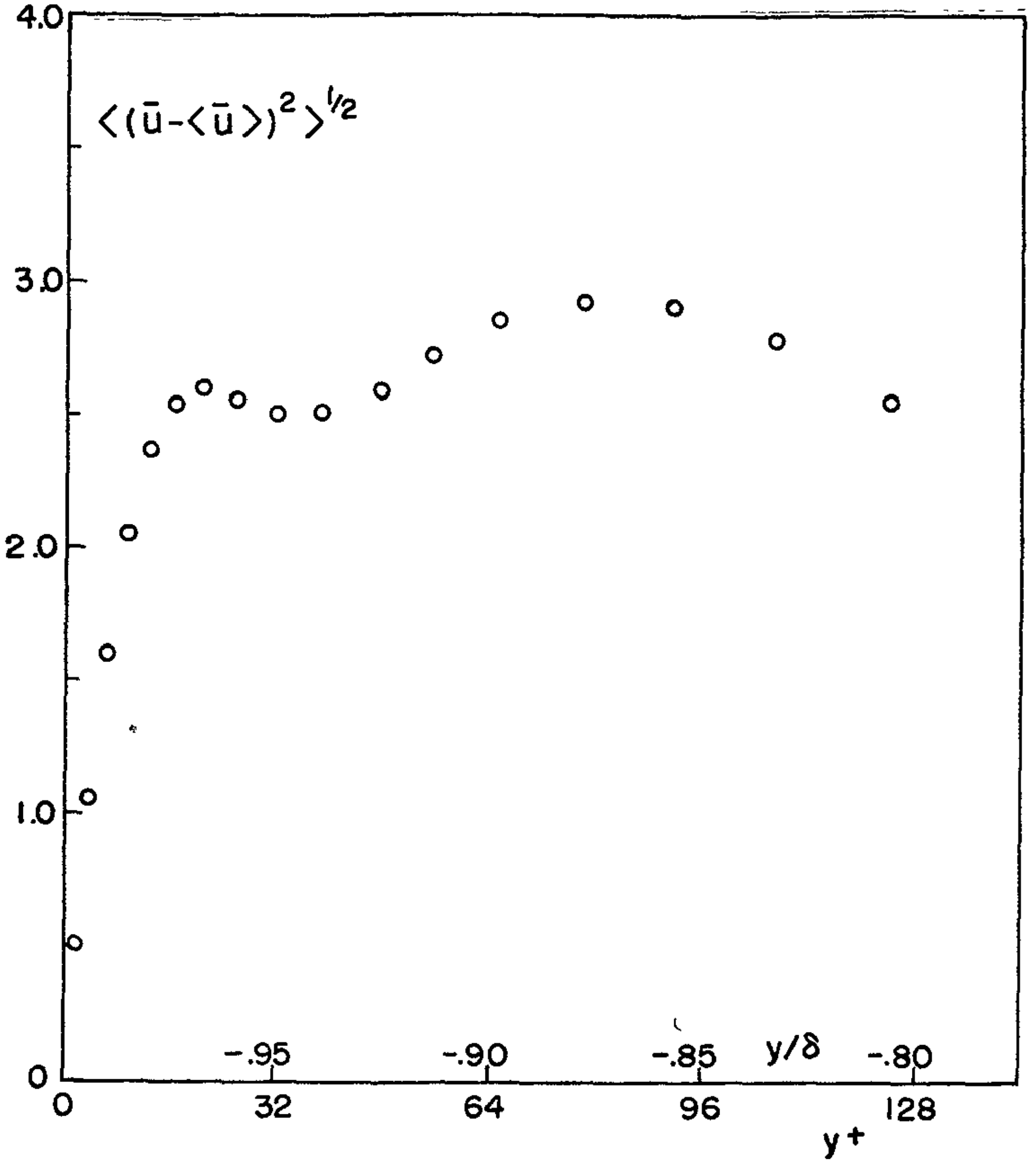

Fig. 4.8. Planar average of the resolvable portion of the stream-wise turbulence intensity in the vicinity of the lower wall at t = 1.05.

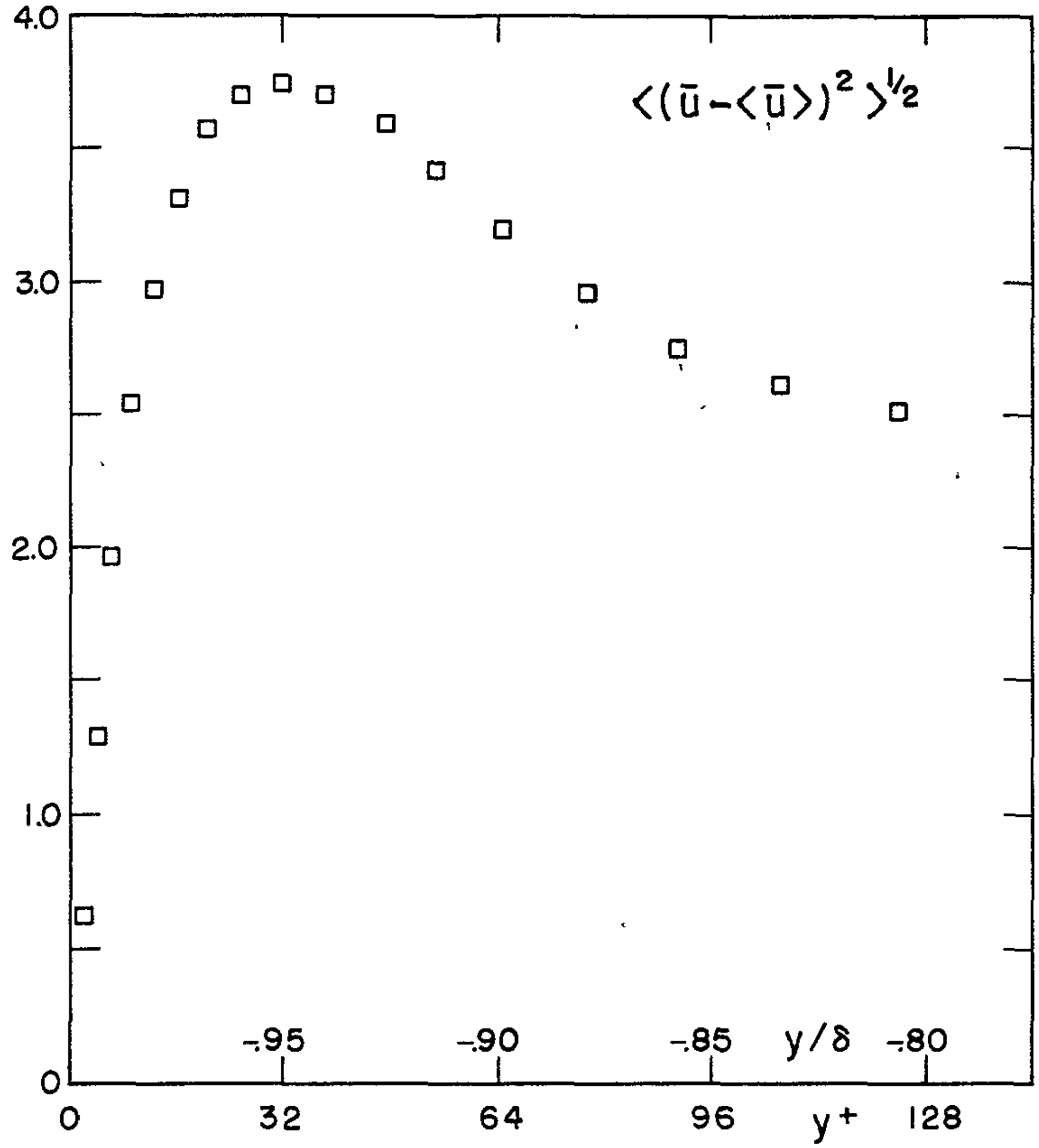

Fig. 4.9. Planar average of the resolvable portion of the streamwise turbulence intensity in the vicinity of the lower wall at $t = 1.425$.

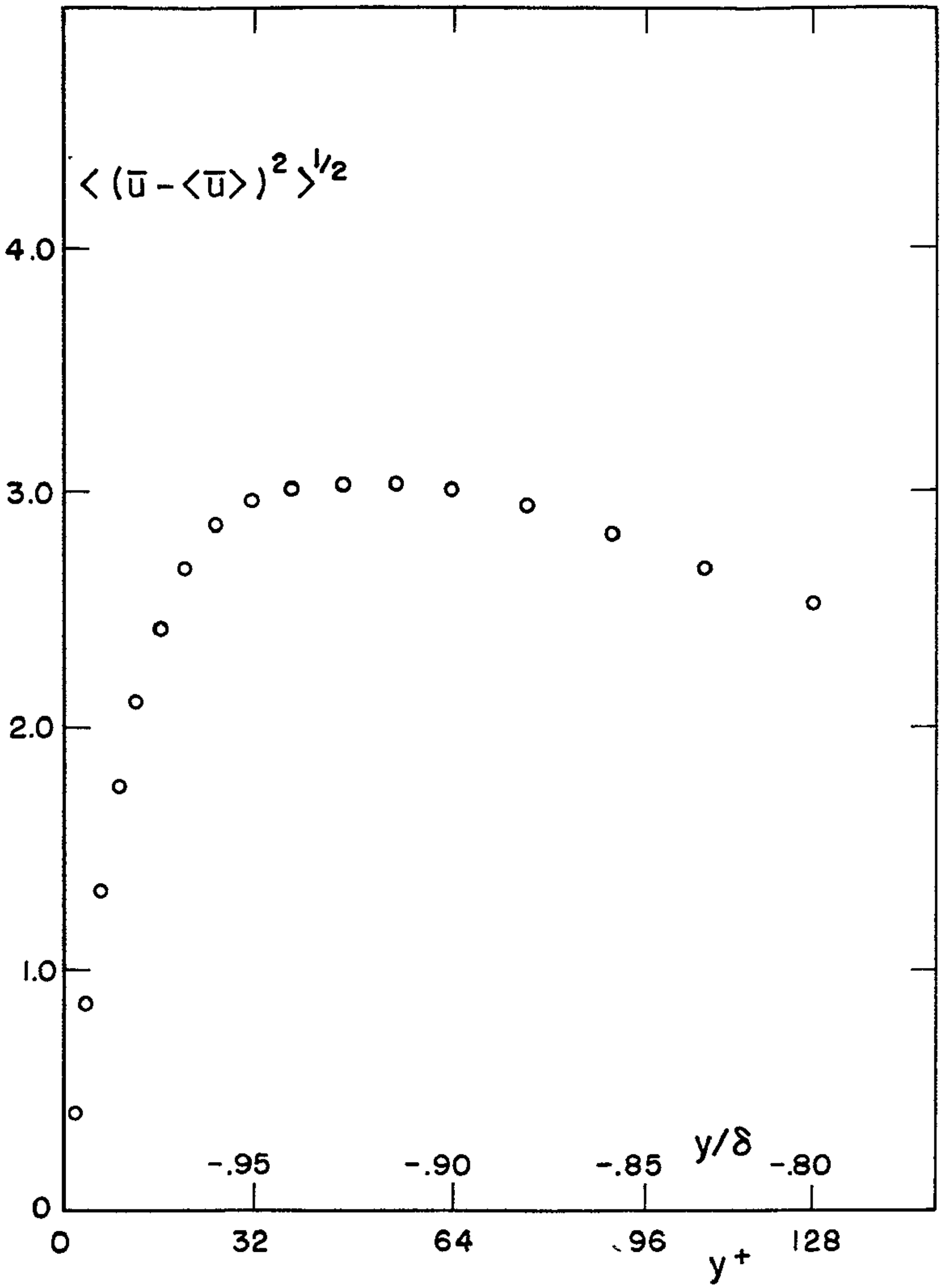

Fig. 4.10. Planar average of the resolvable portion of the streamwise turbulence intensity in the vicinity of the lower wall at $t = 2.025$.

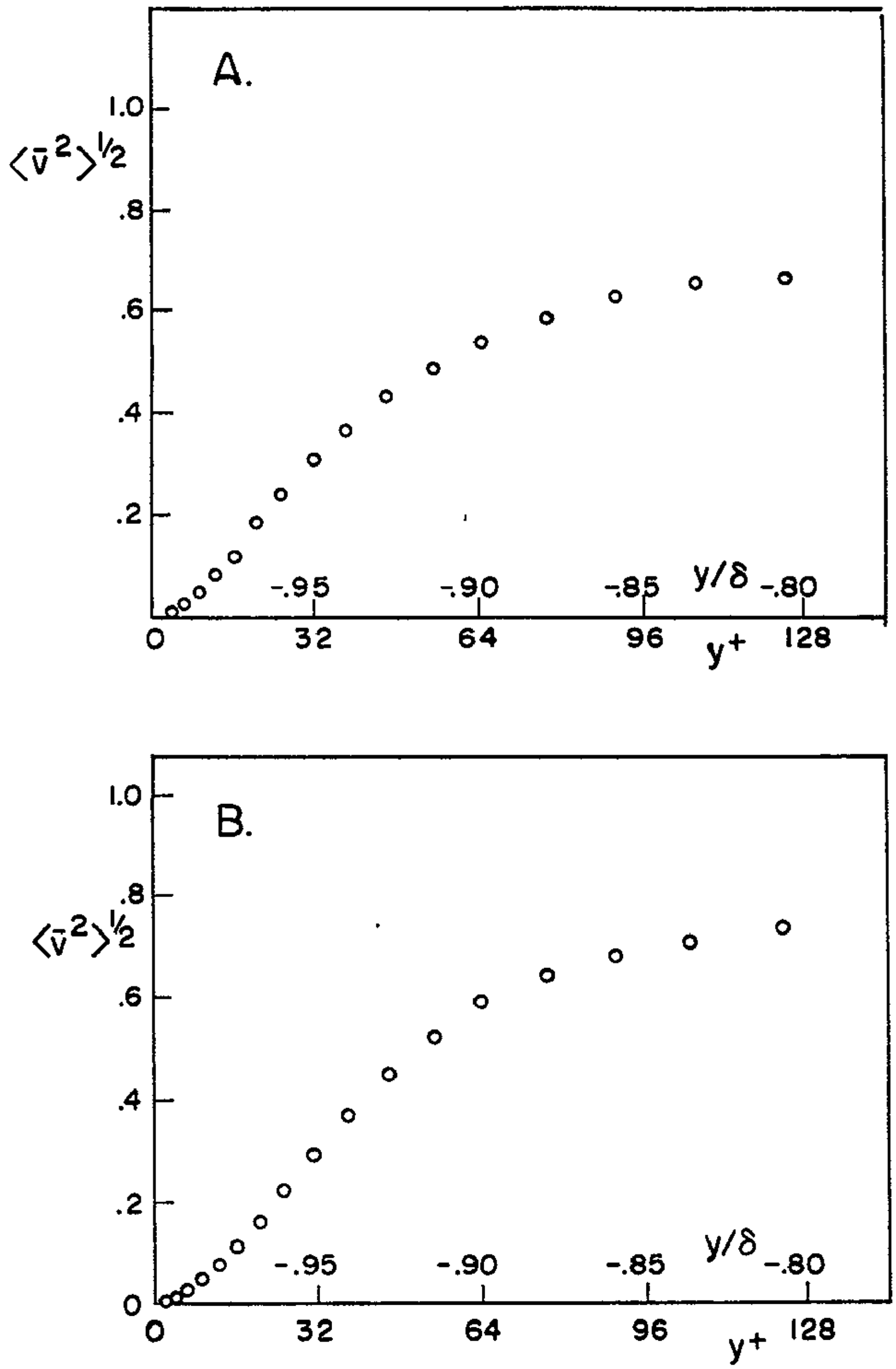

Fig. 4.11. Planar average of the resolvable portion of the vertical turbulence intensity in the vicinity of the lower wall at
A) t = 1.425
B) t = 2.025

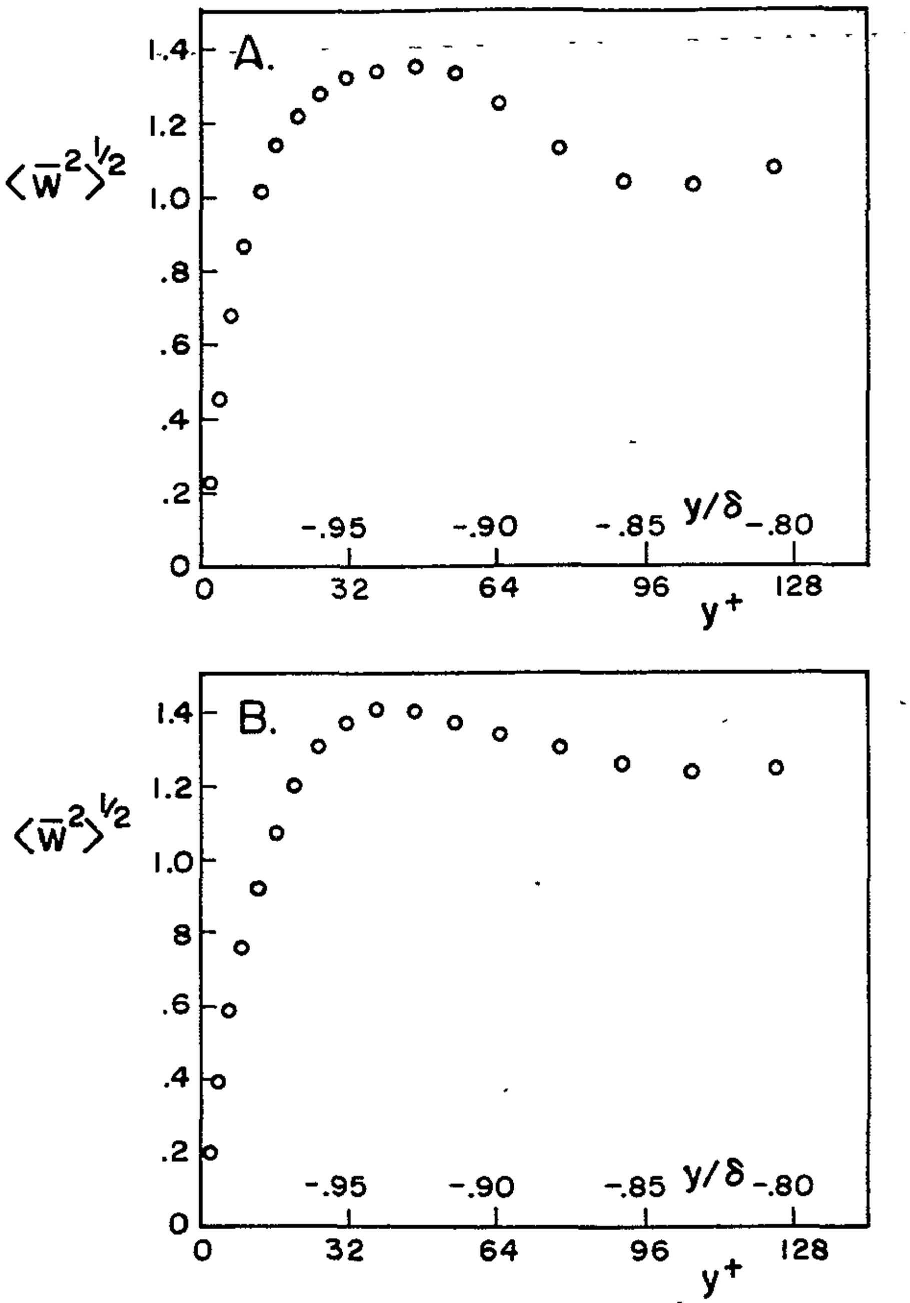

Fig. 4.12. Planar average of the resolvable portion of the spanwise turbulence intensity in the vicinity of the lower wall at
A) t = 1.425
B) t = 2.025

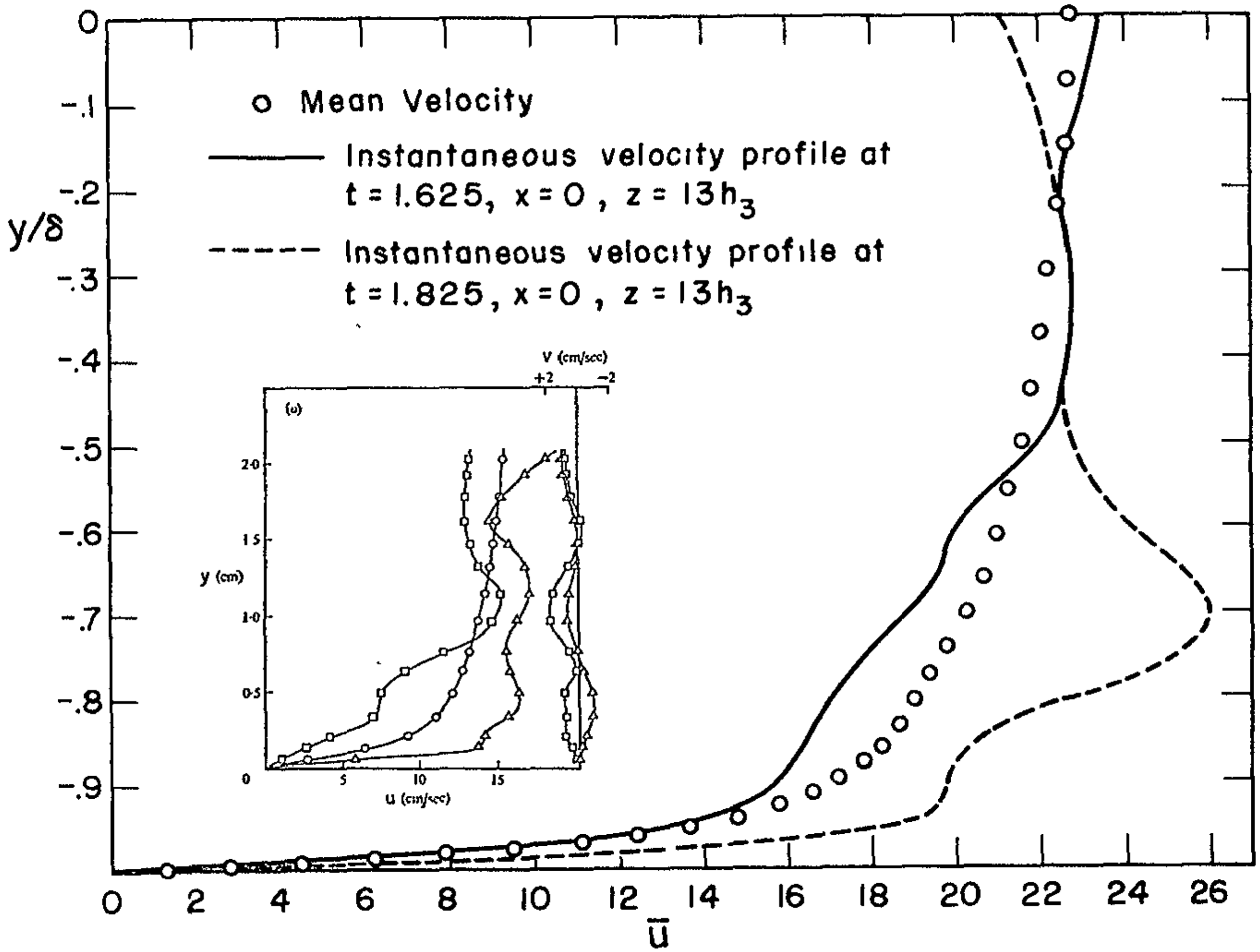

Fig. 4.13. Instantaneous streamwise velocity profiles obtained at one (x,z) location and at two different times. The instantaneous velocity profiles from the measurements of Grass (1971) are included in the left-hand side of the figure.

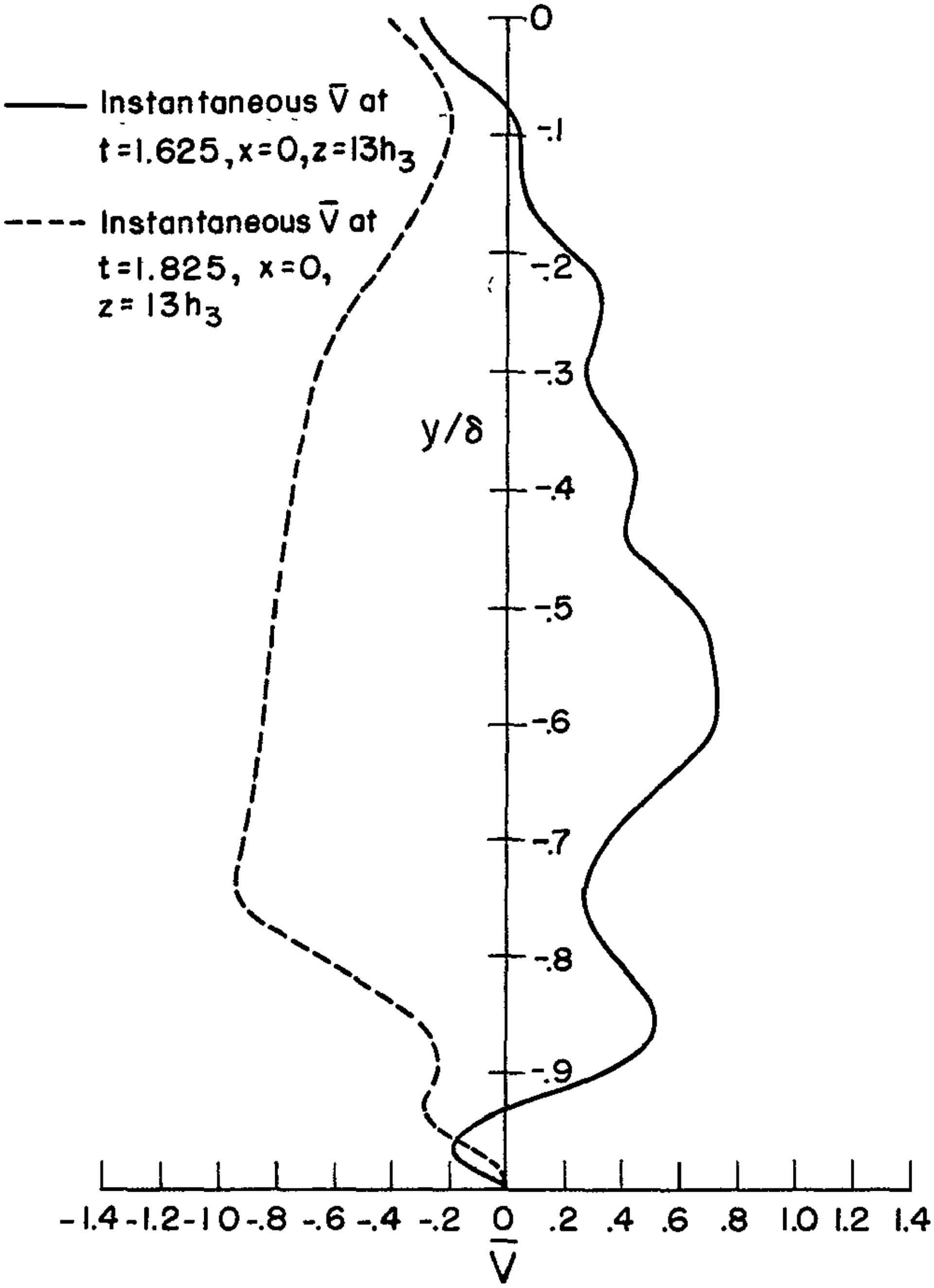

Fig. 4.14. Instantaneous vertical velocity profiles, obtained at the same location and times as in Fig. 4.13.

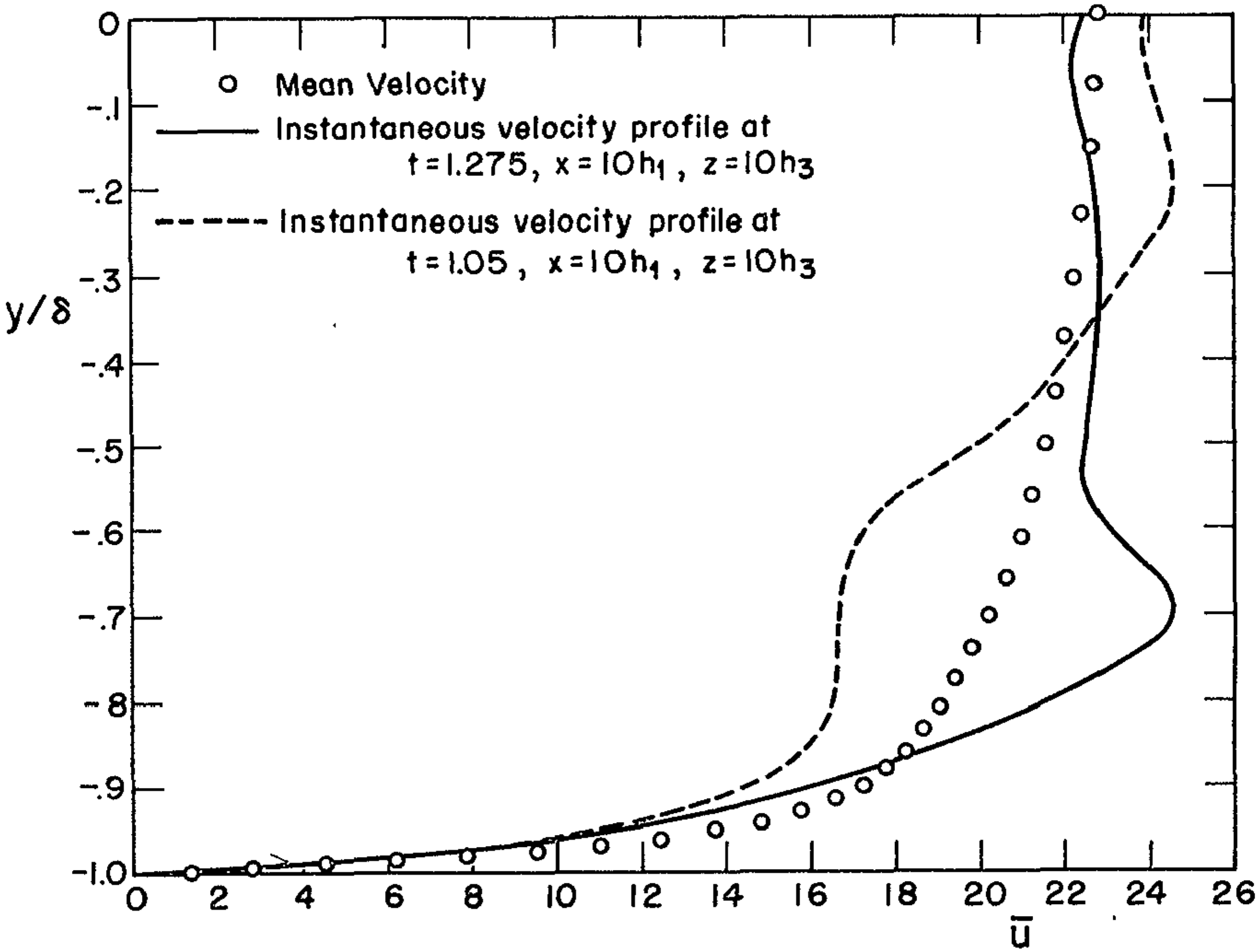

Fig. 4.15. Instantaneous streamwise velocity profiles obtained at one (x,z) location and at two different times.

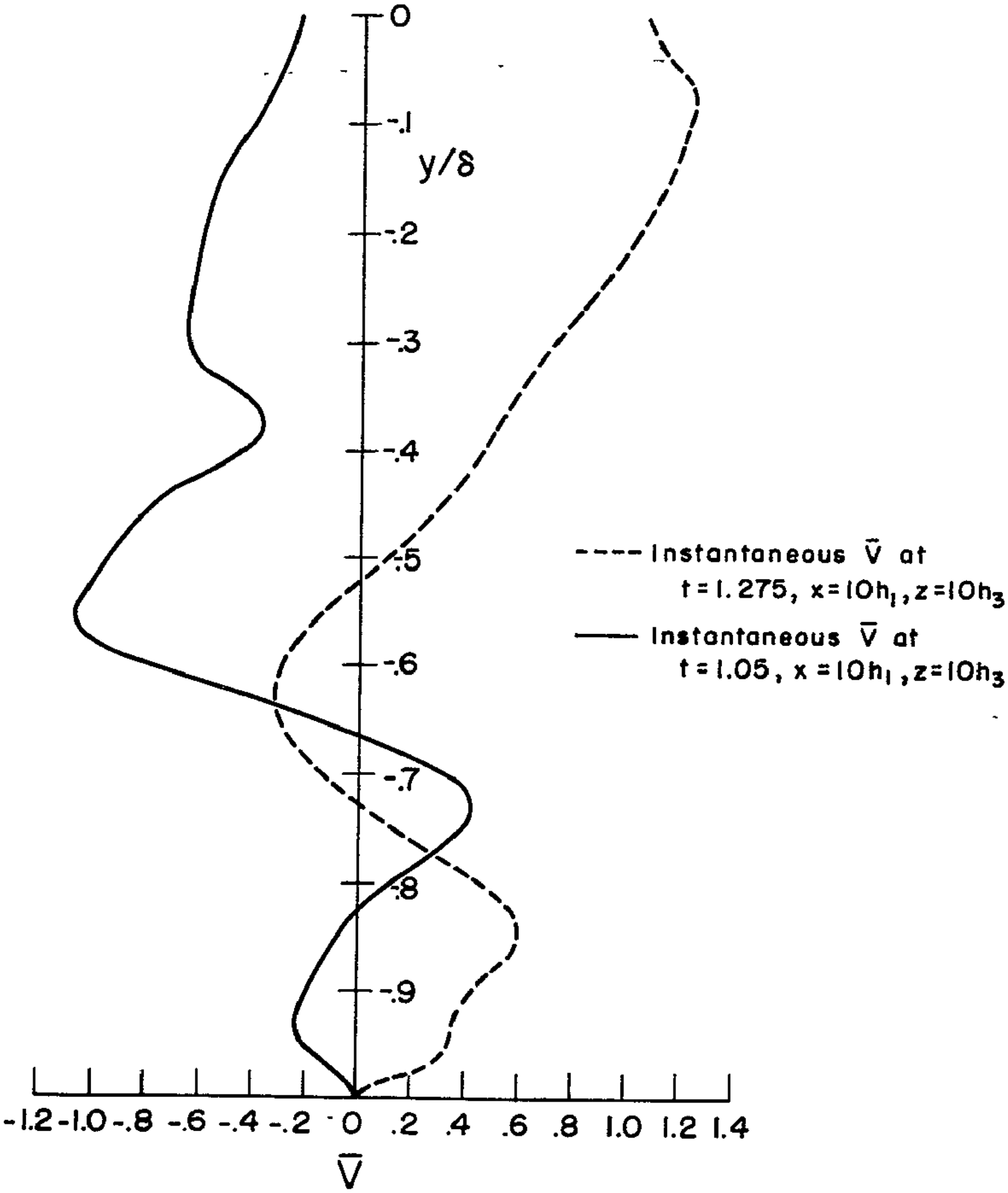

Fig. 4.16. Instantaneous vertical velocity profiles obtained at the same location and times as in Fig. 4.15.

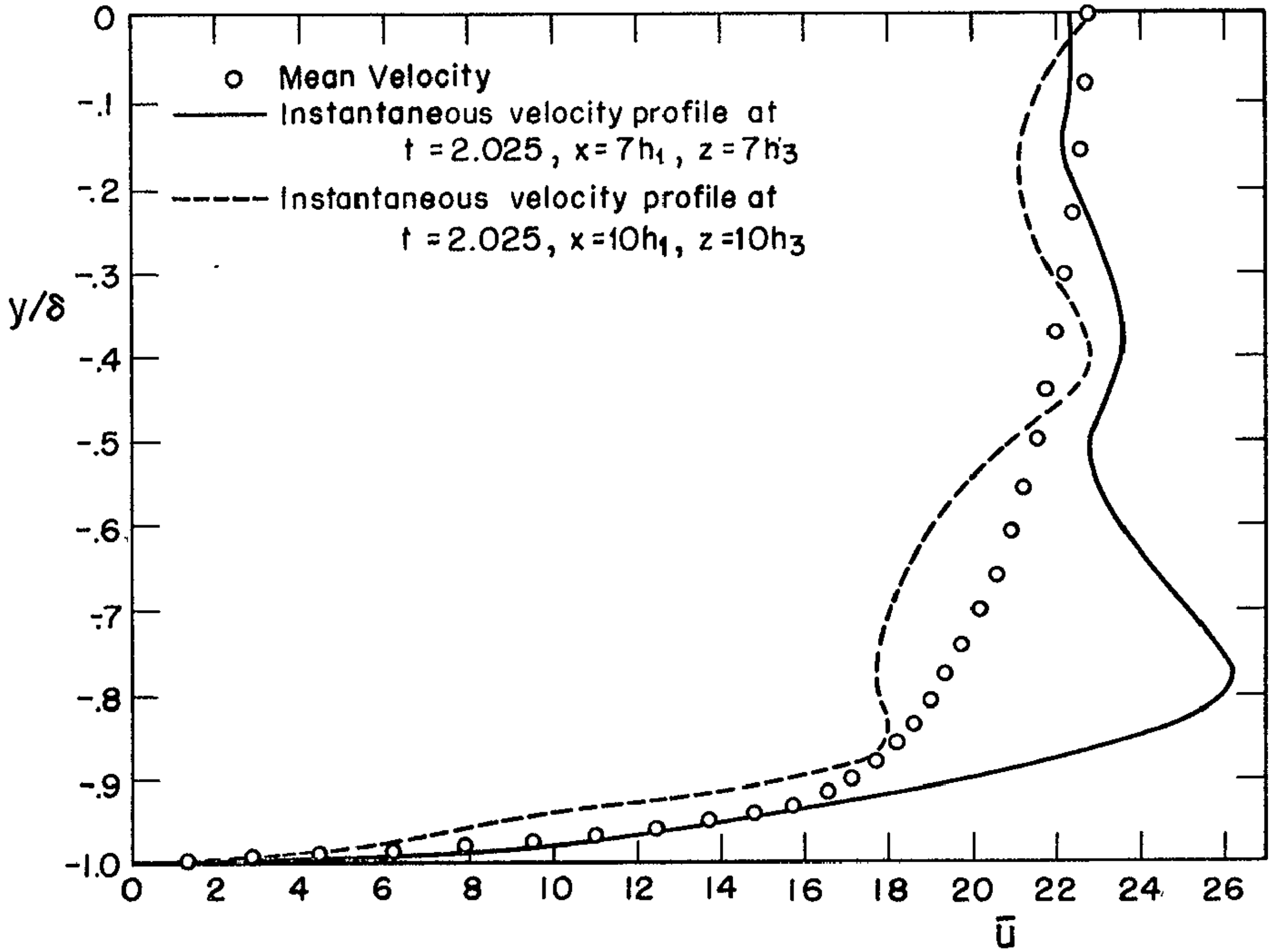

Fig. 4.17. Instantaneous streamwise velocity profiles obtained at the same time and at two different (x,z) locations.

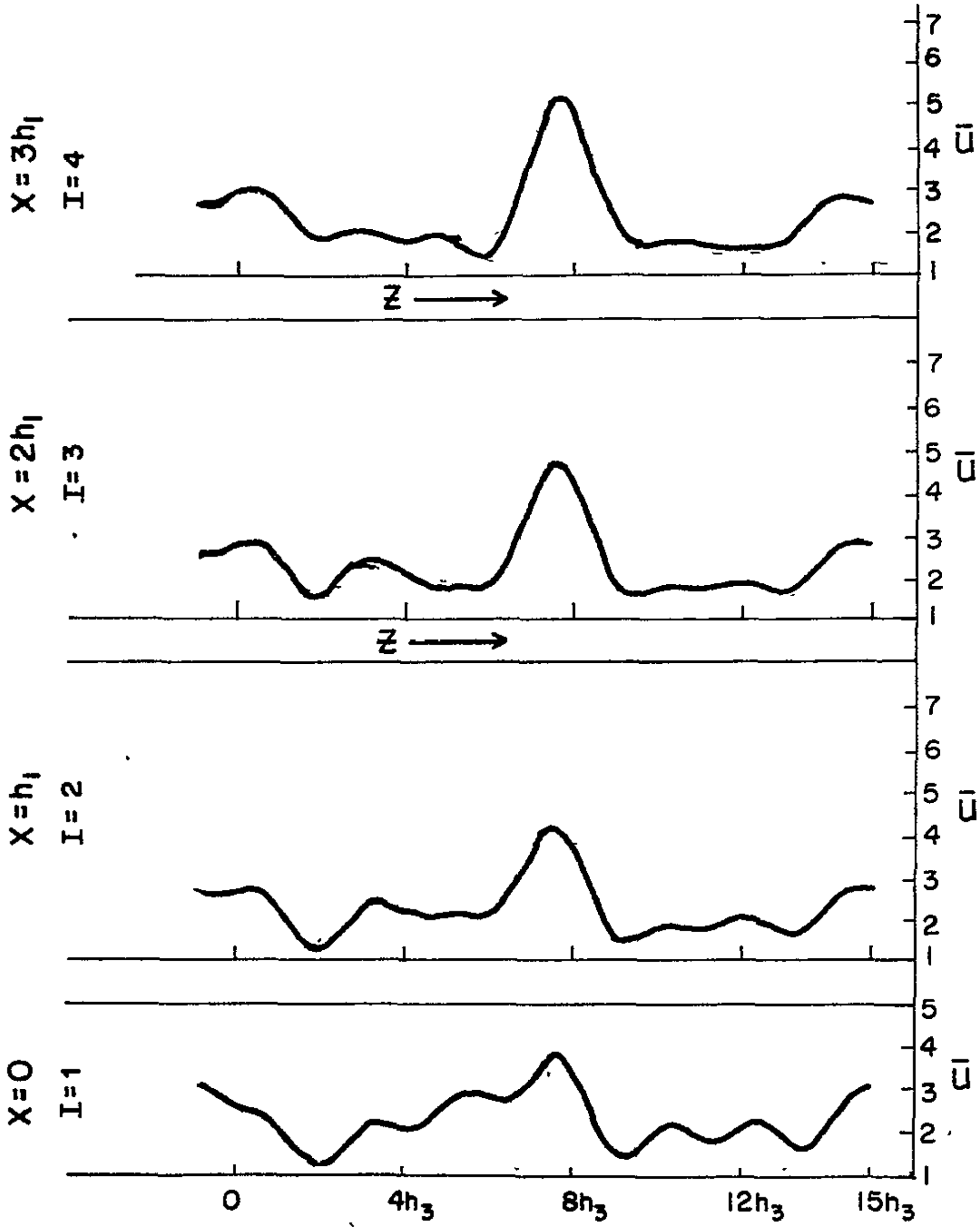

Fig. 4.18. Instantaneous spanwise variations of the $\bar{u}$ at $t = 1.05$, $y^{+} = 3.85$, and at $x = 0$, h_1, $2h_1$, $3h_1$.

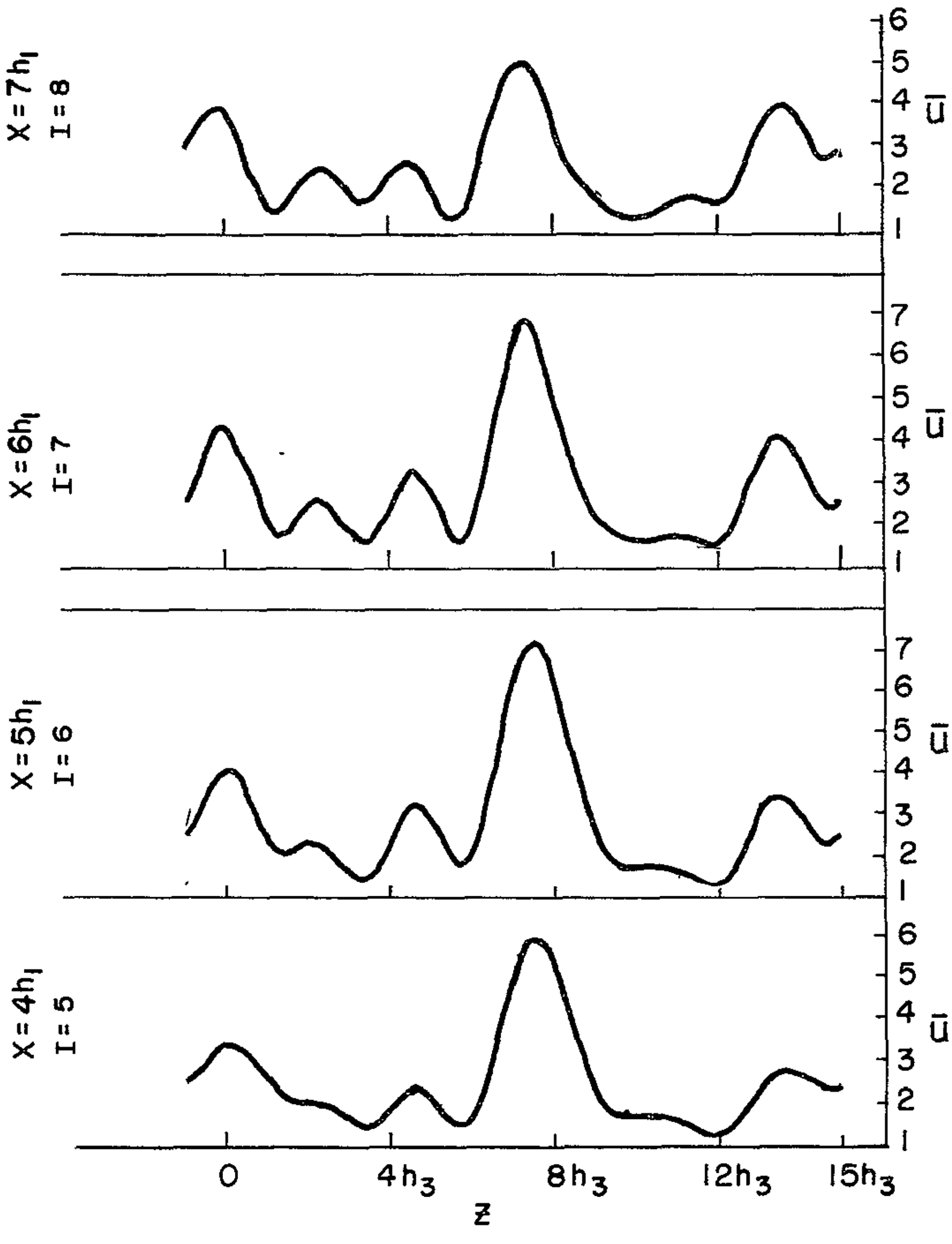

Fig. 4.19. Instantaneous spanwise variations of $\bar{u}$ at $t = 1.05$, $y^+ = 3.85$, and at $x = 4h_1$, $5h_1$, $6h_1$, $7h_1$.

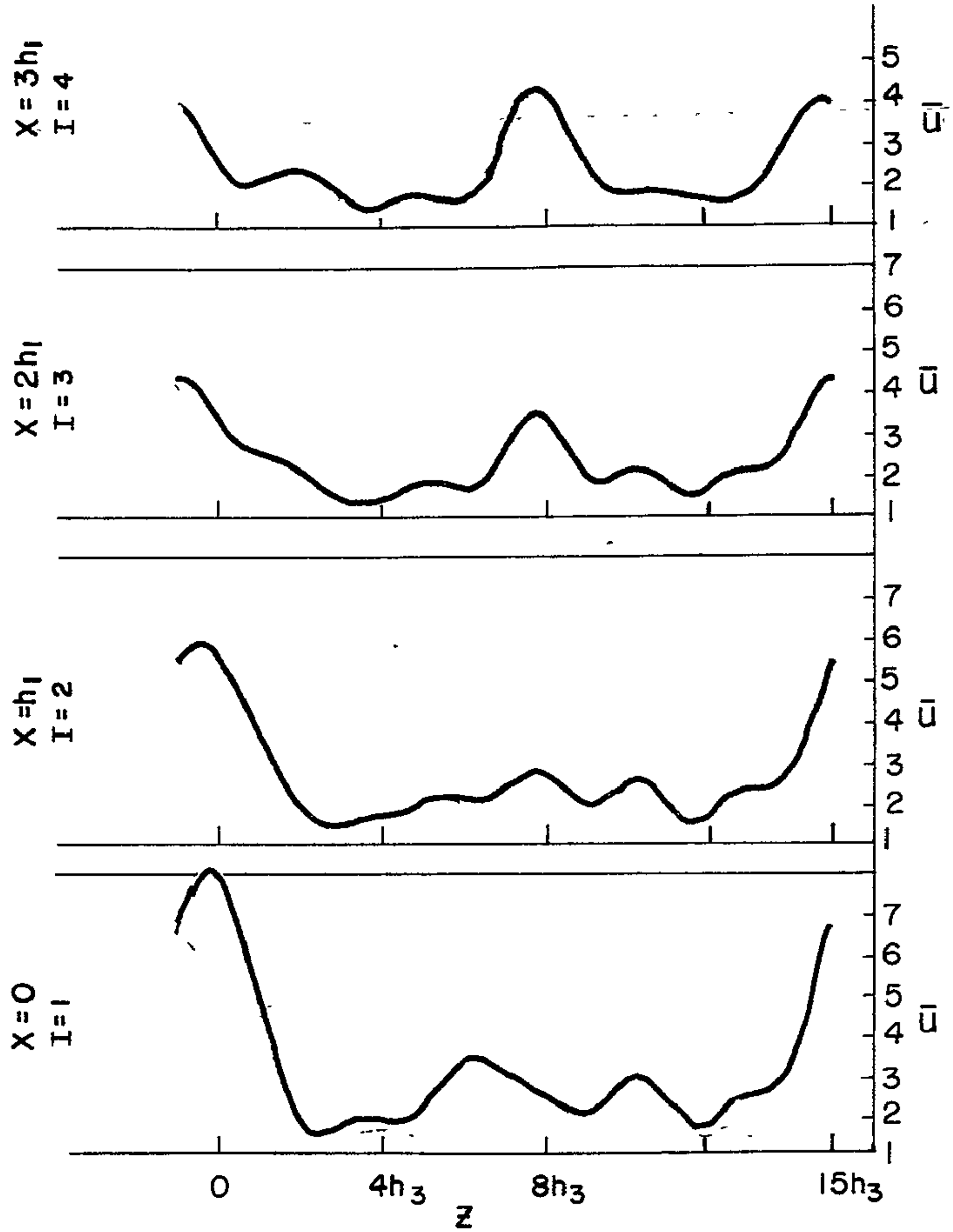

Fig. 4.20. Instantaneous spanwise variations of $\bar{u}$ at $t = 1.425$, $y^+ = 3.85$, and at $x = 0$, h_1, $2h_1$, $3h_1$.

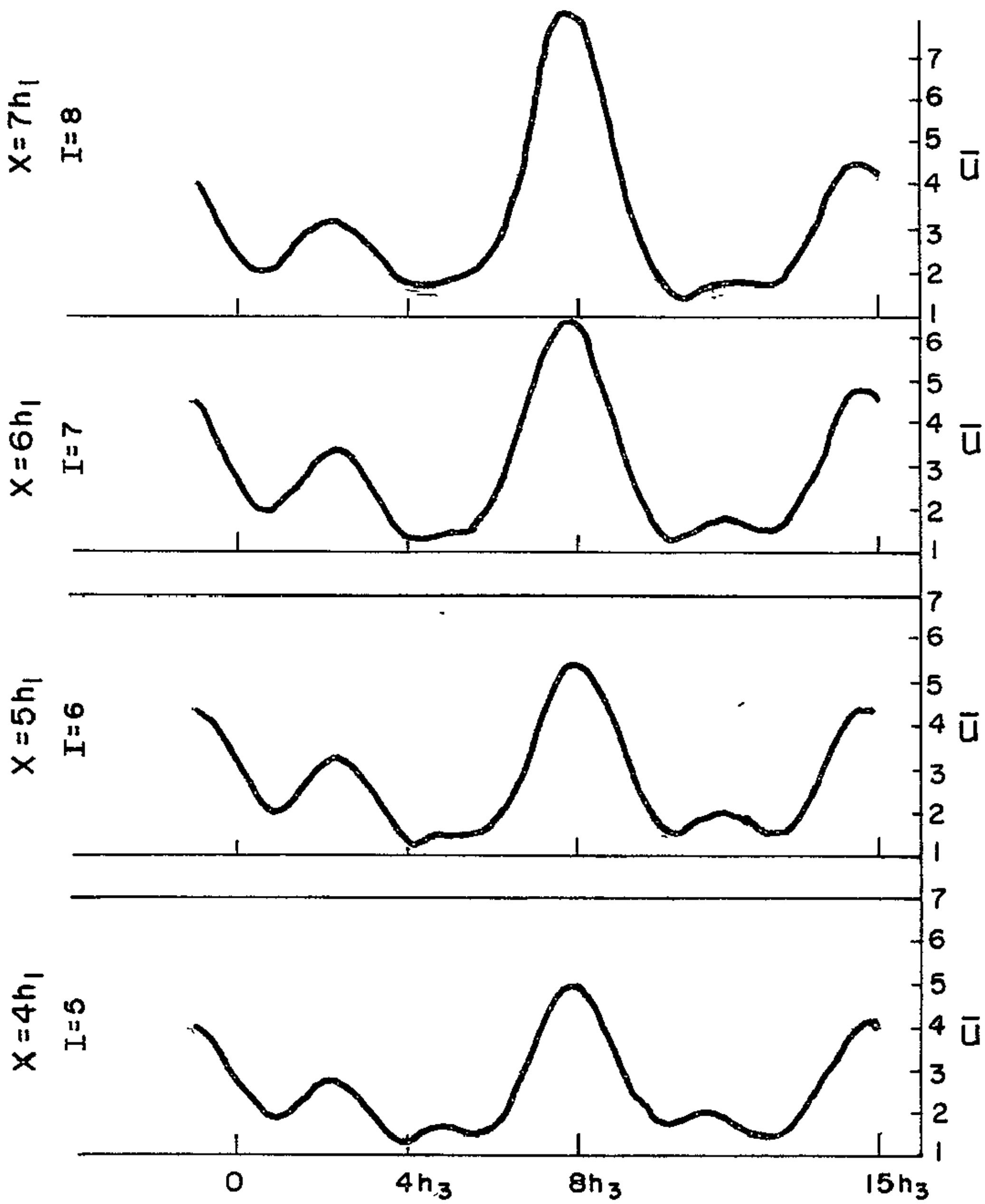

Fig. 4.21. Instantaneous spanwise variations of $\bar{u}$ at $t = 1.425$, $y^+ = 3.85$, and at $x = 4h_1, 5h_1, 6h_1, 7h_1$.

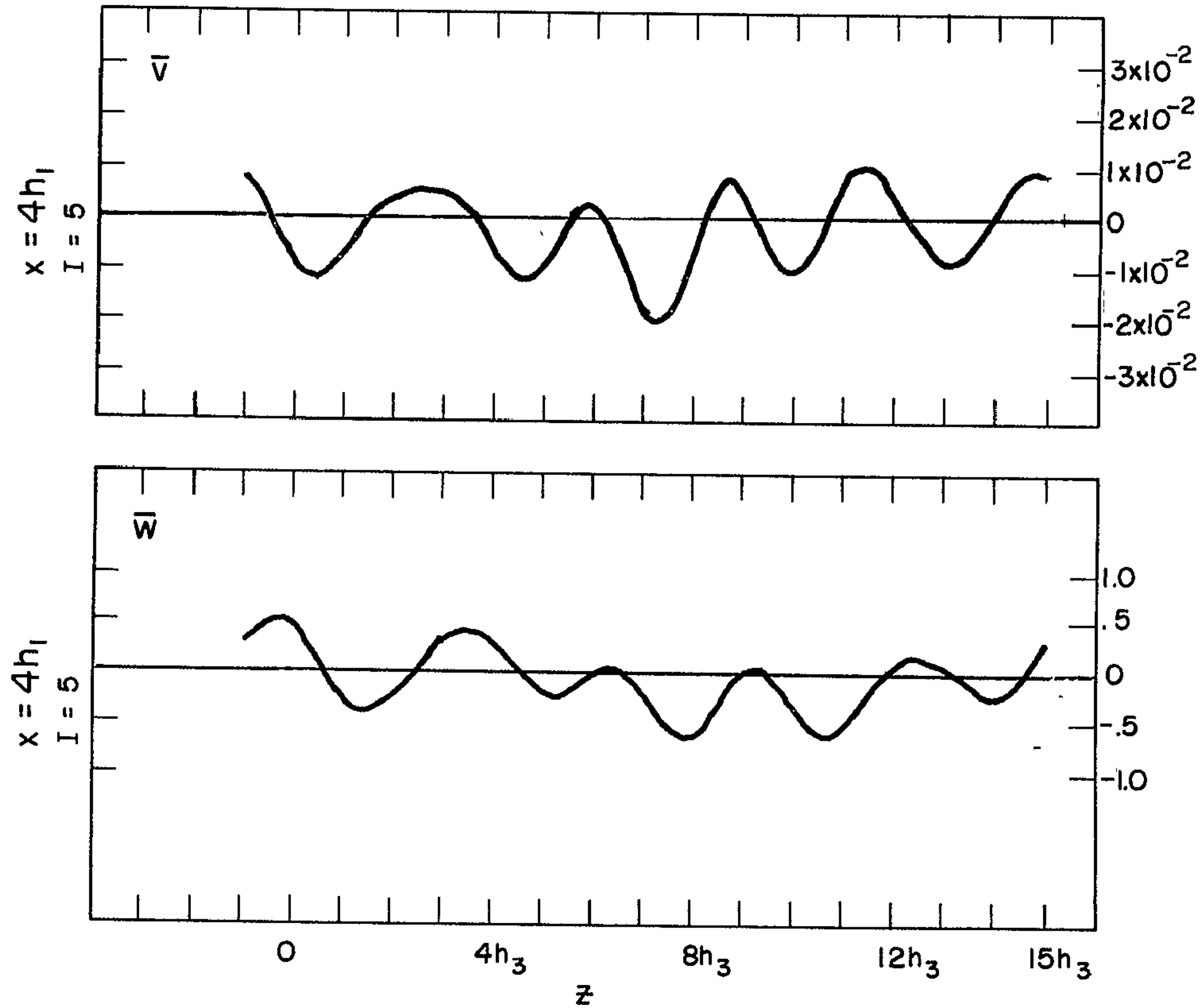

Fig. 4.22. Instantaneous spanwise variation of $\overline{v}$ (upper figure) and $\overline{w}$ (lower figure) at $t = 1.05$, $y^+ = 3.85$, $x = 4h_1$.

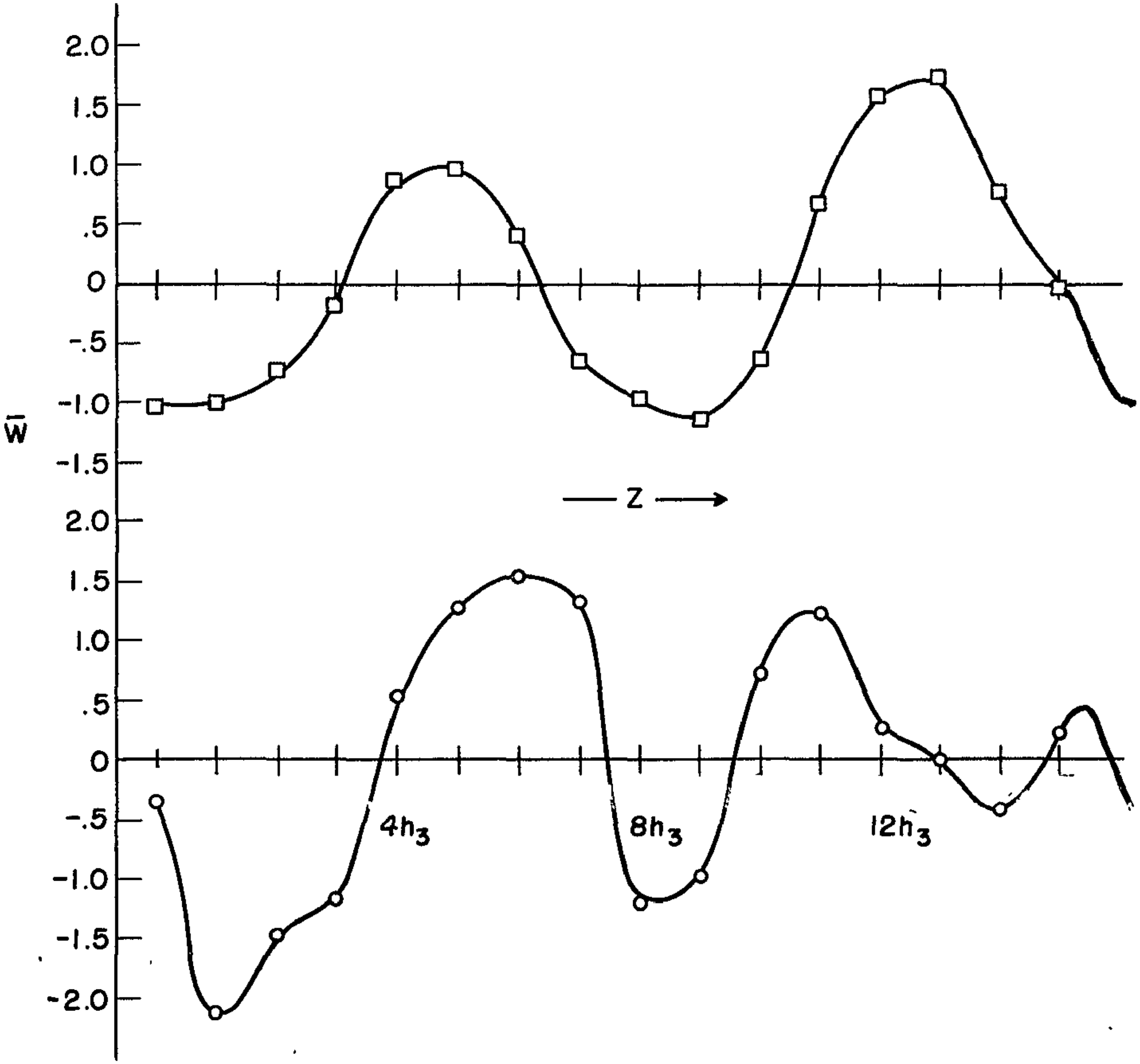

Fig. 4.23. Instantaneous spanwise variations of $\overline{w}$ at $t = 1.05$, $x = 4h_1$, $y = -.807$ (upper figure) and at $t = 1.425$, $x = 4h_1$, $y = -.304$ (lower figure).

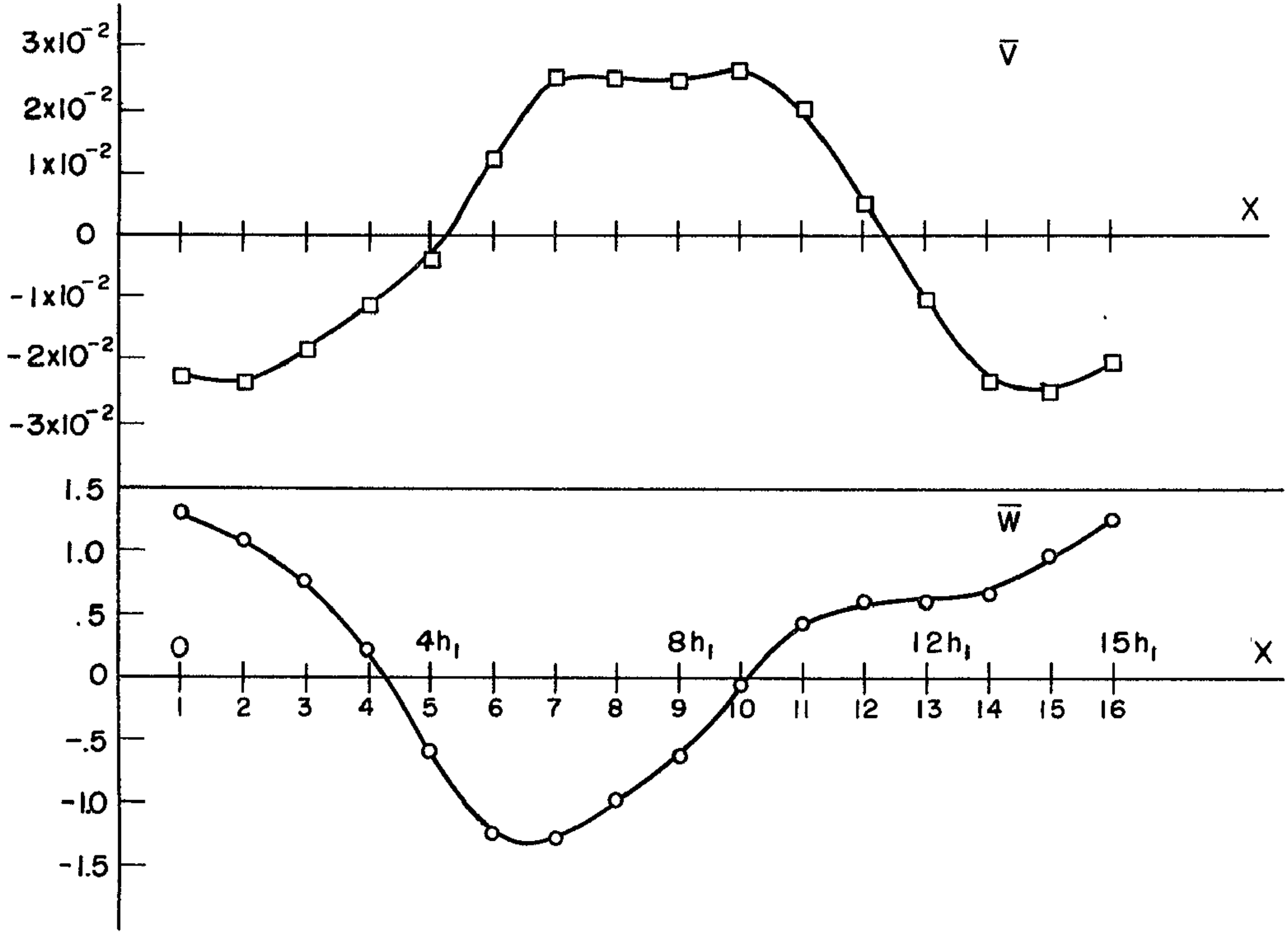

Fig. 4.24. Instantaneous streamwise variations of $\bar{v}$ (upper figure) and $\bar{w}$ (lower figure) at $t = 1.05$, $y^+ = 3.85$, $z = 8h_3$.

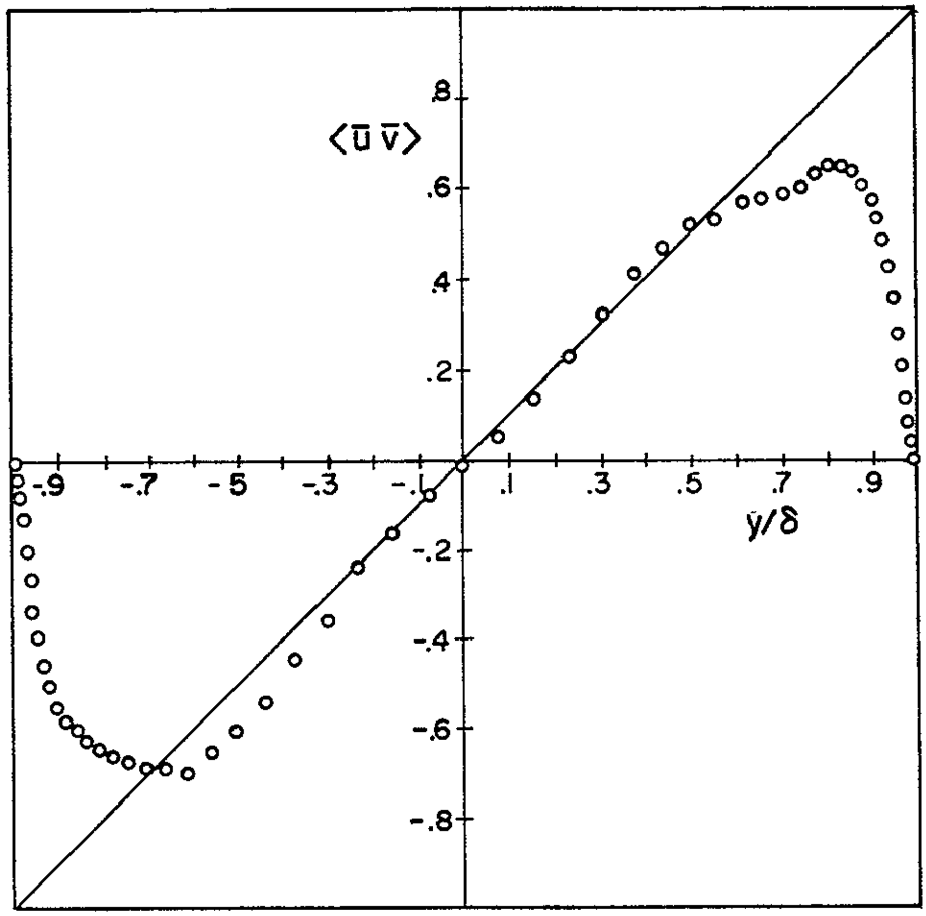

Fig. 4.25. Resolvable portion of turbulence stress averaged in time.

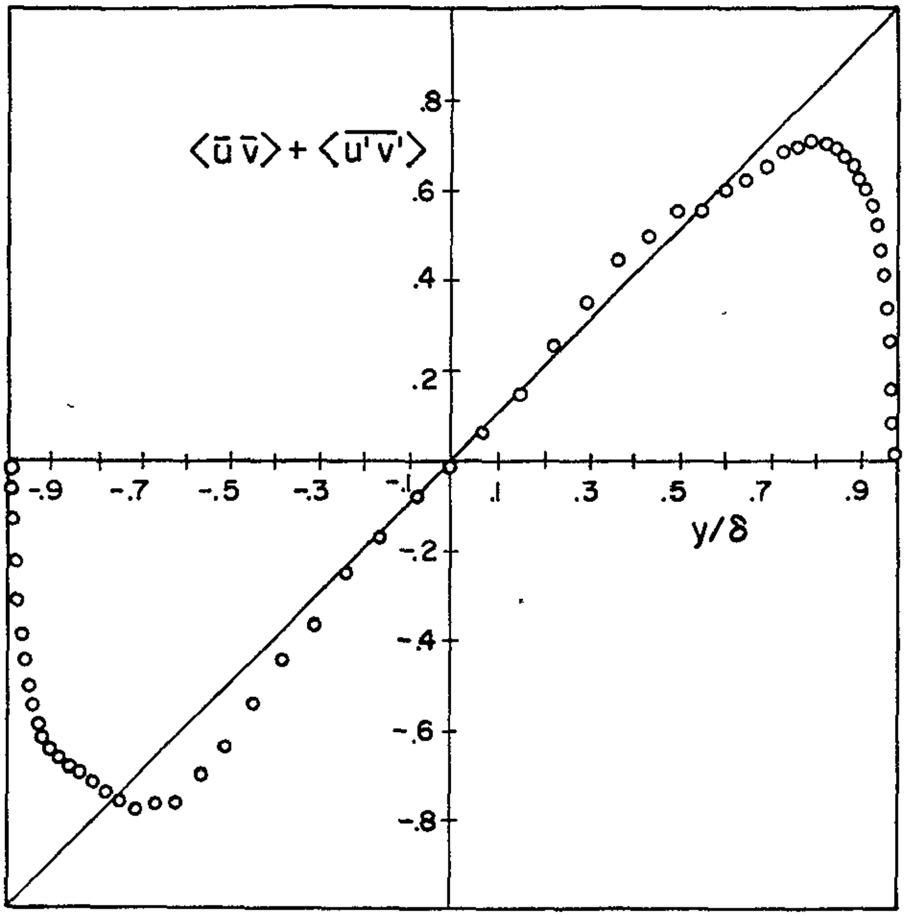

Fig. 4.26a. Total turbulence stress (resolvable portion + SGS contribution), averaged in time.

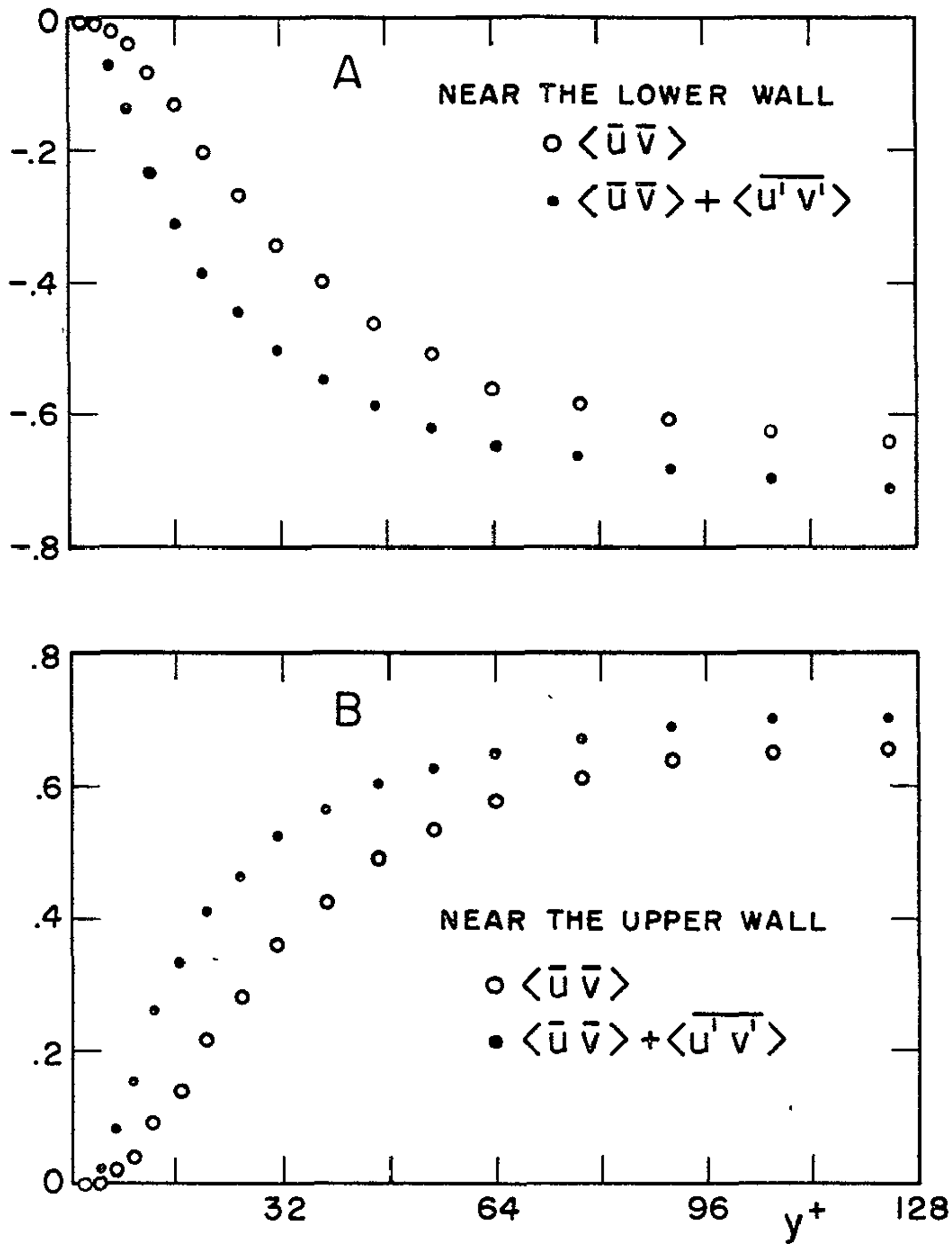

Fig. 4.26b. The resolvable and total turbulence stress in the vicinity of the walls.
A) Near the lower wall
B) Near the upper wall

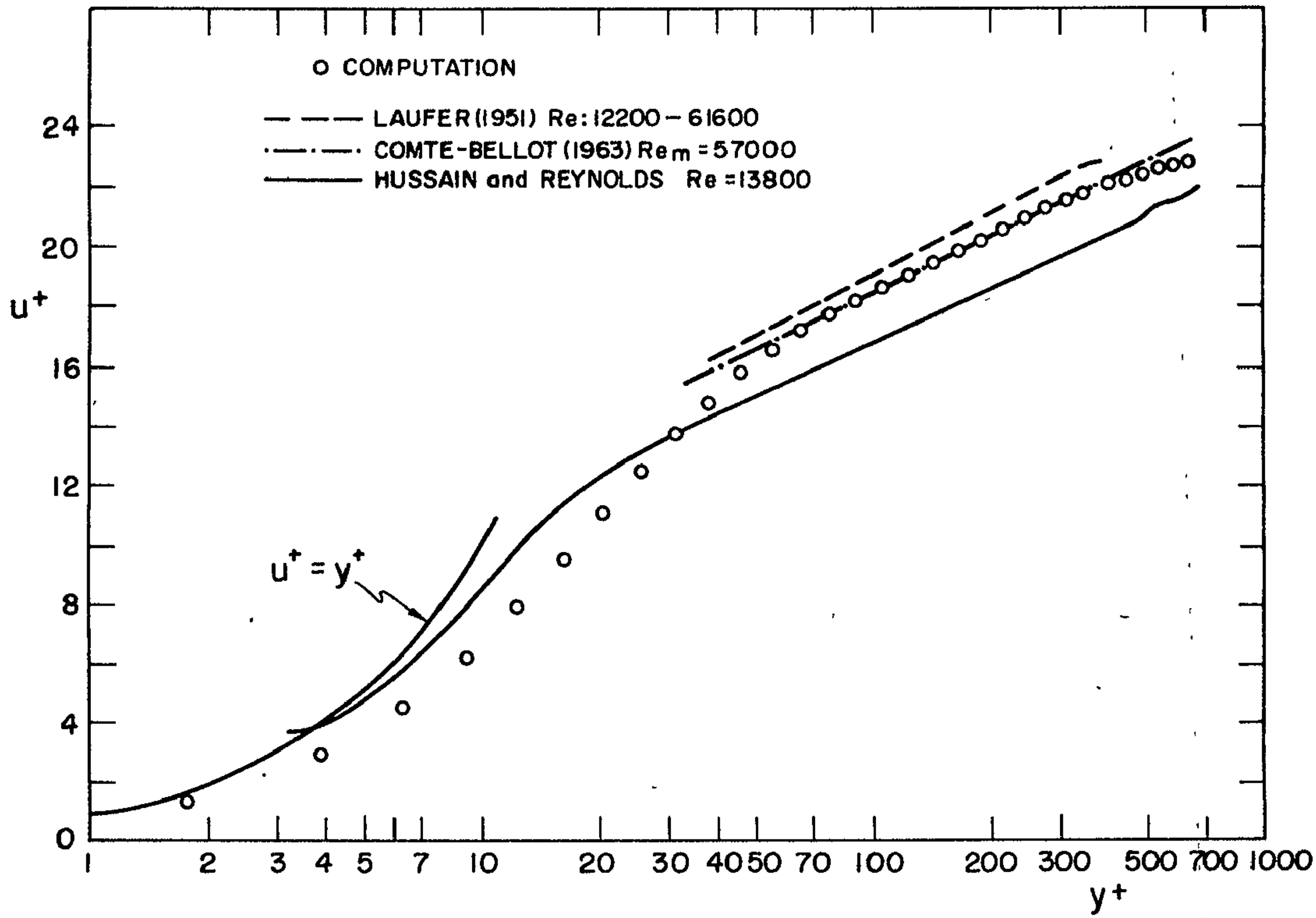

Fig. 4.27. Mean velocity profile. The experimental data of Laufer, Comte-Bellot, and Hussain and Reynolds are included.

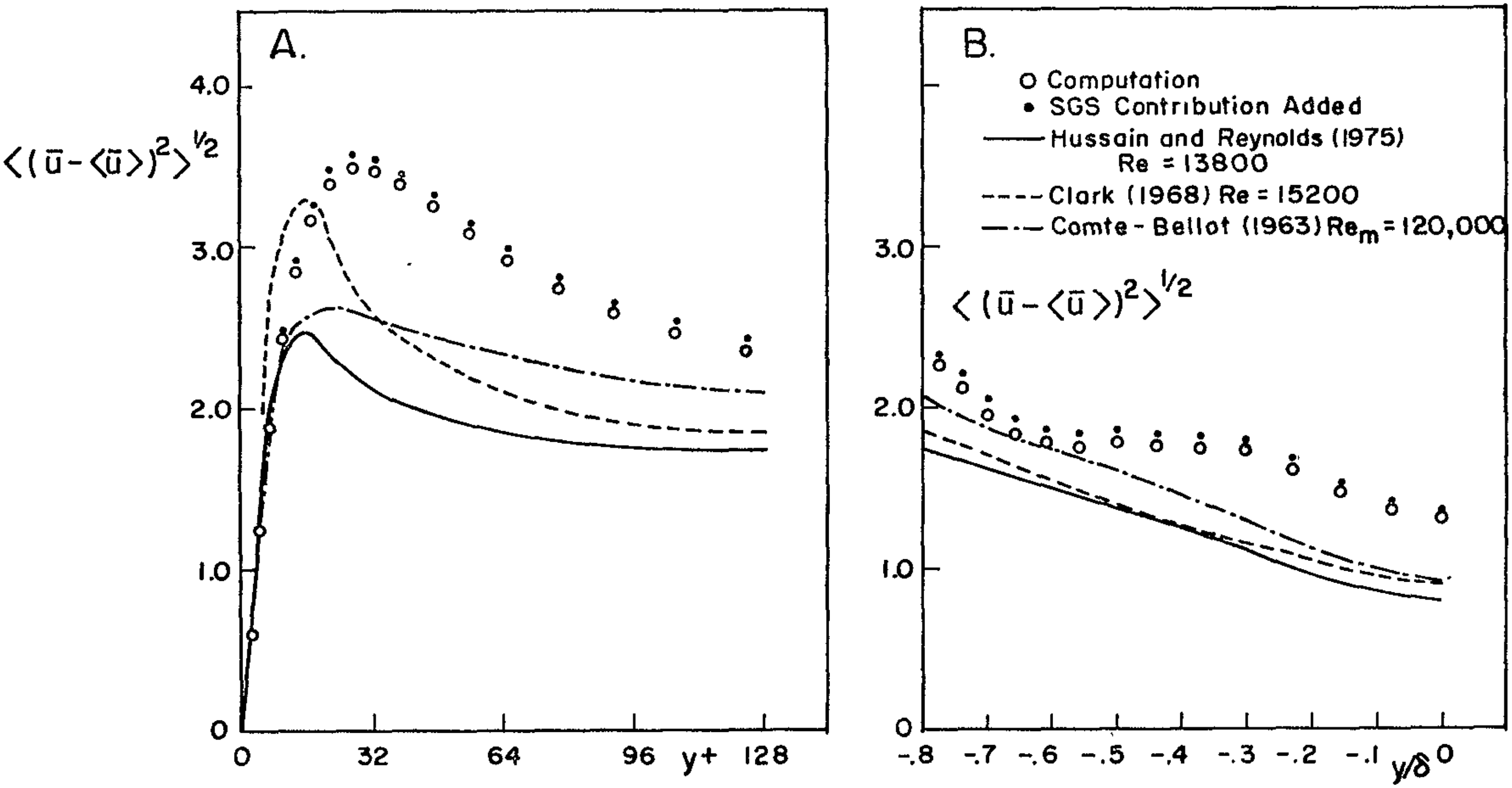

Fig. 4.28. Time-averaged streamwise turbulence intensity in the vicinity of the wall (A) and away from the wall (B).

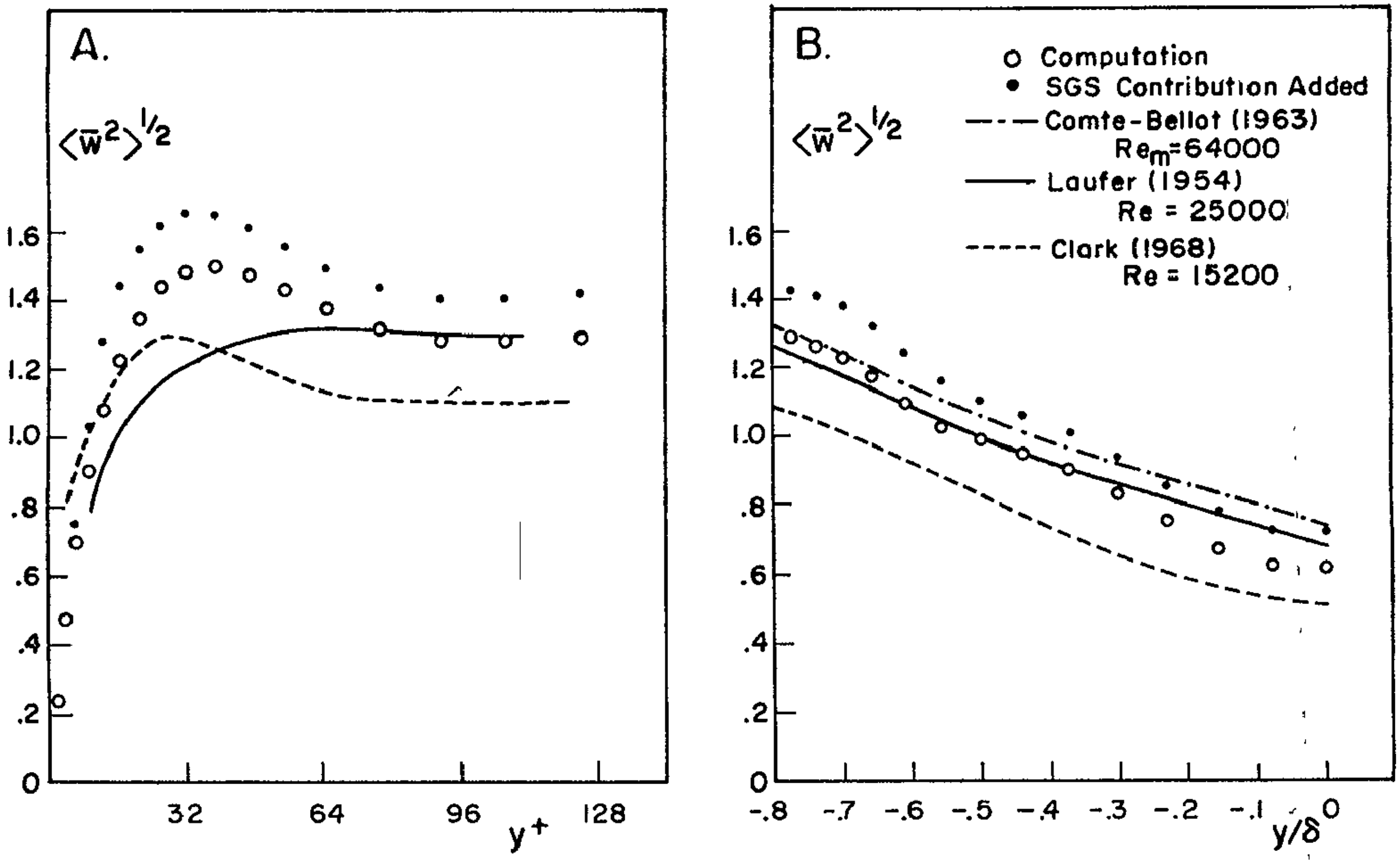

Fig. 4.29. Time-averaged spanwise turbulence intensity in the vicinity of the wall (A) and away from the wall (B).

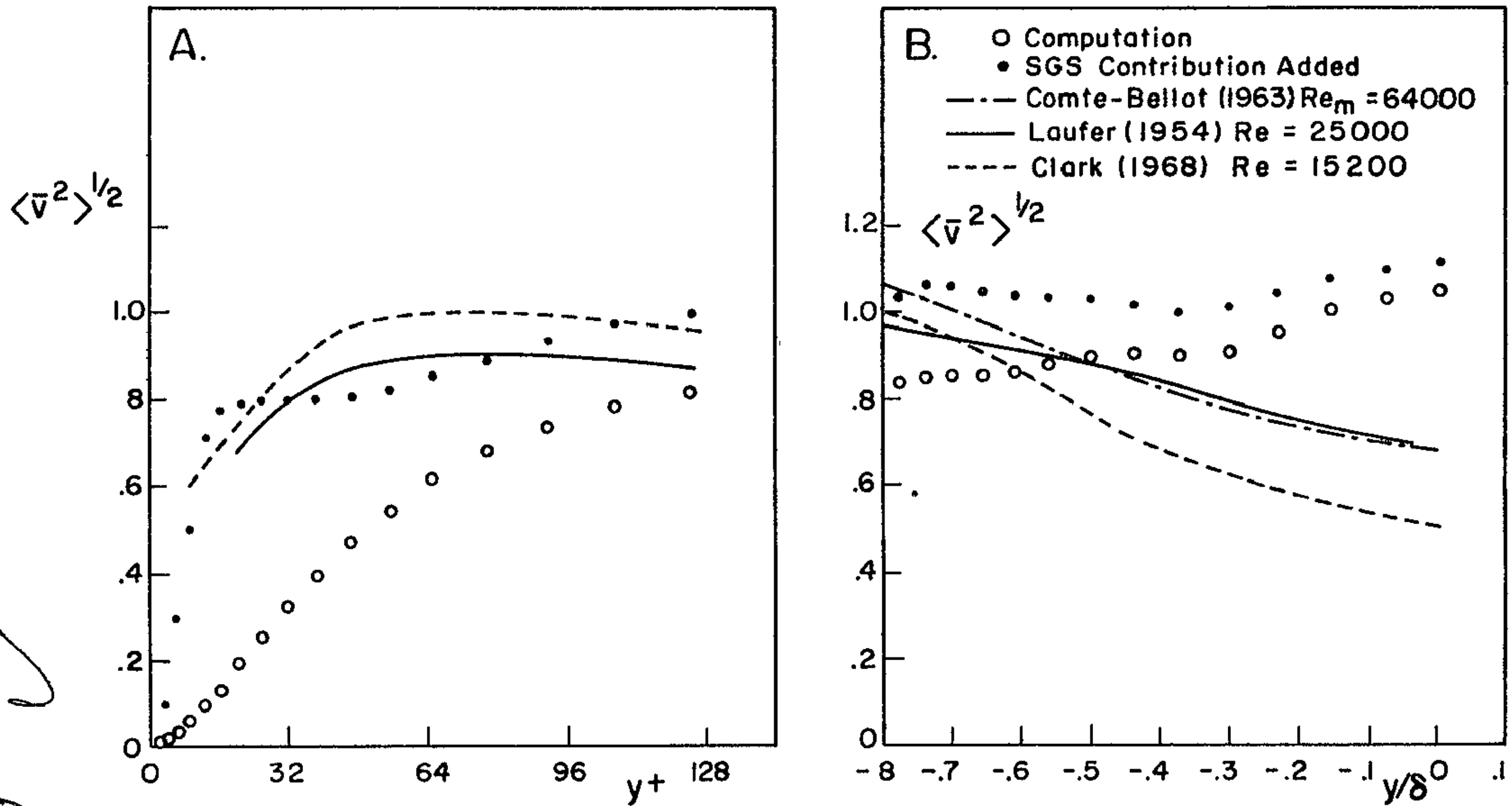

Fig. 4.30. Time-averaged vertical component of turbulence intensity in the vicinity of the wall (A) and away from the wall (B).

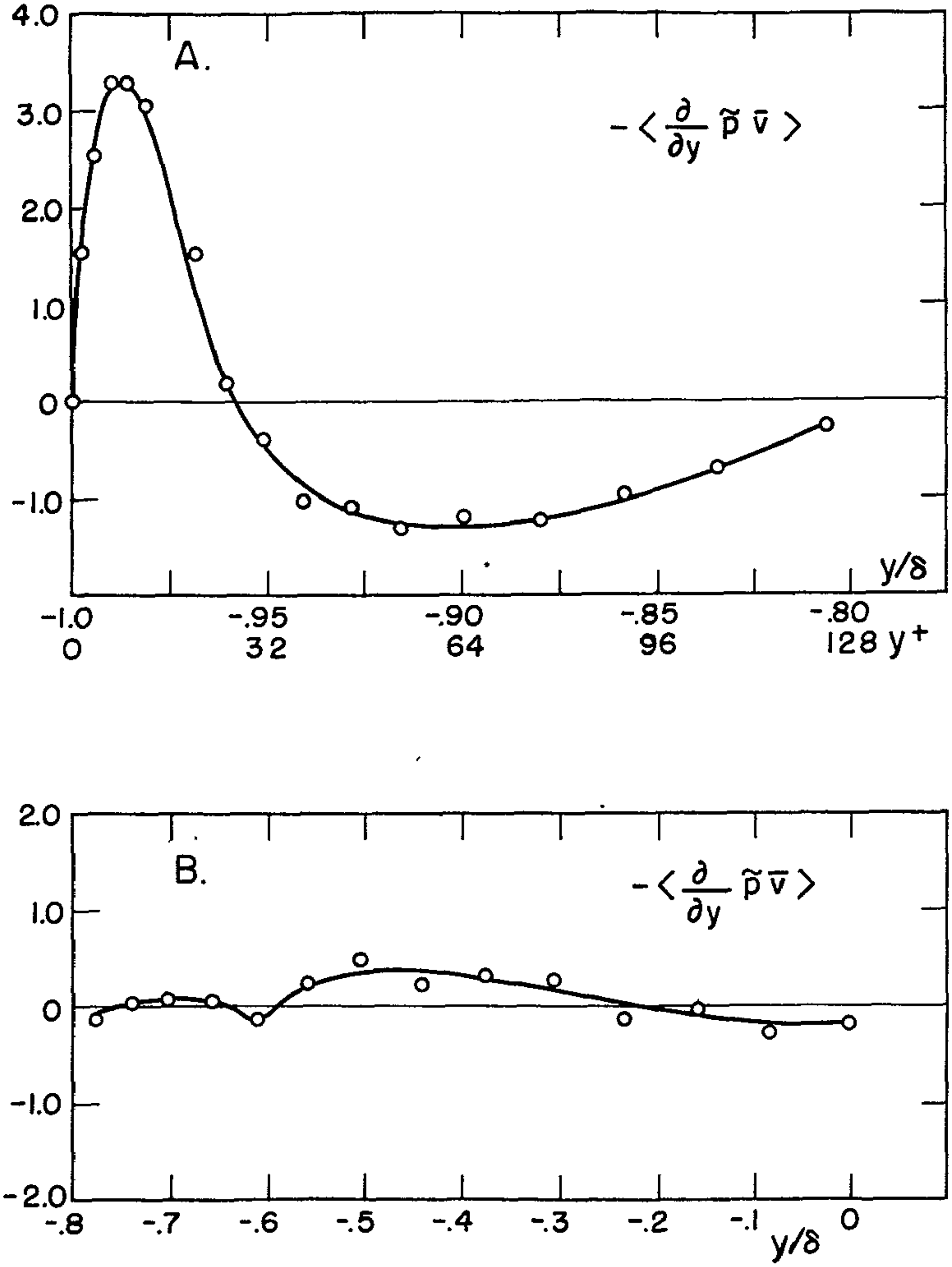

Fig. 4.31. Pressure work term in the vicinity of the wall (A) and away from the wall (B).

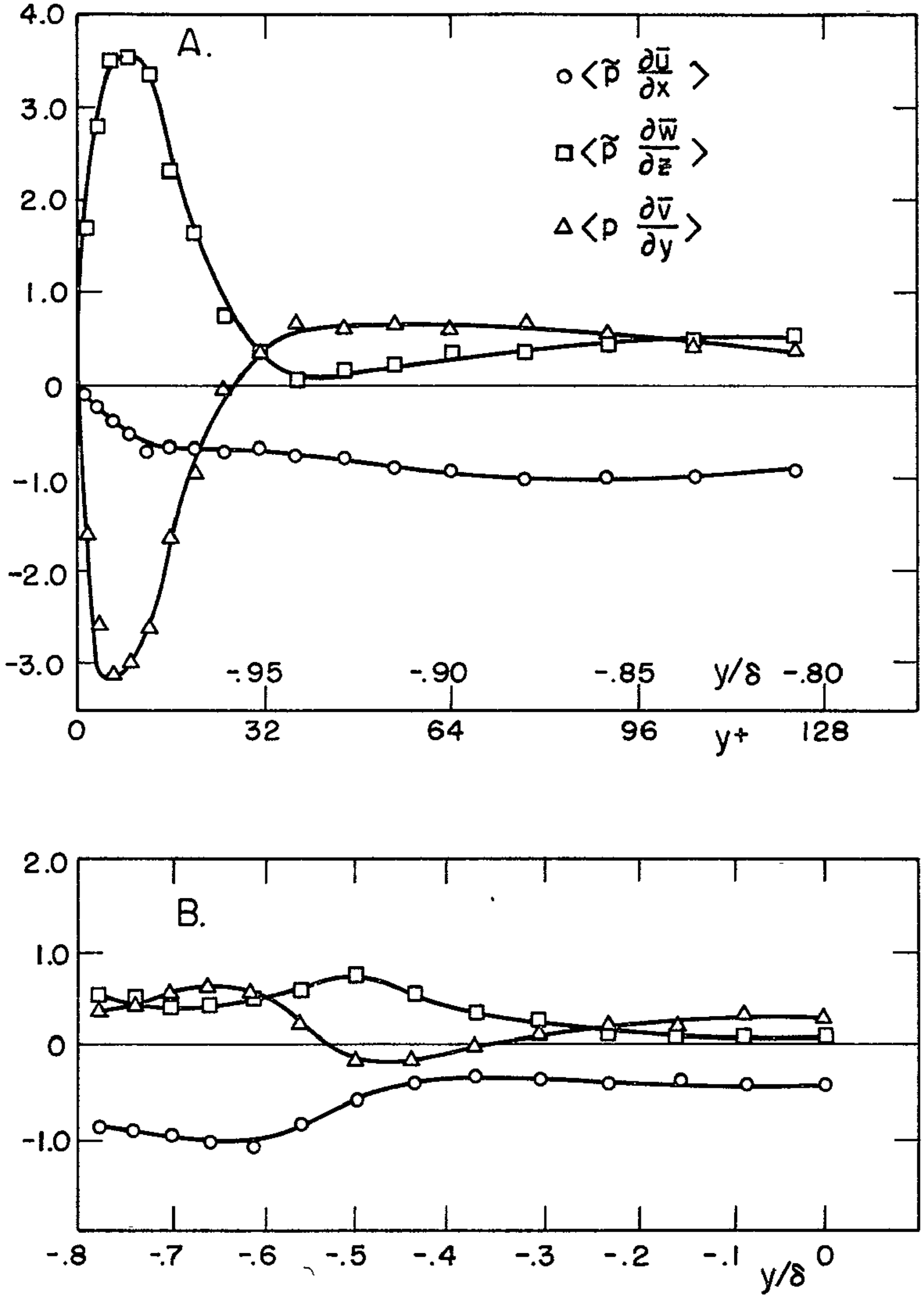

Fig. 4.32. · Pressure velocity-gradient correlations in the vicinity of the wall (A) and away from the wall (B).

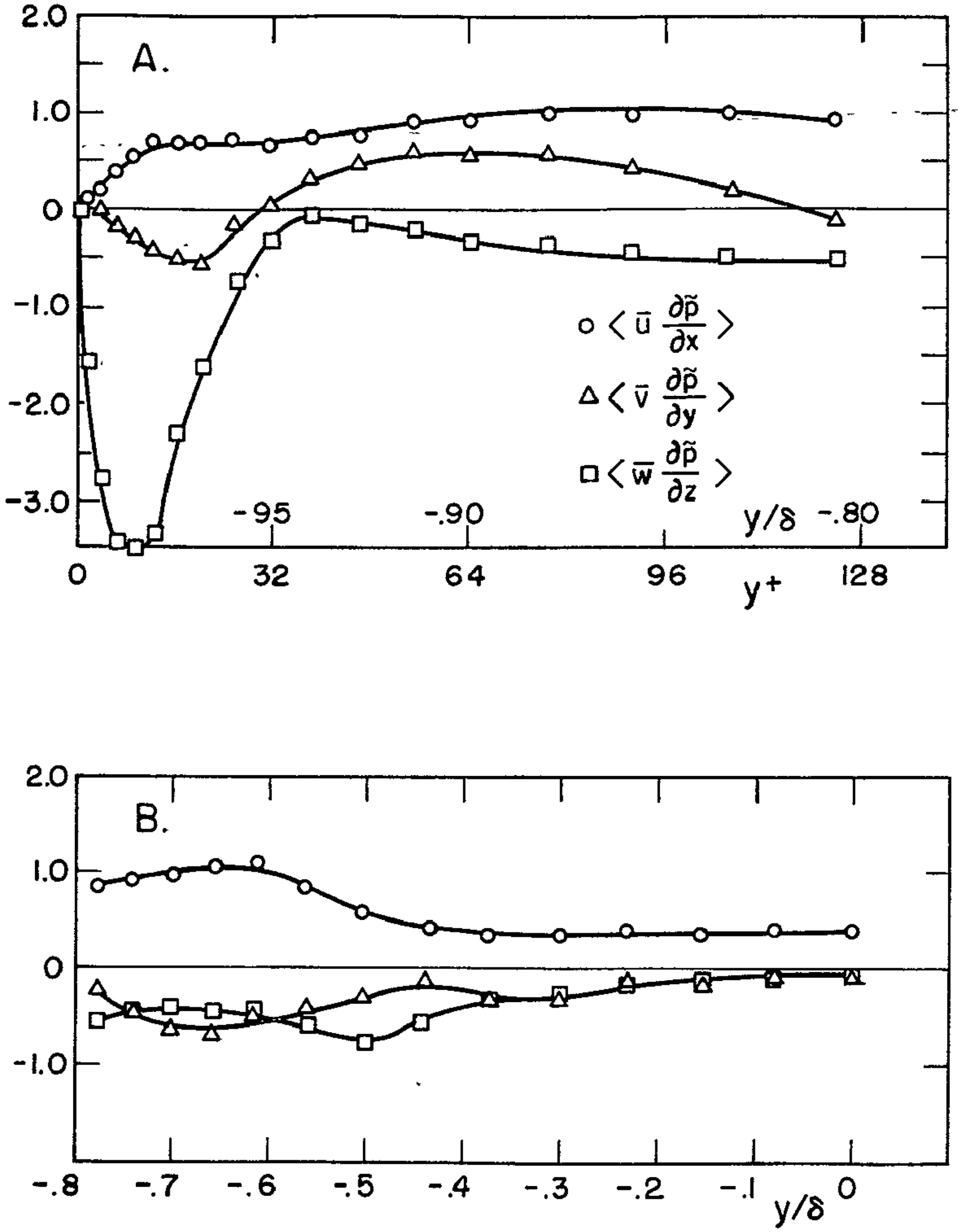

Fig. 4.33. Velocity pressure-gradient correlations in the vicinity of the wall (A) and away from the wall (B).

Appendix A

FILTERING WITH NON-UNIFORM FILTER WIDTH

In this appendix, we briefly discuss non-uniform width filtering and demonstrate its mathematical disadvantages. The use of such filters (non-uniform width) is desirable in the directions in which the flow is inhomogeneous (see Section 2.1). For demonstration, we consider only simple box averaging as the filtering operation.

Let

$$\overline{f}(x) = \frac{1}{(\Delta_+(x) + \Delta_-(x))} \int_{x-\Delta_-(x)}^{x+\Delta_+(x)} f(\xi)\, d\xi \qquad \text{(A.1)}$$

where Δ_+ and Δ_- are the distances from x to its adjacent grid points. They will be treated as continuous variables. Note that here we consider only a one-dimensional case. Differentiating $\overline{f}$ yields:

$$\frac{\partial \overline{f}}{\partial x} = - \frac{\frac{d}{dx}(\Delta_+ + \Delta_-)}{(\Delta_+(x) + \Delta_-(x))^2} \int_{x-\Delta_-}^{x+\Delta_+} f(\xi)\, d\xi$$

$$+ \frac{1}{(\Delta_+ + \Delta_-)} \left[f(x + \Delta_+)\left(1 + \frac{d\Delta_+}{dx}\right) - f(x - \Delta_-)\left(1 - \frac{d\Delta_-}{dx}\right) \right]$$

or

$$\frac{d\overline{f}}{dx} = - \frac{\frac{d}{dx}(\Delta_+ + \Delta_-)}{(\Delta_+ + \Delta_-)} \overline{f} + \frac{f(x + \Delta_+) - f(x - \Delta_-)}{(\Delta_+ + \Delta_-)} \qquad \text{(A.2)}$$

$$+ \frac{1}{(\Delta_+ + \Delta_-)} \left[f(x + \Delta_+) \frac{d\Delta_+}{dx} + f(x - \Delta_-) \frac{d\Delta_-}{dx} \right]$$

Using the definition (A.1), we have:

$$\overline{\frac{\partial f}{\partial x}} = \frac{1}{(\Delta_+ + \Delta_-)} \int_{x-\Delta_-}^{x+\Delta_+} \frac{\partial f}{\partial \xi}\, d\xi = \frac{1}{\Delta_+ + \Delta_-} \left[f(x + \Delta_+) - f(x - \Delta_-) \right] \qquad \text{(A.3)}$$

Substitution of (A.3) in (A.2) yields:

$$\overline{\frac{\partial f}{\partial x}} = \frac{\partial \overline{f}}{\partial x} + \frac{\frac{d}{dx}(\Delta_+ + \Delta_-)}{(\Delta_+ + \Delta_-)} \overline{f} - \frac{1}{(\Delta_+ + \Delta_-)} \left[f(x + \Delta_+) \frac{d\Delta_+}{dx} + f(x - \Delta_-) \frac{d\Delta_-}{dx} \right] \tag{A.4}$$

Thus, it is clear that, in general,

$$\overline{\frac{\partial f}{\partial x}} \neq \frac{\partial \overline{f}}{\partial x} .$$

The The above inequality and the presence of unfiltered quantities in (A.4) renders the use of explicit nonuniform width-filtering extremely difficult.

Appendix B

THE NUMERICAL DIFFICULTY WITH EXPLICIT TIME ADVANCING OF EQUATIONS OF MOTION

In this appendix, we formally demonstrate the numerical difficulty associated with the fully explicit numerical integration of the Navier-Stokes equations (see Section 3.3). Chebyshev polynomials* and Fourier series are used to represent the flow variables in the vertical and horizontal difections, respectively. Consider the governing equations:

$$\dot{u}_i = -\frac{\partial P}{\partial x_i} + H_i \tag{B.1}$$

where H_i contains the transport and diffusion terms and a $\bullet$ over a variable denotes time derivative. Let

$$P = \sum_{n=o}^{N_2}{}'' \sum_{k_1} \sum_{k_3} a_n(k_1,k_3)\ T_n(x_2)\ e^{i(k_1x_1+k_3x_3)} \tag{B.2}$$

$$u_i = \sum_{n=o}^{N_2}{}'' \sum_{k_1} \sum_{k_3} b_{in}(k_1,k_3)\ T_n(x_2)\ e^{i(k_1x_1+k_3x_3)} \tag{B.3}$$

$$H_i = \sum_{n=o}^{N_2}{}'' \sum_{k_1} \sum_{k_3} C_{in}(k_1,k_3)\ T_n(x_2)\, e^{i(k_1x_1+k_3x_3)} \tag{B.4}$$

and

$$\frac{\partial P}{\partial x_2} = \sum_{n=o}^{N_2}{}'' \sum_{k_1} \sum_{k_3} a_n'(k_1,k_3)\ T_n(x_2)\ e^{i(k_1x_1+k_3x_3)} \tag{B.5}$$

where $T_n(x_2)$ is the n^{th}-order Chebychev polynomial of the first kind and the double primes indicate that the first and last terms in the series are to be taken with factor 1/2. Eqns. (B.1), (B.2), (B.4), and (B.5) yield

$$\dot{u}_i = \sum_{m=o}^{N_2}{}'' \sum_{k_1'} \sum_{k_3'} \left\{ \begin{pmatrix} -ik_1'a_m(k_1',k_3') \\ -a_m'(k_1',k_3') \\ -ik_3'a_m(k_1',k_3') \end{pmatrix} + C_{im} \right\} T_m(x_2)\ e^{i(k_1'x_1+k_3'x_3)} \tag{B.6}$$

---

*Other choices are possible.

From (B.3) we readily obtain:

$$\dot{b}_{in}(k_1,k_3) = \frac{2}{N_1N_2N_3} \sum_{j=1}^{N_2-1}{}'' \sum_{x_1} \sum_{x_3} \dot{u}_i(x_1,\cos\theta_j,x_3) \cdot \cos n\theta_j \, e^{-i(k_1x_1+k_3x_3)} \quad \text{(B.7)}$$

where $x_{2_j} = \cos\theta_j$ (see Eqn. (3.12)) and $\theta_j = \pi j/N_2$, $j = 0,1,2,\ldots,N_2$. Note that here we have enforced the no-slip boundary conditions, i.e.,

$$\dot{u}_i(x_1,\theta_j,x_3)\Big|_{j=0,N_2} = 0 \ .$$

Substituting (B.6) into (B.7), we get:

$$\dot{b}_{in}(k_1,k_3) = \frac{2}{N_1N_2N_3} \sum_{j=1}^{N_2-1} \sum_{x_1} \sum_{x_3} \sum_{m=o}^{N_2}{}'' \sum_{k_1'} \sum_{k_3'} \left\{ \begin{pmatrix} -ik_1'a_m(k_1',k_3') \\ -a_m'(k_1',k_3') \\ -ik_3'a_m(k_1',k_3') \end{pmatrix} + C_{im} \right\} \cos\frac{m\pi j}{N_2} \cos\frac{n\pi j}{N_2} \, e^{i(k_1'x_1+k_3'x_3-k_1x_1-k_3x_3)} \quad \text{(B.8)}$$

The use of orthogonality of the expansion functions yields:

$$\dot{b}_{in}(k_1,k_3) = \begin{pmatrix} -ik_1a_n(k_1,k_3) \\ -a_n'(k_1,k_3) \\ -ik_3a_n(k_1,k_3) \end{pmatrix} + C_{in} - \frac{1}{N_2} \sum_{m=o}^{N_2}{}'' \left\{ \begin{pmatrix} -ik_1a_m(k_1,k_3) \\ -a_m'(k_1,k_3) \\ -ik_3a_m(k_1,k_3) \end{pmatrix} + C_{im} \right\} \left[(-1)^{n+m} + 1 \right] \quad \text{(B.9)}$$

The last term in (B.9), which is the result of enforcing the no-slip boundary conditions, is the source of trouble. To make this clear, consider the above equation for $i = 1$:

$$\dot{b}_{1n} = -ik_1a_n(k_1,k_3) + C_{1n} + \alpha(k_1,k_3) + (-1)^n \beta(k_1,k_3) \quad \text{(B.10)}$$

where

$$\alpha(k_1,k_3) = -\frac{1}{N_2} \sum_{m=o}^{N_2}{}'' (-ik_1a_m(k_1,k_3) + C_{1m})$$

$$\beta(k_1,k_3) = -\frac{1}{N_2}\sum_{m=o}^{N_2}{}'' (-ik_1 a_m(k_1,k_3) + C_{1m})(-1)^m$$

Multiplying (B.10) by $T_n(\cos\theta_j)$ and summing over all n yields:

$$\hat{\dot{u}}_1(k_1,x_{2_j},k_3) = -ik_1\hat{P}(k_1,x_{2_j},k_3) + \hat{H}_1(k_1,x_{2_j},k_3) + \alpha(k_1,k_3)\sum_{n=o}^{N_2}{}'' \cos n\theta_j + \beta(k_1,k_3)\sum_{n=o}^{N2}{}'' (-1)^n \cos n\theta_j$$

where ^ over a variable denotes two-dimensional Fourier transform of that variable. But

$$\sum_{n=o}^{N_2}{}'' \cos n\theta_j = \begin{cases} \frac{1}{2}\sin N_2\theta_j \cot\frac{\theta_j}{2} = 0 , & j \neq 0 \\ N_2 - 1 , & j = 0 \end{cases} \quad \text{(B.11)}$$

and

$$\sum_{n=o}^{N_2}{}'' (-1)^n \cos n\theta_j = \begin{cases} -\frac{1}{2}\sin N_2\theta_j \tan\frac{\theta_j}{2} = 0 , & j \neq N_2 \\ N_2 - 1 & j = N_2 \end{cases} \quad \text{(B.12)}$$

Note

$$\theta_j = \frac{\pi j}{N_2} \qquad j = 0,1,2,\ldots,N_2$$

Hence, it has been shown that, unless

$$\alpha(k_1,k_3) = 0 \quad \text{and} \quad \beta(k_1,k_3) = 0 , \qquad \text{(B.13)}$$

the two-dimensional discrete Fourier transform of u_1 is discontinuous at the walls. It should be noted that (B.13) is equivalent to

$$\left.\frac{\partial P}{\partial x}\right|_{x_2=\pm 1} = \left. H_1\right|_{x_2=\pm 1}$$

which is the streamwise momentum equation evaluated at the walls (see Eqn. (3.15)). Similarly, the two-dimensional discrete Fourier transform of u_2 or u_3 is discontinuous unless

$$\left.\frac{\partial P}{\partial y}\right|_{x_2=\pm 1} = \left.H_2\right|_{x_2=\pm 1}$$

or

$$\left.\frac{\partial P}{\partial z}\right|_{x_2=\pm 1} = \left.H_3\right|_{x_2=\pm 1}$$

respectively. Therefore, if Neumann boundary condition is used for the Poisson equation, the Fourier transforms of u_1 and u_3 will have discontinuity at the boundaries. On the other hand, if Dirichlet boundary condition is used, the Fourier transform of u_2 will be discontinuous at the walls. In practice, the presence of discontinuity in the dependent variables results in non-convergent expansions which render a meaningless computation. A remedy for this problem is presented in Section 3.4.

Appendix C

LISTING OF THE COMPUTER PROGRAM FOR THE CALCULATION OF TURBULENT CHANNEL FLOW

```
C******************** COMDECKS *****************************************
C********* BY CALLING A COMDECK THE DIMENSION OR COMMON STATEMENT       *
C********* FOLLOWING THE COMDECK WILL BE PLACED  IN THE CALLING ROUTINE*
C**********************************************************************
*COMDECK C1
      DIMENSION XB(16),YB(16)
*COMDECK C2
      COMMON/AVEDY/RMIU(65)
*COMDECK C3
      COMMON/WV/WAVEX(16),WAVEY(16),WAVEXS(16),WAVEYS(16)
*COMDECK C4
      DIMENSION BETA1(65),BETA2(65)
*COMDECK C5
      DIMENSION RHSV(4,61),AMB(4,4,61),AB(4,4,61),APB(4,4,61),AAUX(4,4,
     161),AMAUX(4,4,61),APAUX(4,4,61)
*COMDECK C6
      COMMON/SECOND/AP2(65),BP2(65),CP2(65)
*COMDECK C7
      COMMON/LAGRNG/AP(65),BP(65),CP(65),APR(65),BPR(65),CPR(65),DPR(65)
     1,EPR(65)
*COMDECK C8
      DIMENSION Z1(16,16),ZM1(16,16),D2(62)
*COMDECK C9
      DIMENSION BC1R(16,16),BC1I(16,16),BCM1R(16,16),BCM1I(16,16)
*COMDECK C10
      COMMON/DAT21/XR(16),XI(16)
*COMDECK C11
      DIMENSION HR(16,16,65)
      LEVEL 2,HR
*COMDECK A1
      COMMON/DATA7/FR(16,16),FI(16,16)
*COMDECK A2
      COMMON   DUDX(16,16,65)
*COMDECK A3
      COMMON/LCM4/DIVC(16,16,65)
      LEVEL 2,DIVC
*COMDECK A4
      COMMON/LARGE2/P(16,16,65)
      LEVEL 2,P
*COMDECK A5
      COMMON/LARGE1/G(16,16,65)
      LEVEL 2,G
*COMDECK A6
      COMMON/LCM2/U(16,16,65),V(16,16,65),W(16,16,65)
      LEVEL 2,U,V,W
*COMDECK A7
      COMMON/LCM1/H1(16,16,65),H2(16,16,65),H3(16,16,65)
      LEVEL 2,H1,H2,H3
*COMDECK A8
      COMMON/LCM3/RU(16,16,65),RV(16,16,65),RW(16,16,65)
      LEVEL 2,RU,RV,RW
*COMDECK A9
      COMMON/STR/ZETA(65),Z(65),RL(65),RM(65),E2,F2,EN,FN,R2,RN,A(65),
     1C(65),D(65),RR2,RRN
*COMDECK A10
      DIMENSION G(16,16,65)
      LEVEL 2,G
*COMDECK A11
      DIMENSION U1(16,16,65),U2(16,16,65),U3(16,16,65)
      LEVEL 2,U1,U2,U3
*COMDECK A12
      DIMENSION U(16,16,65),V(16,16,65),W(16,16,65)
      LEVEL 2,U,V,W
*COMDECK A13
      DIMENSION USUM(65),VSUM(65),WSUM(65)
*COMDECK B1
      COMMON/FLT/FILTX(16),FILTY(16)
*COMDECK B2
      COMMON/EDDY/CV(63)
*COMDECK B3
```

```
      COMMON/RECOVER/FACTOR(65)
*COMDECK B4
      COMMON/HORIAV/U2S(65),V2S(65),W2S(65),SSUM(65),EDYVI(65)
*COMDECK B5
      COMMON/AVEDY/MIU(65)
*COMDECK B6
      DIMENSION U2ST(65),V2ST(65),W2ST(65),UWT(65)
*COMDECK B7
      COMMON/SECOND/AP2(65),BP2(65),CP2(65)
*COMDECK B8
      COMMON/PENTA1/A1(65),B1(65),C1(65),D1(65),E1(65),F1(65)
*COMDECK B9
      DIMENSION U(16,16,65)
      LEVEL2,U
*DECK MAIN
      PROGRAM MAIN(INPUT,OUTPUT,TAPE8,TAPE9,TAPE10,TAPE11)
C*****THIS SUBROUTINE MONITORS THE OVERALL SEQUENCE OF THE COMPUTATION
C***** U,V,W ARE THE VELOCITIES IN STREAMWISE,X,SPANWISE,Y,AND VERTICAL,
C***** Z DIRECTIONS.
      COMMON/LTA1/USUM(65),UTSUM(65),STSUM(65),U2SMT(65),V2SMT(65)
     1,W2SMT(65),PVT(65),PUT(65),PUNST(65),PVNST(65),PWNST(65),PWT(65)
     2,TCONT
      COMMON/LTA2/PDUT(65),PDVT(65),PDWT(65),PDUNT(65),PDVNT(65),PDWNT
     1(65)
      COMMON/SGTT/SGST(65),ETED(65),U2STT(65),V2STT(65),W2STT(65)
     1,TSHGS,TSCNT
      COMMON/COUNT/IICONT
      COMMON/SING/IMR,JMR,IMI,JMI
      COMMON/ADV/NTIME
      DIMENSION A3(61),B3(61),C3(61),D3(61),E3(61)
      COMMON/TINC/DT
      COMMON/PENTA2/XI,QI,GI,YI,QJ,GJ,XN,QIN,GIN,YN,QJN,GJN,Q2,Q3,RC1,
     1RC2,RP1,RP2,RP3,RP4
*CALL C1
*CALL B5
      REAL MIU
      DIMENSION VAUX(4,61)
      DIMENSION AX(3,3,61),APX(3,3,61),AMX(3,3,61),AXX(3,3,61),
     1APXX(3,3,61),AMXX(3,3,61),VH(3,61)
*CALL C3
*CALL C4
*CALL C5
*CALL B7
*CALL C7
      COMMON/BC/CE1,CE2,CE3,CE4,CE5,CE6
      COMMON/IDENTN/CODE
*CALL A1
*CALL A2
*CALL A3
*CALL A4
*CALL A5
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/SCM2/LMAXP1,D1,D2,D9,D4,D5,D6
*CALL A6
*CALL A7
*CALL A8
*CALL A9
      COMMON/SCM3/DELTA1,DELTA2,RE,E
      INTEGER TIME,TEND
      TEND=200
      COF=1.5
      DT=0.001
      NTIME=0
      CODE=2.
      CALL INITIAL
      CALL TRANS
      CC=1./(IMAX*JMAX)
      TP=0.5
      C1=2.0
```

```
      C4=1.0
      LMAXM2=LMAX-2
      LMAXM3=LMAX-3
      LHP1=LMAX/2+1
      ICONT=0
      LCONT=0
      LMAXM1=LMAX-1
      CALL INICON
      NTIME=1
      CALL INITIAL
      DO 300 TIME=1,TEND
      NTIME=TIME
      ICONT=ICONT+1
      IICONT=ICONT-20
      CALL COURANT(DT,NTIME,TEND)
      CALL DIVG
      CALL RHS
      IF(NTIME.EQ.1) GO TO 360
      IF(IICONT.NE.0) GO TO 350
      ICONT=0
  360 CONTINUE
      CALL STAT
  350 CONTINUE
C***** DEFINE THE WAVE NUMBER INDEPENDENT ELEMENTS OF THE BLOCK -
C*****TRIDIAGONAL MATRIX
      DO 600 K=2,LMAX
      BETA1(K)=-C1/(DT*(E+MIU(K)))
      BETA2(K)=-C1/(DT*(E+2.*MIU(K)))
  600 CONTINUE
C***** DEFINE THE ELEMENTS OF THE TRIDIAGONAL MATRIX FOR THE CASE K1=K2=0.
      DO 800 K=1,LMAXM3
      KP2=K+2
      B3(K)=BP2(KP2)+BETA1(KP2)
      A3(K)=AP2(KP2)
  800 C3(K)=CP2(KP2)
      T=(Z(3)-Z(2))*0.5
      AK=1./(TP*DT*BETA2(3))
C***** AMB,AB,APB ARE THE ELEMENTS OF THE BLOCK TRIDIAGONAL MATRIX.
      DO 640 M=1,4
      DO 640 N=1,4
      DO 640 K=1,LMAXM3
      AB(M,N,K)=0.
      AMB(M,N,K)=0.
      APB(M,N,K)=0.
  640 CONTINUE
      DO 645 K=1,LMAXM3
      KP2=K+2
      AB(1,1,K)=BP2(KP2)+BETA1(KP2)
      AB(2,2,K)=AB(1,1,K)
      AB(3,3,K)=BP(KP2)
      AB(4,4,K)=BETA2(KP2)*BP(KP2)*DT*TP
      AB(4,3,K)=BP2(KP2)+BETA2(KP2)
  645 CONTINUE
       AB(4,4,1)=CE2*BETA2(3)*DT*TP
      AB(4,4,LMAXM3)=CE5*BETA2(LMAXM1)*DT*TP
      AB(4,3,1)=BP2(3)+BETA2(3)*CE1
      AB(4,3,LMAXM3)=BP2(LMAXM1)+BETA2(LMAXM1)*CE6
      DO 650 K=1,LMAXM3
      KP2=K+2
      APB(1,1,K)=CP2(KP2)
      APB(2,2,K)=APB(1,1,K)
      APB(3,3,K)=CP(KP2)
      APB(4,4,K)=CP(KP2)*BETA2(KP2)*DT*TP
      APB(4,3,K)=CP2(KP2)
      AMB(1,1,K)=AP2(KP2)
      AMB(2,2,K)=AP2(KP2)
      AMB(4,3,K)=AP2(KP2)
      AMB(4,4,K)=AP(KP2)*BETA2(KP2)*DT*TP
      AMB(3,3,K)=AP(KP2)
  650 CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      AMB(4,4,LMAXM3)=CE4*BETA2(LMAXM1)*DT*TP
      APB(4,4,1)=CE3*BETA2(3)*DT*TP
C***** DEFINE THE ELEMENTS OF THE,K1=0,BLOCK TRIDIAGONAL SYSTEM
      DO 750 M=1,3
      DO 750 N=1,3
      DO 750 K=1,LMAXM3
      AX(M,N,K)=0.
      APX(M,N,K)=0.
      AMX(M,N,K)=0.
  750 CONTINUE
      DO 752 K=1,LMAXM3
      AX(2,2,K)=AB(4,3,K)
      APX(2,2,K)=APB(4,3,K)
      AMX(2,2,K)=AMB(4,3,K)
      AX(3,1,K)=AB(2,2,K)
      APX(3,1,K)=APB(2,2,K)
      AMX(3,1,K)=AMB(2,2,K)
      AX(2,3,K)=AB(4,4,K)
      APX(2,3,K)=APB(4,4,K)
      AMX(2,3,K)=AMB(4,4,K)
      AX(1,2,K)=AB(3,3,K)
      APX(1,2,K)=APB(3,3,K)
      AMX(1,2,K)=AMB(3,3,K)
  752 CONTINUE
C***** DEFINE THE RHS OF THE BLOCK TRIDIAGONAL SYSTEM
      CALL VISCOS(U)
      DO 610 K=3,LMAXM1
      DO 610 J=1,JMAX
      DO 610 I=1,IMAX
      U(I,J,K)=BETA1(K)*(U(I,J,K)+DT*(COF*H1(I,J,K)-0.5*RU(I,J,K)))-
     1DUDX(I,J,K)*C4
  610 CONTINUE
      CALL VISCOS(V)
      DO 615 K=3,LMAXM1
      DO 615 J=1,JMAX
      DO 615 I=1,IMAX
      V(I,J,K)=BETA1(K)*(V(I,J,K)+DT*(COF*H2(I,J,K)-0.5*RV(I,J,K)))-
     1DUDX(I,J,K)*C4
  615 CONTINUE
      CALL VISCOS(W)
      DO 620 K=3,LMAXM1
      DO 620 J=1,JMAX
      DO 620 I=1,IMAX
      W(I,J,K)=BETA2(K)*(W(I,J,K)+DT*(COF*H3(I,J,K)-0.5*RW(I,J,K)))-
     1DUDX(I,J,K)*C4
  620 CONTINUE
C***** FOURRIER TRANSFORM
      DO 625 K=3,LMAXM1
      CALL MOVLEV(U(1,1,K),FR(1,1),IJ)
      CALL FFTX(1.0)
      CALL FFTY(1.0,CC)
      CALL MOVLEV(FR(1,1),U(1,1,K),IJ)
      CALL MOVLEV(FI(1,1),RU(1,1,K),IJ)
      CALL MOVLEV(V(1,1,K),FR(1,1),IJ)
      CALL FFTX(1.0)
      CALL FFTY(1.0,CC)
      CALL MOVLEV(FR(1,1),V(1,1,K),IJ)
      CALL MOVLEV(FI(1,1),RV(1,1,K),IJ)
      CALL MOVLEV(W(1,1,K),FR(1,1),IJ)
      CALL FFTX(1.0)
      CALL FFTY(1.0,CC)
      CALL MOVLEV(FR(1,1),W(1,1,K),IJ)
      CALL MOVLEV(FI(1,1),RW(1,1,K),IJ)
  625 CONTINUE
C****** THE REAL AND IMAGINARY PARTS OF THE FOURIER TRANSFORM OF THE RHS
C****** OF THE BLOCK TRIDIAGONAL MATRIX IS COMPUTED .
C****** NOW DEFINE THE MATRIX ELEMENTS FOR EACH K1 AND K2.
      NHP1X=IMAX/2+1
      NHP1Y=JMAX/2+1
      NHP2X=NHP1X+1
```

```
      NHP2Y=NHP1Y+1
      DO 630 J=1,JMAX
      DO 630 I=1,IMAX
      WAV=WAVEXS(I)+WAVEYS(J)
      IF(I.EQ.1.AND.J.EQ.1) GO TO 662
      IF(I.EQ.1) GO TO 410
      IF(J.EQ.1) GO TO 420
      IF(J.GE.NHP2Y) GO TO 430
  410 IF(J.LT.NHP2Y) GO TO 630
      GO TO 722
  420 IF(I.LT.NHP2X) GO TO 630
      GO TO 440
  430 IF(I.EQ.1.OR.I.EQ.NHP1X) GO TO 630
  440 CONTINUE
      DO 635 K=1,LMAXM3
      KP2=K+2
C****** FIRST SOLVE FOR IMAGINARY PART OF U,V,AND REAL PART OF W AND P.
      RHSV(1,K)=RU(I,J,KP2)
      RHSV(2,K)=RV(I,J,KP2)
      RHSV(3,K)=0.
      RHSV(4,K)=W(I,J,KP2)
  635 CONTINUE
      DO 647 K=1,LMAXM3
      KP2=K+2
      AB(3,1,K)=-WAVEX(I)*IMAX
      AB(3,2,K)=-WAVEY(J)*JMAX
      AB(1,4,K)=-AB(3,1,K)*BETA1(KP2)*DT*TP
      AB(2,4,K)=-AB(3,2,K)*BETA1(KP2)*DT*TP
  647 CONTINUE
C***** REARRANGING THE ROWS FOR CENTRAL DIFFERENCING
      DO 655 M=1,4
      DO 655 K=1,LMAXM3
      AAUX(1,M,K)=AB(3,M,K)
      AAUX(4,M,K)=AB(1,M,K)
      AAUX(3,M,K)=AB(4,M,K)
      APAUX(1,M,K)=APB(3,M,K)
      APAUX(4,M,K)=APB(1,M,K)
      APAUX(3,M,K)=APB(4,M,K)
      AMAUX(1,M,K)=AMB(3,M,K)
      AMAUX(4,M,K)=AMB(1,M,K)
      AMAUX(3,M,K)=AMB(4,M,K)
      AAUX(2,M,K)=AB(2,M,K)
      AMAUX(2,M,K)=AMB(2,M,K)
      APAUX(2,M,K)=APB(2,M,K)
  655 CONTINUE
      DO 310 M=1,4
      DO 310 K=1,LMAXM3
  310 VAUX(M,K)=RHSV(M,K)
      DO 315 K=1,LMAXM3
      RHSV(1,K)=VAUX(3,K)
      RHSV(4,K)=VAUX(1,K)
      RHSV(3,K)=VAUX(4,K)
  315 CONTINUE
      IMR=I
      JMR=J
      CALL MTDAG(AMAUX,AAUX,APAUX,RHSV,4,LMAXM3)
      DO 660 K=3,LMAXM1
      KM2=K-2
      RU(I,J,K)=RHSV(1,KM2)
      RV(I,J,K)=RHSV(2,KM2)
      W(I,J,K)=RHSV(3,KM2)
      G(I,J,K)=RHSV(4,KM2)
  660 CONTINUE
C***** COMPUTE THE REAL PART OF PRESSURE TRANSFORM AT THE WALL.
      G(I,J,2)=QI*G(I,J,3)+GI*G(I,J,4)-(2.*(1.-CE1)/(AP(3)*DT))*W(I,J,3)
      G(I,J,LMAX)=QIN*G(I,J,LMAXM1)+GIN*G(I,J,LMAXM2)-(2.*(1.-CE6)/
     1(CP(LMAXM1)*DT))*W(I,J,LMAXM1)
      GO TO 630
  662 CONTINUE
      G(I,J,3)=T*W(I,J,3)*AK
```

```
      G(I,J,4)=(1.-T*BP(3))*G(I,J,3)/(T*CP(3))
      G(I,J,2)=0.
      G(I,J,1)=-CP(2)*G(I,J,3)/AP(2)
      DO 663 K=4,LMAX
      AK=1./(TP*DT*BETA2(K))
      G(I,J,K+1)=(W(I,J,K)*AK-AP(K)*G(I,J,K-1)-BP(K)*G(I,J,K))/CP(K)
  663 CONTINUE
      DO 664 K=2,LMAX
      W(I,J,K)=0.
      RU(I,J,K)=0.
      RV(I,J,K)=0.
  664 CONTINUE
      GO TO 630
C***** SOLVE WHEN K1=0
  722 CONTINUE
C***** FIRST SOLVE FOR U, SIMPLE TRIDIAGONAL
      DO 724 K=1,LMAXM3
  724 D3(K)=RU(I,J,K+2)
      CALL TRIB(A3,B3,C3,E3,D3,LMAXM3)
      DO 726 K=3,LMAXM1
  726 RU(I,J,K)=D3(K-2)
C****** SOLVE FOR V,W,AND P,BLOCK TRIDIAGONAL
      DO 728 K=1,LMAXM3
      KP2=K+2
      VH(1,K)=0.
      VH(2,K)=W(I,J,KP2)
      VH(3,K)=RV(I,J,KP2)
  728 CONTINUE
      DO 730 K=1,LMAXM3
      KP2=K+2
      AX(1,1,K)=-WAVEY(J)*JMAX
      AX(3,3,K)=-AX(1,1,K)*BETA1(KP2)*DT*TP
  730 CONTINUE
      DO 732 M=1,3
      DO 732 N=1,3
      DO 732 K=1,LMAXM3
      AXX(M,N,K)=AX(M,N,K)
      APXX(M,N,K)=APX(M,N,K)
      AMXX(M,N,K)=AMX(M,N,K)
  732 CONTINUE
      IMR=I
      JMR=J
      CALL MTDAG(AMXX,AXX,APXX,VH,3,LMAXM3)
      DO 734 K=3,LMAXM1
      KM2=K-2
      RV(I,J,K)=VH(1,KM2)
      W(I,J,K)=VH(2,KM2)
      G(I,J,K)=VH(3,KM2)
  734 CONTINUE
C***** COMPUTE THE REAL PART OF PRESSURE TRANSFORM AT THE WALL.
      G(I,J,2)=QI*G(I,J,3)+GI*G(I,J,4)-(2.*(1.-CE1)/(AP(3)*DT))*W(I,J,3)
      G(I,J,LMAX)=QIN*G(I,J,LMAXM1)+GIN*G(I,J,LMAXM2)-(2.*(1.-CE6)/
     1(CP(LMAXM1)*DT))*W(I,J,LMAXM1)
  630 CONTINUE
      DO 665 J=1,JMAX
      DO 665 I=1,IMAX
      IF(I.EQ.1.AND.J.EQ.1) GO TO 810
      WAV=WAVEXS(I)+WAVEYS(J)
      IF(I.EQ.1) GO TO 510
      IF(J.EQ.1) GO TO 520
      IF(J.GE.NHP2Y) GO TO 530
  510 IF(J.LT.NHP2Y) GO TO 665
      GO TO 736
  520 IF(I.LT.NHP2X) GO TO 665
      GO TO 540
  530 IF(I.EQ.1.OR.I.EQ.NHP1X) GO TO 665
  540 CONTINUE
C****** NOW SOLVE FOR REAL PART OF U,V,AND IMAGINARY PART OF W AND P.
      DO 670 K=1,LMAXM3
      KP2=K+2
```

```
      RHSV(1,K)=U(I,J,KP2)
      RHSV(2,K)=V(I,J,KP2)
      RHSV(3,K)=0.
      RHSV(4,K)=RW(I,J,KP2)
  670 CONTINUE
      DO 677 K=1,LMAXM3
      KP2=K+2
      AB(3,1,K)=WAVEX(I)*IMAX
      AB(3,2,K)=WAVEY(J)*JMAX
      AB(1,4,K)=-AB(3,1,K)*BETA1(KP2)*DT*TP
      AB(2,4,K)=-AB(3,2,K)*BETA1(KP2)*DT*TP
  677 CONTINUE
C***** REARRANGING THE ROWS FOR CENTRAL DIFFERENCING
      DO 649 M=1,4
      DO 649 K=1,LMAXM3
      AAUX(1,M,K)=AB(3,M,K)
      AAUX(4,M,K)=AB(1,M,K)
      AAUX(3,M,K)=AB(4,M,K)
      APAUX(1,M,K)=APB(3,M,K)
      APAUX(4,M,K)=APB(1,M,K)
      APAUX(3,M,K)=APB(4,M,K)
      AMAUX(1,M,K)=AMB(3,M,K)
      AMAUX(4,M,K)=AMB(1,M,K)
      AMAUX(3,M,K)=AMB(4,M,K)
      AAUX(2,M,K)=AB(2,M,K)
      AMAUX(2,M,K)=AMB(2,M,K)
      APAUX(2,M,K)=APB(2,M,K)
  649 CONTINUE
      DO 320 M=1,4
      DO 320 K=1,LMAXM3
  320 VAUX(M,K)=RHSV(M,K)
      DO 325 K=1,LMAXM3
      RHSV(1,K)=VAUX(3,K)
      RHSV(4,K)=VAUX(1,K)
      RHSV(3,K)=VAUX(4,K)
  325 CONTINUE
      IMI=I
      JMI=J
      CALL MTDAG(AMAUX,AAUX,APAUX,RHSV,4,LMAXM3)
      DO 690 K=3,LMAXM1
      KM2=K-2
      U(I,J,K)=RHSV(1,KM2)
      V(I,J,K)=RHSV(2,KM2)
      RW(I,J,K)=RHSV(3,KM2)
      DUDX(I,J,K)=RHSV(4,KM2)
  690 CONTINUE
C***** COMPUTE THE IMAGINARY PART OF PRESSURE TRANSFORM AT THE WALL.
      DUDX(I,J,2)=QI*DUDX(I,J,3)+GI*DUDX(I,J,4)-(2.*(1.-CE1)/(AP(3)*DT
     1))*RW(I,J,3)
      DUDX(I,J,LMAX)=QIN*DUDX(I,J,LMAXM1)+GIN*DUDX(I,J,LMAXM2)-(2.
     1*(1.-CE6)/(CP(LMAXM1)*DT))*RW(I,J,LMAXM1)
      GO TO 665
C***** SIMPLE TRIDIAGONAL SOLUTION WHEN K1=0 AND K2=0.
  810 CONTINUE
      DO 820 K=1,LMAXM3
  820 D3(K)=U(I,J,K+2)
      CALL TRIB(A3,B3,C3,E3,D3,LMAXM3)
      DO 825 K=3,LMAXM1
  825 U(I,J,K)=D3(K-2)
      DO 830 K=1,LMAXM3
  830 D3(K)=V(I,J,K+2)
      CALL TRIB(A3,B3,C3,E3,D3,LMAXM3)
      DO 835 K=3,LMAXM1
  835 V(I,J,K)=D3(K-2)
      GO TO 665
C***** SOLVE WHEN K1=0
  736 CONTINUE
C***** FIRST SOLVE FOR U, SIMPLE TRIDIAGONAL
      DO 738 K=1,LMAXM3
  738 D3(K)=U(I,J,K+2)
```

```
      CALL TRIB(A3,B3,C3,E3,D3,LMAXM3)
      DO 740 K=3,LMAXM1
  740 U(I,J,K)=D3(K-2)
C***** SOLVE FOR V,W,AND P,BLOCK TRIDIAGONAL
      DO 742 K=1,LMAXM3
      KP2=K+2
      VH(1,K)=0.
      VH(2,K)=RW(I,J,KP2)
      VH(3,K)=V(I,J,KP2)
  742 CONTINUE
      DO 744 K=1,LMAXM3
      KP2=K+2
      AX(1,1,K)=WAVEY(J)*JMAX
      AX(3,3,K)=-AX(1,1,K)*BETA1(KP2)*DT*TP
  744 CONTINUE
      DO 746 M=1,3
      DO 746 N=1,3
      DO 746 K=1,LMAXM3
      AXX(M,N,K)=AX(M,N,K)
      APXX(M,N,K)=APX(M,N,K)
      AMXX(M,N,K)=AMX(M,N,K)
  746 CONTINUE
      IMI=I
      JMI=J
      CALL MTDAG(AMXX,AXX,APXX,VH,3,LMAXM3)
      DO 748 K=3,LMAXM1
      KM2=K-2
      V(I,J,K)=VH(1,KM2)
      RW(I,J,K)=VH(2,KM2)
      DUDX(I,J,K)=VH(3,KM2)
  748 CONTINUE
      DUDX(I,J,2)=QI*DUDX(I,J,3)+GI*DUDX(I,J,4)-(2.*(1.-CE1)/(AP(3)*DT
     1))*RW(I,J,3)
      DUDX(I,J,LMAX)=QIN*DUDX(I,J,LMAXM1)+GIN*DUDX(I,J,LMAXM2)-(2.
     1*(1.-CE6)/(CP(LMAXM1)*DT))*RW(I,J,LMAXM1)
  665 CONTINUE
      DO 704 J=1,JMAX
      DO 704 I=1,IMAX
      WAV=WAVEXS(I)+WAVEYS(J)
      IF(WAV.GT.0.00001) GO TO 704
      DO 694 K=1,LMAXP1
      RW(I,J,K)=0.
  694 DUDX(I,J,K)=0.
  704 CONTINUE
C***** USE THE FACT THAT THE FLOW VARIABLES ARE REAL TO OBTAIN THE REMAI
C***** -NING FOURIER COEFFICIENTS.
      DO 627 K=2,LMAX
      DO 627 I=1,IMAX
      U(I,NHP1Y,K)=0.
      V(I,NHP1Y,K)=0.
      W(I,NHP1Y,K)=0.
      RU(I,NHP1Y,K)=0.
      RV(I,NHP1Y,K)=0.
      RW(I,NHP1Y,K)=0.
      G(I,NHP1Y,K)=0.
      DUDX(I,NHP1Y,K)=0.
  627 CONTINUE
      DO 629 K=2,LMAX
      DO 629 J=1,JMAX
      U(NHP1X,J,K)=0.
      V(NHP1X,J,K)=0.
      W(NHP1X,J,K)=0.
      RU(NHP1X,J,K)=0.
      RV(NHP1X,J,K)=0.
      RW(NHP1X,J,K)=0.
      G(NHP1X,J,K)=0.
      DUDX(NHP1X,J,K)=0.
  629 CONTINUE
      DO 550 K=2,LMAX
      DO 550 J=NHP2Y,JMAX
```

```
      JJ=JMAX-J+2
      DO 550 I=NHP2X,IMAX
      II=IMAX-I+2
      U(II,JJ,K)=U(I,J,K)
      V(II,JJ,K)=V(I,J,K)
      W(II,JJ,K)=W(I,J,K)
      G(II,JJ,K)=G(I,J,K)
      U(I,JJ,K)=U(II,J,K)
      V(I,JJ,K)=V(II,J,K)
      W(I,JJ,K)=W(II,J,K)
      G(I,JJ,K)=G(II,J,K)
      RU(II,JJ,K)=-RU(I,J,K)
      RV(II,JJ,K)=-RV(I,J,K)
      RW(II,JJ,K)=-RW(I,J,K)
      DUDX(II,JJ,K)=-DUDX(I,J,K)
      RU(I,JJ,K)=-RU(II,J,K)
      RV(I,JJ,K)=-RV(II,J,K)
      RW(I,JJ,K)=-RW(II,J,K)
      DUDX(I,JJ,K)=-DUDX(II,J,K)
  550 CONTINUE
      DO 560 K=2,LMAX
      DO 560 I=NHP2X,IMAX
      II=IMAX-I+2
      U(II,1,K)=U(I,1,K)
      V(II,1,K)=V(I,1,K)
      W(II,1,K)=W(I,1,K)
      G(II,1,K)=G(I,1,K)
      RU(II,1,K)=-RU(I,1,K)
      RV(II,1,K)=-RV(I,1,K)
      RW(II,1,K)=-RW(I,1,K)
      DUDX(II,1,K)=-DUDX(I,1,K)
  560 CONTINUE
      DO 570 K=2,LMAX
      DO 570 J=NHP2X,JMAX
      JJ=JMAX-J+2
      U(1,JJ,K)=U(1,J,K)
      V(1,JJ,K)=V(1,J,K)
      W(1,JJ,K)=W(1,J,K)
      G(1,JJ,K)=G(1,J,K)
      RU(1,JJ,K)=-RU(1,J,K)
      RV(1,JJ,K)=-RV(1,J,K)
      RW(1,JJ,K)=-RW(1,J,K)
      DUDX(1,JJ,K)=-DUDX(1,J,K)
  570 CONTINUE
C***** INVERSE TRANSFORM
      DO 695 K=3,LMAXM1
      CALL MOVLEV(U(1,1,K),FR(1,1),IJ)
      CALL MOVLEV(RU(1,1,K),FI(1,1),IJ)
      CALL FFTX(-1.0)
      CALL FFTY(-1.0,CC)
      CALL MOVLEV(FR(1,1),U(1,1,K),IJ)
      CALL MOVLEV(FI(1,1),RU(1,1,K),IJ)
      CALL MOVLEV(V(1,1,K),FR(1,1),IJ)
      CALL MOVLEV(RV(1,1,K),FI(1,1),IJ)
      CALL FFTX(-1.0)
      CALL FFTY(-1.0,CC)
      CALL MOVLEV(FR(1,1),V(1,1,K),IJ)
      CALL MOVLEV(FI(1,1),RV(1,1,K),IJ)
      CALL MOVLEV(W(1,1,K),FR(1,1),IJ)
      CALL MOVLEV(RW(1,1,K),FI(1,1),IJ)
      CALL FFTX(-1.0)
      CALL FFTY(-1.0,CC)
      CALL MOVLEV(FR(1,1),W(1,1,K),IJ)
      CALL MOVLEV(FI(1,1),RW(1,1,K),IJ)
  695 CONTINUE
      DO 702 K=1,LMAXP1
      DO 703 J=1,JMAX
      DO 703 I=1,IMAX
      FR(I,J)=G(I,J,K)
      FI(I,J)=DUDX(I,J,K)
```

```
  703 CONTINUE
      CALL FFTX(-1.0)
      CALL FFTY(-1.0,CC)
      DO 705 J=1,JMAX
      DO 705 I=1,IMAX
      G(I,J,K)=FR(I,J)
      DUDX(I,J,K)=FI(I,J)
  705 CONTINUE
  702 CONTINUE
C****** STORE DATA (RU,RV,AND RW) FOR NEXT TIME STEP
      CALL PARTIAL(1,G)
      DO 710 K=1,LMAXP1
      DO 710 J=1,JMAX
      DO 710 I=1,IMAX
      RU(I,J,K)=H1(I,J,K)+DUDX(I,J,K)
  710 CONTINUE
      CALL PARTIAL(2,G)
      DO 715 K=1,LMAXP1
      DO 715 J=1,JMAX
      DO 715 I=1,IMAX
      RV(I,J,K)=H2(I,J,K)+DUDX(I,J,K)
  715 CONTINUE
      CALL PARTIAL(3,G)
      DO 720 K=1,LMAXP1
      DO 720 J=1,JMAX
      DO 720 I=1,IMAX
      RW(I,J,K)=H3(I,J,K)+DUDX(I,J,K)
  720 CONTINUE
      CALL LTAVG
      LCONT=LCONT+1
      LLCONT=LCONT-20
      IF(LLCONT.NE.0) GO TO 450
      LCONT=0
      CALL LTPR
  450 CONTINUE
      TP=0.5
      C4=1.0
      C1=2.0
  200 FORMAT(1X,1P9E14.5)
      COF=1.5
      CALL EXTERN(3,1,R2,RR2)
      PRINT 400,TIME
      NHT=TEND/2
      IF(NTIME.EQ.NHT) CALL STAT
  400 FORMAT(3X,* TIME STEP=*,I3)
      IF(NTIME.NE.TEND) GO TO 300
      WRITE(9) U,V,W
      WRITE(9) UTSUM,U2SMT,V2SMT,W2SMT,STSUM,PUT,PVT,PWT,PUNST,PVNST,
     1PWNST,SGST,ETED,U2STT,V2STT,W2STT,TCONT,TSHGS,TSCNT
     2,PDUT,PDVT,PDWT,PDUNT,PDVNT,PDWNT
      CALL STAT
      CALL LTPR
      STOP
  300 CONTINUE
      STOP
      END
*DECK PARTIAL
      SUBROUTINE PARTIAL(M,U)
C***************************************************************************
C THIS SUBROUTINE COMPUTES THE PARTIAL DERIVATIVE OF U . M=1 CORRESPONDS  *
C  TO DERIVATIVE IN THE X-DIRECTION ,M=2 CORRESPONDS TO THE DERIVATIVE    *
C IN THE Y-DIRECTION ,AND M=3 CORRESPONDS TO THE DERIVATIVE IN THE Z-DIREC*
C***************************************************************************
      COMMON/IDENTN/CODE
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
*CALL A2
*CALL A9
*CALL C7
*CALL B9
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*CALL C3
*CALL A1
      LMAXP1=LMAX+1
      DO 20 J=1,JMAX
      DO 20 I=1,IMAX
      DUDX(I,J,1)=0.
      DUDX(I,J,LMAXP1)=0.
   20 CONTINUE
      IF (M.EQ.2) GO TO 30
      IF (M.EQ.3) GO TO 70
C****** DERIVATIVE IN THE X-DIRECTION
      DO 10 L=2,LMAX
      SIGN=1.0
      CALL MOVLEV(U(1,1,L),FR(1,1),IJ)
      CALL FFTX(SIGN)
      DO 15 J=1,JMAX
      DO 15 I=1,IMAX
      DUM=FI(I,J)
      FI(I,J)=WAVEX(I)*FR(I,J)
      FR(I,J)=-WAVEX(I)*DUM
   15 CONTINUE
      SIGN=-1.0
      CALL FFTX(SIGN)
      CALL MOVLEV(FR(1,1),DUDX(1,1,L),IJ)
   10 CONTINUE
      GO TO 300
   30 CONTINUE
C******DERIVATIVE IN THE Y-DIRECTION
      CC=1.0
      DO 35 L=2,LMAX
      SIGN=1.0
      CALL MOVLEV(U(1,1,L),FR(1,1),IJ)
      DO 32 J=1,JMAX
      DO 32 I=1,IMAX
      FI(I,J)=0.0
   32 CONTINUE
      CALL FFTY(SIGN,CC)
      DO 40 J=1,JMAX
      DO 40 I=1,IMAX
      DUM=FI(I,J)
      FI(I,J)=WAVEY(J)*FR(I,J)
      FR(I,J)=-WAVEY(J)*DUM
   40 CONTINUE
      SIGN=-1.0
      CALL FFTY(SIGN,CC)
      CALL MOVLEV(FR(1,1),DUDX(1,1,L),IJ)
   35 CONTINUE
      GO TO 300
   70 CONTINUE
C*****FIRST DERIVATIVE IN THE Z-DIRECTION
      DO 82 J=1,JMAX
      DO 82 I=1,IMAX
      DO 82 K=2,LMAX
      KP1=K+1
      KM1=K-1
      DUDX(I,J,K)=AP(K)*U(I,J,KM1)+CP(K)*U(I,J,KP1)
   82 CONTINUE
   90 CONTINUE
  300 CONTINUE
      RETURN
      END
*DECK FFT
          IDENT  FFT  (A,B,N,ISN)                                      FFT2C  2
          ENTRY  FFT                                                   FFT2C  3
*  RADIX 2 COMPLEX FAST FOURIER TRANSFORM, COMPUTED IN PLACE.          FFT2C  4
*  SEE @ON COMPUTING THE FAST FOURIER TRANSFORM,@ R. SINGLETON,        FFT2C  5
*    COMM. ACM, V.10, N.10, PP.647-654, OCT. 1967.                     FFT2C  6
*  ARRAY A CONTAINS THE REAL COMPONENT OF THE DATA AND RESULT,         FFT2C  7
*    ARRAY B CONTAINS THE IMAGINARY COMPONENT.                         FFT2C  8
*  N, THE NUMBER OF COMPLEX DATA VALUES,                               FFT2C  9
```

```
*    MUST BE A POWER OF 2 AND GREATER THAN 1                               FFT2C 10
*  THE SIGN OF ISN IS THE SIGN OF THE EXPONENTIAL IN THE TRANSFORM.        FFT2C 11
*  THE MAGNITUDE OF ISN IS THE INCREMENT SIZE FOR INDEXING                 FFT2C 12
*    A AND B, AND IS ONE IN THE USUAL CASE.                                FFT2C 13
*  DATA MAY ALTERNATIVELY BE STORED FORTRAN COMPLEX                        FFT2C 14
*    IN A SINGLE ARRAY, IN WHICH CASE THE MAGNITUDE                        FFT2C 15
*    OF ISN IS TWO AND ADDRESS B IS A(2), I.E.                             FFT2C 16
*        CALL FFT2(A,A(2),N,2)                                             FFT2C 17
*    INSTEAD OF                                                            FFT2C 18
*        CALL FFT2(A,B,N,1)                                                FFT2C 19
*  PROGRAM CONTAINS SINE TABLE FOR MAXIMUM N OF 32768                      FFT2C 20
*  6400 TIME FOR N=1024, 220 M.SEC.                                        FFT2C 21
*  6400 TIME FOR N=2**M IS 21.5*N*M MICRO-SEC.                             FFT2C 22
*  6600 TIME FOR N=1024, 44 M.SEC.                                         FFT2C 23
*  6600 TIME FOR N=2**M IS 4.3*N*M MICRO-SEC.                              FFT2C 24
*  RMS ERROR FOR TRANSFORM-INVERSE IS LESS THAN 1.3E-13                    FFT2C 25
*    FOR N=32768 OR SMALLER.                                               FFT2C 26
*  FORTRAN 2.3 SUBROUTINE                                                  FFT2C 27
*  BY R. C. SINGLETON, STANFORD RESEARCH INSTITUTE, NOV. 1968              FFT2C 28
 L100     SX0    B3               NN                                       FFT2C 29
          SB4    B0               KK=0                                     FFT2C 30
          SB3    B3-B7            NN=NN-INC                                FFT2C 31
          AX0    1                KSPAN=NN/2                               FFT2C 32
          SB5    B0               K2=0                                     FFT2C 33
          SB6    X0                                                        FFT2C 34
          SX1    B5               K2=K2                                    FFT2C 35
          EQ     B6,B7,FFT        IF(KSPAN .EQ. INC) RETURN                FFT2C 36
 L110     SB4    B3-B4            KK=NN-KK                                 FFT2C 37
          SB5    B3-B5            K2=NN-K2                                 FFT2C 38
          SA2    B1+B4            EXCHANGE A(KK),A(K2) AND B(KK),B(K2)     FFT2C 39
          SA3    B1+B5                                                     FFT2C 40
          SA4    B2+B4                                                     FFT2C 41
          NX7    X2                                                        FFT2C 42
          SA5    B2+B5                                                     FFT2C 43
          NX6    X3                                                        FFT2C 44
          SA7    A3                                                        FFT2C 45
          SA6    A2                                                        FFT2C 46
          NX7    X4                                                        FFT2C 47
          NX6    X5                                                        FFT2C 48
          SA7    A5                                                        FFT2C 49
          SA6    A4               END OF EXCHANGE                          FFT2C 50
          LT     B6,B4,L110       IF(KSPAN .LT. KK) GO TO L110             FFT2C 51
 L120     SB4    B4+B7            KK=KK+INC                                FFT2C 52
          SB5    B6+B5            K2=KSPAN+K2                              FFT2C 53
          SA2    B1+B4            EXCHANGE A(KK),A(K2) AND B(KK),B(K2)     FFT2C 54
          SA3    B1+B5                                                     FFT2C 55
          SA4    B2+B4                                                     FFT2C 56
          NX7    X2                                                        FFT2C 57
          SA5    B2+B5                                                     FFT2C 58
          NX6    X3                                                        FFT2C 59
          SA7    A3                                                        FFT2C 60
          SA6    A2                                                        FFT2C 61
          NX7    X4                                                        FFT2C 62
          SX0    B6               K=KSPAN                                  FFT2C 63
          NX6    X5                                                        FFT2C 64
          SA7    A5                                                        FFT2C 65
          SA6    A4               END OF EXCHANGE                          FFT2C 66
 L130     AX0    1                K=K/2                                    FFT2C 67
          IX1    X1-X0            K2=K2-K                                  FFT2C 68
          PL     X1,L130          IF(K2 .GE. 0) GO TO L130                 FFT2C 69
          LX0    1                K=K+K                                    FFT2C 70
          SB4    B4+B7            KK=KK+INC                                FFT2C 71
          IX1    X1+X0            K2=K2+K                                  FFT2C 72
          SB5    X1               K2=K2                                    FFT2C 73
          GE     B5,B4,L110       IF(K2 .GE. KK) GO TO L110                FFT2C 74
          LT     B4,B6,L120       IF(KK .LT. KSPAN) GO TO L120             FFT2C 75
 FFT                                                                       FFT2C 76
         SB1    X1                                                         INSR1  1
         SA1    A1+1                                                       INSR1  2
         SB2    X1                                                         INSR1  3
```

```
         SA1     A1+1                                                INSR1   4
         SB3     X1                                                  INSR1   5
         SA1     A1+1                                                INSR1   6
         SB4     X1                                                  INSR1   7
         SA4     B4                 ISN                              FFT2C  77
         MX2     1                  MASK                             FFT2C  78
         SA5     L60                                                 FFT2C  79
         SA3     B3                 N                                FFT2C  80
         LX2     57                                                  FFT2C  81
         PX7     X3                                                  FFT2C  82
         BX6     -X2*X5                                              FFT2C  83
         PL      X4,L10             IF(ISN .GE. 0) GO TO L10         FFT2C  84
         BX6     X2+X5                                               FFT2C  85
         BX4     -X4                INC=-INC                         FFT2C  86
L10      LX3     32                                                  FFT2C  87
         SA6     A5                                                  FFT2C  88
         NX0     B5,X3                                               FFT2C  89
         PX2     X4                                                  FFT2C  90
         SB7     X4                                                  FFT2C  91
         DX7     X2*X7                                               FFT2C  92
         SA1     B5+S               S(M)                             FFT2C  93
         SB3     X7                 NN=INC*N                         FFT2C  94
         SB6     X7                 KSPAN=NN                         FFT2C  95
         EQ      L40                GO TO L40                        FFT2C  96
L20      SA3     CD                                                  FFT2C  97
         RX4     X2*X1              SD*CN                            FFT2C  98
         RX7     X2*X0              SD*SN                            FFT2C  99
         RX5     X3*X0              CD*SN                            FFT2C100
         RX6     X3*X1              CD*CN                            FFT2C101
         RX4     X4-X5                                               FFT2C102
         RX6     X6+X7                                               FFT2C103
         NX5     X4                                                  FFT2C104
         RX7     X1-X6                                               FFT2C105
         RX0     X0+X5                                               FFT2C106
         NX1     X7                                                  FFT2C107
L30      SB5     B6+B4              K2=KSPAN+KK                      FFT2C108
         SA2     B1+B4              A(KK)                            FFT2C109
         SA3     B1+B5              A(K2)                            FFT2C110
         SA4     B2+B4              B(KK)                            FFT2C111
         RX6     X2+X3                                               FFT2C112
         SA5     B2+B5              B(K2)                            FFT2C113
         RX2     X2-X3              RE                               FFT2C114
         SA6     A2                 A(KK)                            FFT2C115
         RX7     X4+X5                                               FFT2C116
         RX3     X1*X2              CN*RE                            FFT2C117
         RX4     X4-X5              IM                               FFT2C118
         SA7     A4                 B(KK)                            FFT2C119
         RX5     X0*X4              SN*IM                            FFT2C120
         RX2     X0*X2              SN*RE                            FFT2C121
         RX6     X3-X5                                               FFT2C122
         RX4     X1*X4              CN*IM                            FFT2C123
         SA6     A3                 A(K2)                            FFT2C124
         RX7     X2+X4                                               FFT2C125
         SB4     B6+B5              KK=KSPAN+K2                      FFT2C126
         SA7     A5                 B(K2)                            FFT2C127
         LT      B4,B3,L30          IF(KK .LT. NN) GO TO L30         FFT2C128
         SB5     B4-B3              K2=KK-NN                         FFT2C129
         BX1     -X1                CN=-CN                           FFT2C130
         SB4     B6-B5              KK=KSPAN-K2                      FFT2C131
         LT      B5,B4,L30          IF(K2 .LT. KK) GO TO L30         FFT2C132
         SB4     B4+B7              KK=KK+INC                        FFT2C133
         SA2     SD                                                  FFT2C134
         LT      B4,B5,L20          IF(KK .LT. K2) GO TO L20         FFT2C135
L40      SB4     B0                 KK=0                             FFT2C136
         SX5     B6                                                  FFT2C137
         AX5     1                  KSPAN=KSPAN/2                    FFT2C138
         SB6     X5                                                  FFT2C139
L50      SB5     B6+B4              K2=KSPAN+KK                      FFT2C140
         SA2     B1+B4              A(KK)                            FFT2C141
         SA3     B1+B5              A(K2)                            FFT2C142
```

```
          SA4    B2+B4              B(KK)                             FFT2C143
          RX6    X2+X3                                                FFT2C144
          SA5    B2+B5              B(K2)                             FFT2C145
          RX7    X2-X3                                                FFT2C146
          SA6    A2                 A(KK)                             FFT2C147
          SA7    A3                 A(K2)                             FFT2C148
          RX6    X4+X5                                                FFT2C149
          SB4    B6+B5              KK=KSPAN+K2                       FFT2C150
          RX7    X4-X5                                                FFT2C151
          SA6    A4                 B(KK)                             FFT2C152
          SA7    A5                 B(K2)                             FFT2C153
          LT     B4,B3,L50          IF(KK .LT. NN) GO TO L50          FFT2C154
          EQ     B6,B7,L100         IF(KSPAN .EQ. INC) GO TO L100     FFT2C155
          SA1    A1                 S(M)                              FFT2C156
          SB4    B7                 KK=INC                            FFT2C157
          RX6    X1*X1                                                FFT2C158
          SA1    A1+1               M=M+1, S(M)                       FFT2C159
          FX6    X6+X6                                                FFT2C160
          SA3    ONE                                                  FFT2C161
          SA6    CD                 CD=2*S(M)**2                      FFT2C162
 L60      BX0    X1                 SN=SD                             FFT2C163
          RX6    X3-X6              CN=1.0-CD                         FFT2C164
          BX7    X0                                                   FFT2C165
          NX1    X6                                                   FFT2C166
          SA7    SD                                                   FFT2C167
          EQ     L30                GO TO L30                         FFT2C168
 S        DATA   9.5873799095977346E-5                                FFT2C169
          DATA   1.9174759731070331E-4                                FFT2C170
          DATA   3.8349518757139559E-4                                FFT2C171
          DATA   7.6699031874270453E-4                                FFT2C172
          DATA   1.5339801862847656E-3                                FFT2C173
          DATA   3.0679567629659763E-3                                FFT2C174
          DATA   6.1358846491544754E-3                                FFT2C175
          DATA   1.2271538285719926E-2                                FFT2C176
          DATA   2.4541228522912288E-2                                FFT2C177
          DATA   4.9067674327418014E-2                                FFT2C178
          DATA   9.8017140329560602E-2                                FFT2C179
          DATA   1.9509032201612827E-1                                FFT2C180
          DATA   3.8268343236508977E-1                                FFT2C181
          DATA   0.7071067811865475                                   FFT2C182
 ONE      DATA   1.0                                                  FFT2C183
 CD                                                                   FFT2C184
 SD                                                                   FFT2C185
          END                                                         FFT2C186
*DECK FFTX
      SUBROUTINE FFTX(SIGN)
C*********************************************************************
C FAST FOURIER TRANSFORM IN X-DIRECTION                               *
C*********************************************************************
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
*CALL A1
*CALL C10
      ISN=-SIGN
      IF (SIGN .LT. 0.) GO TO 3
      DO 2 J=1,JMAX
      DO 1 I=1,IMAX
      FI(I,J)=0.
    1 CONTINUE
    2 CONTINUE
    3 CONTINUE
      DO 100 J=1,JMAX
      DO 110 I=1,IMAX
      XR(I)=FR(I,J)
      XI(I)=FI(I,J)
  110 CONTINUE
      CALL FFT(XR,XI,IMAX,ISN)
      DO 120 I=1,IMAX
      FR(I,J)=XR(I)
      FI(I,J)=XI(I)
  120 CONTINUE
```

```
  100 CONTINUE
      RETURN
      END
*DECK FFTY
      SUBROUTINE FFTY(SIGN,COEF3)
C***********************************************************************
C FAST FOURIER TRANSFORM IN Y-DIRECTION                                *
C***********************************************************************
*CALL A1
*CALL C10
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      ISN=-SIGN
C     Y-TRANSFORM
      DO 100 I=1,IMAX
      DO 110 J=1,JMAX
      XR(J)=FR(I,J)
      XI(J)=FI(I,J)
  110 CONTINUE
      CALL FFT(XR,XI,JMAX,ISN)
      IF(SIGN.LT.0.) GO TO 200
      DO 120 J=1,JMAX
      FR(I,J)=XR(J)
      FI(I,J)=XI(J)
  120 CONTINUE
      GO TO 100
  200 DO 130 J=1,JMAX
      FR(I,J)=XR(J)*COEF3
      FI(I,J)=XI(J)*COEF3
  130 CONTINUE
  100 CONTINUE
      RETURN
      END
*DECK INITIAL
      SUBROUTINE INITIAL
C***********************************************************************
C* THIS SUBROUTINE COMPUTES THE VARIOUS NECESSARY ARRAYS AND CONSTANTS *
C*FOR SGS,PARTIAL,POISON,AND FILTER SUBROUTINES                        *
C***********************************************************************
*CALL A1
      COMMON/ADV/NTIME
*CALL B1
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/SCM2/LMAXP1,D1,D2,D3,D4,D5,D6
      COMMON/SCM3/DELTA1,DELTA2,RE,E
      COMMON/SCM4/CI,CJ,CK,CJK,CIK,CIJ
*CALL C3
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
      REAL NAVG
      C=0.4
      S=2./3.
      PAI=ACOS(-1.)
C***** DELTA1 AND DELTA2 ARE THE MESH SIZES IN X AND Y DIRECTIONS
      DELTA1=PAI/8.
      DELTA2=PAI/12.
      IMAX=16
      JMAX=16
      LMAX=64
      IJ=IMAX*JMAX
      LMAXP1=LMAX+1
      IJK=IMAX*JMAX*LMAXP1
      CI=1./IMAX
      CJ=1./JMAX
      CK=1./LMAXP1
      CJK=1./(JMAX*LMAXP1)
      CIK=1./(IMAX*LMAXP1)
      CIJ=1./(IMAX*JMAX)
      RE=640.25
      E=1./RE
      NHALFX=IMAX/2
      NHALFY=JMAX/2
```

```
      NHP1X=NHALFX+1
      NHP1Y=NHALFY+1
      C100=2.0*PAI/(IMAX*DELTA1)
       C10=2.0*PAI/(JMAX*DELTA2)
      C101=C100/IMAX
      C11=C10/JMAX
C***** DEFINE WAVE NUMBERS.
C***** NOTE THAT WAVEX AND WAVEY ARE SMALLER THAN THE ACTUAL WAVE NUMBERS
C***** BY FACTOR OF IMAX AND JMAX RESPECTIVELY.
      DO 100 I=1,IMAX
      MM=I/NHP1X
      M=MM*IMAX+1
      WAVEX(I)=C101*(I-M)
      WAVEXS(I)=(C100*(I-M))**2
  100 CONTINUE
      WAVEX(NHP1X)=0.
      WAVEXS(NHP1X)=0.
      DO 130 J=1,JMAX
      MM=J/NHP1Y
      M=MM*JMAX+1
      WAVEY(J)=C11*(J-M)
      WAVEYS(J)=(C10*(J-M))**2
  130 CONTINUE
      WAVEY(NHP1Y)=0.
      WAVEYS(NHP1Y)=0.
 1000 FORMAT(1P8E15.7)
      NAVG=2
      IF(NTIME.EQ.0) NAVG=6
      NHP2X=NHP1X+1
      NHP2Y=NHP1Y+1
C*****COMPUTE THE NORMALIZED FOURIER TRANSFORM OF THE FILTER FUNCTION IN X-DIREC
      DO 300 J=1,JMAX
      DO 300 I=1,NHP1X
      FR(I,J)=EXP(-6.*FLOAT(I-1)**2/(NAVG**2))
  300 CONTINUE
      DO 310 J=1,JMAX
      DO 310 I=NHP2X,IMAX
      II=IMAX-I+2
      FR(I,J)=FR(II,J)
  310 CONTINUE
C***** COMPUTE THE NORMALIZATION CONST,AREA.
      AREA=0.
      DO 320 I=1,IMAX
      AREA=AREA+FR(I,1)
  320 CONTINUE
      DO 330 J=1,JMAX
      DO 330 I=1,IMAX
      FR(I,J)=FR(I,J)/AREA
      FI(I,J)=0.
  330 CONTINUE
      CALL FFTX(1.0)
      DO 340 I=1,IMAX
      FILTX(I)=FR(I,1)
  340 CONTINUE
C*****COMPUTE THE NORMALIZED FOURIER TRANSFORM OF THE FILTER FUNCTION IN Y-DIREC
      DO 400 J=1,NHP1Y
      DO 400 I=1,IMAX
      FR(I,J)=EXP(-6.*FLOAT(J-1)**2/(NAVG**2))
  400 CONTINUE
      DO 410 J=NHP2Y,JMAX
      DO 410 I=1,IMAX
      JJ=JMAX-J+2
      FR(I,J)=FR(I,JJ)
  410 CONTINUE
      AREA=0.
      DO 420 J=1,JMAX
      AREA=AREA+FR(1,J)
  420 CONTINUE
      DO 430 J=1,JMAX
      DO 430 I=1,IMAX
```

```
      FR(I,J)=FR(I,J)/AREA
      FI(I,J)=0.
  430 CONTINUE
      CALL FFTY(1.0,1.0)
      DO 440 J=1,JMAX
      FILTY(J)=FR(1,J)
  440 CONTINUE
      FILTX(NHP1X)=0.
      FILTY(NHP1Y)=0.
      PRINT 1000,(WAVEX(L),L=1,IMAX)
      PRINT 1000,(WAVEY(L),L=1,JMAX)
      PRINT 1000,(WAVEXS(L),L=1,IMAX)
      PRINT 1000,(WAVEYS(L),L=1,JMAX)
      RETURN
      END
*DECK INICON
      SUBROUTINE INICON
C***********************************************************************
C*    THIS SUBROUTINE GENERATES THE INITIAL FIELD FOR THE COMPUTATION  *
C***********************************************************************
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      DIMENSION G(161),Y(161),F(65)
      COMMON/SCM3/DELTA1,DELTA2,RE,E
*CALL A8
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
*CALL A1
*CALL A13
      COMMON/SCM4/CI,CJ,CK,CJK,CIK,CIJ
*CALL A11
      EQUIVALENCE (U1,H1),(U2,H2),(U3,H3)
*CALL A2
*CALL A6
*CALL A7
*CALL A9
      PAI=ACOS(-1.)
      LMAXP1=LMAX+1
      LMAXM1=LMAX-1
      DO 210 J=1,JMAX
      DO 210 I=1,IMAX
      U1(I,J,2)=0.
      U2(I,J,2)=0.
      U3(I,J,2)=0.
      U1(I,J,1)=0.
      U2(I,J,1)=0.
      U3(I,J,1)=0.
      U1(I,J,LMAX)=0.
      U2(I,J,LMAX)=0.
      U3(I,J,LMAX)=0.
      U1(I,J,LMAXP1)=0.
      U2(I,J,LMAXP1)=0.
      U3(I,J,LMAXP1)=0.
      U(I,J,2)=0.
      V(I,J,2)=0.
      W(I,J,2)=0.
      U(I,J,1)=0.
      V(I,J,1)=0.
      W(I,J,1)=0.
      U(I,J,LMAX)=0.
      V(I,J,LMAX)=0.
      W(I,J,LMAX)=0.
      U(I,J,LMAXP1)=0.
      V(I,J,LMAXP1)=0.
      W(I,J,LMAXP1)=0.
  210 CONTINUE
C**********************************************************************
C*    THE VELOCITY FIELD FOR THE INITIATION OF THE PROGRAM IS OBTAINED *
C*    FROM THE DISK. THE ORIGINAL VELOCITY FIELD IS GENERATED FROM A   *
C*    SEPARATE PROGRAM (SEE SECTION 4.2 IN THE TEXT).                  *
C*    U1,U2,U3 ARE THE COMPONENTS OF THE VELOCITY FIELD AT TIME STEP N *
C*    RU,RV,AND RW ARE THE INFORMATION AT TIME STEP N-1,NECESSARY FOR  *
```

```
C*    ADAMS BASHFORTH METHOD.                                              *
C**************************************************************************
      READ(8) U1,U2,U3,RU,RV,RW
      DO 25 K=2,LMAX
      DO 25 J=1,JMAX
      DO 25 I=1,IMAX
      U(I,J,K)=U1(I,J,K)
      V(I,J,K)=U2(I,J,K)
      W(I,J,K)=U3(I,J,K)
   25 CONTINUE
      CALL EXTERN(3,1,R2,RR2)
      CALL EXTERN(31,33,RN,RRN)
      PRINT 2000
 1000 FORMAT(1P8E15.7)
 2000 FORMAT(1H1,* VELOCITY IN THE X-DIRECTION ACCROSS THE CHANNEL *)
      PRINT 1000,(U(10,10,K),K=1,LMAXP1)
      RETURN
      END
*DECK CURL
      SUBROUTINE CURL(U,V,W)
C***** THIS SUBROUTINE COMPUTES THE VORTICITY FIELD
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
*CALL A11
      EQUIVALENCE (U1,H1),(U2,H2),(U3,H3)
*CALL A12
*CALL A7
*CALL A2
      LMAXP1=LMAX+1
      CALL PARTIAL(2,W)
      CALL MOVLEV(DUDX(1,1,1),U1(1,1,1),IJK)
      CALL PARTIAL(3,V)
      DO 10 K=1,LMAXP1
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      U1(I,J,K)=U1(I,J,K)-DUDX(I,J,K)
   10 CONTINUE
      CALL PARTIAL(3,U)
      CALL MOVLEV(DUDX(1,1,1),U2(1,1,1),IJK)
      CALL PARTIAL(1,W)
      DO 15 K=1,LMAXP1
      DO 15 J=1,JMAX
      DO 15 I=1,IMAX
      U2(I,J,K)=U2(I,J,K)-DUDX(I,J,K)
   15 CONTINUE
      CALL PARTIAL(1,V)
      CALL MOVLEV(DUDX(1,1,1),U3(1,1,1),IJK)
      CALL PARTIAL(2,U)
      DO 20 K=1,LMAXP1
      DO 20 J=1,JMAX
      DO 20 I=1,IMAX
      U3(I,J,K)=U3(I,J,K)-DUDX(I,J,K)
   20 CONTINUE
      RETURN
      END
*DECK RHS
      SUBROUTINE RHS
C**************************************************************************
C* THIS SUBROUTINE COMPUTES THE RIGHT HAND SIDE OF THE GOVERNING          *
C*EQUATIONS,EXCLUDING THE PRESSURE.                                        *
C**************************************************************************
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
      COMMON/SCM2/LMAXP1,D1,D2,D3,D4,D5,D6
      COMMON/SCM3/DELTA1,DELTA2,RE,E
*CALL A2
*CALL A5
*CALL A6
*CALL A7
      CALL SGS
```

```
C*****MOMENTUM EQUATION IN THE X-DIRECTION
      CALL PARTIAL(1,V)
      DO 10 K=1,LMAXP1
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      G(I,J,K)=V(I,J,K)*DUDX(I,J,K)
   10 CONTINUE
C
      CALL PARTIAL(1,W)
      DO 20 K=1,LMAXP1
      DO 20 J=1,JMAX
      DO 20 I=1,IMAX
      G(I,J,K)=G(I,J,K)+W(I,J,K)*DUDX(I,J,K)
   20 CONTINUE
      CALL PARTIAL(2,U)
      DO 30 K=1,LMAXP1
      DO 30 J=1,JMAX
      DO 30 I=1,IMAX
      G(I,J,K)=G(I,J,K)-V(I,J,K)*DUDX(I,J,K)
   30 CONTINUE
      CALL PARTIAL(3,U)
      DO 40 K=1,LMAXP1
      DO 40 J=1,JMAX
      DO 40 I=1,IMAX
      G(I,J,K)=G(I,J,K)-W(I,J,K)*DUDX(I,J,K)
   40 CONTINUE
      CALL FILTER(G)
      DO 45 K=1,LMAXP1
      DO 45 J=1,JMAX
      DO 45 I=1,IMAX
      H1(I,J,K)=G(I,J,K)+H1(I,J,K)+1.
   45 CONTINUE
C*****COMPUTE THE VISCOUS TERMS IN THE X-MOMENTUM EQUATION
      CALL PARTIAL(1,U)
      CALL MOVLEV(DUDX(1,1,1),G(1,1,1),IJK)
      CALL PARTIAL(1,G)
      DO 50 K=1,LMAXP1
      DO 50 J=1,JMAX
      DO 50 I=1,IMAX
      H1(I,J,K)=H1(I,J,K)+E*DUDX(I,J,K)
   50 CONTINUE
      CALL PARTIAL(2,U)
      CALL MOVLEV(DUDX(1,1,1),G(1,1,1),IJK)
      CALL PARTIAL(2,G)
      DO 55 K=1,LMAXP1
      DO 55 J=1,JMAX
      DO 55 I=1,IMAX
      H1(I,J,K)=H1(I,J,K)+E*DUDX(I,J,K)
   55 CONTINUE
C*****MOMENTUM EQUATION IN THE Y-DIRECTION
      CALL PARTIAL(2,U)
      DO 65 K=1,LMAXP1
      DO 65 J=1,JMAX
      DO 65 I=1,IMAX
      G(I,J,K)=U(I,J,K)*DUDX(I,J,K)
   65 CONTINUE
      CALL PARTIAL(2,W)
      DO 70 K=1,LMAXP1
      DO 70 J=1,JMAX
      DO 70 I=1,IMAX
      G(I,J,K)=G(I,J,K)+W(I,J,K)*DUDX(I,J,K)
   70 CONTINUE
      CALL PARTIAL(3,V)
      DO 75 K=1,LMAXP1
      DO 75 J=1,JMAX
      DO 75 I=1,IMAX
      G(I,J,K)=G(I,J,K)-W(I,J,K)*DUDX(I,J,K)
   75 CONTINUE
      CALL PARTIAL(1,V)
      DO 80 K=1,LMAXP1
```

```
      DO 80 J=1,JMAX
      DO 80 I=1,IMAX
      G(I,J,K)=G(I,J,K)-U(I,J,K)*DUDX(I,J,K)
   80 CONTINUE
      CALL FILTER(G)
      DO 85 K=1,LMAXP1
      DO 85 J=1,JMAX
      DO 85 I=1,IMAX
      H2(I,J,K)=H2(I,J,K)+G(I,J,K)
   85 CONTINUE
C*****COMPUTE THE VISCOUS TERMS IN THE Y-MOMENTUM EQUATION
      CALL PARTIAL(1,V)
      CALL MOVLEV(DUDX(1,1,1),G(1,1,1),IJK)
      CALL PARTIAL(1,G)
      DO 90 K=1,LMAXP1
      DO 90 J=1,JMAX
      DO 90 I=1,IMAX
      H2(I,J,K)=H2(I,J,K)+E*DUDX(I,J,K)
   90 CONTINUE
      CALL PARTIAL(2,V)
      CALL MOVLEV(DUDX(1,1,1),G(1,1,1),IJK)
      CALL PARTIAL(2,G)
      DO 95 K=1,LMAXP1
      DO 95 J=1,JMAX
      DO 95 I=1,IMAX
      H2(I,J,K)=H2(I,J,K)+E*DUDX(I,J,K)
   95 CONTINUE
C*****MOMENTUM EQUATION IN THE Z-DIRECTION
      CALL PARTIAL(3,V)
      DO 105 K=1,LMAXP1
      DO 105 J=1,JMAX
      DO 105 I=1,IMAX
      G(I,J,K)=V(I,J,K)*DUDX(I,J,K)
  105 CONTINUE
      CALL PARTIAL(3,U)
      DO 110 K=1,LMAXP1
      DO 110 J=1,JMAX
      DO 110 I=1,IMAX
      G(I,J,K)=G(I,J,K)+U(I,J,K)*DUDX(I,J,K)
  110 CONTINUE
      CALL PARTIAL(2,W)
      DO 115 K=1,LMAXP1
      DO 115 J=1,JMAX
      DO 115 I=1,IMAX
      G(I,J,K)=G(I,J,K)-V(I,J,K)*DUDX(I,J,K)
  115 CONTINUE
      CALL PARTIAL(1,W)
      DO 120 K=1,LMAXP1
      DO 120 J=1,JMAX
      DO 120 I=1,IMAX
      G(I,J,K)=G(I,J,K)-U(I,J,K)*DUDX(I,J,K)
  120 CONTINUE
      CALL FILTER(G)
      DO 125 K=1,LMAXP1
      DO 125 J=1,JMAX
      DO 125 I=1,IMAX
      H3(I,J,K)=H3(I,J,K)+G(I,J,K)
  125 CONTINUE
C*****COMPUTE THE VISCOUS TERMS IN THE Z-MOMENTUM EQUATION
      CALL PARTIAL(1,W)
      CALL MOVLEV(DUDX(1,1,1),G(1,1,1),IJK)
      CALL PARTIAL(1,G)
      DO 130 K=1,LMAXP1
      DO 130 J=1,JMAX
      DO 130 I=1,IMAX
      H3(I,J,K)=H3(I,J,K)+E*DUDX(I,J,K)
  130 CONTINUE
      CALL PARTIAL(2,W)
      CALL MOVLEV(DUDX(1,1,1),G(1,1,1),IJK)
      CALL PARTIAL(2,G)
```

```
      DO 135 K=1,LMAXP1
      DO 135 J=1,JMAX
      DO 135 I=1,IMAX
      H3(I,J,K)=H3(I,J,K)+E*DUDX(I,J,K)
  135 CONTINUE
      RETURN
      END
*DECK SGS
      SUBROUTINE SGS
C***********************************************************************
C*THIS SUBROUTINE COMPUTES THE EDDY VISCOSITY AND THE SUBGRID SCALE    *
C*TERMS WHICH ARE ADDED TO THE RIGHT HAND SIDE OF THE GOVERNING MOMEN  *
C*-TUM EQUATIONS.THE EDDY VISCOSITY IS SET EQUAL TO ZERO AT THE WALL.  *
C***********************************************************************
      COMMON/ADV/NTIME
      COMMON/SGTT/SGST(65),ETED(65),U2STT(65),V2STT(65),W2STT(65)
     1,TSHGS,TSCNT
      COMMON/TINC/DT
      REAL MIU
      COMMON/COUNT/IICONT
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/SCM2/LMAXP1,D1,D2,D3,D4,D5,D6
      COMMON/INNERC/CVINR(65)
      DIMENSION EDVO(65),EDVI(65)
*CALL A2
*CALL A9
*CALL B2
*CALL B3
*CALL B4
*CALL B5
*CALL A4
*CALL A7
*CALL A6
*CALL A5
      LMAXM1=LMAX-1
      IF(NTIME.NE.1) GO TO 5
      TSCNT=0.
      TSHGS=0.
      DO 2 K=1,LMAXP1
      SGST(K)=0.
      ETED(K)=0.
      U2STT(K)=0.
      V2STT(K)=0.
      W2STT(K)=0.
    2 CONTINUE
    5 CONTINUE
      LHP1=LMAX/2+1
C****** FIRST COMPUTE THE EDDY VISCOSITY,G.
      CALL PARTIAL(1,U)
      DO 10 K=3,LMAXM1
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      G(I,J,K)=DUDX(I,J,K)**2
   10 CONTINUE
      CALL PARTIAL(2,V)
      DO 15 K=3,LMAXM1
      DO 15 J=1,JMAX
      DO 15 I=1,IMAX
      G(I,J,K)=G(I,J,K)+DUDX(I,J,K)**2
   15 CONTINUE
      CALL PARTIAL(3,W)
      DO 20 K=3,LMAXM1
      DO 20 J=1,JMAX
      DO 20 I=1,IMAX
      G(I,J,K)=G(I,J,K)+DUDX(I,J,K)**2
   20 CONTINUE
      CALL PARTIAL(2,U)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(1,V)
```

ORIGINAL PAGE IS
OF POOR QUA

```
      DO 25 K=3,LMAXM1
      DO 25 J=1,JMAX
      DO 25 I=1,IMAX
      G(I,J,K)=2.*G(I,J,K)+(DUDX(I,J,K)+P(I,J,K))**2
   25 CONTINUE
      CALL PARTIAL(2,W)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(3,V)
      DO 30 K=3,LMAXM1
      DO 30 J=1,JMAX
      DO 30 I=1,IMAX
      G(I,J,K)=G(I,J,K)+(DUDX(I,J,K)+P(I,J,K))**2
   30 CONTINUE
      CALL PARTIAL(1,W)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(3,U)
      BMAX=0.
      DO 35 K=3,LMAXM1
      DO 35 J=1,JMAX
      DO 35 I=1,IMAX
      CCC=G(I,J,K)+(DUDX(I,J,K)+P(I,J,K))**2
      H2(I,J,K)=CV(K)*SQRT(CCC)
      H1(I,J,K)=CVINR(K)*CCC
   35 CONTINUE
      CC=1./(IMAX*JMAX)
C***** COMPUTE THE PLANAR AVERAGE OF INNER AND OUTER LAYER MODELS.
      DO 900 K=3,LMAXM1
      EDVO(K)=0.
      EDVI(K)=0.
      DO 910 J=1,JMAX
      DO 910 I=1,IMAX
      EDVO(K)=EDVO(K)+H2(I,J,K)
      EDVI(K)=EDVI(K)+H1(I,J,K)
  910 CONTINUE
      EDVO(K)=EDVO(K)*CC
      EDVI(K)=EDVI(K)*CC
  900 CONTINUE
      CR=1.0
      MMM=0
      DO 915 K=3,LHP1
      IF(EDVI(K).GT.EDVO(K)) MMM=2
      IF(MMM.EQ.2) GO TO 915
      IF(EDVI(K).LT.EDVO(K)) KCROS1=K
  915 CONTINUE
      MMM=0
      DO 920 K=LHP1,LMAXM1
      KK=LMAXM1-K+LHP1
      IF(EDVI(KK).GT.EDVO(KK)) MMM=2
      IF(MMM.EQ.2) GO TO 920
      IF(EDVI(KK).LT.EDVO(KK)) KCROS2=KK
  920 CONTINUE
      PRINT 925,KCROS1,KCROS2
  925 FORMAT(5X,* CROSS OVER POINTS OF INNER AND OUTER LAYER*,2I5)
      PRINT 930
  930 FORMAT(/,20X,* PLANE AVERAGE OF INNER LAYER MODEL *)
      PRINT 200,(EDVI(K),K=3,LMAXM1)
      PRINT 935
  935 FORMAT(/,20X,* PLANE AVERAGE OF OUTER LAYER MODEL *)
      PRINT 200,(EDVO(K),K=3,LMAXM1)
      DO 940 K=3,KCROS1
      DO 940 J=1,JMAX
      DO 940 I=1,IMAX
      G(I,J,K)=H1(I,J,K)*CR
  940 CONTINUE
      KCROS3=KCROS1+1
      KCROS4=KCROS2-1
      DO 945 K=KCROS3,KCROS4
      DO 945 J=1,JMAX
      DO 945 I=1,IMAX
      G(I,J,K)=H2(I,J,K)
```

```
  945 CONTINUE
      DO 950 K=KCROS2,LMAXM1
      DO 950 J=1,JMAX
      DO 950 I=1,IMAX
      G(I,J,K)=H1(I,J,K)*CR
  950 CONTINUE
      DO 40 J=1,JMAX
      DO 40 I=1,IMAX
      G(I,J,1)=0.
      G(I,J,2)=0.
      G(I,J,LMAX)=0.
      G(I,J,LMAXP1)=0.
   40 CONTINUE
  200 FORMAT(1X,1P9E14.5)
C****** COMPUTE THE AVERAGE OF EDDY VISCOSITY IN X-Y PLANES
      DO 600 K=1,LMAXP1
      MIU(K)=0.
      DO 610 J=1,JMAX
      DO 610 I=1,IMAX
      MIU(K)=MIU(K)+G(I,J,K)
  610 CONTINUE
      MIU(K)=MIU(K)/(IMAX*JMAX)
  600 CONTINUE
      PRINT 190
  190 FORMAT(10X,* AVERAGE EDDY VISCOSITY *)
      PRINT 200,(MIU(K),K=2,LMAX)
C***** COMPUTE THE VISCOUS INSTABILITY CRITERION.
      BMAX=0.
      DO 400 K=3,LHP1
      KM1=K-1
      DO 400 J=1,JMAX
      DO 400 I=1,IMAX
      VIS=((Z(K)-Z(KM1))**2)/(ABS(G(I,J,K)-MIU(K)))
      VIS=DT/VIS
      IF(VIS.LT.BMAX) GO TO 400
      BMAX=VIS
      IDUM2=I
      JDUM2=J
      KDUM2=K
  400 CONTINUE
      DMAX=0.
      DO 500 K=LHP1,LMAXM1
      KP1=K+1
      DO 500 J=1,JMAX
      DO 500 I=1,IMAX
      VIS=((Z(KP1)-Z(K))**2)/(ABS(G(I,J,K)-MIU(K)))
      VIS=DT/VIS
      IF(VIS.LT.DMAX) GO TO 500
      DMAX=VIS
      IDUM1=I
      JDUM1=J
      KDUM1=K
  500 CONTINUE
      PRINT 510,BMAX,IDUM1,JDUM1,KDUM1,DMAX,IDUM2,JDUM2,KDUM2
  510 FORMAT(1X,* VIS INSTABILITY *,1P1E14.5,3I5,5X,1P1E14.5,3I5)
C*****EDDY VISCOSITY IS COMPUTED,NOW COMPUTE THE SUBGRID SCALE TERMS
      CALL PARTIAL(1,U)
      DO 60 K=1,LMAXP1
      DO 60 J=1,JMAX
      DO 60 I=1,IMAX
      P(I,J,K)=2.*G(I,J,K)*DUDX(I,J,K)
   60 CONTINUE
      CALL PARTIAL(1,P)
      DO 62 K=1,LMAXP1
      DO 62 J=1,JMAX
      DO 62 I=1,IMAX
      H1(I,J,K)=DUDX(I,J,K)
   62 CONTINUE
      CALL PARTIAL(2,U)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
```

```
      CALL PARTIAL(1,V)
      DO 64 K=1,LMAXP1
      DO 64 J=1,JMAX
      DO 64 I=1,IMAX
      P(I,J,K)=G(I,J,K)*(P(I,J,K)+DUDX(I,J,K))
   64 CONTINUE
      CALL PARTIAL(2,P)
      DO 66 K=1,LMAXP1
      DO 66 J=1,JMAX
      DO 66 I=1,IMAX
      H1(I,J,K)=H1(I,J,K)+DUDX(I,J,K)
   66 CONTINUE
      CALL PARTIAL(3,U)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(1,W)
      DO 68 K=1,LMAXP1
      DO 68 J=1,JMAX
      DO 68 I=1,IMAX
      P(I,J,K)=P(I,J,K)+DUDX(I,J,K)
   68 CONTINUE
C****** CALCULATE SGS CONTRIBUTIONS TO REYNOLDS STRESS AND INTENSITIES.
C****** ALSO AVERAGE THEM IN TIME.
      TSHGS=TSHGS+1
      DO 92 K=1,LMAXP1
      SSUM(K)=0.
      DO 94 J=1,JMAX
      DO 94 I=1,IMAX
      SSUM(K)=SSUM(K)+P(I,J,K)*G(I,J,K)
   94 CONTINUE
      SSUM(K)=-SSUM(K)/(IMAX*JMAX)
      SGST(K)=SGST(K)+SSUM(K)
   92 CONTINUE
      IF(NTIME.EQ.1) GO TO 360
      IF(IICONT.NE.0) GO TO 350
  360 CONTINUE
      DO 98 K=1,LMAXP1
      EDYVI(K)=0.
      DO 102 J=1,JMAX
      DO 102 I=1,IMAX
      EDYVI(K)=EDYVI(K)+G(I,J,K)**2
  102 CONTINUE
      EDYVI(K)=EDYVI(K)*FACTOR(K)/(IMAX*JMAX)
   98 CONTINUE
      CALL PARTIAL(1,U)
      DO 104 K=1,LMAXP1
      U2S(K)=0.
      DO 106 J=1,JMAX
      DO 106 I=1,IMAX
      U2S(K)=U2S(K)+G(I,J,K)*DUDX(I,J,K)
  106 CONTINUE
      U2S(K)=U2S(K)*2./(IMAX*JMAX)
      U2S(K)=EDYVI(K)-U2S(K)
  104 CONTINUE
      CALL PARTIAL(2,V)
      DO 108 K=1,LMAXP1
      V2S(K)=0.
      DO 110 J=1,JMAX
      DO 110 I=1,IMAX
      V2S(K)=V2S(K)+G(I,J,K)*DUDX(I,J,K)
  110 CONTINUE
      V2S(K)=V2S(K)*2./(IMAX*JMAX)
      V2S(K)=EDYVI(K)-V2S(K)
  108 CONTINUE
      CALL PARTIAL(3,W)
      DO 112 K=1,LMAXP1
      W2S(K)=0.
      DO 114 J=1,JMAX
      DO 114 I=1,IMAX
      W2S(K)=W2S(K)+G(I,J,K)*DUDX(I,J,K)
  114 CONTINUE
```

```
      W2S(K)=W2S(K)*2./(IMAX*JMAX)
      W2S(K)=EDYVI(K)-W2S(K)
  112 CONTINUE
      TSCNT=TSCNT+1
      DO 220 K=3,LMAXM1
      ETED(K)=ETED(K)+EDYVI(K)
      U2STT(K)=U2STT(K)+U2S(K)
      V2STT(K)=V2STT(K)+V2S(K)
      W2STT(K)=W2STT(K)+W2S(K)
  220 CONTINUE
  350 CONTINUE
      CALL PARTIAL(3,P)
      DO 70 K=1,LMAXP1
      DO 70 J=1,JMAX
      DO 70 I=1,IMAX
      H1(I,J,K)=H1(I,J,K)+(G(I,J,K)-MIU(K))*DUDX(I,J,K)
   70 CONTINUE
      CALL PARTIAL(3,G)
      DO 71 K=1,LMAXP1
      DO 71 J=1,JMAX
      DO 71 I=1,IMAX
      H1(I,J,K)=H1(I,J,K)+DUDX(I,J,K)*P(I,J,K)
   71 CONTINUE
      CALL PARTIAL(1,W)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(3,P)
      DO 715 K=1,LMAXP1
      DO 715 J=1,JMAX
      DO 715 I=1,IMAX
      H1(I,J,K)=H1(I,J,K)+MIU(K)*DUDX(I,J,K)
  715 CONTINUE
C***** Y-MOMENTUM EQUATION.
      CALL PARTIAL(1,V)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(2,U)
      DO 72 K=1,LMAXP1
      DO 72 J=1,JMAX
      DO 72 I=1,IMAX
      P(I,J,K)=G(I,J,K)*(P(I,J,K)+DUDX(I,J,K))
   72 CONTINUE
      CALL PARTIAL(1,P)
      CALL MOVLEV(DUDX(1,1,1),H2(1,1,1),IJK)
      CALL PARTIAL(2,V)
      DO 74 K=1,LMAXP1
      DO 74 J=1,JMAX
      DO 74 I=1,IMAX
      P(I,J,K)=2.*G(I,J,K)*DUDX(I,J,K)
   74 CONTINUE
      CALL PARTIAL(2,P)
      DO 76 K=1,LMAXP1
      DO 76 J=1,JMAX
      DO 76 I=1,IMAX
      H2(I,J,K)=H2(I,J,K)+DUDX(I,J,K)
   76 CONTINUE
      CALL PARTIAL(3,V)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(2,W)
      DO 78 K=1,LMAXP1
      DO 78 J=1,JMAX
      DO 78 I=1,IMAX
      P(I,J,K)=P(I,J,K)+DUDX(I,J,K)
   78 CONTINUE
      CALL PARTIAL(3,P)
      DO 80 K=1,LMAXP1
      DO 80 J=1,JMAX
      DO 80 I=1,IMAX
      H2(I,J,K)=H2(I,J,K)+(G(I,J,K)-MIU(K))*DUDX(I,J,K)
   80 CONTINUE
      CALL PARTIAL(3,G)
      DO 81 K=1,LMAXP1
```

```
      DO 81 J=1,JMAX
      DO 81 I=1,IMAX
      H2(I,J,K)=H2(I,J,K)+DUDX(I,J,K)*P(I,J,K)
   81 CONTINUE
      CALL PARTIAL(2,W)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(3,P)
      DO 815 K=1,LMAXP1
      DO 815 J=1,JMAX
      DO 815 I=1,IMAX
      H2(I,J,K)=H2(I,J,K)+MIU(K)*DUDX(I,J,K)
  815 CONTINUE
C****** Z-MOMENTUM EQUATION.
      CALL PARTIAL(1,W)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(3,U)
      DO 82 K=1,LMAXP1
      DO 82 J=1,JMAX
      DO 82 I=1,IMAX
      P(I,J,K)=G(I,J,K)*(P(I,J,K)+DUDX(I,J,K))
   82 CONTINUE
      CALL PARTIAL(1,P)
      CALL MOVLEV(DUDX(1,1,1),H3(1,1,1),IJK)
      CALL PARTIAL(2,W)
      CALL MOVLEV(DUDX(1,1,1),P(1,1,1),IJK)
      CALL PARTIAL(3,V)
      DO 84 K=1,LMAXP1
      DO 84 J=1,JMAX
      DO 84 I=1,IMAX
      P(I,J,K)=G(I,J,K)*(P(I,J,K)+DUDX(I,J,K))
   84 CONTINUE
      CALL PARTIAL(2,P)
      DO 86 K=1,LMAXP1
      DO 86 J=1,JMAX
      DO 86 I=1,IMAX
      H3(I,J,K)=H3(I,J,K)+DUDX(I,J,K)
   86 CONTINUE
      CALL PARTIAL(3,W)
      DO 88 K=1,LMAXP1
      DO 88 J=1,JMAX
      DO 88 I=1,IMAX
      P(I,J,K)=2.*DUDX(I,J,K)
   88 CONTINUE
      CALL PARTIAL(3,P)
      DO 90 K=1,LMAXP1
      DO 90 J=1,JMAX
      DO 90 I=1,IMAX
      H3(I,J,K)=H3(I,J,K)+(G(I,J,K)-MIU(K))*DUDX(I,J,K)
   90 CONTINUE
      CALL PARTIAL(3,G)
      DO 91 K=1,LMAXP1
      DO 91 J=1,JMAX
      DO 91 I=1,IMAX
      H3(I,J,K)=H3(I,J,K)+DUDX(I,J,K)*P(I,J,K)
   91 CONTINUE
      DO 100 J=1,JMAX
      DO 100 I=1,IMAX
      H1(I,J,1)=0.
      H1(I,J,2)=0.
      H2(I,J,1)=0.
      H2(I,J,2)=0.
      H3(I,J,1)=0.
      H3(I,J,2)=0.
      H1(I,J,LMAX)=0.
      H2(I,J,LMAX)=0.
      H3(I,J,LMAX)=0.
      H1(I,J,LMAXP1)=0.
      H2(I,J,LMAXP1)=0.
      H3(I,J,LMAXP1)=0.
  100 CONTINUE
```

```
      RETURN
      END
*DECK FILTER
      SUBROUTINE FILTER(HR)
C***********************************************************************
C* THIS SUBROUTINE FILTERS A THREE DIMENSIONAL ARRAY IN X AND Y DIRECTIONS
C***********************************************************************
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/SCM2/LMAXP1,D1,D2,D3,D4,D5,D6
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
*CALL C11
*CALL A1
*CALL B1
      CC=1./(IMAX*JMAX)
      DO 20 L=1,LMAXP1
      CALL MOVLEV(HR(1,1,L),FR(1,1),IJ)
      CALL FFTX(1.0)
      CALL FFTY(1.0,1.0)
      DO 30 I=1,IMAX
      DO 30 J=1,JMAX
      FR(I,J)=FR(I,J)*FILTX(I)*FILTY(J)
      FI(I,J)=FI(I,J)*FILTX(I)*FILTY(J)
   30 CONTINUE
      CALL FFTY(-1.0,CC)
      CALL FFTX(-1.0)
      CALL MOVLEV(FR(1,1),HR(1,1,L),IJ)
   20 CONTINUE
      RETURN
      END
*DECK STAT
      SUBROUTINE STAT
C***********************************************************************
C* THIS SUBROUTINE COMPUTES THE STATISTICS OF THE FLOW FOR OUTPUT.     *
C***********************************************************************
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
      COMMON/SCM2/LMAXP1,D1,D2,D3,D4,D5,D6
      COMMON/SCM4/CI,CJ,CK,CJK,CIK,CIJ
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
*CALL B4
*CALL B6
*CALL A6
*CALL A9
      PRINT 2000
 2000 FORMAT(1H1)
      PRINT 1100
 1100 FORMAT(1X,* UAVG IN X-Y*,4X,*VAVG IN X-Y*,3X,*WAVG IN X-Y*,1X,*
     1U2AVG IN XY*,3X,*V2AVG IN XY*,3X,*W2AVG IN XY*,3X,*Q2AVG IN XY*
     1,3X,*TURB SHEAR*,7X,*Z*)
      UTOT=0.
      VTOT=0.
      WTOT=0.
      U2TOT=0.
      V2TOT=0.
      W2TOT=0.
      QTOT=0.
      PAI=ACOS(-1.)
      DO 100 K=1,LMAXP1
      USUM=0.
      VSUM=0.
      WSUM=0.
      DO 110 J=1,JMAX
      DO 110 I=1,IMAX
      USUM=USUM+U(I,J,K)
      VSUM=VSUM+V(I,J,K)
      WSUM=WSUM+W(I,J,K)
  110 CONTINUE
      USUM=USUM*CIJ
      VSUM=VSUM*CIJ
      WSUM=WSUM*CIJ
      SHEAR=0.
```

```
      U2SUM=0.
      V2SUM=0.
      W2SUM=0.
      DO 120 J=1,JMAX
      DO 120 I=1,IMAX
      U2SUM=U2SUM+(U(I,J,K)-USUM)**2
      V2SUM=V2SUM+(V(I,J,K)-VSUM)**2
      W2SUM=W2SUM+(W(I,J,K)-WSUM)**2
      SHEAR=SHEAR+(U(I,J,K)-USUM)*(W(I,J,K)-WSUM)
  120 CONTINUE
      Q=(U2SUM+V2SUM+W2SUM)*CIJ*.5
      U2SUM=SQRT(U2SUM*CIJ)
      V2SUM=SQRT(V2SUM*CIJ)
      W2SUM=SQRT(W2SUM*CIJ)
      SHEAR=SHEAR*CIJ
      PRINT 1000,USUM,VSUM,WSUM,U2SUM,V2SUM,W2SUM,Q,SHEAR,Z(K)
      U2ST(K)=SQRT(U2SUM**2+U2S(K))
      V2ST(K)=SQRT(V2SUM**2+V2S(K))
      W2ST(K)=SQRT(W2SUM**2+W2S(K))
      UWT(K)=SHEAR+SSUM(K)
      UTOT=UTOT+USUM
      VTOT=VTOT+VSUM
      WTOT=WTOT+WSUM
      U2TOT=U2TOT+U2SUM
      V2TOT=V2TOT+V2SUM
      W2TOT=W2TOT+W2SUM
      QTOT=QTOT+Q
  100 CONTINUE
      UTOT=UTOT*CK
      VTOT=VTOT*CK
      WTOT=WTOT*CK
      U2TOT=U2TOT*CK
      V2TOT=V2TOT*CK
      W2TOT=W2TOT*CK
      QTOT=QTOT*CK
      PRINT 1200
 1200 FORMAT(///,1X,* UTOT IN X-Y    VTOT IN X-Y    WTOT IN X-Y    U2TOT
     1 IN X-Y    V2TOT IN X-Y    W2TOT IN X-Y    TURB ENERGY *)
      PRINT 1000,UTOT,VTOT,WTOT,U2TOT,V2TOT,W2TOT,QTOT
 1000 FORMAT(1P9E14.5)
      PRINT 200
  200 FORMAT(//,5X,* INSTANTENEOUS U*)
      PRINT 210,(U(8,8,K),K=1,LMAXP1)
  210 FORMAT(1X,1P9E14.5)
      PRINT 300
  300 FORMAT(////,30X,* SGS CONTRIBUTIONS ADDED*)
      PRINT 310
  310 FORMAT(1X,*  SGS ENERGY*,4X,* TOTALU2S *,5X,* TOTAL V2S *,3X,* TOT
     1ALW2S *,3X,*TOTAL SHEAR*,3X,* PLANE*)
      LMAXM1=LMAX-1
      DO 320 K=3,LMAXM1
      PRINT 330,EDYVI(K),U2ST(K),V2ST(K),W2ST(K),UWT(K),K
  320 CONTINUE
  330 FORMAT(1X,1P5E14.5,I6)
      RETURN
      END
*DECK TRANS
      SUBROUTINE TRANS
C***********************************************************************
C* THIS SUBROUTINE COMPUTES THE VARIOUS TRANSFORMATION QUANTITIES      *
C***********************************************************************
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/LENGTH/LSCALE(65)
      REAL LSCALE
      COMMON/INNERC/CVINR(65)
      COMMON/SCM3/DELTA1,DELTA2,RE,E
      COMMON/RANGE/LMAXM1,LMAXM2,LMAXM3,LMAXM4,LMAXM5
      COMMON/TINC/DT
      COMMON/BC/CE1,CE2,CE3,CE4,CE5,CE6
*CALL B2
```

```
*CALL B3
*CALL A9
*CALL C7
*CALL B7
*CALL B8
      COMMON/PENTA2/XI,QI,GI,YI,QJ,GJ,XN,QIN,GIN,YN,QJN,GJN,Q2,Q3,
     1RC1,RC2,RP1,RP2,RP3,RP4
      COMMON/ZERO/C3,C4
      COMMON/IDENTN/CODE
      LMAXM1=LMAX-1
      LMAXM2=LMAX-2
      LMAXM3=LMAX-3
      LMAXM4=LMAX-4
      LMAXM5=LMAX-5
      LMAXP1=LMAX+1
      LHP1=LMAX/2+1
C***** MESH STRECHING TRANSFORMATION
      P=0.98346
      TANIP=0.5*ALOG((1.+P)/(1.-P))
      PINV=1./P
      P2=P**2
      DO 5 J=1,LMAXP1
      ZETA(J)=-1.+2.*(J-2)/(LMAX-2)
      DUM1=ZETA(J)*TANIP
      Z(J)=PINV*TANH(DUM1)
      RL(J)=(2.*P2/TANIP)*((COSH(DUM1))**3)*(SINH(DUM1))
      RM(J)=P2*((COSH(DUM1))**4)/(TANIP**2)
    5 CONTINUE
      DELTA3=ZETA(2)-ZETA(1)
      E2=RL(2)/(2.*DELTA3*RE)
      F2=RM(2)/((DELTA3**2)*RE)
      EN=RL(LMAX)/(2.*DELTA3*RE)
      FN=RM(LMAX)/((DELTA3**2)*RE)
      R2=(F2+E2)/(F2-E2)
      RN=(FN-EN)/(FN+EN)
      RR2=1./(E2-F2)
      RRN=-1./(EN+FN)
      PRINT 20
   20 FORMAT(6X,*ZETA*,12X,*Z*,14X,*RL*,14X,*RM*)
      DO 30 K=1,LMAXP1
      PRINT 40,ZETA(K),Z(K),RL(K),RM(K)
   30 CONTINUE
   40 FORMAT(1X,1P4E15.7)
      PRINT 50,E2,F2,EN,FN,R2,RN,DELTA3
   50 FORMAT(1X,//,1P7E14.5)
      CC=0.2
C***** COMPUTE THE LENGTH SCALE FOR THE SGS MODEL
      VONK=0.4
      DFILT1=2.*DELTA1
      DFILT2=2.*DELTA2
      POWER=1./3.
      DO 300 K=3,LHP1
      KM1=K-1
      DW=(Z(K)-Z(2))*VONK
      GRID=Z(K)-Z(KM1)
      LSCALE(K)=(AMIN1(DW,0.1,DFILT1))*(AMIN1(DW,0.1,DFILT2))*(AMIN1
     1(DW,0.1,GRID))
      LSCALE(K)=LSCALE(K)**POWER
  300 CONTINUE
      DO 310 K=LHP1,LMAXM1
       KK=LMAXM1-K+LHP1
      KKP1=KK+1
      DW=(Z(LMAX)-Z(KK))*VONK
      GRID=Z(KKP1)-Z(KK)
      LSCALE(KK)=(AMIN1(DW,0.1,DFILT1))*(AMIN1(DW,0.1,DFILT2))*(AMIN1
     1(DW,0.1,GRID))
      LSCALE(KK)=LSCALE(KK)**POWER
  310 CONTINUE
      CINER=(CC**2)/(VONK*27.)
      DO 320 K=3,LMAXM1
```

```
      CV(K)=(CC*LSCALE(K))**2
      CVINR(K)=CINER*RE*(LSCALE(K))**4
  320 CONTINUE
      PRINT 330
  330 FORMAT(//,20X,* COEFFICIENT OF INNER SGS*)
      PRINT 120,(CVINR(K),K=3,LMAXM1)
      PRINT 340
  340 FORMAT(//,20X,* SUBGRID LENGTH SCALE*)
      PRINT 120,(LSCALE(K),K=3,LMAXM1)
      PRINT 110
  110 FORMAT(20X,* COEFFICIENT OF SGS *)
      PRINT 120,(CV(K),K=3,LMAXM1)
  120 FORMAT(1X,1P9E14.5)
      FAC=226*(CC**2)/3.
      FACTOR(1)=0.
      FACTOR(2)=0.
      FACTOR(LMAX)=0.
      FACTOR(LMAXP1)=0.
      DO 100 K=3,LMAXM1
      FACTOR(K)=FAC/CV(K)
  100 CONTINUE
      DO 12 J=2,LMAX
      H1=Z(J)-Z(J-1)
      H2=Z(J+1)-Z(J)
C***** ARRAYS FOR FINITE DIFFERENCE IN Z-DIRECTION
      AP(J)=-1./(Z(J+1)-Z(J-1))
      BP(J)=0.
      CP(J)=-AP(J)
C*********** DEFINE THE COEFFICIENTS FOR SECOND DERIVATIVE IN Z DIREC
      AP2(J)=2./(H1*(H1+H2))
      BP2(J)=-2./(H1*H2)
      CP2(J)=2./(H2*(H1+H2))
      PRINT 80,AP(J),BP(J),CP(J), AP2(J),BP2(J),CP2(J)
   12 CONTINUE
C*** CONSTANTS FOR THE BLOCK TRI-DIAGONAL MATRIX IN THE MAIN PROGRAM
      T=0.5*(Z(3)-Z(2))
      CE1=1.-AP(3)*T*E*DT*0.5*(CP2(2)-AP2(2)*CP(2)/AP(2))/(1.+T*AP(3))
      CE2=BP(3)+AP(3)*(1.-T*BP(3))/(1.+T*AP(3))
      CE3=CP(3)-AP(3)*T*CP(3)/(1.+T*AP(3))
      T=0.5*(Z(LMAX)-Z(LMAXM1))
      CE4=AP(LMAXM1)+CP(LMAXM1)*T*AP(LMAXM1)/(1.-T*CP(LMAXM1))
      CE5=BP(LMAXM1)+CP(LMAXM1)*(1.+T*BP(LMAXM1))/(1.-T*CP(LMAXM1))
      CE6=1.+CP(LMAXM1)*T*E*DT*0.5*(AP2(LMAX)-CP2(LMAX)*AP(LMAX)
     1/CP(LMAX))/(1.-T*CP(LMAXM1))
      T=0.5*(Z(3)-Z(2))
      C3=(1.-T*BP(3))/CP(3)
      C4=(T*CP(3)/(1.-T*BP(3)))
      Q=1./(1.+T*AP(3))
      XI=-T*Q
      QI=(1.-T*BP(3))*Q
      GI=-T*CP(3)*Q
      YI=(1.+BP(2)*T*Q)/AP(2)
      QJ=(BP(2)*(T*BP(3)-1.)*Q-CP(2))/AP(2)
      GJ=BP(2)*T*CP(3)*Q/AP(2)
      T=0.5*(Z(LMAX)-Z(LMAXM1))
      Q=1./(1.-T*CP(LMAXM1))
      XN=T*Q
      QIN=(1.+T*BP(LMAXM1))*Q
      GIN=T*AP(LMAXM1)*Q
      YN=(1.-BP(LMAX)*T*Q)/CP(LMAX)
      QJN=-(BP(LMAX)*(1.+T*BP(LMAXM1))*Q+AP(LMAX))/CP(LMAX)
      GJN=-T*AP(LMAXM1)*BP(LMAX)*Q/CP(LMAX)
   80 FORMAT(1X,1P3E15.7,5X,1P3E15.7)
   90 FORMAT(1X,1P5E15.7)
      RETURN
      END
*DECK VISCOS
      SUBROUTINE VISCOS(U)
C***** THIS SUBROUTINE COMPUTES THE SECOND DERIVATIVE OF U IN THE Z-DIRECTION
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
```

```
*CALL A2
*CALL B7
*CALL B9
*CALL A9
      LMAXP1=LMAX+1
      DELTA3=2./(LMAX-2.)
      DO 20 J=1,JMAX
      DO 20 I=1,IMAX
      DUDX(I,J,1)=0.
      DUDX(I,J,LMAXP1)=0.
   20 CONTINUE
      DO 30 K=2,LMAX
      DO 30 J=1,JMAX
      DO 30 I=1,IMAX
      KP1=K+1
      KM1=K-1
      DUDX(I,J,K)=AP2(K)*U(I,J,KM1)+BP2(K)*U(I,J,K)+CP2(K)*U(I,J,KP1)
   30 CONTINUE
      RETURN
      END
*DECK EXTERN
      SUBROUTINE EXTERN(L1,L2,R,RR)
C ***********************************************************************
C * THIS SUBROUTINE FIXES THE EXTERNAL VALUES OF THE U AND V AND W       *
C * NOTE THAT THE EXTERNAL VALUES OF U AND V WILL NOT ENTER INTO THE     *
C * COMPUTATION.  AND THEY ARE UNNECESSARY                               *
C ***********************************************************************
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
      COMMON/SCM3/DELTA1,DELTA2,RE,E
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
*CALL A1
*CALL C3
*CALL A6
*CALL C7
      LMAXP1=LMAX+1
      LMAXM1=LMAX-1
      DO 90 J=1,JMAX
      DO 90 I=1,IMAX
      W(I,J,1)=-CP(2)*W(I,J,3)/AP(2)
   90 W(I,J,LMAXP1)=-AP(LMAX)*W(I,J,LMAXM1)/CP(LMAX)
      DO 97 J=1,JMAX
      DO 97 I=1,IMAX
      U(I,J,1)=0.
      V(I,J,1)=0.
      U(I,J,LMAXP1)=0.
      V(I,J,LMAXP1)=0.
   97 CONTINUE
   95 CONTINUE
      RETURN
      END
*DECK MTDAG
      SUBROUTINE MTDAG(AM,A,AP,V,N,K)
C           SOLVES COUPLED TRI-DIAGONAL ALGEBRAIC EQUATIONS
C        AM(I,J,L)X(J,L-1)+A(I,J,L)X(J,L)+AP(I,J,L)X(J,L+1)=Y(I,L)
C          (SUM OVER J IN EACH EQUATION)
C          (I,J,L)  I IS EQUATION TYPE, J IS VARIABLE TYPE, L IS NODE
C          AT CALL V(I,L)=X(I,L)  Y(J,L) IS RETURNED IN V(J,L)
C          THE AM,A,AP ARRAYS ARE RETURNED AS GARBAGE
      REAL AM(N,N,K),A(N,N,K),AP(N,N,K),V(N,K)
      COMMON/SING/IMR,JMR,IMI,JMI
C        ELIMINATE TO OBTAIN A SEQUENTIALLY SOLVABLE FORM
      DO 20 LX=1,K
      L=K-LX+1
      LM=L-1
      DO 18 J=1,N
      C=A(J,J,L)
      IF (C.EQ.0.)  GO TO 80
C          ELIMINATE X(J,L) FROM ALL EQUATIONS OTHER THAN ITS OWN
      DO 16 I=1,N
C          ELIMINATE X(J,L) FROM THE EQUATION FOR THE NODE L-1
```

```
      IF (L.EQ.1)  GO TO 12
      F=AP(I,J,LM)
      IF (F.EQ.0.0)  GO TO 12
      F=F/C
      DO 6 J1=1,N
      A(I,J1,LM)=A(I,J1,LM)-F*AM(J,J1,L)
    6 AP(I,J1,LM)=AP(I,J1,LM)-F*A(J,J1,L)
      V(I,LM)=V(I,LM)-F*V(J,L)
C          ELIMINATE X(J,L) FROM OTHER EQUATIONS AT THIS NODE L
   12 IF (I.EQ.J)  GO TO 16
      F=A(I,J,L)
      IF (F.EQ.0.0)  GO TO 16
      F=F/C
      DO 14 J1=1,N
      A(I,J1,L)=A(I,J1,L)-F*A(J,J1,L)
      IF (L.EQ.1)  GO TO 14
      AM(I,J1,L)=AM(I,J1,L)-F*AM(J,J1,L)
   14 CONTINUE
      V(I,L)=V(I,L)-F*V(J,L)
   16 CONTINUE
   18 CONTINUE
   20 CONTINUE
C          CARRY OUT THE BACK SOLUTION
      DO 30 L=1,K
      LM=L-1
      DO 28 I=1,N
      C=A(I,I,L)
      IF (C.EQ.0.0)  GO TO 80
      F=V(I,L)
      IF (L.EQ.1)  GO TO 28
      DO 24 J1=1,N
   24 F=F-AM(I,J1,L)*V(J1,LM)
   28 V(I,L)=F/C
   30 CONTINUE
      RETURN
   80 PRINT 90
      PRINT 10,IMR,JMR,IMI,JMI
   10 FORMAT(4X,4I5)
      RETURN
   90 FORMAT(///,10X,* MTDAG MATRIX IS SINGULAR *)
      END
*DECK DIVG
      SUBROUTINE DIVG
C     THIS SUBROUTINE COMPUTES THE DIVERGENCE OF VELOCITY FIELD
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF
*CALL A2
*CALL A6
*CALL A5
      CALL PARTIAL(1,U)
      CALL MOVLEV(DUDX(1,1,1),G(1,1,1),IJK)
      CALL PARTIAL(2,V)
      DO 10 K=2,LMAX
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      G(I,J,K)=G(I,J,K)+DUDX(I,J,K)
   10 CONTINUE
      CALL PARTIAL(3,W)
      DO 20 K=2,LMAX
      DO 20 J=1,JMAX
      DO 20 I=1,IMAX
      G(I,J,K)=G(I,J,K)+DUDX(I,J,K)
   20 CONTINUE
      BMAX=0.
      DO 30 K=2,LMAX
      DO 30 J=1,JMAX
      DO 30 I=1,IMAX
      IF(ABS(G(I,J,K)).GT.BMAX) BMAX=ABS(G(I,J,K))
   30 CONTINUE
      PRINT 40,BMAX
```

```
   40 FORMAT(2X,* MAX DIVERGENCE=*,1P1E15.7)
      RETURN
      END
*DECK COURANT
      SUBROUTINE COURANT(DT,NTIME,TEND)
C***** THIS SUBROUTINE MONITORS THE COURANT NUMBER
*CALL A9
*CALL A6
      COMMON/SCM3/DELTA1,DELTA2,RE,E
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
      LMAXM1=LMAX-1
      LHP1=LMAX/2+1
      BMAX=0.
      DO 51 K=3,LHP1
      KM1=K-1
      DO 51 J=1,JMAX
      DO 51 I=1,IMAX
      CMAX1=ABS(W(I,J,K))/(Z(K)-Z(KM1))+ABS(U(I,J,K)/DELTA1)+ABS(V(I,J,
     1K))/DELTA2
      IF(CMAX1.LT.BMAX) GO TO 51
      BMAX=CMAX1
      IDUM1=I
      JDUM1=J
      KDUM1=K
   51 CONTINUE
      DMAX=0.
      DO 56 K=LHP1,LMAXM1
      KP1=K+1
      DO 56 J=1,JMAX
      DO 56 I=1,IMAX
      CMAX2=ABS(W(I,J,K))/(Z(KP1)-Z(K))+ABS(U(I,J,K))/DELTA1+ABS(
     1V(I,J,K))/DELTA2
      IF(CMAX2.LT.DMAX) GO TO 56
      DMAX=CMAX2
      IDUM2=I
      JDUM2=J
      KDUM2=K
   56 CONTINUE
      BMAX=BMAX*DT
      DMAX=DMAX*DT
      PRINT 61,BMAX,IDUM1,JDUM1,KDUM1,DMAX,IDUM2,JDUM2,KDUM2
   61 FORMAT(2X,* COURRANT *,1P1E14.5,3I5,1P1E14.5,3I5)
      IF(BMAX.GT.0.35.OR.DMAX.GT.0.35) NTIME=TEND
      RETURN
      END
*DECK LTAVG
      SUBROUTINE LTAVG
C***** THIS SUBROUTINE COMPUTES THE RUNNING TIME AVERAGE OF VARIOUS
C***** STATISTICAL QUANTITIES.
      COMMON/SCM4/CI,CJ,CK,CJK,CIK,CIJ
      COMMON/DATA9/IMAX,JMAX,LMAX,NHALFX,NHALFY
*CALL A2
*CALL A3
*CALL A4
*CALL A5
*CALL A6
      COMMON/RANGE/LMAXM1,LMAXM2,LMAXM3,LMAXM4,LMAXM5
      COMMON/SCM2/LMAXP1,D1,D2,D9,D4,D5,D6
      COMMON/LTA1/USUM(65),UTSUM(65),STSUM(65),U2SMT(65),V2SMT(65)
     1,W2SMT(65),PVT(65),PUT(65),PUNST(65),PVNST(65),PWNST(65),PWT(65)
     2,TCONT
      COMMON/LTA2/PDUT(65),PDVT(65),PDWT(65),PDUNT(65),PDVNT(65),PDWNT
     1(65)
      COMMON/ADV/NTIME
      IF(NTIME.NE.1) GO TO 5
      TCONT=0.
      DO 2 K=3,LMAXM1
      UTSUM(K)=0.
      U2SMT(K)=0.
      V2SMT(K)=0.
```

```
      W2SMT(K)=0.
      STSUM(K)=0.
      PUT(K)=0.
      PVT(K)=0.
      PWT(K)=0.
      PUNST(K)=0.
      PVNST(K)=0.
      PWNST(K)=0.
      PDUT(K)=0.
      PDVT(K)=0.
      PDWT(K)=0.
      PDUNT(K)=0.
      PDVNT(K)=0.
      PDWNT(K)=0.
    2 CONTINUE
    5 CONTINUE
      TCONT=TCONT+1
      DO 10 K=3,LMAXM1
      USUM(K)=0.
      DO 15 J=1,JMAX
      DO 15 I=1,IMAX
      USUM(K)=USUM(K)+U(I,J,K)
   15 CONTINUE
      USUM(K)=USUM(K)*CIJ
      UTSUM(K)=UTSUM(K)+USUM(K)
   10 CONTINUE
      DO 20 K=3,LMAXM1
      U2SUM=0.
      V2SUM=0.
      W2SUM=0.
      SSUM=0.
      DO 25 J=1,JMAX
      DO 25 I=1,IMAX
      U2SUM=U2SUM+(U(I,J,K)-USUM(K))**2
      V2SUM=V2SUM+V(I,J,K)**2
      W2SUM=W2SUM+W(I,J,K)**2
      SSUM=SSUM+W(I,J,K)*(U(I,J,K)-USUM(K))
   25 CONTINUE
      U2SUM=U2SUM*CIJ
      V2SUM=V2SUM*CIJ
      W2SUM=W2SUM*CIJ
      SSUM=SSUM*CIJ
      U2SMT(K)=U2SMT(K)+U2SUM
      V2SMT(K)=V2SMT(K)+V2SUM
      W2SMT(K)=W2SMT(K)+W2SUM
      STSUM(K)=STSUM(K)+SSUM
   20 CONTINUE
      DO 30 K=1,LMAXP1
      DO 30 J=1,JMAX
      DO 30 I=1,IMAX
      P(I,J,K)=(U(I,J,K)**2+V(I,J,K)**2+W(I,J,K)**2)/2.
   30 CONTINUE
      CALL FILTER(P)
      DO 35 K=1,LMAXP1
      DO 35 J=1,JMAX
      DO 35 I=1,IMAX
      DIVC(I,J,K)=G(I,J,K)-P(I,J,K)
   35 CONTINUE
      CALL PARTIAL(1,DIVC)
      DO 40 K=3,LMAXM1
      PU=0.
      DO 45 J=1,JMAX
      DO 45 I=1,IMAX
      PU=PU+DUDX(I,J,K)*U(I,J,K)
   45 CONTINUE
      PU=PU*CIJ
      PUT(K)=PUT(K)+PU
   40 CONTINUE
      CALL PARTIAL(1,G)
      DO 50 K=3,LMAXM1
```

```
      PUNS=0.
      DO 55 J=1,JMAX
      DO 55 I=1,IMAX
      PUNS=PUNS+DUDX(I,J,K)*U(I,J,K)
   55 CONTINUE
      PUNS=PUNS*CIJ
      PUNST(K)=PUNST(K)+PUNS
   50 CONTINUE
      CALL PARTIAL(2,DIVC)
      DO 60 K=3,LMAXM1
      PV=0.
      DO 65 J=1,JMAX
      DO 65 I=1,IMAX
      PV=PV+DUDX(I,J,K)*V(I,J,K)
   65 CONTINUE
      PV=PV*CIJ
      PVT(K)=PVT(K)+PV
   60 CONTINUE
      CALL PARTIAL(2,G)
      DO 70 K=3,LMAXM1
      PVNS=0.
      DO 75 J=1,JMAX
      DO 75 I=1,IMAX
      PVNS=PVNS+DUDX(I,J,K)*V(I,J,K)
   75 CONTINUE
      PVNS=PVNS*CIJ
      PVNST(K)=PVNST(K)+PVNS
   70 CONTINUE
      CALL PARTIAL(3,DIVC)
      DO 80 K=3,LMAXM1
      PW=0.
      DO 85 J=1,JMAX
      DO 85 I=1,IMAX
      PW=PW+DUDX(I,J,K)*W(I,J,K)
   85 CONTINUE
      PW=PW*CIJ
      PWT(K)=PWT(K)+PW
   80 CONTINUE
      CALL PARTIAL(3,G)
      DO 90 K=3,LMAXM1
      PWNS=0.
      DO 95 J=1,JMAX
      DO 95 I=1,IMAX
      PWNS=PWNS+DUDX(I,J,K)*W(I,J,K)
   95 CONTINUE
      PWNS=PWNS*CIJ
      PWNST(K)=PWNST(K)+PWNS
   90 CONTINUE
      CALL PARTIAL(1,U)
      DO 100 K=3,LMAXM1
      PDU=0.
      PDUN=0.
      DO 105 J=1,JMAX
      DO 105 I=1,IMAX
      PDU=PDU+DUDX(I,J,K)*DIVC(I,J,K)
      PDUN=PDUN+DUDX(I,J,K)*G(I,J,K)
  105 CONTINUE
      PDU=PDU*CIJ
      PDUN=PDUN*CIJ

      PDUN=PDUN*CIJ
      PDUT(K)=PDUT(K)+PDU
      PDUNT(K)=PDUNT(K)+PDUN
  100 CONTINUE
      CALL PARTIAL(2,V)
      DO 110 K=3,LMAXM1
      PDV=0.
      PDVN=0.
      DO 115 J=1,JMAX
      DO 115 I=1,IMAX
```

```
      PDV=PDV+DUDX(I,J,K)*DIVC(I,J,K)
      PDVN=PDVN+DUDX(I,J,K)*G(I,J,K)
  115 CONTINUE
      PDV=PDV*CIJ
      PDVN=PDVN*CIJ
      PDVT(K)=PDVT(K)+PDV
      PDVNT(K)=PDVNT(K)+PDVN
  110 CONTINUE
      CALL PARTIAL(3,W)
      DO 120 K=3,LMAXM1
      PDWN=0.
      PDW=0.
      DO 125 I=1,IMAX
      DO 125 J=1,JMAX
      PDW=PDW+DUDX(I,J,K)*DIVC(I,J,K)
      PDWN=PDWN+DUDX(I,J,K)*G(I,J,K)
  125 CONTINUE
      PDW=PDW*CIJ
      PDWN=PDWN*CIJ
      PDWT(K)=PDWT(K)+PDW
      PDWNT(K)=PDWNT(K)+PDWN
  120 CONTINUE
      RETURN
      END
*DECK LTPR
      SUBROUTINE LTPR
C***** THIS SUBROUTINE PRINTS LONG TIME AVERAGES AT DESIGNATED INTERVALS
      COMMON/RANGE/LMAXM1,LMAXM2,LMAXM3,LMAXM4,LMAXM5
      COMMON/LTA1/USUM(65),UTSUM(65),STSUM(65),U2SMT(65),V2SMT(65)
     1,W2SMT(65),PVT(65),PUT(65),PUNST(65),PVNST(65),PWNST(65),PWT(65)
     2,TCONT
      COMMON/LTA2/PDUT(65),PDVT(65),PDWT(65),PDUNT(65),PDVNT(65),PDWNT
     1(65)
      COMMON/SGTT/SGST(65),ETED(65),U2STT(65),V2STT(65),W2STT(65)
     1,TSHGS,TSCNT
      PRINT 10,TCONT,TSHGS,TSCNT
   10 FORMAT(//,10X,* COUNTERS *,1P3E14.5)
      F1=1./TCONT
      F2=1./TSHGS
      F3=1./TSCNT
      DO 20 K=3,LMAXM1
      A1=UTSUM(K)*F1
      A2=U2SMT(K)*F1
      A3=V2SMT(K)*F1
      A4=W2SMT(K)*F1
      A5=STSUM(K)*F1
      A6=PUT(K)*F1
      A7=PVT(K)*F1
      A8=PWT(K)*F1
      PRINT 30,A1,A2,A3,A4,A5,A6,A7,A8,K
   20 CONTINUE
      PRINT 40
   40 FORMAT(/////)
      DO 50 K=3,LMAXM1
      A1=PUNST(K)*F1
      A2=PVNST(K)*F1
      A3=PWNST(K)*F1
      A4=SGST(K)*F2
      A5=ETED(K)*F3
      A6=U2STT(K)*F3
      A7=V2STT(K)*F3
      A8=W2STT(K)*F3
      PRINT 30,A1,A2,A3,A4,A5,A6,A7,A8,K
   50 CONTINUE
   30 FORMAT(3X,1P8E14.5,I5)
      PRINT 40
      DO 60 K=3,LMAXM1
      A1=PDUT(K)*F1
      A2=PDVT(K)*F1
      A3=PDWT(K)*F1
```

```
      A4=PDUNT(K)*F1
      A5=PDVNT(K)*F1
      A6=PDWNT(K)*F1
      PRINT 70,A1,A2,A3,A4,A5,A6,K
   60 CONTINUE
   70 FORMAT(1P6E14.5,I5)
      RETURN
      END
```

References

Abbott, D. E. (1978), Proceedings of the Workshop on Coherent Structures in Turbulent Boundary Layers. To be published.

Blackwelder, R. F., and R. E. Kaplan (1976), "On the Wall Structure of the Turbulent Boundary Layer," J. Fluid Mech.," 76, 89.

Blottner, F. G. (1974), "Nonuniform Grid Method for Turbulent Boundary Layers," Fourth Int. Conf. on Num. Fluid Dynamics.

Clark, J. A. (1968), "A Study of Incompressible Turbulent Boundary Layers in Channel Flow," J. Basic. Engrg., 90, 455.

Clark, J. A., and E. Markland (1970), "Vortex Structures in Turbulent Boundary Layers," Aeronaut. J., 74, 243.

Comte-Bellot, G. (1963), "Contribution a l'etude de la Turbulence de Conduite," Doctoral Thesis, University of Grenoble.

Corino, E. R., and R. S. Brodkey (1969), "A Visual Investigation of the Wall Region in Turbulent Flow," J. Fluid Mech., 37, 1.

Deardorff, J. W. (1970), "A Numerical Study of Three-Dimensional Turbulent Channel Flow at Large Reynolds Number," J. Fluid Mech., 41, 453.

Deardorff, J. W. (1973), "The Use of Subgrid Transport Equations in a Three-Dimensional Model of Atmospheric Turbulence," J. Fluid Engrg., Sept. 1973, 429.

Ferziger, J. H., U. B. Mehta, and W. C. Reynolds (1977), "Large Eddy Simulation of Homogeneous Isotropic Turbulence," Proc. Symp. on Turbulent Shear Flows, Penn. State.

Fox, D. G., and S. A. Orszag (1973), "Pseudospectral Approximation to Two-Dimensional Turbulence," J. Comp. Phys., 11, No. 4, Apr. 1973.

Fox, L., and I. B. Parker (1968), Chebyshev Polynomials in Numerical Analysis, Oxford: University Press.

Grass (1971), "Structural Features of Turbulent Flow over Smooth and Rough Boundaries," J. Fluid Mech., 50, 233.

Hinze, J. O. (1975), Turbulence, McGraw-Hill, Inc., 2nd ed.

Hussain, A. K. M. F., and W. C. Reynolds (1975), "Measurements in Fully Developed Turbulent Channel Flow," J. Fluid Engrg., 97, Dec. 1975, 568.

Kim, H. T., S. J. Kline, and W. C. Reynolds (1971), "The Production of Turbulence Near a Smooth Wall in a Turbulent Boundary Layer," J. Fluid. Mech., 50, 133.

Klebanoff, P. S. (1954), "Characteristics of Turbulence in a Boundary Layer with Zero Pressure Gradient," NACA Tech. Note, 3178.

Kline, S. J., W. C. Reynolds, F. A. Schraub, and P. W. Runstadler (1967), "The Structure of Turbulent Boundary Layers," J. Fluid Mech., 30, 741.

Kline, S. J., and P. W. Runstadler (1959), "Some Preliminary Results of Visual Studies of Wall Layers of the Turbulent Boundary Layer," J. Appl. Mech., 2, 166.

Kwak, D., W. C. Reynolds, and J. H. Ferziger (1975), "Three-Dimensional, Time-Dependent Computation of Turbulent Flows," Report No. TF-5, Mechanical Engineering Department, Stanford Univ.

Laufer, J. (1951), "Investigation of Turbulent Flow in a Two-Dimensional Channel," NACA Report 1053.

Laufer, J. (1954), "The Structure of Turbulence in Fully Developed Pipe Flow," NACA Rept. 1174.

Laufer, J. (1975), "Annual Review of Fluid Mechanics," 2, 95.

Lee, L. H., and W. C. Reynolds (1967), "On the Approximate and Numerical Solution of Orr-Sommerfeld Problems," Quart. J. Mech. and Appl. Math., XX, 1, Feb. 1967.

Leonard, A. (1974), "On the Energy Cascade in Large-Eddy Simulations of Turbulent Fluid Flows," Adv. in Geophysics, 18A, 237.

Lilley, G. M. (1960), College Aeron. Cranfield, Co. A Rep., 133

Lilley, G. M., and T. H. Hodgson (1960), AGARD Rep. 276.

Lilly, D. K. (1965), "On the Computational Stability of Numerical Solutions of Time-Dependent Nonlinear Geophysical Fluid Dynamic Problems," Monthly Weather Rev., 93, No. 1, Jan. 1965.

Lilly, D. K. (1967), "The Representation of Small-Scale Turbulence in Numerical Simulation Experiments," Proc. of the IBM Scient. Comp. Symp. on Env. Sciences, IBM Form No. 320-1951.

Lin, C. C. (1955), The Theory of Hydrodynamic Stability, Cambridge Univ. Press.

Mansour, N. N., P. Moin, W. C. Reynolds, and J. H. Ferziger (1977), "Improved Methods for Large Eddy Simulation of Turbulence," Proc. Symp. on Turbulent Shear Flows, Penn. State.

Mehta, U. B. (1977), "Dynamic Stall of an Oscillating Airfoil," AGARD Fluid Dynamics Panel Symp. on Unst. Aerodynamics, Ottawa, Canada.

Moin, P., N. N. Mansour, U. B. Mehta, W. C. Reynolds, and J. H. Ferziger (1978), "Improvements in Large Eddy Simulation Technique: Special Methods and High-Order Statistics," Report No. TF-10, Mech. Engrg. Dept., Stanford Univ.

Narahari Rao, K., R. Narasimha, and M. A. Badri Nara Yanan, (1971), "The 'Bursting' Phenomenon in a Turbulent Boundary Layer," J. Fluid Mech., 48, 339.

Orszag, S. A. (1971), "Galerkin Approximation to Flows within Slabs, Spheres, and Cylinders," Phys. Rev. Letters, 26, 1100.

Orszag, S. A. (1972), "Comparison of Pseudospectral and Spectral Approximation," Studies in Appl. Math., Vol. LI, No. 3, Sept. 1972, 253.

Phillips, N. A. (1959), "An Example of Nonlinear Computational Instability," The Atmosphere and Sea in Motion, Rockefeller Inst. Press, New York.

Richtmyer, R. D., and K. W. Morton (1967), Difference Methods for Initial Value Problems, New York: Interscience, 2nd ed.

Runstadler, P. W., S. J. Kline, and W. C. Reynolds (1963), "An Investigation of the Flow Structure of the Turbulent Boundary Layer, Report No. MD-8, Dept. of Mech. Engrg., Stanford Univ.

Schumann, U. (1973), "Ein Verfahren zur direckten numerischen turbulenter Strömungen in Platten und Ringspaltkanälen und über seine Anwendung zur Untersuchung von Turbulenzmodellen," Universität Karlsruhe (NASA Tech. Translation, NASA TTF 15, 391).

Shaanan, S., J. H. Ferziger, and W. C. Reynolds (1975), "Numerical Simulation of Turbulence in the Presence of Shear," Report No. TF-6, Mech. Engrg. Dept., Stanford Univ.

Smagorinsky, J. (1963), "General Circulation Experiments with the Primitive Equations," Monthly Weather Rev., 91, 99.

Smagorinsky, J., S. Manake, and J. L. Holloway (1965), Monthly Weather Rev., 93, 727.

Townsend, A. A. (1951), "The Structure of the Turbulent Boundary Layer," Proc. Cambridge Phil. Soc., 47, 375.

Townsend, A. A. (1956), The Structure of Turbulent Shear Flow, Cambridge University Press.

Willmarth, W. W., and C. E. Wooldridge (1962), "Measurement of the Fluctuating Pressure at the WAll beneath a Thick Turbulent Boundary Layer," J. Fluid Mech., 14; Corrigendum: J. Fluid Mech., 21 (1965).

Willmarth, W. W., and C. E. Wooldridge (1963), "Measurements of the Correlation between the Fluctuating Velocities and the Fluctuating Wall Pressure in a Thick Turbulent Boundary Layer," AGARD Rep. 456.

Willmarth, W. W. (1975), "Structure of Turbulence in Boundary Layers," Advances in Applied Mechanics (ed. by C. S. Yih), Academic Press, New York, Vol. 15.