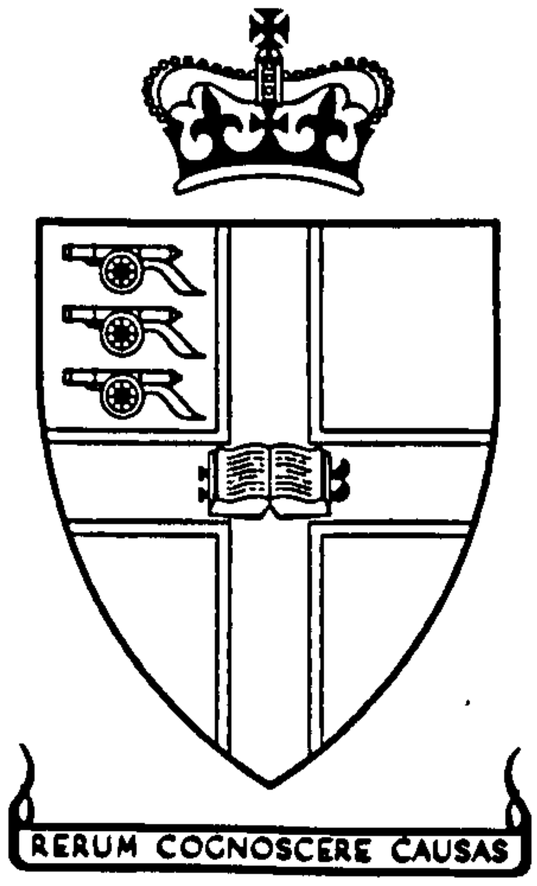
CRANFIELD UNIVERSITY

Royal Military College of Science

School of Engineering and Applied Science



PhD Thesis

Academic Year 1996-7

A. K. SENIOR

A NUMERICAL STUDY OF RESISTANCE IN A ROUGH WALLED CHANNEL FLOW WHERE THE RATIO OF ROUGHNESS LENGTH SCALE TO THE DEPTH OF FLOW VARIES OVER A WIDE RANGE

Academic Supervisor: Dr. D. G. Rhodes
Industrial Supervisor: Dr. P. G. Samuels

June 1997

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

ABSTRACT

Numerical calculations were performed over a variety of two-dimensional rib roughness configurations in which the ratio of flow depth to roughness height was varied from 1.1 to 40. Periodically fully developed flow was achieved by employing periodic boundary conditions and the effect of turbulence was accounted for by a two-layer model.

These calculations were used to test the hypothesis that any rough wall resistance may be reduced to an equivalent wall shear stress located on a plane wall. The position of the plane wall is determined by a novel method of prediction obtained by consideration of streamwise force moments. The resistance is then determined by three dynamically significant length scales: the first (y_0) specifies the position of the equivalent plane wall, the second is the depth of flow h and the third is similar to Nikuradse's sand grain roughness k_s. The latter length scale is however depth dependent and a universal relationship is postulated:

$$\frac{k_s}{h-y_0} = \mathrm{F}\left(\frac{k_s}{k_{s\infty}}\right)$$

where $k_{s\infty}$ is the asymptotic value of k_s at very large flow depths. For the calculation of friction factor, a resistance equation is proposed of the form typical of fully rough flows. These postulates are supported by the numerical model results though further work including physical experiments is required to confirm them.

Before applying the two-layer model to this problem it was tested on smooth rectangular duct flows and Schlichting's (1936) long angle roughness experiments. The opportunity was taken to further explore these flows, and in addition calculations were carried out for Grass *et al*'s (1991) open channel rib roughness experiments.

The periodic boundary conditions were also applied to a laminar counter-flow plate-fin heat exchanger. A novel source-sink arrangement for heat flux was developed in order to implement these boundary conditions.

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

ACKNOWLEDGEMENTS

I am indebted to many people who have assisted me during the course of this study:

My academic supervisor Dr. D. G. Rhodes provided expert guidance throughout the project and his advice has been invaluable.

I would like to acknowledge the assistance of my industrial supervisor, Dr. P. G. Samuels and the financial support provided by Hydraulics Research (HR) Wallingford Ltd. I would also like to thank all those at HR Wallingford who helped me during my interesting and informative industrial placements there.

Additionally I would like to express my gratitude to the staff of the Computing Service at RMCS, particularly Mr. C. Prowting for his endeavours in the installation and maintenance of the PHOENICS CFD software, and his assistance with the UNIX operating system. I am also grateful for the help provided by the User Support team at CHAM, particularly Dr. M. Malin, Mr J. Edwards and Mr. J. Ludwig.

For the extended time required to complete this study, some funding was provided by the School of Engineering and Applied Science, and I am grateful to Professor A. Brown for his approval.

Financial support was provided by my parents, who have also given tremendous moral support throughout. Finally I would like to thank Miss M Day who has suffered my long working hours and has helped me to keep a rational outlook during the project.

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

CONTENTS

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

NOTATION

Flow Variables

p = instantaneous pressure [N m^{-2}]

P = time averaged pressure [N m^{-2}]

P^* = "local pressure" i.e. periodically distributed pressure obtained by subtraction of linear ("reduced") pressure gradient [N m^{-2}], or approximate solution for P in pressure correction method [N m^{-2}]

P' = pressure correction [N m^{-2}]

p' = fluctuating component of turbulent pressure [N m^{-2}] *or* linear ("reduced") part of pressure distribution in fully developed periodic flow [N m^{-2}]

u, v, w = instantaneous velocity components in the x, y and z directions [m s^{-1}]

U, V, W = time averaged velocity components in the x, y and z directions [m s^{-1}] (or U_α for $\alpha = 1, 2$ or 3)

U^*, V^*, W^* = approximate solutions for U, V, W in pressure correction method [m s^{-1}]

U', V', W' = velocity corrections in pressure correction method [m s^{-1}]

u', v', w' = turbulent fluctuating components of velocity in the x, y and z directions [m s^{-1}]

$\boldsymbol{u}$ = instantaneous velocity vector [m s^{-1}]

$\boldsymbol{U}$ = time averaged velocity vector [m s^{-1}]

$\boldsymbol{u}'$ = turbulent fluctuating component of velocity vector [m s^{-1}]

$\overline{U}$ = cross-sectional average velocity [m s^{-1}]

$\hat{U}$ = maximum velocity [m s^{-1}]

General Latin Symbols

A = cross-sectional area of duct [m^2] *or* area of cell face [m^2] *or* coefficient in universal velocity profile *or* coefficient in universal k_s equation

a = coefficient in finite volume equations [kg s^{-1}] *or* coefficient in resistance equation

B = coefficient in universal velocity profile *or* coefficient in universal k_s equation

b = source in finite volume equations [(units of ϕ) × kg s^{-1}] *or* duct semi-width [m] *or* coefficient in resistance equation

C = Chezy coefficient [$m^{1/2}$ s^{-1}] *or* PHOENICS boundary condition coefficient [kg m^{-3} s^{-1}]

d = pipe diameter [m]

d^U = diffusion coefficient for U velocity [kg s^{-1}]

E = roughness parameter used in law of the wall

f = friction factor defined by $8\tau / \rho \overline{U}^2$

g = acceleration due to gravity [m s^{-2}]

h = height of duct [m] *or* depth of flow [m]

Δh_f = head loss due to friction in a pipe [m]

J = total (convection and diffusion) momentum flux [kg m^{-1} s^{-2}]

k = turbulent kinetic energy [J kg^{-1}]

k_0 = roughness length scale [m]

k_h = height of roughness element [m]

k_l = length of roughness element [m]

k_s = Nikuradse sand grain size [m]

$k_{s\infty}$ = Nikuradse sand grain size at infinite flow depth (small scale roughness) [m]

l = period length for roughness pattern [m] *or* streamwise length scale [m]

l_m = mixing length [m]

l_r = length of separated flow region [m]

m = coefficient in universal k_s equation

m^U = mass flow rate for U velocity [kg s^{-1}]

n = Mannings roughness coefficient [s $m^{-1/3}$] *or* coefficient in universal k_s equation

P_w = wetted perimeter of open channel [m]

q = volume flow rate per unit width [m^2 s^{-1}]

R = radius of pipe [m]

$R_h = A/P_w$ = hydraulic radius [m]

S = source in general differential equation [(units of ϕ) × kg s^{-1}]

$\boldsymbol{S}$ = source per unit mass in general differential equation [(units of ϕ) × s^{-1}]

s = spacing between roughness elements [m]

S_f = friction slope

$\%SF_w$ = shear force on the side wall of a smooth rectangular duct as a percentage of the total perimeter shear force

T = temperature [°C]

$T_{O\text{-}O}$ = moment or torque about axis O-O [Nm]

t = time [s]

$T_{\alpha\beta}$ = Reynolds stress [N m^{-2}]

$U_\tau = \sqrt{\tau_w/\rho}$ = shear velocity [m s^{-1}]

V = PHOENICS boundary condition value [units of ϕ]

(NB: also used for time averaged velocity component in the y direction)

y_0 = rough wall datum level measured from the base of the roughness elements, i.e. from the plane $y = 0$ [m]

General Greek Symbols

Γ = exchange coefficient [kg m^{-1} s^{-1}]

δ = boundary layer thickness [m]

$\delta_{\alpha\beta}$ = Kronecker delta operator; $\delta_{\alpha\beta} = 1$ for $\alpha = \beta$; $\delta_{\alpha\beta} = 0$ for $\alpha \neq \beta$

ε = dissipation rate of turbulent kinetic energy [J kg^{-1} s^{-1}]

κ = von Karman's constant

$\mu_{eff} = \mu + \mu_t$ = effective viscosity [kg $m^{-1}s^{-1}$]

μ_t = eddy viscosity [kg $m^{-1}s^{-1}$]

μ = dynamic viscosity [kg m^{-1} s^{-1}]

ν = kinematic viscosity [m^2 s^{-1}] = μ/ρ

Π = profile parameter for Coles' wake function

ρ = density [kg m^{-3}]

σ = normal stress [N m^{-2}]

τ = shear stress [N m^{-2}]

Φ = approximate solution of general variable

ϕ = general specific variable for a conserved property

ω = Coles' wake function

Geometric Variables

x = streamwise co-ordinate

y = vertical transverse co-ordinate

z = horizontal transverse co-ordinate

P = "present" cell centre

N, S, E, W, H, L= cell centres to the north, south, east, west, high and low sides of the "present" cell

Subscripts / Superscripts

$_{n, s, e, w, h, l}$ = north, south, east, west, high and low face of cell P

$_{nb}$ = neighbouring cells

$_{up}$ = upstream face

$_{dwn}$ = downstream face

$_{w}$ = at a wall

$_{mf}$ = main flow region, i.e. from the top of the roughness elements to the top of the duct.

$_{g.m.l.}$ = geometric mean level of the roughness elements

$_{e.s.m.}$ = datum level obtained by equating streamwise force moments

$_{l.v.p.}$ = logarithmic velocity profile

$_{t}$ = turbulent

$_{l}$ = laminar

$_{\infty}$ = free stream

Dimensionless numbers

Re = Reynolds number based on depth of flow = $\frac{Uh}{\nu}$

$\mathbf{Re_r}$ = roughness Reynolds number = $\frac{U_\tau k_s}{\nu}$

P = Peclet number = $\frac{\rho U l}{\Gamma}$ where l is the length scale of a computational cell

Relative depth = $\frac{h}{k_h}$

Y^+ = Non-dimensional distance from the wall = $U_\tau y / \nu$

U^+ = Non-dimensional velocity = U / U_τ

PHOENICS Co-ordinate System and Grid Nomenclature

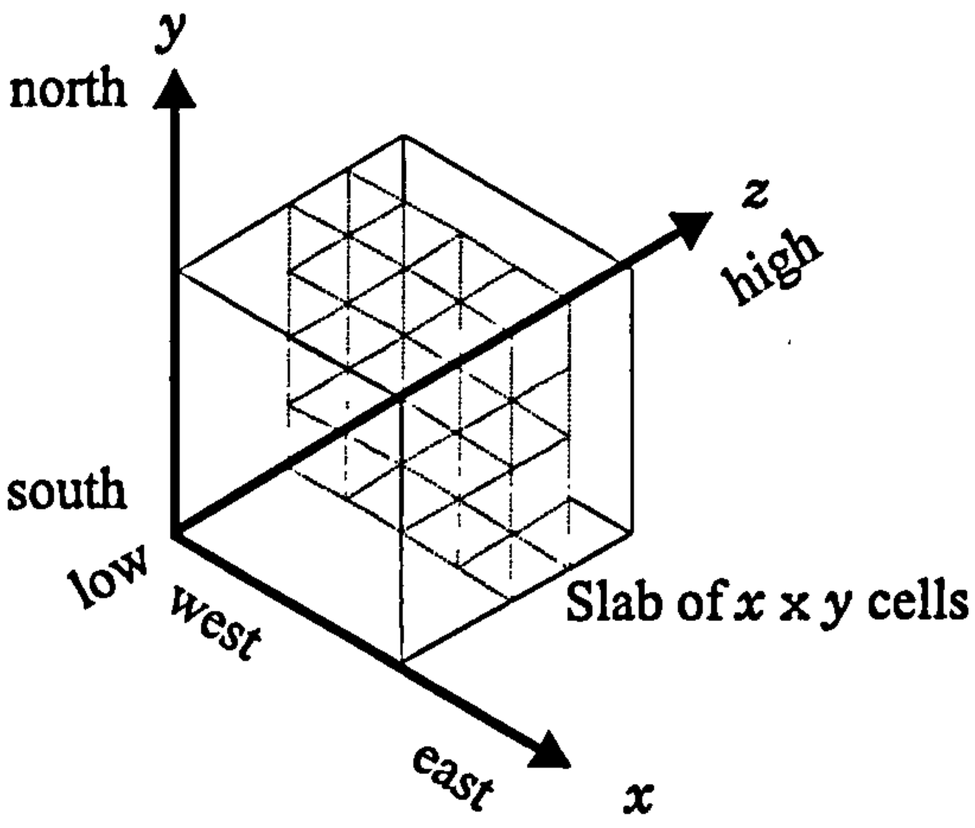


Figure (i)*: PHOENICS co-ordinate system*

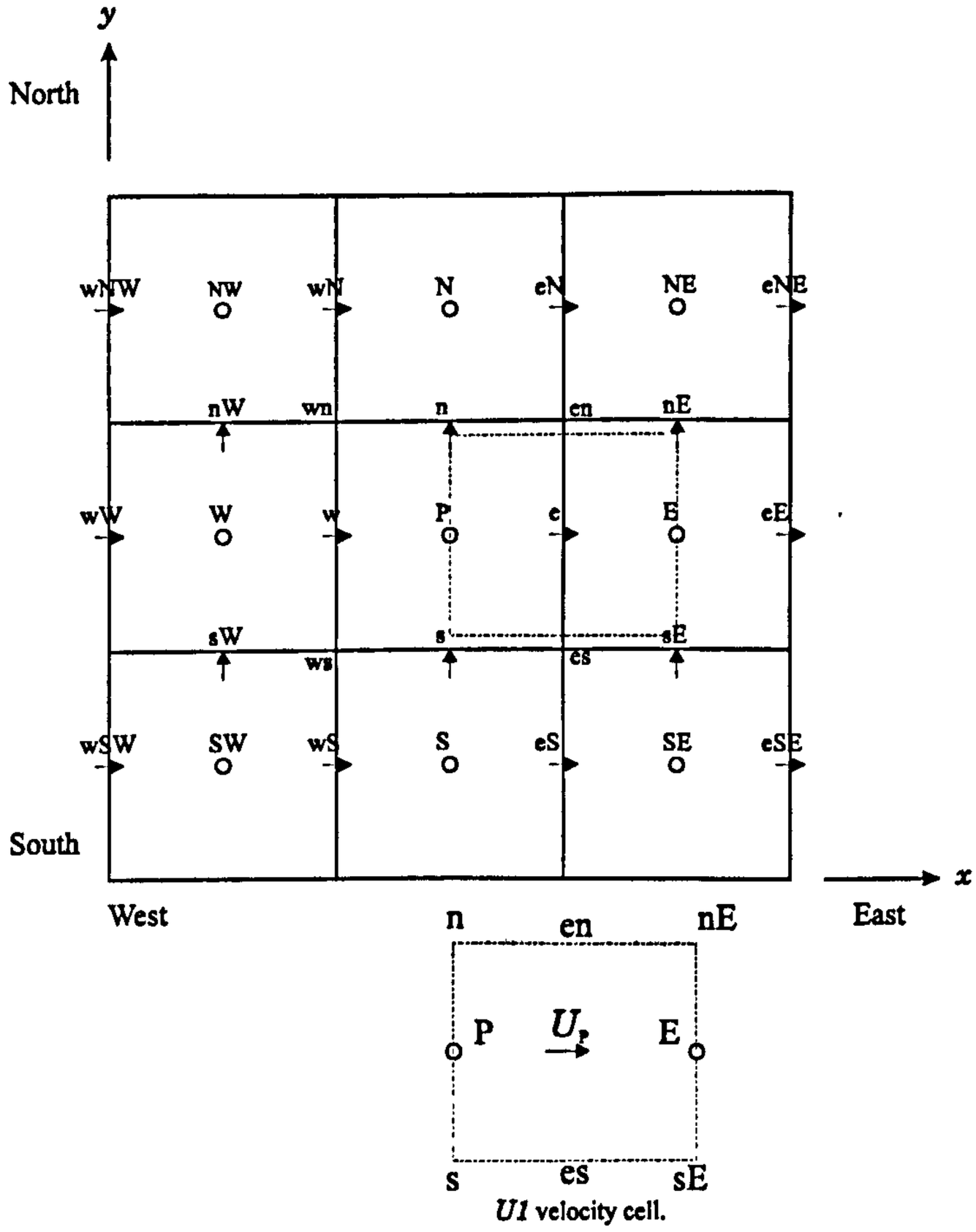


Figure (ii)*: Grid nomenclature for the x-y plane*

Common PHOENICS Variables

P1 = first phase time averaged pressure [Pa]

U1, V1, W1 = first phase time averaged velocity in the x, y and z directions [m s^{-1}]

KE= kinetic energy of turbulence

EP = dissipation rate of turbulence

RHO1 = density of first phase

NX = total number of cells in the x direction

NY = total number of cells in the y direction

NZ = total number of cells in the z direction

Convention for Distances

The distance between the nodal points P and E is expressed as the modulus of the vector |PE| and similarly the distance between the west and east face of cell P is |we|. Between the points e and eE the convention |e(eE)| is used.

Convention for Stress Components

A positive stress corresponds to a positive force acting on the positive face of a control volume or alternatively a negative force acting on the negative face of a control volume. The first subscript denotes the direction of the normal to the surface on which the force acts and the second subscript denotes the direction in which the force acts. Note that τ_{yy} and τ_{xx} are direct stresses and may be written as σ_y and σ_x .

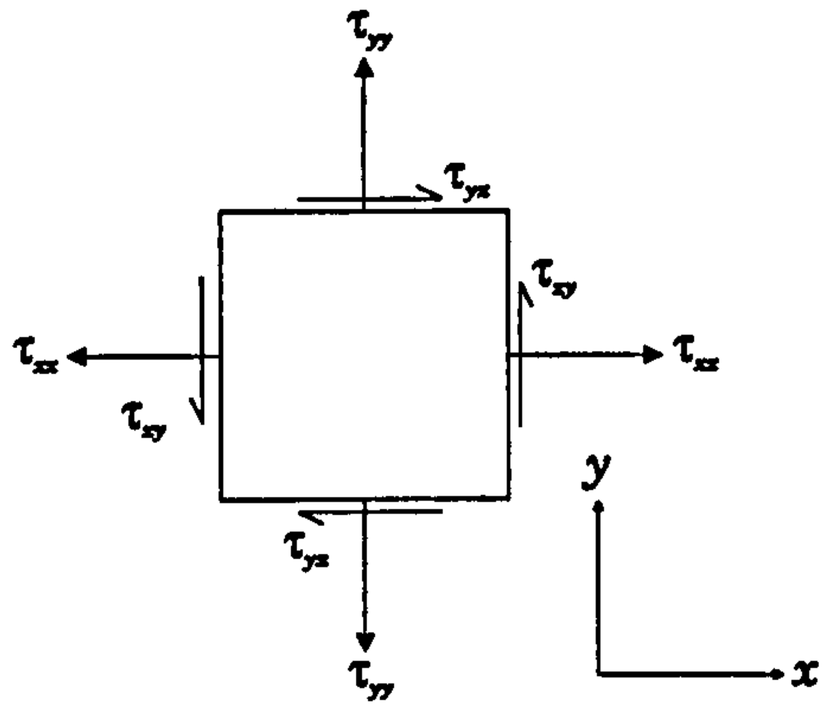


Figure (iii)*: Two-dimensional stress components*

1. INTRODUCTION

Fundamental to river management is the ability to predict the water level in a river or channel for a given flow rate, bed slope, cross-sectional geometry and channel roughness. The variation of flow rate (discharge) with water level (stage), termed the stage-discharge relationship, is essential information for the vital task of flood warning. In the UK, storm weather forecasting and a knowledge of the stage-discharge relationships for the local catchment area form the leading part of the new flood warning system to be implemented following recommendations by the Ministry of Agriculture, Fisheries and Food (MAFF) (Pritchard 1996). Due to increased media attention in the past decade (e.g. flooding of the Mississippi, USA, *Economist magazine* 1993), flood forecasting has become increasingly high profile, focusing public attention on the vital role of the hydraulic engineer.

The stage-discharge relationship in a river is governed by the resistive forces acting on the channel bed and banks. These forces are also responsible for the process of erosion which is a major problem for rivers world-wide (e.g. Lawler 1994). Boundary forces are the primary production mechanism for turbulent mixing, which controls the cross-stream transport of sediment, heat, chemicals and other pollutants. The prediction of pollutant concentrations plays a large part in the management of effluent discharges in our urbanised and industrial society. A knowledge of the boundary forces which occur in river channels is also essential to the design of new waterways, the scope of which may vary from re-routing natural rivers to the design of new irrigation canals, including the development of recreational water facilities (e.g. white water kayak courses, Cetina and Rajar 1993).

Tighter restrictions have been imposed on all new waterways under the Wildlife and Countryside Act of 1981 in which the National Rivers Authority (now the Environment Agency) and internal drainage boards have a duty to *further* the conservation of wildlife and the landscape, so far as is consistent with the primary function of the channel. Conservation of natural features (e.g. meanders, pools and riffles) increases the

complexity of hydraulic calculations by making the channel form less regular. Examples of these features are included in the three recent river restoration projects reported by Vivash *et al* (1996).

1.1 Determining Resistance

The determination of resistance to flow in river channels is complicated by two factors:

1. Turbulence. A fundamental problem shared by all turbulent flows is that the mathematical representation of the fluid mechanics (the Navier Stokes and continuity equations) cannot be solved analytically. Direct numerical solutions can now be obtained for simple low Reynolds number flows using the largest computers available. However the ability to use direct numerical simulation on flows typically found in river channels is unlikely in the foreseeable future.

2. Roughness irregularity. It is doubtful whether a single parameter can be used to represent the extreme range of roughness elements which may occur in river flows. Such a parameter would need to account for factors such as the shape, size, distribution and surface texture of the roughness elements, which may take the form of solid rock, natural vegetation or debris. The scale of roughness elements may vary from small sand grains to large rocks. The channels which contain these roughness elements may vary from small streams to rivers many metres deep.

However, empirical methods relating the flow rate, cross-sectional geometry and bed slope via a resistance coefficient have been in use since the 18th century. These include Manning's equation and the Chezy equation which are still used today (described in Chapter 2). The resistance coefficients employed in these equations are determined in the field from personal judgement by trained and experienced engineers, aided by tables and photographic databases (e.g. French 1986). They depend on the Reynolds number of the flow, boundary roughness and the shape of the channel cross-section. Errors in predicting the resistance coefficients of ±15 % are quoted by Aguire-Pe and Fyentes (1990) and French (1986).

Alternatively, the resistance equation may be obtained by the Darcy-Weisbach equation and boundary layer analysis. Provided that the roughness elements are of small scale in comparison with the flow depth, the bed generated turbulence creates a mean flow structure adequately described by the theoretically based logarithmic law of the wall and the velocity defect law. Integration of the latter leads to a resistance equation (e.g. Schlichting 1960) for the friction factor f.

By comparing velocity profiles over various roughness configurations with the standard profile produced by Nikuradse (1933) for flow over sand grains, an equivalent sand grain diameter (k_s) may be determined. This length scale can be used as a measure of resistance and is taken to be independent of flow depth, provided that the roughness is of small scale. However, when the roughness length scale is not small compared with the depth of flow, the assumptions on which this theory is based are no longer valid. Given that large scale boundary roughness is frequently encountered in natural river flows, this breakdown of classical rough boundary layer theory makes the prediction of channel resistance problematical. For this reason, the prediction of resistance in such flows is a potentially fruitful area of research and is the subject of the study reported here.

The study of fluid flow over large scale roughness elements is also applicable to various other facets of engineering. Similar flows are encountered in channels transporting a wide variety of fluids through factories and process plants. Boundary layer flows which obey the same resistance laws include air cooled integrated circuit boards and air flow over crops (e.g. Paeschke 1937).

1.2 Aim of Project

The primary aim of this study was to investigate the application of boundary layer theory to obtain a resistance equation applicable to flow over depth scale roughness, i.e. in which the size of the roughness elements are of the same order of magnitude as the depth of flow.

1.3 Methodology

Numerical analysis is becoming increasingly popular as a tool for simulating fluid flow and in some cases may have advantages over physical modelling in the laboratory (Patankar 1980). In particular, computational modelling is becoming more cost effective and ever faster solutions on higher density grids are becoming available through progress made in hardware and software design. The ability to evaluate flow variables that are not easily obtainable by physical measurement was considered to be important in the present work, in which the distribution of local boundary shear stress and static pressure was required. Although a computational model is no better than the mathematical representation of the fluid mechanics on which it depends and the results have to be validated against experimental studies, in the present application it was considered that even if the computational model fell short of a completely faithful reproduction of the physical fluid dynamics, this drawback would be compensated by the degree of consistency attainable from one application to the next.

It was for these reasons that this investigation was performed using a computational fluid dynamics (CFD) code based on the Reynolds averaged equations in conjunction with a turbulence model. A commercially available program was employed to eliminate the time required to develop and write the CFD code, which would have simply replicated the commercially used techniques. Roughness elements were chosen to be of a simple geometric shape, periodically distributed in the streamwise direction. The possibility of modelling the complexity of the free surface boundary condition was considered but for the present application it was decided that a simple plane of symmetry would be sufficient.

To validate the numerical model, simulations were performed in which the results were compared with experimental studies on flow in three-dimensional smooth ducts and two-dimensional roughness configurations.

New criteria were used to define numerical convergence, based on analysis of the hydrodynamic forces acting within the fluid and on the boundaries. Analysis of hydrodynamic forces also enabled streamwise force moments to be used in a novel

prediction method to determine the datum position of the rough wall. This was an integral part of a proposed new method of predicting resistance over depth scale roughness.

1.4 Organisation of Thesis

A review of previous research into hydraulic resistance is presented in Chapter 2 to give the theoretical background to the main part of this study. The general theory of Reynolds averaged numerical solutions is given in Section 2-2, with the particular techniques employed by the computational model PHOENICS (CHAM Ltd.) presented in Section 2-3. For readers particularly interested in the PHOENICS code, a brief guide is given in Appendix 1.

During the course of this study, several theoretical developments were carried out to aid computation and analysis. These included the implementation of periodic boundary conditions, the calculation of period averaged transverse shear stress, a force balance within the domain, various methods of calculating k_s values, and a novel method to determine the rough wall datum position by consideration of streamwise moments. These techniques are presented in Chapter 3.

Before commencing on the study of depth scale roughness, the turbulence model employed was tested against the physical experiments of Schlichting (1936). This work is set out in Chapter 4 together with additional calculations carried out on the roughness configuration of Grass *et al* (1991). A series of simulations over rough boundaries in which the depth of flow was varied from very deep to very shallow is presented in Chapter 5, together with the analysis which leads to a relationship between the roughness length scale k_s and the depth of flow.

Chapter 6 discusses two other applications of the numerical methods used within this study, namely three-dimensional flow in a smooth rectangular duct, and laminar flow in a counterflow plate-fin heat exchanger. The smooth rectangular duct calculations were in fact conducted early on in the study with the primary aim of performing simple

simulations to validate the computer code. However the opportunity was taken to examine the distribution of boundary shear stress and the effect of the side wall with varying duct aspect ratios. The heat transfer in a plate-fin heat exchanger was conducted as a feasibility study for a potential commercial application.

Conclusions drawn from each of the investigations reported are presented in Chapter 7.

For the reader's convenience, descriptive figures are located within the text, and figures illustrating results are presented at the end of each chapter. The notation used within this thesis is set out at the beginning of the document.

Throughout the thesis an *italic* font is used for mathematical variables, and an ARIAL font is used for variables employed in the computer code.

1.5 Using Commercial Software

This concluding note is addressed to the reader who may be unacquainted with the difficulties incurred by the application of commercial CFD codes. It is offered with the intention of correcting the possible impression that using PHOENICS or an equivalent package entails no more than lifting a model "off the shelf" and simply running it.

No doubt this practice occurs, and may be common place among inexperienced users especially in industry, but in the author's view it is fraught with danger. It is relatively easy to model a flow and produce a plausible output, especially when judged on the appearance of the velocity distribution, but in other respects the results may be severely flawed.

A large proportion of this project period was spent by the author in understanding the numerical methods of PHOENICS and painstakingly developing the procedures set out in Chapter 3. Great emphasis was placed on checking every solution for the conservation of momentum flux, ensuring that the source representing the pressure gradient was balanced by the sink representing the resistance at the boundary.

These procedures had to be designed and programmed by the user in the PHOENICS subroutine called GROUND. The author accepts full responsibility for this work, and of course for the analysis of the computational results set out in Chapters 4 to 7.

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

2. THEORETICAL BACKGROUND AND LITERATURE REVIEW

2.1 Hydraulic Resistance

The study of hydraulic resistance is essentially concerned with defining the relationship between the flow rate carried by a channel and the corresponding boundary shear stress developed in it. A comprehensive solution to this problem has not yet been found for turbulent flow over rough boundaries, despite the large amount of research dedicated to the subject.

2.1.1 Empirical Resistance Equations

In open channels the flow rate may be related to the bed slope and the hydraulic radius, and various equations of the form

$$\overline{U} = a R_h{}^b S_f{}^{1/2} \qquad \textit{Equation 2-1}$$

have been presented, where R_h is the hydraulic radius, S_f is the friction slope (equal to the bed slope for uniform flow), b is a constant and a is a resistance coefficient. The Chezy equation follows from dimensional analysis in which $\tau \propto \rho U^2$, giving

$$\overline{U} = C R_h{}^{1/2} S_f{}^{1/2} \qquad \textit{Equation 2-2}$$

where C is the Chezy coefficient which has to be determined empirically.

The Manning equation, developed in 1889 is another empirical formula resulting from a curve fitting process to experimental data:

$$\overline{U} = \frac{1}{n} R_h{}^{\frac{2}{3}} S_f{}^{1/2} \qquad \textit{Equation 2-3}$$

Since the Chezy and Manning equations describe the same phenomena, the coefficients C and n are related:

$$C = \frac{1}{n} R_h^{\frac{1}{6}}$$ *Equation 2-4*

For turbulent flow in a pipe, Darcy (1803 to 1858) found the head loss due to friction to be

$$h_f = \frac{\delta P}{\rho g} = \frac{f \delta x \overline{U}^2}{2 d g}$$ *Equation 2-5*

where f is the Darcy-Weisbach friction factor[1], δx is the length of the pipe and d is the diameter. The wall shear stress may be related to f by using

$$\tau_w = R_h \frac{dP}{dx}$$ *Equation 2-6*

where $R_h = A/P_w$, which is equal to $d/4$ for a pipe (in which A is the cross-sectional area and P_w is the perimeter of the pipe). Equation 2-5 may then be written as

$$\frac{\tau_w}{R_h} = \frac{dP}{dx} = \frac{f \rho \overline{U}^2}{2d}$$ *Equation 2-7*

or

$$f = \frac{8\tau_w}{\rho \overline{U}^2}$$ *Equation 2-8*

The friction factor can also be used in open channels by relating the friction slope S_f to the bed shear stress:

$$\tau_w = \rho g R_h S_f$$ *Equation 2-9*

from which Equation 2-8 may be written as:

$$S_f = \frac{f \overline{U}^2}{8 g R_h}$$ *Equation 2-10*

The relationship between f and n, and between f and C, may then also be found:

$$n = R_h^{1/6} \sqrt{\frac{f}{8g}}$$ *Equation 2-11*

---

[1] For convenience f will simply be termed as the friction factor for the remainder of this thesis.

$$C = \sqrt{\frac{8g}{f}}$$ *Equation 2-12*

To complete the representation of resistance, the coefficients C, n or f need to be determined for each channel or duct. These may be obtained by charts, for example the Moody chart (e.g. Massey 1989), tables or data bases. French (1986) presents three methods for evaluating Manning's n for open channel flow and quotes an accuracy in determining the flow rate of ± 15%. The choice of C, n or f is subject to personal preference, though some authors have found advantages of one over the other. Sayre and Albertson (1963) show the Chezy C to be accurate over a wider range of roughness than Manning's n, which has been used by numerous authors (e.g. Coon, 1994, Jha *et al*, 1994, Einstein and Barbarossa, 1952 and Miller *et al*, 1994). In the review by the ASCE task force (1961) the use of Manning's n is only advised for "fully rough channels", for which n is nearly constant. However the use of f is recommended for most situations because more correlated data exists.

2.1.2 The Universal Velocity Profile

Resistive forces are transmitted from the boundaries through the fluid by the action of viscous and turbulent shear. These shear layers are directly related to the velocity profile, shown in Figure 2-1 for turbulent flow over a smooth bed. Analysis of velocity profiles leads to a more theoretically sound method of determining resistance.

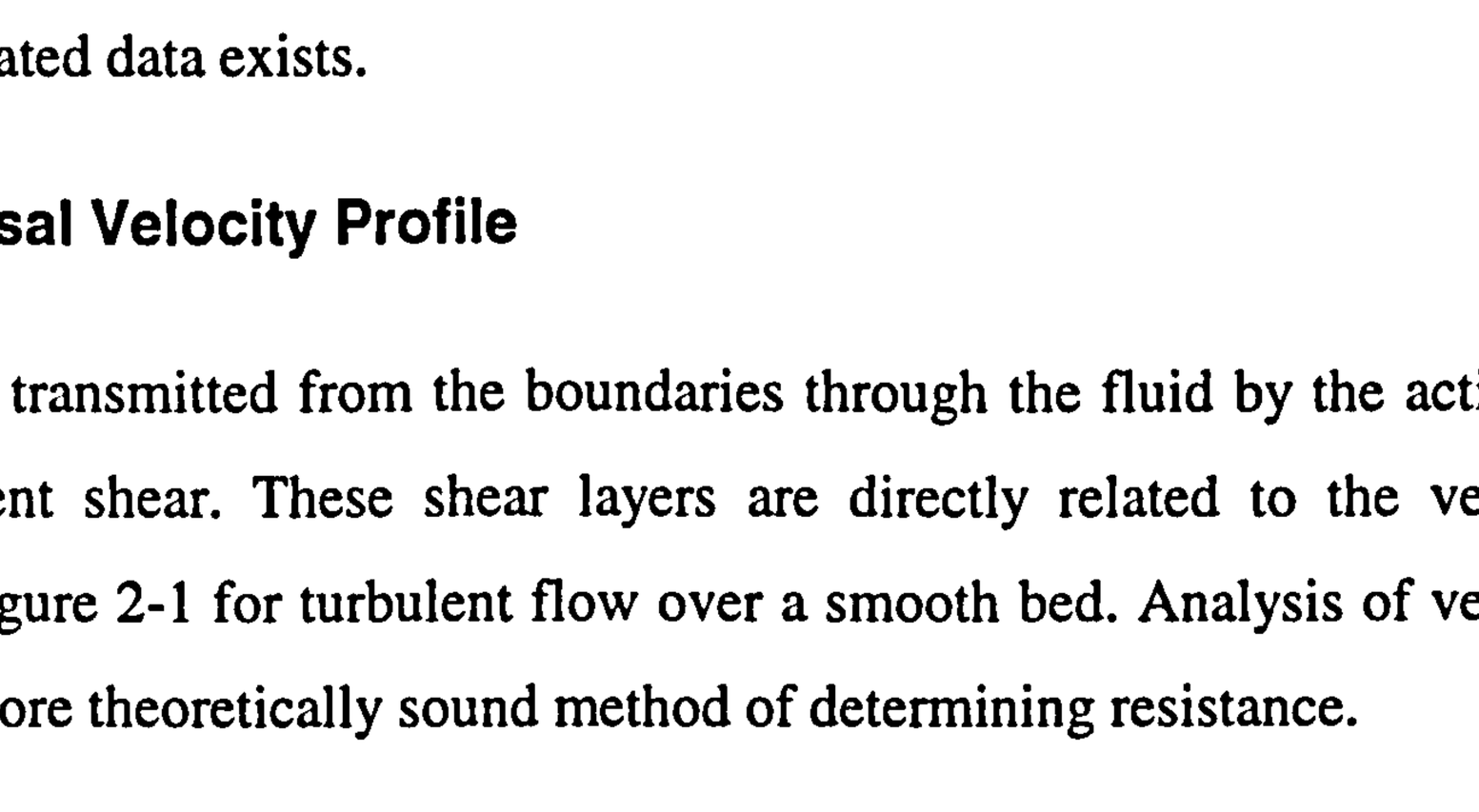


Figure 2-1: Typical smooth wall turbulent velocity profile

In the viscous region adjacent to the solid boundary Newton's law of viscosity gives

$$\tau = \mu \frac{dU}{dy}$$ *Equation 2-13*

which on integration and assuming a constant shear stress results in

$$\frac{U}{U_\tau} = \frac{U_\tau y}{\nu}$$ *Equation 2-14*

The velocity profile in turbulent flow may be derived using Prandtl's mixing length hypothesis to describe turbulent shear (τ_t):

$$\tau_t = \rho l_m^2 \left|\frac{d\overline{U}}{dy}\right| \frac{d\overline{U}}{dy}$$ *Equation 2-15*

Assuming that in the region near a wall, the mixing length l_m is proportional to the distance from the wall

$$l_m = \kappa y$$ *Equation 2-16*

and that the shear stress is constant, $\tau = \tau_w$ then (e.g. Schlichting 1960)

$$U = {}^{1}\!/_{\kappa}\, U_\tau \ln y + c$$ *Equation 2-17*

In spite of the fact that these assumptions are valid only in the near wall region of a zero pressure gradient boundary layer, Equation 2-17 is commonly applied over the complete depth of a fully developed channel (h) or pipe of radius R, resulting in the velocity defect law:

$$\frac{\hat{U} - U}{U_\tau} = \frac{1}{\kappa} \ln \frac{h}{y}$$ *Equation 2-18*

where for this case h is the depth of flow in a channel. Alternatively the constant of integration, c, in Equation 2-17 may be evaluated from the condition that $U = 0$ at a certain distance from the wall, which is of the same order of magnitude as the thickness of the laminar sub-layer. This leads to the well known law of the wall

$$\frac{U}{U_\tau} = {}^{1}\!/_{\kappa}\, \ln \frac{U_\tau y}{\nu} - {}^{1}\!/_{\kappa} \ln \beta$$ *Equation 2-19*

The constant of proportionality, κ in Equation 2-16 is von Karman's constant, generally taken to be 0.4. The coefficient β in Equation 2-19 depends on the nature of the

boundary surface and, for flow in smooth walled pipes, was found to be 0.111 by Nikuradse (1933). Thus for flow over a smooth boundary the universal velocity distribution is

$$\frac{U}{U_\tau} = 5.5 + 2.5 \ln\left(\frac{U_\tau y}{\nu}\right) \qquad \textit{Equation 2-20}$$

or generally

$$\frac{U}{U_\tau} = A + B \log\left(\frac{U_\tau y}{\nu}\right) \qquad \textit{Equation 2-21}$$

where $A = 5.5$ and $B = 5.75 \left(= {}^{1}\!/_{\kappa} \times {}^{1}\!/_{\log e}\right)$. This is shown in Figure 2-2 together with the velocity distribution in the laminar sub layer.

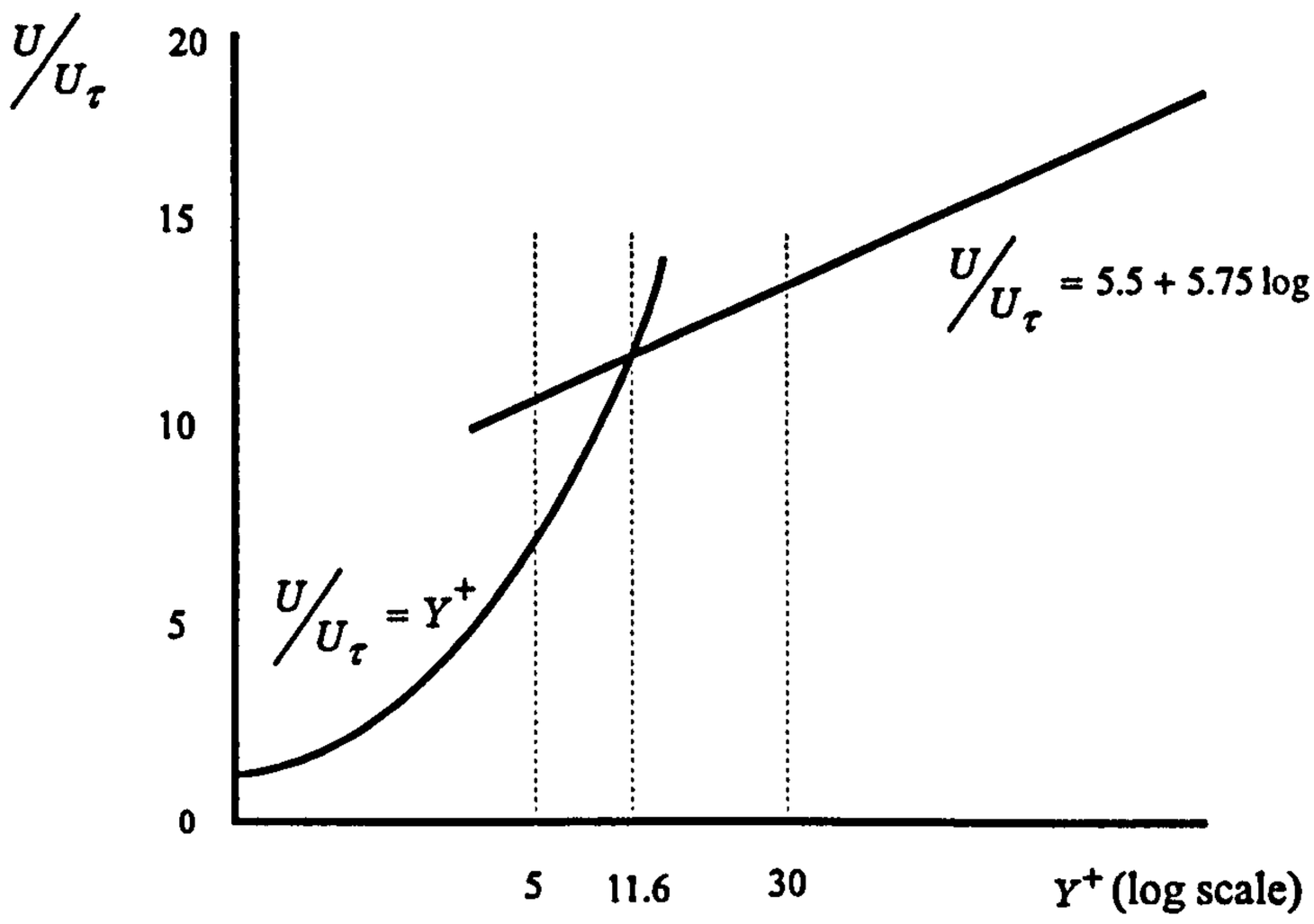


Figure 2-2: Universal velocity profile

Using the relationship between f and the wall shear stress

$$f = {}^{8\tau_w}\!/_{\rho\overline{U}^2} \qquad \textit{Equation 2-22}$$

then integration of the velocity defect law (Equation 2-18) leads to the Prandtl universal resistance law (e.g. Schlichting 1960)

$$\frac{1}{\sqrt{f}} = 2.0 \log\left(\mathbf{Re}\sqrt{f}\right) - 0.8 \qquad \textit{Equation 2-23}$$

which may also be written as:

$$\frac{1}{\sqrt{f}} = 2.0 \log\left(\mathbf{Re}\sqrt{f}/2.51\right)$$ *Equation 2-24*

2.1.3 Rough Walls

For boundaries of increasing roughness the coefficient A in Equation 2-21 is known to decrease. Hama (1954) describes the action of the roughness as being equivalent to a velocity reduction (ΔU);

$$\frac{U}{U_\tau} = B \log\left(\frac{U_\tau (y - y_0)}{\nu}\right) + A - \frac{\Delta U}{U_\tau}$$ *Equation 2-25*

where A and B are the coefficients used in the smooth wall relationship and y_0 is the datum position of the rough wall. Perry and Joubert (1963) note that the effect of the roughness is similar to viscosity and is confined to a thin region adjoining the surface. The roughness effect can therefore be accounted for by using a modified coefficient of viscosity, ν_e;

$$\frac{U}{U_\tau} = B \log\left(\frac{U_\tau (y - y_0)}{\nu_e}\right) + A$$ *Equation 2-26*

Equation 2-19 may also be written in the form (Jayatilleke 1969)

$$U/U_\tau = 1/\kappa \ln\left(\frac{U_\tau y E}{\nu}\right)$$ *Equation 2-27*

where E is a roughness parameter. For smooth walls, E = 8.6 and for rough boundaries E is

$$E = 29.7/\mathbf{Re}_r$$ *Equation 2-28*

where $\mathbf{Re}_r$ is the roughness Reynolds number $(k_s U_\tau/\nu)$. The roughness length scale (k_s) is the equivalent sand grain roughness, described below.

As an alternative to using $\frac{U_\tau y}{\nu}$, Hosni, Coleman and Taylor (1993) suggest that the y co-ordinate should be made non-dimensional with respect to the momentum thickness. For the same roughness configurations, plots of U/U_τ against y/δ_{mt} do indeed show a single curve for all values of Reynolds number. For different roughness patterns

however the presented data points are coincident at high values of y, but the correlation is not maintained closer to the wall.

In Nikuradse's (1933) famous experiments on pipes uniformly roughened with sand grains, the friction factor was found to be a function of Reynolds number and k_s/d only, where k_s was the diameter of the sand grains and d was the pipe diameter. At higher Reynolds numbers the flow become independent of viscosity and was a function of k_s/d alone. Since

$$\frac{(y-y_0)U_\tau}{\nu} \equiv \frac{(y-y_0)}{k_s}\frac{U_\tau k_s}{\nu} \qquad \textit{Equation 2-29}$$

then the velocity distribution for a pipe in the fully rough regime was written as:

$$\frac{U}{U_\tau} = A + B\log\left(\frac{(y-y_0)}{k_s}\right) \qquad \textit{Equation 2-30}$$

where A now differs from that used in Equation 2-21 and is a characteristic function of the roughness concerned. For the particular case of Nikuradse's sand grain experiment, the constant A in Equation 2-30 was found to be 8.48, the value of B remaining at 5.75.

Taylor *et al* (1985) and Grass *et al* (1991) use the logarithmic velocity distribution in the form (using current notation):

$$\frac{U}{U_\tau} = B\log\frac{y-y_0}{k_0} \qquad \textit{Equation 2-31}$$

in which the coefficient A of Equation 2-30 is incorporated within the general roughness length scale (k_0). As expected, Monin and Yaglom (1971) found this to be a constant within the logarithmic region.

Granville and Taylor (1984) refer to the two different forms of the universal velocity profile as 'Reynolds number mode' (Equation 2-21) and 'relative roughness mode' (Equation 2-30).

A resistance law for rough pipes may be derived in a similar manner to Equation 2-23 (e.g. Schlichting 1960)

$$\frac{1}{\sqrt{f}} = 2.0 \log\left(R/k_s\right) - a \qquad \textit{Equation 2-32}$$

where R is the radius of the pipe and the constant a also depends on the roughness Reynolds number. For Nikuradse's sand grain experiments, a was found to be 1.74 at high Reynolds numbers.

2.1.4 Equivalent Sand Roughness

Schlichting (1936) proposed that flow in ducts of unknown roughness could be calculated through the use of an equivalent sand roughness parameter k_s whereby the resistance of a duct under investigation is equal to an equivalent duct lined with sand grains of diameter k_s. Any small scale roughness could then be given a k_s value and the velocity profile described by

$$\frac{U}{U_\tau} = 8.48 + 5.75 \log\left(\frac{(y - y_0)}{k_s}\right) \qquad \textit{Equation 2-33}$$

The friction factor was then calculated knowing the equivalent sand roughness from

$$\frac{1}{\sqrt{f}} = 2.0 \log\left(R/k_s\right) - 1.74 \qquad \textit{Equation 2-34}$$

Colebrook (1939) combined the smooth wall resistance law (Equation 2-23) with Equation 2-34 to give the Colebrook-White equation for transitional flow

$$\frac{1}{\sqrt{f}} = -2.0 \log\left(\frac{k_s}{7.4R} + \frac{2.51}{\mathbf{Re}\sqrt{f}}\right) \qquad \textit{Equation 2-35}$$

A discussion on the merits of the Colebrook-White equation is given by Matthew (1990). It is used extensively for predicting f but its implicit nature means that a numerical solution is required. Haaland (1983) has reviewed some approximate explicit formulae and recommended

$$\frac{1}{\sqrt{f}} = -1.8 \log\left[\left(k_s/7.4R\right)^{1.11} + 6.9/\mathbf{Re}\right] \qquad \textit{Equation 2-36}$$

The resistance equation for rough flow, Equation 2-34 was also modified by French (1986) for use in open channels

$$\frac{1}{\sqrt{f}} = 2.0 \log\left(\frac{12R_h}{k_s}\right)$$ *Equation 2-37*

In his review of channel flow, Ackers (1958) concluded that the Colebrook-White equation was the best available method for determining flow rate, however he suggested that the k_s value for channel flow should be increased by approximately 20% over the value used in pipes. The k_s value for a surface is taken to be constant at high Reynolds numbers and may be obtained from charts for various materials commonly found in open channel and engineering flows (e.g. French 1986 or Massey 1989) or from analysis of the velocity profile (e.g. Knight, Alhamid and Yuen 1992, Kumar and Roberson 1980 and Schlichting 1936).

When used as an equivalent sand roughness the k_s parameter is simply a roughness length scale which has no physical interpretation and should be regarded in the same manner as the general roughness length scale k_0 employed by Taylor *et al* (1985) and Grass *et al* (1991) (Equation 2-31). Indeed the two length scales are related by equating Equation 2-31(assuming B = 5.75) to Equation 2-33 which leads to

$$\frac{k_0}{k_s} = 10^{(-8.48/5.75)} \approx 1/30$$ *Equation 2-38*

In order to obtain k_s for practical purposes various correlations of k_s as a function of the roughness geometry have been proposed. Keulegan (1938) suggested

$$5.75 \log k_s = 8.5 - c + 5.75 \log k_h$$ *Equation 2-39*

however the constant c is still unknown. Kironoto and Graf (1994), Miller and Wenzel (1985) and Parker and Peterson (1980) calculated k_s from a representative diameter of random roughness elements, using $k_s = d_{50}$, d_{90} and $2d_{90}$ respectively. (d_n is the diameter of bed particles for which $n\%$ are finer.) The k_s parameter has also been related to the Chezy resistance coefficient (Aguire-Pe and Fyentes 1990) and the Manning's n (Einstein and Barbarossa 1952).

The k_s parameter is now widely used for all types of flow and is increasingly used as the roughness parameter in computational models (e.g. Cebeci and Chang 1978, Christoph 1981, Miller and Wenzel 1985 , Johns, Soulsby and Xing 1993).

2.1.5 Alternatives to the Universal Velocity Profile

The derivation of the Prandtl log law (Equation 2-17) implies that it is only valid in the near wall region. Experimental evidence (e.g. Hama 1954, Kirksoz 1989, Bergeron 1994 and Ferro and Baiamonte 1994) indicates that two regions exist in the velocity profile. The inner, or wall region consists of the viscous sub-layer and a logarithmic region (Equation 2-19). The velocity distribution is controlled by the wall shear stress, wall roughness, distance from the wall, the density and the viscosity of the fluid. The inner region extends out from the wall to a distance of 10 - 20% of the boundary layer thickness (ASCE task force 1963, Perry and Joubert 1963, Bergeron 1994, Ferro and Baiamonte 1994) or up to $Y^+ = 130$ (Jayatilleke 1969). The outer region extends inwards from the edge of the boundary layer and is described by the velocity defect law (Equation 2-18). The velocities are controlled by turbulent shearing and the depth of flow. For the high Reynolds numbers usually encountered in channel flow, the two regions overlap.

Hama (1954) uses the equations in Table 2-1 to describe the two regions.

subdivision	y range	velocity profile law
logarithmic region	$32.5 \le \frac{yU_\tau}{\nu}$ $\frac{yU_\tau}{\delta^*\hat{U}} \le 0.045$	$\frac{U}{U_\tau} = 5.6\log\frac{U_\tau y}{\nu} + 4.9$ or $\frac{\hat{U}-U}{U_\tau} = -\left(5.6\log\frac{yU_\tau}{\delta^*\hat{U}} + 0.6\right)$
outer region	$0.15 \le {}^{y}/_{\delta} \le 1.0$	$\frac{\hat{U}-U}{U_\tau} = 9.6\left(1-{}^{y}/_{\delta}\right)^2$
Where: δ = boundary layer thickness = $0.30\ \delta^*\,{}^{\hat{U}}/_{U_\tau}$ δ^* = displacement thickness		

Table 2-1: Turbulent velocity distribution after Hama (1954)

The complete turbulent boundary layer may alternatively be described using a wake function such as that given by Coles (1956)

$$\frac{U}{U_\tau} = \frac{1}{\kappa}\ln\frac{U_\tau(y-y_0)}{\nu} + B + \frac{\Pi}{\kappa}\omega\left(\frac{y-y_0}{\delta}\right) \qquad \textit{Equation 2-40}$$

For zero pressure gradient boundary layers over a smooth wall the profile parameter (Π) was calculated as 0.55 and the wake function ω as

$$\omega\left(\frac{y-y_0}{\delta}\right) = 2\sin^2\left(\frac{\pi(y-y_0)}{2\delta}\right) \qquad \textit{Equation 2-41}$$

The Coles wake function has also been found to apply in open channel flow over smooth boundaries by Nezu and Rodi (1986) and by Kirksoz (1989) using a profile parameter of 0.2 and 0.1 respectively. The lower values of profile parameter obtained in open channels was explained by Nezu and Rodi as being due to the absence of an intermittence region in open channel flow[2]. The deviation from the logarithmic law for such small values of Π is however very slight and is within the bounds of scatter in the data reported by Kirksoz.

For open channel flow over rough boundaries, Kirksoz obtained so much scatter in his data that a good estimate of the wake function was found impossible. The velocity profile over two-dimensional block roughness elements measured by Grass *et al* (1991) did however show significant deviations from the logarithmic law in the outer region, though a wake function was not applied.

Investigating flow over gravel bed rivers, Ferro and Baiamonte (1994) demonstrated the alternative wake function suggested by Dean (1978) to be more accurate than either the Coles wake function or that suggested by Finley *et al* (1966). This however was due to the more complex mathematical formulation of Dean's function which allowed the maximum streamwise velocity to be located below the free surface. This dip in the location of maximum velocity has been reported by many authors (e.g. Nezu and Rodi

---

[2] The intermittence region refers to a region within a zero pressure gradient boundary layer, near the free stream, which alternates between laminar and turbulent flow (e.g. Tritton 1977).

1985) and was explained by the existence of secondary currents in channels of narrow aspect ratio.

Two different flow regions were identified by Aguire-Pe and Fyentes (1990) for flow in steep rough streams. The first zone, close to the top of the roughness elements, contained overlapping wakes generated by the roughness elements. The velocity in this region was assumed to be uniform. Above this a logarithmic region was identified. Two similar regions were found by Bathurst (1988), however as the roughness elements became larger the velocity profile ceased to follow a logarithmic profile (Marchland, Jarret and Jones 1984), typically forming an 'S' shaped curve (Bathurst 1988 and Ferro and Baiamonte 1994). Bathurst gave the conditions for an 'S' shaped profile as

1. channel slopes to be above 1% and the ratio of depth to d_{84} to be between 1 and 4.

2. the bed material to have a non-uniform size distribution.

The latter criterion is required to provide the physical space between the major boulders for development of the lower zone.

2.1.6 Flow Classification

Morris (1954) reasoned that the generation, spreading and dissipation of vortices was the major source of frictional loss in rough flows. Because the vortices were generated at the tip of each roughness element, the longitudinal spacing of the roughness elements should be the roughness dimension of greatest importance. Two non-dimensional parameters utilising the roughness spacing were suggested to replace the Nikuradse relative roughness h/k_s:

1. roughness index $= \dfrac{\text{roughness spacing}}{\text{element height}} = \dfrac{l - k_l}{k_h}$

2. relative roughness spacing $= \dfrac{\text{depth of flow}}{\text{roughness spacing}} = \dfrac{h}{l - k_l}$

where l is the period length. Three flow regimes were then identified. With decreasing period length these were:

1. Isolated roughness: The wake zones and vortex generation zone were developed and dissipated before the next roughness element was reached. The roughness index was suggested as the relevant non-dimensional parameter.

2. Wake interference flow: Zones of separation, vortex generation and dissipation were not completely dissipated before the next roughness element was reached. The relative roughness spacing was suggested as the relevant non-dimensional parameter.

3. Quasi-smooth or skimming flow: Dead water with stable vortices was present between each roughness element. For this flow regime the roughness index was again suggested as the non-dimensional parameter.

Knight and Macdonald (1979) extended this into a six fold classification

1. Smooth turbulent flow.

2. Semi-smooth turbulent flow.

3. Non-uniform hyper-turbulent flow.

4. Uniform hyper-turbulent flow.

5. Semi-quasi-smooth flow.

6. Quasi-smooth flow.

Perry *et al* (1969) described two classes of roughness configuration from examination of the friction factor - Reynolds number relationship. The roughness configurations which followed the Nikuradse scheme, in which $f = \mathrm{F}\left(\mathrm{Re}, {}^{k_h}/_{d}\right)$, they termed '$k$' type roughness. However they noted that examples of rough flows in the quasi-smooth regime had been reported (e.g. Streeter and Chu 1949 and Ambrose 1956) for which the friction factor - Reynolds number characteristics were insensitive to the parameter k_h/d, and the roughness function was $\frac{U_\tau d}{\nu}$ rather than of $\frac{U_\tau k_h}{\nu}$. As the length scale (d) used

within the roughness function was associated with the outer flow, this appeared to be inconsistent with Nikuradse's law of the wall (Equation 2-30) and the roughness was accordingly termed 'd' type roughness.

2.1.7 Datum Levels

Moore (1951) showed that the logarithmic velocity distribution could only be applied to rough walls if a hypothetical velocity origin was located some distance (y_0) above the true bed but below the height of the roughness elements. The position of this datum level might then be considered as a measure of the interaction between the mean flow and the roughness. Perry *et al* (1969) suggested that y_0 should be related to the roughness function but pointed out that no such relation had been found.

$l - k_l / k_h$			
Raju and Garde (1970)	Adachi (1964)	Knight and Macdonald (1979)	y_0 / k_h
> 5.0	160	> 6.94	0
/	80	/	0.05
/	40	/	0.15
3.75	20	/	0.25
/	10	/	0.4
/	/	5.21	0.5
/	5	/	0.77
2.5	4.1	3.47	1.0

Table 2-2: Typical values for rough wall datum level

The most popular method of determining y_0 is to use the geometric mean level (g.m.l.) (e.g. Schlichting 1936, Simpson 1972, Ferro and Baimonte 1994). Other authors also find y_0 from the geometry of the roughness elements and the flow classification. Some examples are given in Table 2-2 and in Alhamid (1991). Morris (1954) suggested that the datum level should be located on the plane corresponding to the top of the roughness elements, using the argument that the turbulent vortices were generated at this point.

A more theoretically sound method of calculating y_0 is from analysis of the logarithmic region which results from plotting U/U_τ against log $(y - y_0)$. Clauser (1956) points out that a shift in the position of the datum plane causes the straight line logarithmic region to become curved. The correct value for y_0 should therefore be that which gives the best straight line through the data points. This "progressive origin shift" method was also employed by Perry and Joubert (1963) and Grass *et al* (1991) and in different forms by Coleman *et al* (1984) and Pyle and Novak (1981). Coleman *et al* used linear least squares regression to obtain a constant value of k_0 for his data set $y - y_0$ using the expression

$$k_0 = (y - y_0)e^{\left(-\kappa U/U_\tau\right)} \qquad \textit{Equation 2-42}$$

which is simply a re-arrangement of Equation 2-31 in the natural logarithmic form. Implicit in the linear least squares technique is the assumption that the shear velocity (based on the bed shear velocity) is constant. More accurately, the value of shear stress acting on the datum plane will vary with the location of the datum plane, requiring the use of non-linear regression techniques to obtain y_0.

Pyle and Novak found that adjusting y_0 resulted in a straight line logarithmic region with an altered slope, rather than a curve as suggested by Clauser. The datum level employed in their investigation was accordingly one which resulted in a constant slope (and therefore κ) for all the roughness concentrations investigated. The chosen value of κ was 0.285, employed because this was found to be the constant value obtained for low

roughness concentrations before the datum shift was applied. This value of κ is however significantly lower than the range typically quoted ($0.42 \geq \kappa \geq 0.32$) for flow over rough boundaries (Schlichting 1936) and suggests that a datum shift was also required for the low roughness concentrations.

Three alternative methods were used by Knight, Alhamid and Yuen (1992): the geometric mean level, a velocity datum and a shear datum. The velocity datum was found by integrating measured velocities, comparing the flow rate with an orifice plate discharge and suitably adjusting the datum. However, for each set of velocity profiles the value of y_0 obtained contained a large amount of scatter and this method was discarded. The shear datum was obtained by comparing the cross-sectional mean boundary shear stress determined from the measured velocity profiles with that determined from the energy slope. Both of these methods are however subject to large experimental errors. For example, in experimental studies on a smooth duct, for which the position of the boundary is known, Rhodes (1991) reported errors of up to 6.8% in matching the integrated boundary shear stress with the friction slope. For roughened boundaries higher errors would be expected which could lead to an error in the datum shift of the same order of magnitude as the height of the roughness elements. Rhodes reported considerably lower errors in matching the cross-section mean velocities, for which large scatter was observed in the experiments of Knight, Alhamid and Yuen.

2.1.8 Further Methods of Calculating Resistance

The resistance to flow over a roughened boundary may be classified into two processes: skin friction and form drag. Skin friction is produced by viscous shear stresses acting on surfaces parallel to the primary flow direction, while form drag is produced by direct stresses acting on surfaces which are normal to that direction. Form drag usually dominates and is characterised by flow separation and vortex shedding. It may be accounted for using a drag coefficient of the form

$$C_D = \frac{F}{\frac{1}{2}\rho U^2 A} \qquad \textit{Equation 2-43}$$

where F is the force on the element typically found by integration of the pressures over each roughness element of area A normal to the flow.

Einstein and Barbarossa (1952) calculated the total resistance to flow in natural streams by summing the form drag (Equation 2-43), for each individual large scale roughness element, and the shear force due to the underlying bed material. The latter small scale roughness was termed grain resistance following Nikuradse's experiments and was accounted for by using Manning's n as a function of k_s:

$$n = \frac{\sqrt[6]{k_s}}{29.3} \qquad \textit{Equation 2-44}$$

Shen *et al* (1990) also calculated the resistance by summing the form drag and skin friction, using a shear coefficient to account for the grain resistance and a drag coefficient for the form drag.

Kumar and Roberson (1980) defined the flow as either semi-rough, in which the resistance consisted of skin friction and form drag, or fully rough in which the effects of viscous shear at the boundaries was neglected and form drag accounted for the total resistance. Using known logarithmic velocity distribution laws and the theoretical linear shear stress distribution, an algorithm to predict the resistance was presented. Similar algorithms are also suggested by Roberson (1970) and Lewis (1975). These analytical approaches are however limited to the flow regimes on which the governing equations are based. Thus, for example the wake generated by a large scale roughness element upstream will not be taken into account.

Alternatively, flow through a duct may be determined by modelling the duct numerically. The effect of rough boundaries may then be accounted for by three methods:

1. Refined flow method. The computational domain includes the geometric representation of each roughness element. A fine grid is usually required to accurately predict flow over individual roughness elements (e.g. Liu *et al* 1995 and Chang and Mills 1991).

2. Discrete element approach. The effect of a collection of elements is taken into account by including a form drag term in the momentum equation and including an allowance for the blockage effect of the roughness elements. Models have been presented by Finson and his co-workers (1979, 1979, 1982), Christoph and Pletcher (1983), Christoph (1981) and Taylor, Coleman and Hodge (1985).

3. Wall laws. By specifying a roughness parameter (such as C, n, f or k_s) a wall function (e.g. Cebeci and Chang 1978) or resistance function (e.g. Miller and Wenzel 1985) may be used within a numerical model to account for the effect of a rough wall.

2.2 Computational Hydraulics

Numerical modelling is now widely used to simulate fluid flow in open channels and closed conduits. Due to the wide range of applications, models are available with varying degrees of complexity.

In numerical modelling the governing differential equations of fluid flow are approximated by a set of algebraic equations, and an algorithm is prescribed for solving the latter. The numerical solution consists of a set of numbers from which the distribution of the dependant variable ϕ can be constructed. In this sense a numerical method is akin to a laboratory experiment in which a set of instrument readings at a finite number of measurement stations gives the distribution of the measured quantity. A computational model consists of the computer coding required to implement the numerical algorithm and thereby solve the equations with appropriate boundary conditions.

The rest of Section 2.2 gives a general background into the various differential equations used to model fluid flow in current computational river models and the techniques available for solving these equations are then presented. Section 2.3 contains specific details of the commercial software PHOENICS which is used in this study.

2.2.1 Differential Equations for Fluid Flow

Classical fluid dynamics uses the continuity and Navier-Stokes equations to describe fluid motion. For incompressible, isothermal flow these are

$$\frac{\partial u_\beta}{\partial x_\beta} = 0 \qquad \textit{Equation 2-45}$$

$$\frac{\partial u_\alpha}{\partial t} + u_\beta \frac{\partial u_\alpha}{\partial x_\beta} = -\frac{1}{\rho}\frac{\partial p}{\partial x_\alpha} + \nu \frac{\partial^2 u_\alpha}{\partial x_\beta \partial x_\beta} + S_\alpha \qquad \textit{Equation 2-46}$$

These equations and those which follow are written in Cartesian tensor notation which is briefly described in Appendix 2. The Navier-Stokes equations in full Cartesian notation are given in Appendix 3.

2.2.2 Representation of Turbulence

Theoretically, it is possible to solve the Navier-Stokes equations directly, using Direct Numerical Simulation (DNS). Indeed, this has been achieved for laminar flows and turbulent flows of low Reynolds number in simple geometries. However flow in conduits are generally highly turbulent, and often with complex geometries. Such high Reynolds number turbulent motion contains a hierarchy of eddies which, at their smallest scale[3], can only be resolved by a computational mesh size which is beyond the storage capacity and processing speed of the most powerful computers currently available.

An alternative to DNS is Large Eddy Simulation (LES) in which the large eddies are resolved by the grid and are therefore calculated from the Navier-Stokes equations, but the unresolved smaller eddies are modelled. The success of this method is based on the premise that the large scale eddies in the flow are directly affected by the boundary conditions, whereas the smaller scale eddies have more universal characteristics and are thus more amenable to modelling. Because the smallest eddies are modelled, the computational cells can be much larger than the Kolmogorov length scale, and a lower number of grid cells may be used for LES than are required for DNS. However large

---

[3] The Kolmogorov scale. See e.g. Tennekes and Lumley (1978) or Willcox (1994).

numbers of cells are still required. Willcox (1994) for example suggested that approximately 1×10^7 grid points are required to model a channel flow with a Reynolds number of 61,600 compared with 1.5×10^8 cells for DNS.

Because hydraulic engineers are not usually concerned with the detailed turbulent motion but only with its effect on the mean flow, an approximation can be made by averaging the turbulent velocity fluctuations over a time scale much larger than that of the random motions (but the time scale should not be long enough to average out bulk, time dependent flows, such as tides or flood waves). Using this method the flow properties of velocity and pressure may be defined in terms of a mean (time averaged) component and a randomly fluctuating component:

$$u = U + u'$$
$$p = P + p'$$

Equation 2-47

After substitution into Equation 2-45 and Equation 2-46, this gives the time averaged continuity equations and the Reynolds equations which in tensor form are (an expanded representation is given in Appendix 3):

$$\frac{\partial U_\beta}{\partial x_\beta} = 0$$

Equation 2-48

$$\frac{\partial U_\alpha}{\partial t} + U_\beta \frac{\partial U_\alpha}{\partial x_\beta} = -\frac{1}{\rho}\frac{\partial P}{\partial x_\alpha} + \frac{1}{\rho}\frac{\partial T_{\alpha\beta}}{\partial x_\beta} + \nu \frac{\partial^2 U_\alpha}{\partial x_\beta \partial x_\beta} + S_\alpha$$

Equation 2-49

where

$$T_{\alpha\beta} = -\rho(\overline{u'_\alpha u'_\beta})$$

This averaging process creates the Reynolds stresses ($T_{\alpha\beta}$) which are additional unknowns. The equations do not form a closed set, and further equations are required to model the Reynolds stresses. Such turbulence models are based on hypotheses concerning turbulent processes and require empirical input in the form of functions or constants; they do not simulate the details of the turbulent motion but only the effect of turbulence on the mean-flow behaviour. A comprehensive review of the turbulence models used in hydraulic engineering is given by Rodi (1984) and Willcox (1994), from which certain aspects have been selected for discussion here.

One of the most common ways to model the Reynolds stresses is to use the Boussinesq *eddy-viscosity* concept. Using an analogy with laminar flows, it assumes that the turbulent stresses are proportional to the mean velocity gradients at that point in the flow:

$$-\rho(\overline{u'_\alpha u'_\beta}) = \mu_t \left(\frac{\partial U_\alpha}{\partial x_\beta} + \frac{\partial U_\beta}{\partial x_\alpha}\right) - \frac{2}{3}\rho k \delta_{\alpha\beta} \qquad \textit{Equation 2-50}$$

where μ_t is the eddy viscosity. This is not a property of the fluid (unlike the laminar dynamic viscosity) as it varies throughout the flow field. The eddy-viscosity is of course still unknown, so that the problem has now changed from modelling the Reynolds stresses to finding a value for the eddy viscosity. Models which use the eddy-viscosity concept generally use the result obtained by dimensional analysis (e.g. Tennekes and Lumley 1978) where the eddy viscosity is given as the product of a single length scale, a single velocity scale and the density. A review of the use of these models in practical river channel applications is presented in Senior (1994). Such models may be classed according to their complexity.

Zero-equation models

Zero-equation models use no transport equations to model the eddy viscosity. The eddy viscosity may be either prescribed as a constant, or related to the time averaged flow distribution. It is well known (e.g. Rodi 1984) that μ_t has a nearly parabolic distribution with depth in open channel flow and the use of a constant eddy viscosity leads to unrealistic velocity fields. However many river and tidal computational models which are used to simulate large bodies of water use a depth averaged eddy viscosity within a two-dimensional depth averaged model.

In Prandtl's mixing length model the eddy viscosity is a function of a prescribed length scale (l_m) and the local velocity gradient (e.g. Schlichting 1960):

$$\mu_t = \rho\, l_m^2 \left|\frac{\partial U}{\partial y}\right| \qquad \textit{Equation 2-51}$$

As the flow becomes more complex, the value of l_m becomes increasing difficult to define.

One-equation models

The models in this group use a single equation for the transport of a turbulence property to determine the velocity scale. Most commonly used is the equation for the kinetic energy of turbulence, k. The square root of k gives the velocity scale required so that

$$\mu_t = c_\mu \rho k^{1/2} l_m \qquad \textit{Equation 2-52}$$

where c_μ is a constant derived from empirical data.

One-equation models still require the specification of a length scale as for zero-equation models, but take better account of turbulent transport.

Two-equation models

These contain two transport equations from which the velocity and length scales may be determined. Several models have been proposed (e.g. Rodi 1984 or Willcox 1994) but by far the most popular is the k-ε model. The kinetic energy of turbulence is used to find the velocity scale, as employed in the one-equation model, and the length scale is derived from the dissipation rate of turbulent kinetic energy, ε (e.g. Tennekes and Lumley 1978):

$$l_m = \frac{c_D k^{3/2}}{\varepsilon} \qquad \textit{Equation 2-53}$$

The constant c_D is usually set to unity (Launder and Spalding 1974). Combining Equation 2-52 and Equation 2-53 gives an expression in which the length scale has been eliminated:

$$\mu_t = c_\mu \rho \frac{k^2}{\varepsilon} \qquad \textit{Equation 2-54}$$

The transport equation for k is;

$$U_\alpha \rho \frac{\partial k}{\partial x_\alpha} = \frac{\partial}{\partial x_\alpha}\left(\frac{\mu_t}{\sigma_k}\frac{\partial k}{\partial x_\alpha}\right) + \mu_t \frac{\partial U_\alpha}{\partial x_\beta}\left(\frac{\partial U_\alpha}{\partial x_\beta} + \frac{\partial U_\beta}{\partial x_\alpha}\right) - \rho\varepsilon \qquad \textit{Equation 2-55}$$

and the transport equation for the dissipation rate of turbulent kinetic energy, ε, is;

$$U_\alpha \rho \frac{\partial \varepsilon}{\partial x_\alpha} = \frac{\partial}{\partial x_\alpha}\left(\frac{\mu_t}{\sigma_k}\frac{\partial \varepsilon}{\partial x_\alpha}\right) + c_1 \frac{\varepsilon}{k}\mu_t \frac{\partial U_\alpha}{\partial x_\beta}\left(\frac{\partial U_\alpha}{\partial x_\beta} + \frac{\partial U_\beta}{\partial x_\alpha}\right) - c_2 \rho \frac{\varepsilon^2}{k} \qquad \textit{Equation 2-56}$$

The empirical constants c_μ, σ_k, σ_ε, c_1, c_2 have been evaluated from physical measurements of various turbulent flows and are given in Table 2-3. (Launder and Spalding 1974).

c_μ	σ_k,	σ_ε	c_1	c_2
0.09	1.0	1.314	1.44	1.92

Table 2-3: Constants used in the k-ε model

By the use of these semi-empirical techniques, Equations 2-50, 2-54, 2-55 and 2-56 together with the continuity and Reynolds equations (Equations 2-48 and 2-49) form a closed set and may be solved.

Equations 2-55 and 2-56 are only valid for high Reynolds numbers and give an isotropic eddy viscosity, which is used in Equation 2-50 to calculate all of the Reynolds stresses $(\overline{u'_\alpha u'_\beta})$. Very near to the wall the local Reynolds number is low and the standard k-ε model is invalid. Empirically based *wall functions* are commonly used to bridge the gap between the wall and the fully turbulent region, but these are not accurate near separation and stagnation points. For example Senior and Aroussi (1992) found that a wall function used in conjunction with the k-ε model under-predicted the re-attachment length behind a backward facing step by up to 20%. To overcome this limitation, low Reynolds number forms of the k-ε model have been developed (e.g. Jones and Launder 1972). Another expedient is the two-layer model.

The two-layer model

A two-layer form of the k-ε turbulence model (Rodi 1991) has been found to perform more accurately than the standard k-ε model, by resolving the flow in the viscosity affected near wall region, thus obviating the need for empirical wall functions. It does however require a greater number of cells and therefore an increase in storage capacity and computational time.

The domain is split into two 'layers', a near wall layer and an outer fully turbulent layer and different turbulence models are used in each layer. In the fully turbulent region of the flow, the standard high Reynolds number k-ε turbulence model is used. Close to the wall the one-equation model of Norris and Reynolds (1975) is used. In the latter model, the equation for ε (Equation 2-56) is no longer employed and instead the length scale l_m is determined by a simple algebraic expression:

$$l_m = \kappa C_\mu^{-3/4} y \left(1 - \exp\left(-0.0198 \mathbf{Re_y}\right)\right) \qquad \textit{Equation 2-57}$$

where C_μ is 0.09, $\kappa = 0.41$ and $\mathbf{Re_y} = k^{1/2} \frac{y}{\nu}$. The velocity scale again comes from a transport equation for the turbulent kinetic energy k. The dissipation rate of turbulent kinetic energy, appearing as a (negative) source term in the kinetic energy transport equation, is determined from:

$$\varepsilon = C_d k^{\frac{3}{2}} \left(1 + \frac{5.3}{\mathbf{Re_y}}\right) \qquad \textit{Equation 2-58}$$

where the coefficient $C_d = 0.1643$. Thus the eddy viscosity now reads:

$$\nu_t = C_\mu k^{\frac{1}{2}} l_m \qquad \textit{Equation 2-59}$$

When implemented in the PHOENICS CFD code the one-equation model is matched with the high Reynolds number k-ε model at locations where $\mathbf{Re_y} = 350$.

Reynolds stress and algebraic stress models

In certain flow situations the assumption of an isotropic eddy viscosity inherent in the standard k-ε model is too crude. For example the turbulence driven secondary motion observed in rectangular ducts is not predicted by isotropic eddy viscosity models. In order to allow for the development of different Reynolds stresses and to account properly for their transport, models have been developed which employ transport equations for the individual stresses $(\overline{u'_\alpha u'_\beta})$.

From the Reynolds equations, Reynolds stress transport equations may be developed which express the fact that the distribution of the Reynolds stress is influenced by the

physical processes of convective and diffusive transport, stress production, pressure-strain and viscous dissipation (Rodi 1984). The diffusion, pressure-strain and dissipation terms contain correlations for which model approximations must be introduced in order to close the Reynolds stress transport equations.

The Reynolds stresses are obtained by the solution of six partial differential equations. With the continuity equation and the solution of various additional model terms, the numerical solution is not trivial, thus making the model computationally demanding and restricting its application. For this reason, efforts have been made to simplify the Reynolds stress transport equations into algebraic relationships. Various ways of performing this have been tried, the most successful models including the turbulent kinetic energy term k and its rate of dissipation ε. These are obtained by means of transport equations, so these algebraic stress models are particular forms of k-ε model, but having the facility to compute the turbulence driven secondary flow production terms, which the standard k-ε model does not have.

Choice of turbulence model

For the application of modelling flow over periodically spaced roughness elements, a two-layer model was chosen for the following reasons.

1. When boundary layer separation occurs on a bluff body, the flow streamlines can be roughly described in terms of a wake region surrounded by a free stream. The free stream pressure is imposed on the wake region which generally creates a pressure distribution on the downstream face of the bluff body much lower than that on its upstream face. This is the mechanism of form drag.

   It follows that if the streamlines bounding the wake region are known, the pressure distribution around the bluff body may be calculated approximately using irrotational flow theory. Haque and Mahmood (1983) did this when they used finite element analysis to calculate the irrotational flow around sand dunes and thereby predict the drag force. In the present case it was required that the turbulence model should give a good prediction of the streamlines bounding

separated flow regions, and for this reason a turbulence model was sought which had a good track record in predicting boundary layer separation and re-attachment.

2. The full Reynolds stress transport model satisfied these criteria and was available in PHOENICS. However, its PHOENICS implementation used a wall function rather than integrating the conservation equations right up to the boundary, and this would have given a poor prediction of wall shear stresses in the recirculating region behind a roughness element.

The facility to predict turbulence driven secondary flows, as the Reynolds stress model does, was immaterial to the present application. Although it was originally intended to model flow over three-dimensional roughness elements, their staggered arrangement would have precluded the development of turbulence driven secondary flows of any significance. For flows over two-dimensional roughness elements turbulence driven secondary flows are absent from the physical flow.

For these reasons the full Reynolds Stress transport model offered little to the present application, while imposing high demands upon computational resources.

3. As previously mentioned, it was considered essential that the numerical model should integrate the conservation equations right up to the wall. This eliminated the standard k-ε model and focused attention on low Reynolds number k-ε models and two-layer models (using the k-ε model in the fully turbulent region) both of which had a good reputation for predicting separation and reattachment.

After testing the PHOENICS implementation of the low Reynolds number k-ε model (Lam and Bremhorst 1981) and the two-layer model (Rodi 1991) it was decided to adopt the latter. Rodi's model was successfully applied with a far less stringent requirement for fine grid cells near the wall. The Lam-Bremhorst model required particularly fine resolution of the high gradient in ε in the near wall region.

2.2.3 Simplifications to the Differential Equations

A variety of approximations may be made to the original Reynolds equations enabling the computer model to be simpler, less demanding on computational resources and therefore more economic. For a given flow problem the choice of approximations employed will depend on factors such as geometry, expected flow pattern, accuracy required, cost and the computational resources available. Typical approximations include:

1. Time dependence. Neglecting the time dependent term $\partial U_\alpha / \partial t$ in the Reynolds equations (Equation 2-49) gives a steady state solution.

2. Reduced dimensionality. Averaging the flow over a channel cross-section greatly reduces the numerical complexity and leads to the one-dimensional St. Venant equation (e.g. French 1986). Low processing requirements enable long reaches of rivers to be modelled quickly and cheaply. (e.g. Neat *et al* 1989). Two-dimensional depth averaging of the Reynolds equations leads to the shallow water equations (e.g. Ligget 1994) which may be used to model shorter reaches of river (e.g. Schymitz *et al* 1983). A typical vertical plane can also be modelled in two dimensions if the solution of transverse distributions is not required, for example on the centre line of a channel when the influence of the side walls can be neglected.

2.2.4 Numerical Solutions

Algebraic equations are derived from the original set of differential equations by discretizing the flow variables over space and time (if time dependent). Thus at a general point P_P the dependent variables such as velocity are given in terms of conditions at adjacent points, P_E and P_W (where the subscripts E and W denote points to the east and west) and at time t, which are located at a finite distance and time away. The space co-ordinate is often elliptic for irregular channel flows, i.e. points either side of P_P are used, but time is parabolic in that only the previous time is used in the present calculation. The continuous nature of the physical flow is thus approximated by values at a finite number of specific points.

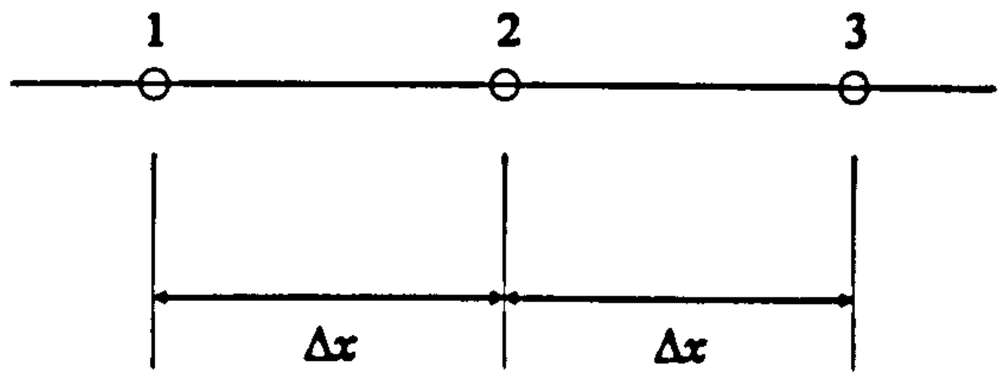


Figure 2-3: Finite difference line

The algebraic equations, also referred to as discretized equations, can be derived in various forms. These result from different methods of derivation and the different profiles used for specifying the variation of ϕ between the points. A discussion of all the schemes shown here may be found in Peyret and Taylor (1986).

1. Finite difference methods. The finite difference equations are typically derived from a truncated Taylor series. For example consider the grid points along a single line as shown in Figure 2-3. For grid point 2, located mid way between points 1 and 3 the Taylor series gives

$$\phi_1 = \phi_2 - \Delta x \left(\frac{d\phi}{dx}\right)_2 + \frac{1}{2}(\Delta x)^2 \left(\frac{d^2\phi}{dx^2}\right)_2 - \ldots$$

$$\phi_3 = \phi_2 + \Delta x \left(\frac{d\phi}{dx}\right)_2 + \frac{1}{2}(\Delta x)^2 \left(\frac{d^2\phi}{dx^2}\right)_2 + \ldots$$

Equation 2-60

Adding and subtracting these equations gives expressions for the second and first derivatives of ϕ at point 2 respectively. The substitution of such expressions into the differential equations leads to the finite difference equations, which may have a variety of forms, for example Preissmann's box scheme and the Abbott 6 point scheme, both of which are discussed at some length by Cunge, Holly and Verwey (1980).

2. Finite element methods. Finite element analysis may use variational formulation or the method of weighted residuals. The general variable ϕ is integrated over triangles, quadrilaterals or hexagons and gives a profile of ϕ over each element.

3. Finite volume method. Also known as the *finite domain* or *control volume* method. The domain is divided into a number of non-overlapping control volumes (cells) such that there is one control volume surrounding each point. The

differential equations are integrated over each control volume. Piecewise profiles expressing the variation of ϕ between the grid points are used to evaluate the required integrals but are not part of the eventual solution.

4. Method of characteristics. This method has been developed from inviscid supersonic theory and uses the fact that such flows can be characterised by families of intersecting lines along which disturbances propagate. These lines, called the *characteristics* of the flow, are used to construct the grid. The position of the lines, and therefore the grid shape, are unknown at the start of the calculation. As the grid adapts during the solution it is possible for the grid to collapse onto itself in areas of high gradients, causing this method to fail.

2.2.5 Boundary Conditions

Boundary conditions represent the physical conditions which occur at the boundaries of the computational domain.

At solid boundaries all velocity components, including the turbulent fluctuations are zero. These conditions may be specified directly if the high gradients which occur in the viscosity affected sub-layer are resolved by very fine cells. Alternatively a wall function such as that presented in Section 2.3.4 can be employed which specifies the flow conditions outside of the viscosity affected region.

The free surface of open channel flow is most simply represented by a plane of zero shear stress. This may be improved by damping the turbulent fluctuations normal to the free surface (e.g. Nezu and Rodi 1986, Celik and Rodi 1984). Various models have been presented which may be used to calculate the position of the free surface and may be used to simulate waves (e.g. Hirt and Nichols 1981, Maxwell 1977 and Johns, Soulsby and Xing 1993).

If mass enters or leaves the domain, values need to be specified for the rate at which this occurs. Any properties pertaining to the incoming fluid, for example temperature or turbulence, will also need to be specified. The mass flow rate can be defined directly through the use of a velocity profile, or in some open channel models the water levels

may be specified at inlet and outlet. When modelling a pressure driven duct, the mass flow rate is most effectively defined by specifying a lower pressure distribution at exit than at inlet. The incoming fluid will be transporting momentum into the domain which will need to be explicitly defined together with any other properties convected into the domain.

2.3 Mathematical Basis of PHOENICS

This section explains the equations which are used within the PHOENICS CFD code and the numerical methods used to solve them. Only the material directly relevant to this study is presented. A more general description of the PHOENICS equations is given in the PHOENICS manual TR99 (Rosten and Spalding 1987). A more in depth discussion of many of the numerical methods summarised in this section is given by Patankar (1980).

During the course of this study several errors were found within the code, these are presented in Appendix 4 together with the coding used to correct the problem.

2.3.1 Differential Equations

PHOENICS, in common with several other CFD codes, predicts the distribution of a general specific (i.e. per unit mass) variable ϕ which obeys conservation laws of the form:

$$\underbrace{\frac{\partial \phi}{\partial t}}_{\text{unsteady term}} + \underbrace{\operatorname{div}(\mathbf{u}\phi)}_{\text{convection term}} = \underbrace{{}^{1}\!/\!_{\rho}\operatorname{div}(\Gamma \operatorname{grad}\phi)}_{\text{diffusion term}} + S \qquad \textit{Equation 2-61}$$

where Γ is the exchange coefficient and S is the source term for the general variable ϕ. (indeed these should be written Γ_ϕ and S_ϕ but this would lead to too many subscripts in later work).

The conservation laws used by PHOENICS in this study are illustrated in Table 2-4, each determined by the definition of ϕ.

Conservation Law	Specific variable ϕ	Exchange coefficient Γ	Types of source
Conservation of mass (Continuity Equation)	1.0	0	Source of mass per unit volume
Conservation of momentum (Navier-Stokes or Reynolds equations)	u, v, w (laminar) U, V, W (turbulent)	μ (laminar) $\mu_{eff}=\mu_t+\mu$ (turbulent)	May contain: 1) Sources of momentum 2) pressure gradients 3) gravitational forces
Conservation of turbulence kinetic energy	k	$\Gamma = \mu_{eff}/\sigma_k$ where: $\sigma_k = 1.0$	Generation of k per unit volume
Conservation of turbulent kinetic energy dissipation rate	ε	$\Gamma = \mu_{eff}/\sigma_\varepsilon$ where: $\sigma_\varepsilon = 1.314$	Generation of ε per unit volume

Table 2-4: Conservation Laws

2.3.2 Discretization Equations

Section 2.2.4 mentions various numerical methods by which the governing differential equations of a problem are changed into a set of algebraic equations. PHOENICS uses the *Control Volume* formulation, often termed the *Finite Volume* or *Finite Domain* method. The discretization equations (or Finite Volume Equations, FVE) obtained express the conservation principle for ϕ over a finite control volume, just as the differential equation expresses it for an infinitesimally small control volume. Some characteristics of this method are discussed below.

1. An attractive feature is that the resulting solution gives integral conservation over any group of cells, and consequently over the whole calculation domain, for any size of cell and not just in the limiting case of infinitesimally small grid cells. This differs from the Finite Difference method which expresses the conservation principle only for an infinitesimal control volume.

2. The average value of ϕ for a particular cell may be represented by the value at one point located at the centre of the cell, termed the *grid node* or the *grid point.*

3. The solution consists only of values of ϕ at these grid points and gives no information about the profile of ϕ between them. Thus although piece-wise profiles over each cell are used to evaluate the integrals, the assumed profile may then be forgotten. The Finite Element method on the other hand also gives the assumed profile across the cell as part of the solution.

After integration, the FVE for each variable may be cast into the following form:

$$a_P\phi_P = a_E\phi_E + a_W\phi_W + a_N\phi_N + a_S\phi_S + a_H\phi_H + a_L\phi_L + a_T\phi_T + b \qquad \textit{Equation 2-62}$$

or

$$a_P\phi_P = \sum_{nb=E,W,N,S,H,L,T} a_{nb}\phi_{nb} \quad + b$$

where the subscript P represents the current node and E, W, N, S, H, L and T represent the neighbouring nodes (subscript nb) to the east, west, north, south, high, low and previous time as defined by Figure (ii). The coefficients of the FVE's represent the influence from one cell to its neighbour and are:

$$a_E = \max\left(0, d_e - \alpha|m_e|\right) + \max(0, -m_e)$$

$$a_W = \max\left(0, d_w - \alpha|m_w|\right) + \max(0, m_w)$$

$$a_N = \max\left(0, d_n - \alpha|m_n|\right) + \max(0, -m_n)$$

$$a_S = \max\left(0, d_s - \alpha|m_s|\right) + \max(0, m_s)$$

$$a_H = \max\left(0, d_h - \alpha|m_h|\right) + \max(0, -m_h)$$

$$a_L = \max\left(0, d_l - \alpha|m_l|\right) + \max(0, m_l)$$

$$a_T = \frac{\rho_T (\Delta x \Delta y \Delta z)_T}{\Delta_t}$$

$$a_P = a_E + a_W + a_N + a_S + a_H + a_L + a_T$$

$+(-S_2)$ a source term contribution

$+\left(\rho \Delta x \Delta y \Delta z / \Delta t\right)_P$ a term for false time step relaxation

+ boundary conditions

The dominant effect of upstream values due to convection is taken into account through the use of the hybrid scheme. This scheme and a theoretical background of the FVE coefficients is given in Appendix 5.

The last term in Equation 2-62 represents the source term given by

$$b = S_1 \quad + {\phi_P \rho (\Delta x \Delta y \Delta z)_T}/{\delta t} \quad + \text{boundary conditions} \qquad \textit{Equation 2-63}$$

This contains part, or all, of the information about the source of the entity ϕ. Any additional terms or external influences which are not yet accounted for may also be added to the source term. Thus the complete source term may contain several individual terms, some of which were mentioned in Table 2-4. In PHOENICS the total source is expressed as a linear function of ϕ_P:

$$S = S_1 + S_2 \phi_P \qquad \textit{Equation 2-64}$$

S_2 is the gradient of the linear function and is thus:

$$S_2 = \left(\frac{\partial S}{\partial \phi} \right)_P \qquad \textit{Equation 2-65}$$

and $S_1 = S - S_2 \phi_P$.

After discretization, the linear part of the source term, $-S_2$, becomes part of the a_P term and S_1 contributes to the b term in Equation 2-62. Also included within the b term are sources due to boundary conditions, and relaxation when the false time step method is used.

2.3.3 Solution Procedure

Although Equation 2-62 has the appearance of a linear equation, it is in general non-linear. The coefficients may themselves depend on the values of ϕ, and since ϕ can represent a number of physical quantities the coefficients for one meaning of ϕ may be influenced by some other quantity ϕ. Thus to obtain ϕ for each grid point, an iterative procedure needs to be employed. This consists of a number of *sweeps* throughout the computational domain. A single sweep visits each grid point and updates the value of ϕ

for all variables. Many sweeps are usually required until a converged solution is obtained, i.e. when the values of ϕ from one sweep to the next change by an amount smaller than the specified convergence criterion.

For steady two-dimensional simulations, PHOENICS possesses two solution procedures, the Jacobi point by point and the slab-wise simultaneous procedure.

Jacobi point by point

This procedure visits each grid node once in a defined order, replacing the current value of ϕ_P with a new value calculated from:

$$\phi_P = \frac{\sum a_{nb}\phi^*_{nb} + b}{a_P} \qquad \textit{Equation 2-66}$$

where ϕ^*_{nb} denotes the value of ϕ at some neighbouring point from the previous sweep.

The point by point scheme is conceptually easy, however information is transmitted at one grid interval per iteration and therefore convergence may be slow.

Slab-wise simultaneous

PHOENICS defines a slab as a two-dimensional array of grid cells in x and y shown in Figure (ii). The FVE may then be written as:

$$a_{P(i,j)}\phi_{(i,j)} = a_{E(i,j)}\phi_{(i+1,j)} + a_{W(i,j)}\phi_{(i-1,j)} + a_{N(i,j)}\phi_{(i,j+1)} + a_{S(i,j)}\phi_{(i,j-1)} + b_{(i,j)} \qquad \textit{Equation 2-67}$$

The procedure is a variant of the Tri-Diagonal Matrix Algorithm (TDMA) and solves the ϕ values for the whole slab simultaneously.

For domains in three dimensions, a slab consists of x and y cells at a constant IZ value. The solution procedure steps through each IZ slab in turn. The influence of the neighbouring z cells to a slab are represented by the coefficients a_H and a_L which, for the slab by slab method, are included inside the b term. Alternatively the whole field solution procedure (Spalding 1980) which is an extension of the slab by slab procedure, may be employed for three-dimensional grids.

To formulate the procedure it is noted that at the boundaries of a slab the coefficients will be zero, i.e.

$$a_{S(i,1)} = 0 \qquad a_{N(i,M)} = 0$$

$$a_{W(1,j)} = 0 \qquad a_{E(N,j)} = 0$$

so for example applying Equation 2-67 to $i = 1$ and $j = 1$ will give:

$$\phi_{(1,1)} = \frac{a_{N(1,1)}}{a_{P(1,1)}}\phi_{(1,2)} + \frac{a_{E(1,1)}}{a_{P(1,1)}}\phi_{(2,1)} + b_{(1,1)} \qquad \textit{Equation 2-68}$$

$$= \mathrm{F}^{N}_{(1,1)}\phi_{(1,2)} + \mathrm{F}^{E}_{(1,1)}\phi_{(2,1)} + b_{(1,1)}$$

where F represents a functional relationship. Postulate that for all grid cells, ϕ may be given in terms of its north and east neighbours, i.e.:

$$\phi_{(i,j)} = \mathrm{F}^{N}_{(i,j)}\phi_{(i,j+1)} + \mathrm{F}^{E}_{(i,j)}\phi_{(i+1,j)} + \mathrm{F}^{b}_{(i,j)} \qquad \textit{Equation 2-69}$$

The following two relations may then also be written:

$$\phi_{(i-1,j)} = \mathrm{F}^{N}_{(i-1,j)}\phi_{(i-1,j+1)} + \mathrm{F}^{E}_{(i-1,j)}\phi_{(i,j)} + \mathrm{F}^{b}_{(i-1,j)} \qquad \textit{Equation 2-70}$$

and

$$\phi_{(i,j-1)} = \mathrm{F}^{N}_{(i,j-1)}\phi_{(i,j)} + \mathrm{F}^{E}_{(i,j-1)}\phi_{(i+1,j-1)} + \mathrm{F}^{b}_{(i,j-1)} \qquad \textit{Equation 2-71}$$

Substituting Equation 2-69, Equation 2-70 and Equation 2-71 into Equation 2-67 gives:

$$\begin{aligned} a_{P(i,j)}\phi_{(i,j)} = {} & a_{S(i,j)}\left[\mathrm{F}^{N}_{(i,j-1)}\phi_{(i,j)} + \mathrm{F}^{E}_{(i,j-1)}\phi_{(i+1,j-1)} + \mathrm{F}^{b}_{(i,j-1)}\right] \\ & + a_{N(i,j)}\phi_{(i,j+1)} \\ & + a_{W(i,j)}\left[\mathrm{F}^{N}_{(i,j-1)}\phi_{(i-1,j+1)} + \mathrm{F}^{E}_{(i-1,j)}\phi_{(i,j)} + \mathrm{F}^{b}_{(i-1,j)}\right] \\ & + a_{E(i,j)}\phi_{(i+1,j)} + b_{(i,j)} \end{aligned} \qquad \textit{Equation 2-72}$$

Putting this equation into the same form as Equation 2-69:

$$\begin{aligned} \phi_{(i,j)}\left[a_{P(i,j)} - a_{S(i,j)}\mathrm{F}^{N}_{(i,j-1)} - a_{W(i,j)}\mathrm{F}^{E}_{(i-1,j)}\right] & \\ = a_{N(i,j)}\phi_{(i,j+1)} + a_{E(i,j)}\phi_{(i+1,j)} & \\ + a_{S(i,j)}\left[\mathrm{F}^{E}_{(i,j-1)}\phi_{(i+1,j-1)} + \mathrm{F}^{b}_{(i,j-1)}\right] & \qquad \textit{Equation 2-73} \\ + a_{W(i,j)}\left[\mathrm{F}^{N}_{(i,j-1)}\phi_{(i-1,j+1)} + \mathrm{F}^{b}_{(i-1,j)}\right] & \\ + b_{(i,j)} & \end{aligned}$$

then

$$\phi_{(i,j)} = \mathrm{F}^{\mathrm{N}}_{(i,j)}\phi_{(i,j+1)} + \mathrm{F}^{\mathrm{E}}_{(i,j)}\phi_{(i+1,j)} + \mathrm{F}^{b}_{(i,j)} \qquad \textit{Equation 2-74}$$

if:

$$\mathrm{F}^{\mathrm{N}}_{(i,j)} = \frac{a_{\mathrm{N}(i,j)}}{d_{(i,j)}}$$

$$\mathrm{F}^{\mathrm{E}}_{(i,j)} = \frac{a_{\mathrm{E}(i,j)}}{d_{(i,j)}}$$

$$\mathrm{F}^{b}_{(i,j)} = \frac{a_{\mathrm{S}(i,j)}\left(\mathrm{F}^{\mathrm{E}}_{(i,j-1)}\phi_{(i+1,j-1)} + \mathrm{F}^{b}_{(i,j-1)}\right) + a_{\mathrm{W}(i,j)}\left(\mathrm{F}^{\mathrm{N}}_{(i,j-1)}\phi_{(i-1,j+1)} + \mathrm{F}^{b}_{(i-1,j)}\right) + b_{(i,j)}}{d_{(i,j)}}$$

$$d_{(i,j)} = a_{\mathrm{P}(i,j)} - a_{\mathrm{S}(i,j)}\mathrm{F}^{\mathrm{N}}_{(i,j-1)} - a_{\mathrm{W}(i,j)}\mathrm{F}^{\mathrm{E}}_{(i-1,j)}$$

Equation 2-74 gives the value of ϕ for all grid points in terms of the neighbouring cells to the north and to the east (however, see note below).

The slab by slab procedure is then as follows:

1. Calculate $\mathrm{F}^{\mathrm{N}}_{(i,j)}, \mathrm{F}^{\mathrm{E}}_{(i,j)}, d_{(i,j)}, \mathrm{F}^{b}_{(i,j)}$ for j = 1 to NY and i = 1 to NX.

   Note: the calculation of $\mathrm{F}^{b}_{(i,j)}$ involves the terms $\phi_{(i+1,\,j-1)}$ and $\phi_{(i-1,\,j+1)}$. These are unknown in the current sweep and therefore the approximate values obtained from the previous sweep are used. Further iterations over the slab will improve the calculated value of $\mathrm{F}^{b}_{(i,j)}$.

2. Compute ϕ from Equation 2-74 for j = NY to 1 and i = NX to 1, i.e. starting from the north east corner of the domain.

Relaxation of solution procedure

In the iterative solution of the algebraic equations it is often desirable to speed up (*over relax*) or to slow down (*under relax*) the changes from one iteration to the next. The two methods of performing this in the PHOENICS code are *linear relaxation* and the *false time step* method.

Linear relaxation

The general FVE (Equation 2-62) may be written as

$$\phi_P = \phi_P^* + \left(\frac{\sum a_{nb}\phi_{nb} + b}{a_P} - \phi_P^* \right) \qquad \textit{Equation 2-75}$$

where ϕ_P^* is the value of ϕ from the previous iteration and the contents of the parentheses represent the change in ϕ_P produced by the current iteration. This change may be altered in a linear manner by the introduction of a relaxation factor α

$$\phi_P = \phi_P^* + \alpha \left(\frac{\sum a_{nb}\phi_{nb} + b}{a_P} - \phi_P^* \right) \qquad \textit{Equation 2-76}$$

False time step or inertia method

The FVE may alternatively be written as

$$(a_P + i)\phi_P = \sum a_{nb}\phi_{nb} + b + i\phi_P^* \qquad \textit{Equation 2-77}$$

where i is the false time step or inertia relaxation coefficient. If i takes a very small value then Equation 2-77 reduces to the standard FVE and the relaxation will have no effect. The name "false time step" is due to the analogy between iterations converging to a steady value and an unsteady flow becoming steady. Note that the term $i\phi_P^*$ is analogous to $a_T\phi_T$ if $\phi_P^* = \phi_T$ and $i = a_T$ i.e. $i = \frac{\rho \Delta x \Delta y \Delta z}{\Delta t}$. This method is employed for the momentum equation as the false time interval allows for the influence of inertia in the fluid.

2.3.4 Boundary Conditions and Sources

All differential equations require boundary conditions to make the solution determinate. PHOENICS applies these boundary conditions as additional sources applied over a range of cells (a PATCH in PHOENICS). For example, the cells on one edge of the computational domain may contain sources to model the effects of a solid wall with friction on the external faces of these cells. Boundary conditions must be supplied at all external faces of the domain, but they may also be supplied internally to account for the effects of, for example, an internal wall. If the user does not explicitly define boundary

conditions for all the external faces of the domain, PHOENICS will automatically supply a boundary condition of zero diffusion flux, i.e. all gradients of ϕ are set to zero. This is equivalent to a plane of symmetry.

Sources to define boundary conditions are specified in the form:

$$S = T(C + \|m\|)\ (V - \phi_P) \qquad \textit{Equation 2-78}$$

where:

T is a multiplier dependent on the PATCH type. This is commonly the volume of the cell for which the units of T are [m^3].

V is the value of the boundary condition to be specified (dimensions are that of ϕ).

C is the coefficient of the boundary condition. Assuming a volume PATCH type, the dimensions of C are [kg m^{-3} s^{-1}].

m is the mass flow rate through the PATCH type. Again assuming a volume PATCH type, the dimensions of m are [kg m^{-3} s^{-1}].

The units of the source term are therefore [ϕ kg s^{-1}]. The operator $\|\ \|$ returns the maximum of zero and the quantity enclosed. The presence of $\|m\|$ accounts for the mass created in the current PATCH by the P1 boundary condition. For the majority of boundary conditions no mass is created, thus $m = 0$ and the source is:

$$S = TC\ (V - \phi_P) \qquad \textit{Equation 2-79}$$

or

$$S = TCV - TC\phi_P \qquad \textit{Equation 2-80}$$

This may be put into the same form as Equation 2-64 by putting $S_1 = T\ C\ V$ and $S_2 = -TC$.

The following are some examples of frequently encountered types of boundary condition. These include the fixed source, the fixed value, the mass flow and the wall boundary condition.

Fixed source boundary condition

If a fixed flux of a variable ϕ is required then the coefficient C in Equation 2-79 may be set to a small number (e.g. 2×10^{-10}) and the value of the boundary condition (V) set to the required flux (J_ϕ) divided by the coefficient:

$$V = \frac{J_\phi}{C} \qquad \textit{Equation 2-81}$$

The source term (Equation 2-80) then becomes (ignoring the multiplier T):

$$S = 10^{-10}\frac{J_\phi}{10^{-10}} - 10^{-10}\phi_P \approx J_\phi \qquad \textit{Equation 2-82}$$

PHOENICS supplies the variable FIXFLU which when used as the coefficient supplies a value of 2×10^{-10} and automatically divides V by 2×10^{-10}.

Fixed value boundary condition

If it is desired that ϕ should take the value of V at the cell in question, the C is set to a large number, e.g. 2×10^{10} and the FVE (Equation 2-62) then becomes:

$$\phi_P = \frac{a_N + a_S + \quad \ldots \quad + a_L + (2\times10^{10} \times V)}{a_N + a_S + \quad \ldots \quad + a_L - (-2\times10^{10})} \qquad \textit{Equation 2-83}$$

in which all terms are negligible compared with the source terms giving

$$\phi_P = \frac{2\times10^{10} \times V}{2\times10^{10}} \quad \approx V \qquad \textit{Equation 2-84}$$

The PHOENICS variable FIXVAL of value 2×10^{10} may be used as the coefficient to achieve this.

Mass flow boundary condition

The conservation law for mass flow is the continuity equation but, as can be seen from Table 2-4, there is no variable ϕ for this law. To overcome this, the variable P1 is used as a flag for the continuity equation (note that the P1 variable does not have its own conservation law). The concept used is that of pressure as a driving force for mass flow; an inflow (for example) of mass is the result of a difference between the external pressure (the value) and the internal pressure (P1 at the current cell).

Thus to model a fixed flux of mass into a domain, the variable used is P1 and the coefficient is FIXFLU. If mass is flowing into the domain (or being "created" within the cells of the current PATCH) then other fluid properties (such as turbulence) are consequently convected into the domain. To account for this the source of ϕ (where ϕ is the convected quantity, e.g. turbulence kinetic energy) is used in the form of Equation 2-78 or

$$S = TC\ (V - \phi_P) + T\ \|m\|\ (V - \phi_P) \qquad \textit{Equation 2-85}$$

The first term accounts for diffusive inflow and is usually neglected by setting C=0. The second term represents the convective inflow of the property ϕ. The use of the operator $\|\ \|$ on m ensures that only positive values of m will give rise to convected inflow. This procedure may be performed in PHOENICS by setting the coefficient equal to ONLYMS (ONLY MasS flow) and ϕ to the required convected quantity.

The continuity equation may also be used as a method of specifying boundary conditions for pressures directly. Thus to fix the pressure at a particular cell to a desired value, the coefficient FIXVAL (2×10^{-10}) could be used with the variable P1. However, CHAM (1987) report that convergence problems may arise due to round off error and suggest the use of FIXP (numerical value of 1.0) as the coefficient. Intermediate values of the coefficient will change the "stiffness" of the fixed pressure boundary condition.

Wall boundary condition

The following example shows how the streamwise velocity, parallel to a wall with friction, is reduced to zero by applying a (negative) source of momentum flux per unit area of $\mu\, \mathrm{d}U/\mathrm{d}y$.

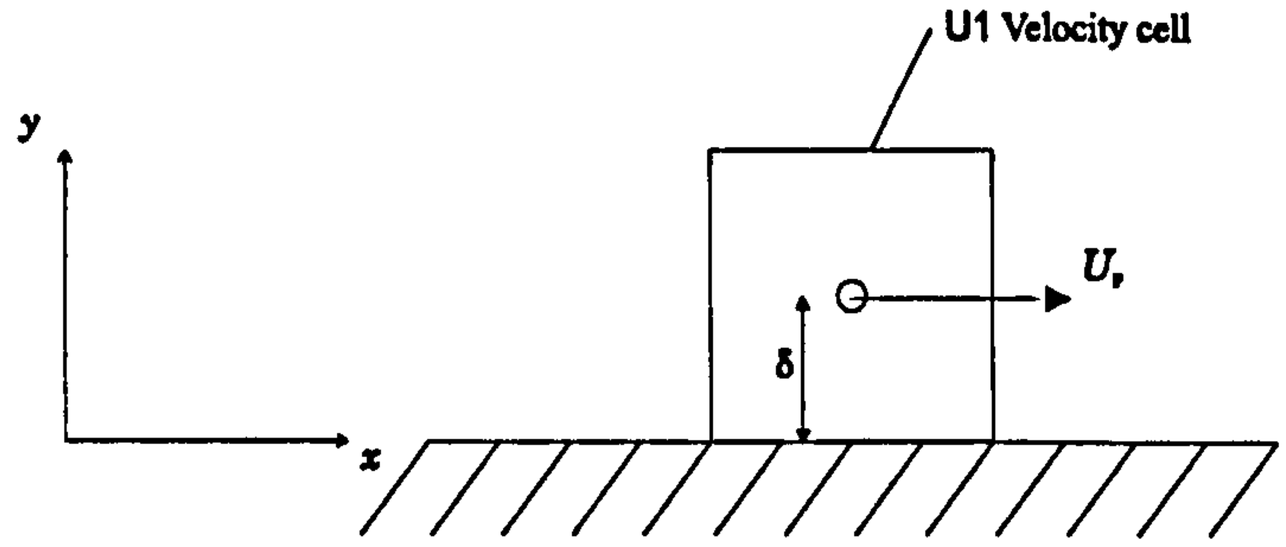


Figure 2-4: Wall boundary condition velocity cell

For the cell shown in Figure 2-4 having wall area A the source of U (i.e. specific momentum) required is

$$S = \mu A \frac{U}{\delta} \qquad \textit{Equation 2-86}$$

which can be expressed in linear form using:

$$C = \frac{\mu A}{\delta} \quad \text{and} \quad V = 0$$

PHOENICS however provides several PATCH types which supply the factor $\frac{\mu A}{\delta}$. For the orientation in Figure 2-4 it is the south face of the cell on which the wall boundary condition is applied, so the PATCH type used is SWALL. Thus the PHOENICS user should supply a coefficient C equal to 1.0 and a value V equal to zero.

Alternatively, the wall function employed by PHOENICS is that suggested by Jayatilleke (1969)

$$\frac{U}{U_\tau} = \frac{1}{\kappa} \ln\left(\frac{E y U_\tau}{\nu}\right) \qquad \textit{Equation 2-87}$$

to prescribe the streamwise velocity in the near wall cell. The roughness parameter E may be defined in terms of k_s by the user (Equation 2-28). The turbulence quantities are specified by:

$$k = \frac{U_\tau^2}{\sqrt{0.09}} \qquad \textit{Equation 2-88}$$

$$\varepsilon = \frac{0.09^{0.75} \times k^{1.5}}{\kappa y} \qquad \textit{Equation 2-89}$$

2.3.5 Calculation of the Flow Field

Equation 2-62 is the FVE for a general variable ϕ which may be used to calculate the distribution of ϕ at the grid points for a given flow field. To calculate the flow field, conservation of momentum (Navier-Stokes or Reynolds equations) is used with $\phi = U$, V and W representing the specific momentum in the three co-ordinate directions. From Table 2-4 and Equation A2.3 it can be seen that the source term includes the pressure gradient term, $-{}^{dP}/_{dx}$ which is unknown. A discussion of this problem is given in Appendix 6. A two-fold solution is employed by PHOENICS with a *staggered grid* arrangement of cells, whereby scalar variables such as pressure and turbulence quantities are stored at cell centres and velocity components are stored on cell faces. This is shown in Figure (ii). The momentum equations are then solved together with the continuity equation using the SIMPLEST scheme. Details of the staggered grid and SIMPLEST are also given in Appendix 6.

2.3.6 Grid Methods

Introduction

The size, shape and distribution of grid cells for a given simulation will depend on the following factors:

1. Representation of geometry. The computational domain and any internal geometric features are defined only at cell boundaries and therefore any small scale geometric features will require small cells. Complex non-orthogonal geometries may be approximately represented by orthogonal geometries using small cells. Alternatively curvilinear grids may be used, which in PHOENICS may be orthogonal or non-orthogonal and are termed Body Fitted Co-ordinates (BFC).

2. Resolution of results. A finer grid will be required in regions of particular interest and where gradients in the flow are large.

3. Numerical accuracy. To solve the original differential equations accurately the control volume needs to be as small as possible, towards the limit of infinitely small cells from which the differential equations originate. Areas of non-linearity and high gradients require particularly small cells to ensure that the finite volume equations accurately represent the differential equations.

   Numerical inaccuracies may also arise from control volumes which have widely differing length scales, either from one control volume to the next or from cells which have an extreme ratio of length to height.

4. Turbulence model and flow pattern. Further conditions on cell size are imposed by the turbulence model employed and expected flow structure. For example in order that the viscous region of a boundary layer is represented, the two-layer model requires the centre of the cell nearest to a boundary to be located inside the sub-layer. The standard k-ε model on the other hand employs a wall function to bridge the sub-layer and dictates that the first cell must be located outside the viscosity affected region.

If all of the above factors are met to an acceptable degree then the solution should be insignificantly affected by the grid, a condition termed *grid independence*. For many practical purposes however the computational time and storage requirements to obtain a sufficiently grid independent solution are not available and a compromise has to be made.

The density of cells may often be safely reduced in some regions of a domain. Using conventional grid techniques, non-uniform distributions of cells are allowed in each co-ordinate direction, though each line of cells must be continuous. Multi-block gridding (MBG) is a technique which allows individual cells to be subdivided to allow increased resolution in isolated regions of the domain. This method was first introduced in version 2.1 (March 1995) of PHOENICS and due to its promising benefits the feature was investigated (see Appendix 7). However errors in the coding and inflexibility in defining sub-divided grids prohibited its use in the current project.

Both BFC and MBG's require increased computational resources and can lead to divergence of the solution procedure, which may outweigh the advantages of these techniques.

Non-uniform cartesian grids

PHOENICS specifies the position of each grid cell in terms of a cumulative fraction of the total domain in each co-ordinate direction. This is defined within the Q1 file using the XFRAC, YFRAC and ZFRAC arrays. Taking the x direction as an example, if the cells 1 to i occupy a distance of x then cell i has an XFRAC of

$$\mathrm{XFRAC}(i) = \frac{x}{\mathrm{XLAST}} \qquad \textit{Equation 2-90}$$

where XLAST is the total length of the domain.

Using this method of defining grids allows great flexibility but is tedious if each cell is defined separately. PHOENICS includes built-in functions to facilitate the generation of uniform grids and expanding/contracting grids which follow a power law. For the latter case each cell is defined by (using the x direction as an example)

$$\mathrm{XFRAC}(i) = \left(\frac{i}{\mathrm{NY}}\right)^{k} \qquad \textit{Equation 2-91}$$

An expansion factor, k, of greater than one produces an expanding grid. If k is less than one a contracting grid is generated.

Geometric law

Modelling the viscous sub-layer requires very small cells adjacent to the wall. For this application a geometric expansion gives an improved cell distribution. A geometric progression is defined as the series

$$\Delta_1,\ \Delta_1 k,\ \Delta_1 k^2,\ \Delta_1 k^3,\ \ldots\ ,\ \Delta_1 k^{n-1} \qquad \textit{Equation 2-92}$$

for which the sum of the first n terms is given by

$$s = \sum_{i=1}^{n} \frac{\Delta_1 \left(k^i - 1\right)}{k-1} \qquad \textit{Equation 2-93}$$

Here Δ_1 is the thickness of the first cell, k is the expansion factor and n is the number of cells.

For ease of generating a grid with geometric and uniform distributions the XFRAC, YFRAC and ZFRAC arrays were written using a simple FORTRAN program and then imported into the Q1 file.

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

3. METHODOLOGY

3.1 Introduction

During the course of this study, several techniques were developed to aid the modelling of flow over rough boundaries and the subsequent analysis.

For all of the investigations reported, periodic boundary conditions were employed to achieve fully developed flow, while minimising computational resources. Two methods of implementing periodic boundary conditions within the PHOENICS code were developed, one of which is recommended for use.

In addition to the usual procedure of monitoring the history of residuals in the finite volume equations, and monitoring each variable at specific locations, other convergence criteria were developed which had a more obvious physical basis. These were derived from an analysis of the Reynolds equations and were used to:

1. Confirm the theoretically linear cross-stream distribution of period averaged shear stress.

2. Confirm that the Reynolds equations employed within PHOENICS satisfied a strict force balance.

To analyse flow over rough boundaries, several techniques were developed to obtain the equivalent sand roughness or k_s value from a standard velocity profile or resistance equation.

Finally a new technique to calculate the position of an equivalent plane wall to replace a rough boundary was developed. This involved a novel application of streamwise force moments over a rough bed.

Discussion of these techniques forms the subject of this chapter.

3.2 Periodic Boundary Conditions

3.2.1 Introduction

In duct and open channel flow studies (both physical and computational) the area of interest usually lies in the fully developed region. Fully developed flow is an asymptotic condition and various criteria have been employed to define an approximate flow development length. For example Gessner's (1981) criterion of 60 times the hydraulic diameter employed by Nezu and Rodi (1986) for flow in smooth open channels was probably at the low end of the range.

The thickness of a turbulent boundary layer over a smooth plate at the streamwise location x was given by Schlichting (1955) as

$$\frac{\delta}{x} = 0.37\left(\frac{U_\infty x}{\nu}\right)^{-1/5} \qquad \textit{Equation 3-1}$$

where U_∞ is the free stream velocity. This can be used to estimate the boundary layer thickness on a rough wall by applying the work of Perry and Joubert (1963) in which it was shown that the effect of roughness can be accounted for by using a modified viscosity which increases with higher degrees of roughness. Equation 3-1 with a modified (larger) viscosity predicts that a rough wall boundary layer would develop more rapidly, as would be expected given the larger cross-stream turbulent velocity scale. For a given duct, the streamwise location at which boundary layers from each wall meet can therefore be determined, though a further length of duct is then required for the velocity profiles to fully develop. To the author's knowledge there has been no published application of this method as a means of predicting the required development length, however Samuels (1997) used a similar technique to predict the development length for the SERC flood channel facility at Hydraulics Research Ltd, Wallingford.

In CFD the provision of a development length requires additional cells in the primary flow direction that are unused in the final analysis. This makes corresponding demands on computer storage and run time which may be alleviated by the following alternative methods:

1. using a parabolic flow approximation.

2. inputting accurate profiles for all variables at the inlet to the domain.

3. using a *periodic* or *cyclic* boundary condition.

Parabolic flow approximation

Flow may be classed as parabolic if the primary velocity at every point in the fluid is in one direction only, the curvature of streamlines is minimal and the convective/diffusive transport in the cross-stream direction is negligible. Such flows may be simulated by a single slab solution procedure which greatly reduces the storage and processing requirements. Flows over roughness elements however do not satisfy these criteria but require influences to propagate in all spatial directions and are thus elliptic in nature.

Inlet velocity profiles

The difficulty in specifying correct inlet profiles is that universal distributions are generally limited to the primary velocity and only known for a very limited number of flows. For example, in a simple two-dimensional boundary layer on a smooth wall, the primary velocity distribution may be approximated by the $^1/_7{}^{th}$ power law

$$U(y)=\left(y/h\right)^{\frac{1}{7}}\times\hat{U} \qquad \textit{Equation 3-2}$$

or the universal velocity profile (Equation 2-21). However this method is inherently unsuitable for investigating unknown flow fields. Never-the-less, approximate distributions (such as Equation 3-2) may be usefully employed to give the initial values in an iterative procedure.

Periodic boundary conditions

In the application of periodic boundary conditions the value of each variable at the outlet is used as the inlet boundary condition for the next stage in the computation. Thus the computational domain consisting of an integer number of periods can be viewed as a short length of duct repeated indefinitely. This method allows the flow to develop naturally and works equally well for all domains which have a repetitive streamwise distribution.

Patankar *et al* (1977) describe the method mathematically in the form:

$$U(x,y)=U(x+l,y)=U(x+2l,y) \qquad \text{etc.} \qquad \textit{Equation 3-3}$$

for the velocity field and for the pressure field as:

$$P(x,y)-P(x+l,y)=P(x+l,y)-P(x+2l,y) \qquad \text{etc.} \qquad \textit{Equation 3-4}$$

where l is the period length. It is therefore the pressure drop which is periodic, rather than the pressure. The pressure may be thought of as being the sum of two components, a "local" pressure, $P*$ and a linear "reduced" pressure gradient $\mathrm{dP'}/_{\mathrm{dx}}$

$$P(x,y)=P^{*}(x,y)+P'_{\mathrm{in}}+x.\mathrm{dP'}/_{\mathrm{dx}} \qquad \textit{Equation 3-5}$$

where P'_{in} is the reduced pressure at inlet. The local pressure $P*$ is periodic

$$P^{*}(x,y)=P^{*}(x+l,y)=P^{*}(x+2l,y) \qquad \text{etc.} \qquad \textit{Equation 3-6}$$

Within the PHOENICS code, cyclic boundary conditions may be written and controlled by the user or the in-built facility may be employed.

3.2.2 User Controlled Cyclic Boundary Conditions

In PHOENICS a duct flow is most effectively modelled by defining a source of mass flux (and associated momentum flux) at inlet and a pressure distribution at exit (Section 2.3.4). Due to the slab-wise solution procedure employed by PHOENICS, a converged solution is most quickly reached if the predominant flow direction is chosen to be the same as the direction in which the slabs of cells are solved, i.e. the z direction. (e.g. PHOENICS library case No. 290).

A duct based on these boundary conditions may be made periodic by simply extracting the partly developed velocity and turbulence profiles at exit and repeating the calculation with these profiles as inlet boundary conditions. The exit pressure profile is set equal to the pressure distribution at inlet, minus the pressure drop between the duct entrance and exit. It is convenient to set the average value of the outlet pressure distribution equal to zero, allowing the pressure distribution throughout the rest of the domain to adjust to this datum. A similar method of implementing periodic boundary conditions was employed by Beale (1989) for laminar flow through a plate fin heat exchanger.

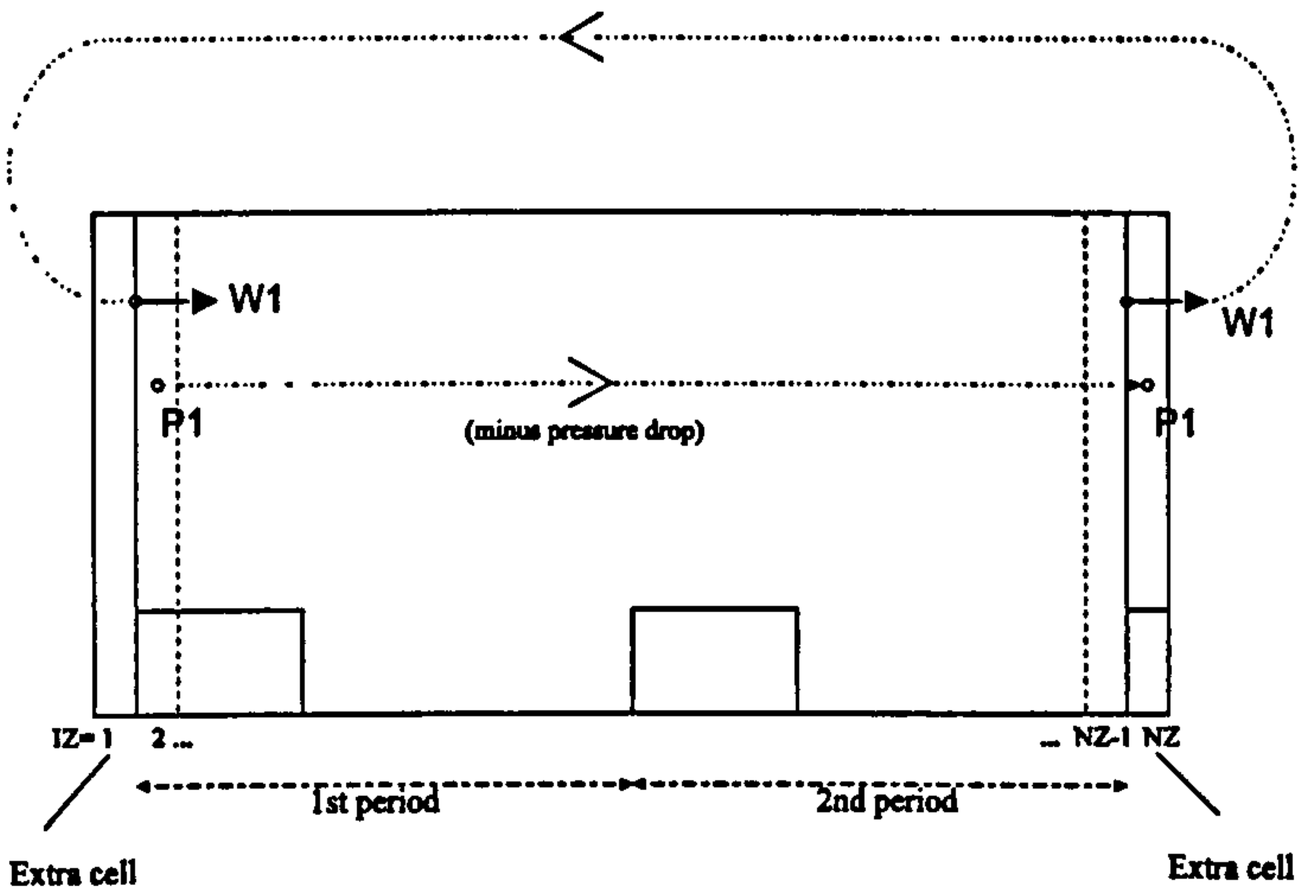


Figure 3-1: Flow rate boundary condition location of recycled variables

For flow over periodically spaced roughness elements, Figure 3-1 shows that an extra cell needs to be added at both inlet and outlet to ensure that the variables are recycled to the correct locations. Thus the streamwise velocity is recycled from IZ=NZ−1 to IZ=1 and the pressure and turbulence quantities from IZ=2 to IZ=NZ. Note that the W1 velocity cannot be calculated on the downstream boundary (Section 2.3.5).

FORTRAN code to transfer the arrays of W1, P1, KE and EP may be written by the user in the GROUND subroutine of PHOENICS (Appendix 8).

Testing the user controlled cyclic boundary conditions

Using the long angle (case XX) roughness configuration described by Schlichting (1936), the cyclic boundary conditions were tested for periodically fully developed flow and efficient convergence. Figure 3-8 shows example profiles of streamwise velocity (W1) taken at the inlet face of the first period and the outlet face of the second period. The velocity profile at inlet was less developed then the profile at exit, indicating that further recycles were required to achieve fully developed flow. The convergence of each variable is indicated by the history of spot values throughout the solution procedure, given in Figure 3-9 and 3-10 which show the monitoring values of the W1 and P1 variables at cell IZ=NZ/2, IY=NY/2. The discontinuities show when recycling took place. The frequency of recycling (set by the variable NF) was adjusted to ensure that any oscillations triggered by the recycling had decayed before the next recycle. In the tests described, all the variables were recycled every 3,000 sweeps.

In order to reduce the large number of iterations required, various schemes were investigated in which recycling of particular variables, for example pressure, was switched off for part of the solution procedure. The effect of relaxation was also investigated to give an optimum rate of convergence.

A more fundamental problem associated with this method is indicated by the profiles of the pressure variable at the start and end of the domain (Figure 3-11). For fully developed flow the profiles should be identical in shape, separated by the pressure drop ΔP. The average value of the exit pressure profile should also be zero. An examination of the code during the solution procedure revealed that the exit pressure distribution was calculated correctly within GROUND but that the coefficient used to fix[1] this distribution to the required field values was too low. As a consequence the inlet pressure profile also developed incorrectly.

Increasing the exit pressure coefficient from 1 to 1000 caused the solution to oscillate, as indicated by the spot values of P1 and W1 in Figures 3-12 and 3-13. Further, profiles of W1 and P1 at inlet and outlet (Figures 3-14 and 3-15) show that fully developed flow was again not achieved, though the inlet distribution of pressure was more realistic.

Finally, after prolonged investigation, it was concluded that a pressure coefficient could not be found which was able to give a converged solution of fully developed flow for the domains tested and further development of this method was discontinued.

3.2.3 PHOENICS Controlled Cyclic Boundary Conditions

The PHOENICS XCYCLE command was designed for modelling flow in circular cross-sections using a segment defined in the cylindrical co-ordinate system. The circumferential direction is defined as the x direction, which therefore prescribes the direction in which recycling takes place. In the present application, recycling was required in the primary flow direction and therefore this had to be the x direction.

---

[1] See Section 2.3.4 for details on pressure boundary conditions.

In the XCYCLE facility, the flow domain is bounded by two x-planes which are linked by the convection and diffusion coefficients in the finite volume equation (Equation 2-62 and Appendix 5). In the present rough wall application the x-planes are an integer number of periods apart. The values of each variable within a cell in the first x-slab are thus dependent on its immediate neighbours and the corresponding cell in the last x-slab. Similarly the values of variables in the last x-slab are dependent on the current values of the first x-slab. The user has no control over this process and therefore it is the "local" pressure P^* which is recycled and not P. Therefore the "reduced" pressure gradient that drives the flow has to be modelled by other means. This is achieved by applying a source of momentum flux per unit volume to each cell in the domain, equivalent to applying a body force per unit volume. In this method, the magnitude of the "local" pressure[2] P^* is completely independent of any external boundary conditions and simply requires a reference datum to be specified at some arbitrary location within the domain. For incompressible flow the value of the datum is most conveniently defined to be zero. Generally this was achieved here by setting P^* to zero in one cell in the first x-slab.

An advantage of this method is that there are no discontinuities within the re-cycling procedure and no additional FORTRAN coding is required. However, to model a duct with a specified flow rate, adjustment of the momentum flux source is required. Initially this was achieved by comparing the latest computed value of the flow rate with the required value and adjusting the pressure gradient in a linear manner:

$$\frac{dP'}{dx}_{new} = \frac{dP'}{dx}_{old} \times \frac{\text{Required flow rate}}{\text{Computed flow rate}} \qquad \textit{Equation 3-7}$$

Linear relaxation was also incorporated into this procedure, which was performed at a specified frequency of iterations using GROUND coding (Appendix 11). However, if

---

[2] The term "local" pressure will be simplified to "pressure" for the remainder of the thesis.

the friction factor is independent of Reynolds number, the pressure gradient is proportional to the square of the average velocity and the effect of using a linear relationship is therefore to increase the degree of relaxation.

Due to oscillations in the pressure gradient obtained using this method (Chapter 5), an alternative procedure was developed. Starting with an estimated source of momentum flux, a nearly converged solution was obtained (error between driving and resistive forces of 5%). The required momentum flux source was then more accurately calculated by

$$\frac{dP'}{dx}_{new} = \frac{dP'}{dx}_{old} \times \left(\frac{\text{Required flow rate}}{\text{Computed flow rate}} \right)^2 \qquad \textit{Equation 3-8}$$

and the solution continued. After confirming that the error between the required and computed flow rates was less than 5%, further iterations were performed until full convergence was achieved, as defined in Section 3.3.

The PHOENICS controlled method of cyclic boundary conditions was tested on flows in a smooth walled rectangular duct and also using the long angle roughness configurations of Schlichting (1936). In all tests momentum was conserved and accurate periodic distributions were obtained for all variables. The results for the smooth duct tests are presented in Section 6.2 and those for Schlichting's roughness plates XX, XXI and XXII are given in Section 4.1.

3.3 Convergence Criteria

Due to the iterative nature of computational fluid dynamics, convergence criteria are required to specify the desired degree of numerical accuracy. A converged solution is commonly accepted when the residuals of the finite volume equations fall below specified values, and when further iterations do not produce appreciable changes in the values of each variable at specific monitoring locations.

For the investigations reported here, additional convergence criteria were used based on the flow physics rather than on the numerics of the solution. This ensured that the

calculated flow was physically realistic and provided more enlightening evidence of numerical accuracy. The two criteria employed were:

1. That the period averaged shear stress profile in the transverse direction should be linear. The derivation of period averaged shear stress is discussed in Section 3.3.1.

2. That the driving force acting on the fluid should be equal and opposite to the sum of the resistive forces exerted by the duct boundary, to within 0.1%. The calculation of this force balance is discussed in Section 3.3.2.

3.3.1 Period Averaged Shear Stress

Governing equations

The Reynolds equations, using the eddy viscosity concept to model the Reynolds stresses, are given in Equation A3-9. For a two-dimensional flow in the x-y plane, the x component is

$$\frac{\partial}{\partial x}\left[\underbrace{P+\rho U^2-2\Gamma\frac{\partial U}{\partial x}+{}^2\!/_3\,\rho k}_{-\sigma_x}\right]=\frac{\partial}{\partial y}\left[\underbrace{-\rho UV+\Gamma\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)}_{\tau_{yx}}\right] \qquad \textit{Equation 3-9}$$

or

$$\frac{-\partial\sigma_x}{\partial x}=\frac{\partial\tau_{yx}}{\partial y} \qquad \textit{Equation 3-10}$$

Applying the continuity equation for incompressible flow gives (Appendix 3):

$$\frac{\partial}{\partial x}\left[\underbrace{P+\rho U^2-\Gamma\frac{\partial U}{\partial x}+{}^2\!/_3\,\rho k}_{-\sigma'_x}\right]=\frac{\partial}{\partial y}\left[\underbrace{-\rho UV+\Gamma\frac{\partial U}{\partial y}}_{\tau'_{yx}}\right] \qquad \textit{Equation 3-11}$$

It is in this form that PHOENICS solves the momentum equations. Note that one of the $\Gamma\,{}^{\partial U}\!/_{\partial x}$ terms has disappeared from the left hand side of Equation 3-10, as has the $\Gamma\,{}^{\partial V}\!/_{\partial x}$ term from the right hand side. Thus, although this equation is exact, it no

longer describes the complete set of stresses over a finite control volume. The terms within the derivatives will be denoted the "reduced" stresses:

$$\tau'_{yx} = -\rho UV + \Gamma \frac{\partial U}{\partial y} \qquad \textit{Equation 3-12}$$

$$-\sigma'_x = P + \rho U^2 - \Gamma \frac{\partial U}{\partial x} + \tfrac{2}{3}\rho k \qquad \textit{Equation 3-13}$$

and

$$\frac{\partial \tau'_{yx}}{\partial y} = \frac{-\partial \sigma'_x}{\partial x} \qquad \textit{Equation 3-14}$$

Physical interpretation

1. Consider the general flow given by $U = U(y)$ as shown in Figure 3-2. If the y momentum flux is positive there is a net transfer of momentum in the +ve y direction, from level 1 to level 2. Therefore the fluid at level 2 will tend to accelerate as though it were dragged in the positive x direction by the lower fluid and the fluid at level 1 will experience a resistance in the −ve x direction. This is equivalent to a −ve shear stress on the +ve y face of the fluid at level 1. Thus the total convective and diffusive flux in the positive y direction represents a negative reduced shear stress in PHOENICS, i.e.

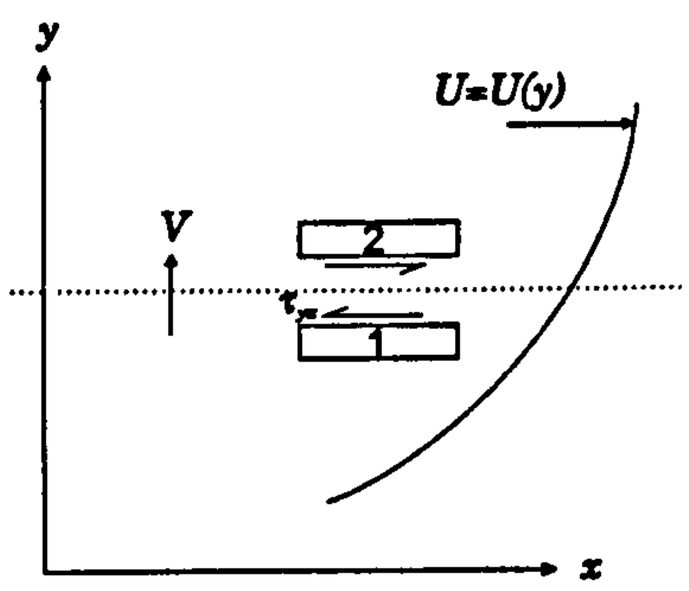


Figure 3-2: Flux and shear stress

$$J_y = -\tau'_{yx}.$$

2. Equation 3-10 may also be derived by examining the force balance over a finite control volume as shown in Figure 3-3.

   Summation of the forces gives

$$\delta x(\tau_{yx} + \delta\tau_{yx}) + \delta y(\sigma_x + \delta\sigma_x) - \delta x.\tau_{yx} - \sigma_x.\delta y = 0$$

   i.e.

$$\delta\tau_{yx}.\delta x + \delta\sigma_x.\delta y = 0$$

or

$$\partial\tau_{xy}/\partial y = -\partial\sigma_x/\partial x$$

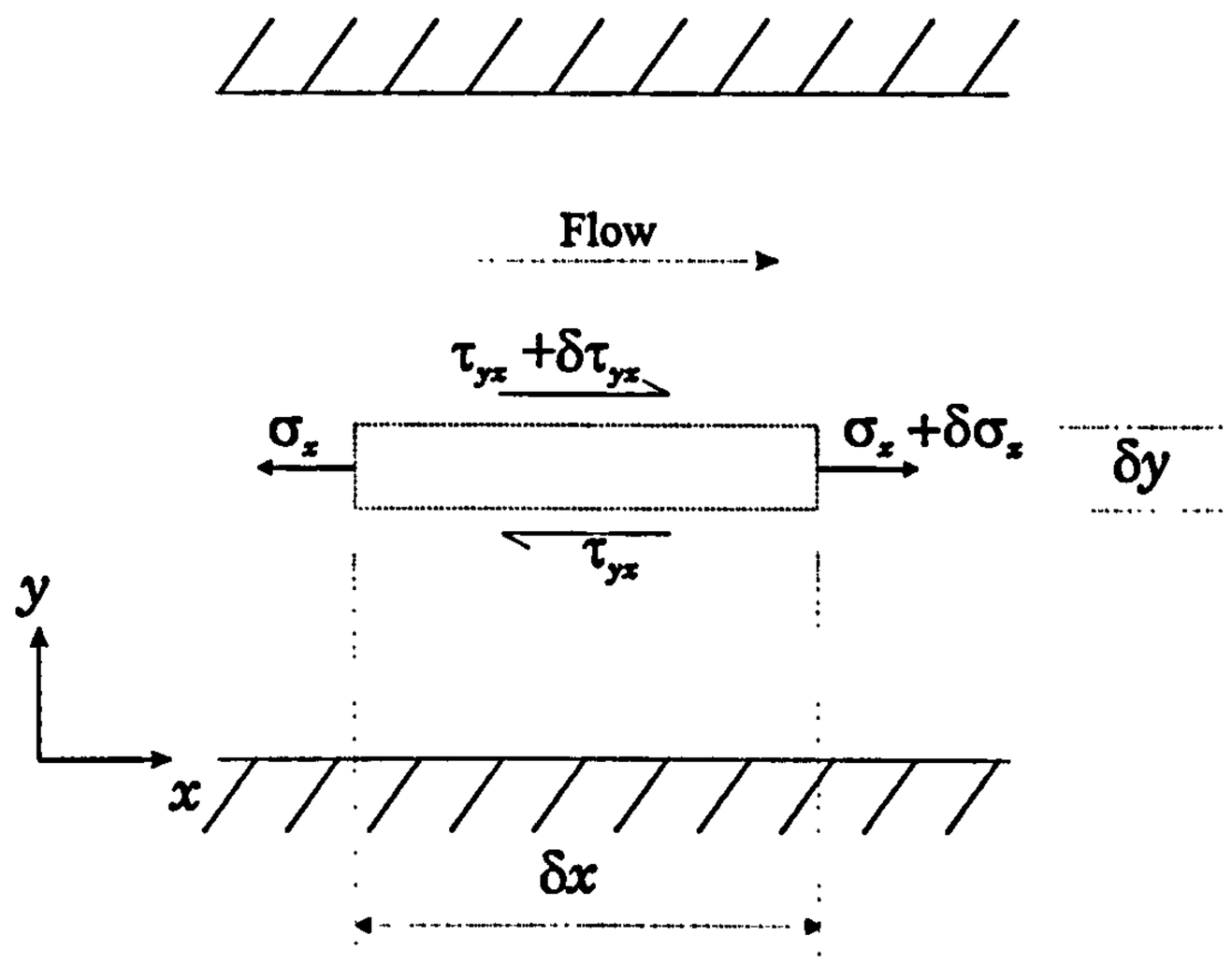


Figure 3-3: Shear stress control volume.

3. For periodically developed flow, the terms $\rho U^2 - \Gamma\frac{\partial U}{\partial x} + \frac{2}{3}\rho k$ in Equation 3-13 are identical on the upstream and downstream faces of the period. Equation 3-14 then reduces to

$$\frac{d\overline{\tau'_{yx}}}{dy} = -\frac{\Delta P}{l} \qquad \textit{Equation 3-15}$$

where $\overline{\tau'_{yx}}$ is the period averaged value of τ'_{yx} and $-\Delta P$ is the pressure drop over a period length. Thus the gradient of the period averaged reduced shear stress profile is constant and equal to the average pressure gradient over one period.

4. The reduced stress τ'_{yx} is approximately equal to the true shear stress in the majority of the flow region. However, immediately above the roughness block and in the recirculation region, the term $\partial N/\partial x$ will be significant and τ'_{yx} will not be an accurate representation of the shear stress.

5. It is interesting to note that when the omitted term $\Gamma \partial N/\partial x$ was included in the calculation of τ_{yx} then the profile of shear stress was found to be no longer linear. This is shown in Figure 3-16 using the roughness test RCDA3 (Section 5.3) as an example. The reason for this discrepancy can be seen by plotting a period distribution of the transverse velocity V1 at a y plane above the roughness element. Figure 3-17 shows a very high value of V1 behind the leading edge of the roughness element. This area of the domain does not contain sufficient cells to accurately determine the distribution of the peak value. When the gradient $\partial N/\partial x$ is calculated (Figure 3-18 and 3-19), gross errors will be obtained. Although Figure 3-17 shows continuity to be satisfied over a period, the integrated value of $\partial N/\partial x$ in Figure 3-18 is non zero.

Implementation within PHOENICS

The reduced stress τ'_{yx} and thus its period averaged value $\overline{\tau'_{yx}}$ was found by two methods.

1. Most simply, each term in Equation 3-12 was calculated by extracting the values of U, V, Γ and the co-ordinates for each cell. The gradients were obtained by taking the difference between adjacent cells. Integrating τ'_{yx} over one period length for each y plane gave a distribution of $\overline{\tau'_{yx}}$. The GROUND subroutine MANEFF (Appendix 11) shows how this was implemented within PHOENICS. This method uses the same principles as those in the central difference scheme (Appendix 5).

2. In calculating the flow field, PHOENICS uses the hybrid scheme to solve Equation 3-11. In recognition of this, the reduced stress τ'_{yx} was also calculated using the convection and diffusion coefficients used within the finite volume equations (Equation 2-62 and Appendix 5).

Implementation using convection and diffusion coefficients

Using the general variable Φ, Patankar (1980 Equation 5.54) gives the integrated total flux for the east face of a scalar control volume as

$$J_e = \rho U_e \Delta y \Phi_P + a_e(\Phi_P - \Phi_E) \qquad \textit{Equation 3-16}$$

The integrated total flux for variable U over the north face ('en' in Figure (ii)) of the U1 velocity cell is therefore

$$J_n = \rho V_{en} \Delta x U_P + a_N(U_P - U_N) \qquad \textit{Equation 3-17}$$

where $J_n = \int J_y . dx$ or $-\int \tau'_{xy} . dx$ and a_N is the convection and diffusion coefficient in the finite volume equations. Note that due to the staggered grid, U_P and U_N are located on faces 'e' and 'en'. Thus the average shear stress over one roughness period is

$$\begin{aligned} \overline{\tau'_{yx}} &= \frac{1}{l}\sum \tau'_{yx} . \Delta x \\ &= -\frac{1}{l}\sum_{IX=1}^{PNX} J_n \\ &= \frac{1}{l}\sum_{IX=1}^{PNX}\left[a_N(U_N - U_P) - m^U_{en} . U_P\right] \end{aligned} \qquad \textit{Equation 3-18}$$

where Δx is the length of one cell and PNX is the number of cells in the x direction over a roughness period of length l. Using chapters 4 and 5 of Rosten and Spalding (1987) and the conventions described in Notation, then the remaining terms in Equation 3-18 have the following meanings:

U_P is the U1 velocity of the P cell, located at the centre of face 'e', in accordance with the staggered cell arrangement (Appendix 6). Similarly, U_N applies to the U1 velocity of the cell N.

m^U_{en} is the mass flux (kg s^{-1}) across the north face of the U1 velocity cell, i.e. the face 'en' in Figure (ii) : $m^U_{en} = {(m_n + m_{nE})}/{2}$,

where $m_n = \rho V_n A_n$ and $m_{nE} = \rho V_{nE} A_{nE}$.

a_N is the coefficient which represents the combined effect of convection and molecular/ turbulent diffusion through the north face of the U1 velocity cell: $a_N = \max\left(0, d_{en}^U - \alpha\left|m_{en}^U\right|\right) + \max\left(0, -m_{en}^U\right)$

where $\alpha = 0.5$ for the hybrid scheme and d_{en}^U is the diffusion coefficient (kg s^{-1}) across the north face of the U1 velocity cell, i.e. $d_{en}^U = {(d_n + d_{nE})}/{2}$ where $d_n = {\Gamma_n A_n}/{|PN|}$ and $d_{nE} = {\Gamma_{nE} A_{nE}}/{|E(NE)|}$. The exchange coefficient, Γ (kg m^{-1} s^{-1}) may be determined by arithmetic or harmonic averaging. It is the latter which is used for the momentum equations, for example

$$\Gamma_n = \frac{2|PN|}{\left({|ns|}/{\Gamma_P} + {|n(nN)|}/{\Gamma_N}\right)}$$

In PHOENICS the convection flux (m) is stored as an *inflow* to the relevant velocity cell. Thus for the U1 velocity cell P, the flow across face 'en' may be either

1. Flowing from north to south: thus the flux is in the -ve y direction and stored as an inflow to the U1 cell P across its north face. The PHOENICS variable is LD11 (when NDIREC=1) and the user GROUND variable is CUN (convection of U1 over north face).

2. Flowing from south to north: thus the flux is in the +ve y direction and stored as an inflow to the U1 cell N across its south face. The PHOENICS variable is LD12 (when NDIREC=1) and the user GROUND variable is CUS (convection of U1 over south face).

These GROUND variables are extracted from the F-array in group 8 of GROUND. The diffusion coefficient through the north face of the U1 velocity cell (d_{en}^U) is extracted from group 9 of GROUND using the PHOENICS variable LAN (when NDIREC=1) and put into the user GROUND array DUN.

The rest of the coding to obtain $\overline{\tau'_{yx}}$ is put into the subroutine EFFSS, called from section 8, group 19 of GROUND (Appendix 11). This first calculates the total convection flux array (CON= m_{en}^{U}) by summing the arrays CUS and CUN, taking into account the negative direction of CUN. Next the terms

$$\mathrm{SDN}=(U_{\mathrm{N}}-U_{\mathrm{P}})\left[\max\left(0,d_{en}^{U}-\alpha\left|m_{en}^{U}\right|\right)\right] \qquad \textit{Equation 3-19}$$

$$\mathrm{SCN}=(U_{\mathrm{N}}-U_{\mathrm{P}})\left[\max\left(0,-m_{en}^{U}\right)\right] \qquad \textit{Equation 3-20}$$

$$\mathrm{SFN}=-m_{en}^{U}\times U_{\mathrm{P}} \qquad \textit{Equation 3-21}$$

are calculated, noting that $\left[\max\left(0,-\mathrm{m}_{en}^{U}\right)\right]=\mathrm{CUN}$. These are combined to give

$$\mathrm{SHR}=\mathrm{SDN}+\mathrm{SCN}+\mathrm{SFN}$$

or

$$\mathrm{SHR}=(U_{\mathrm{N}}-U_{\mathrm{P}})\left[\max\left(0,d_{en}^{U}-\alpha\left|m_{en}^{U}\right|\right)+\max\left(0,-m_{en}^{U}\right)\right]-m_{en}^{U}U_{\mathrm{P}} \qquad \textit{Equation 3-22}$$

The shear force over the north face of each U1 velocity cell (SHR = $-J_{\mathrm{n}}$) is then summed over the complete period of the rough duct and divided by the period length (Equation 3-18).

Conclusions

As can be seen from Figure 3-16 there was no discernible difference between methods 1 and 2 of calculating $\overline{\tau'_{yx}}$ for the application considered. The reason for this is that although PHOENICS uses the hybrid scheme, the absolute values of cell Peclet numbers were below 2 for the majority of cells and therefore the numerical procedure effectively used the central difference scheme for its cross-stream momentum flux calculations.

3.3.2 Force Balance Over One Period

The forces acting on a fluid flowing over a rough boundary for one period are shown in

Figure 3-4:

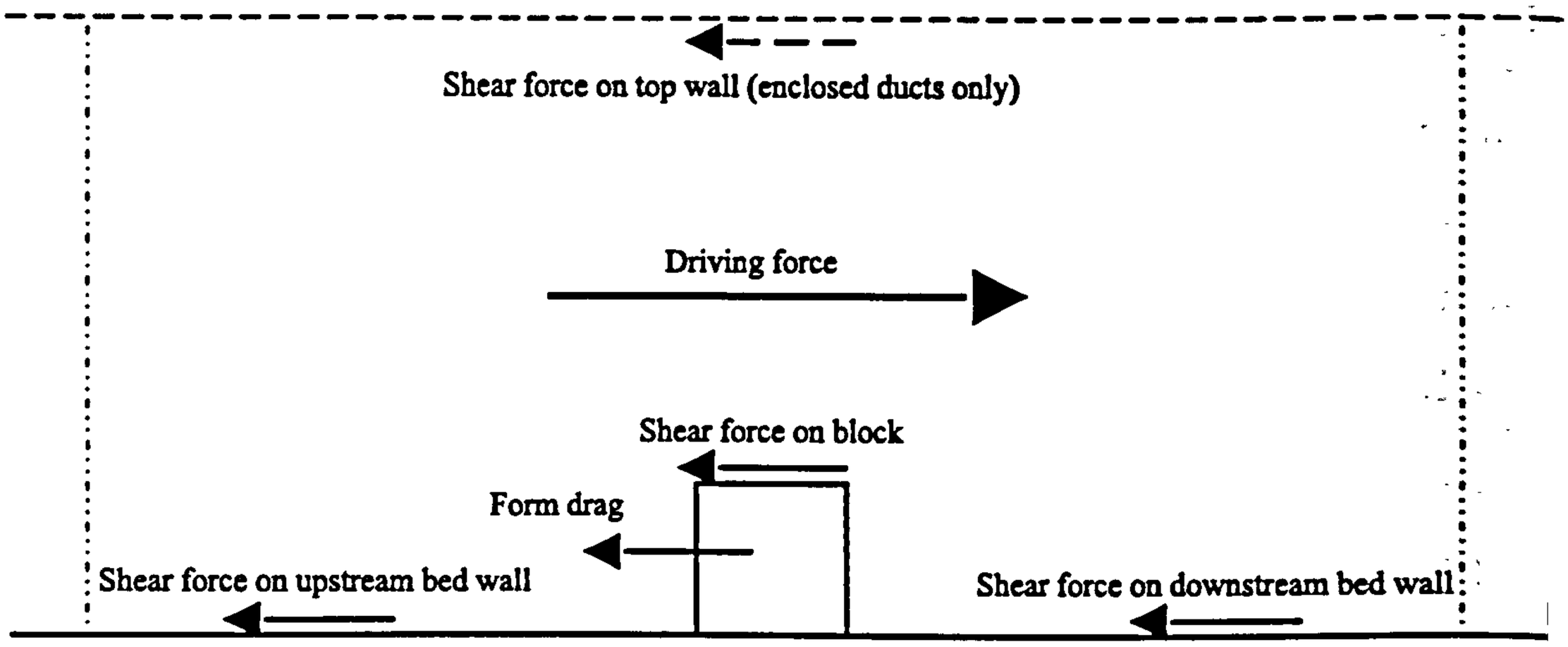


Figure 3-4: Force balance over one period

The driving force is modelled by a source of streamwise momentum flux per unit volume added to each cell in the flow domain. This is equivalent to a pressure drop ΔP over the period length, from which the driving force may be calculated as

$$\text{Driving force} = \Delta P \times \text{cross-sectional area of duct} \qquad \textit{Equation 3-23}$$

The resistive force is the sum of the bed shear forces, the shear force on the top of the roughness element and the form drag (or pressure induced force) on the roughness element.

Wall Shear Forces

The shear forces on the fluid due to the top wall (enclosed ducts only), the bed and the horizontal surface of the roughness elements are calculated in the GROUND subroutine SHFOR (Appendix 11). Use is made of the PHOENICS variable PVSTRS (a block location of the F-array; see Appendix 1) which stores the magnitude of the shear stress multiplied by the fluid density. For the two-layer model where the first cell is located within the laminar sub-layer, this is calculated internally within PHOENICS from

$$\left|\tau_w/\rho\right| = \nu \frac{\left|\left(U1_{(IX=W,JY=1)} + U1_{(IX=P,JY=1)}\right)\right|/2}{1/2\,\Delta y} \qquad \textit{Equation 3-24}$$

where Δy is the y dimension of the first cell from the wall, P is the current IX cell and W is the neighbouring cell to the west.

The direction of the shear force is found by examination of the direction of the local average U1 velocity, $\left(U1_{(IX=W,JY=1)} + U1_{(IX=P,JY=1)}\right)/2$. If this is greater than zero then the

shear force exerted by the boundary on the fluid will act in the –ve x direction. The force on each surface is obtained by summing the product of the shear stress and cell area for each cell adjacent to the wall under consideration.

Pressure Force on a Block

The difference between the integrated pressures on the upstream and downstream faces of the block gives rise to a hydrodynamic force acting on the block. This force also acts on the fluid, causing a "bluff body" form drag or resistance force.

Using the differential form of the Reynolds equations in the x direction (Equation 3-9), the direct stress in the x direction is

$$-\sigma_x = P + \rho U^2 - 2\Gamma\frac{\partial U}{\partial x} + \tfrac{2}{3}\rho k \qquad \textit{Equation 3-25}$$

Due to the staggered grid arrangement employed by PHOENICS, the pressure is not calculated at the boundary of a cell and therefore Equation 3-25 cannot be evaluated next to the solid boundary. To find the direct stress acting on the wall, a force balance must be taken over a control volume consisting of the cells adjacent to the solid boundary.

Integrating Equation 3-10 with respect to x and y gives

$$(\tau_{yx,2} - \tau_{yx,1})\Delta x = -(\sigma_{x,2} - \sigma_{x,1})\Delta y$$

where $\tau_{yx,1}$ and $\tau_{yx,2}$ occur at co-ordinates y and $y + \Delta y$ respectively and $\sigma_{x,1}$ and $\sigma_{x,2}$ occur at co-ordinates x and $x + \Delta x$. Rearranging gives

$$(\tau_{yx,2} - \tau_{yx,1})\Delta x + (\sigma_{x,2} - \sigma_{x,1})\Delta y = 0$$

and substituting

$$\tau_{yx,1} = \tau_s \quad \text{and} \quad \tau_{yx,2} = \tau_n$$
$$\sigma_{x,1} = \sigma_P \quad \text{and} \quad \sigma_{x,2} = \sigma_e$$

gives

$$(\tau_n - \tau_s)\Delta x + (\sigma_e - \sigma_P)\Delta y = 0 \qquad \textit{Equation 3-26}$$

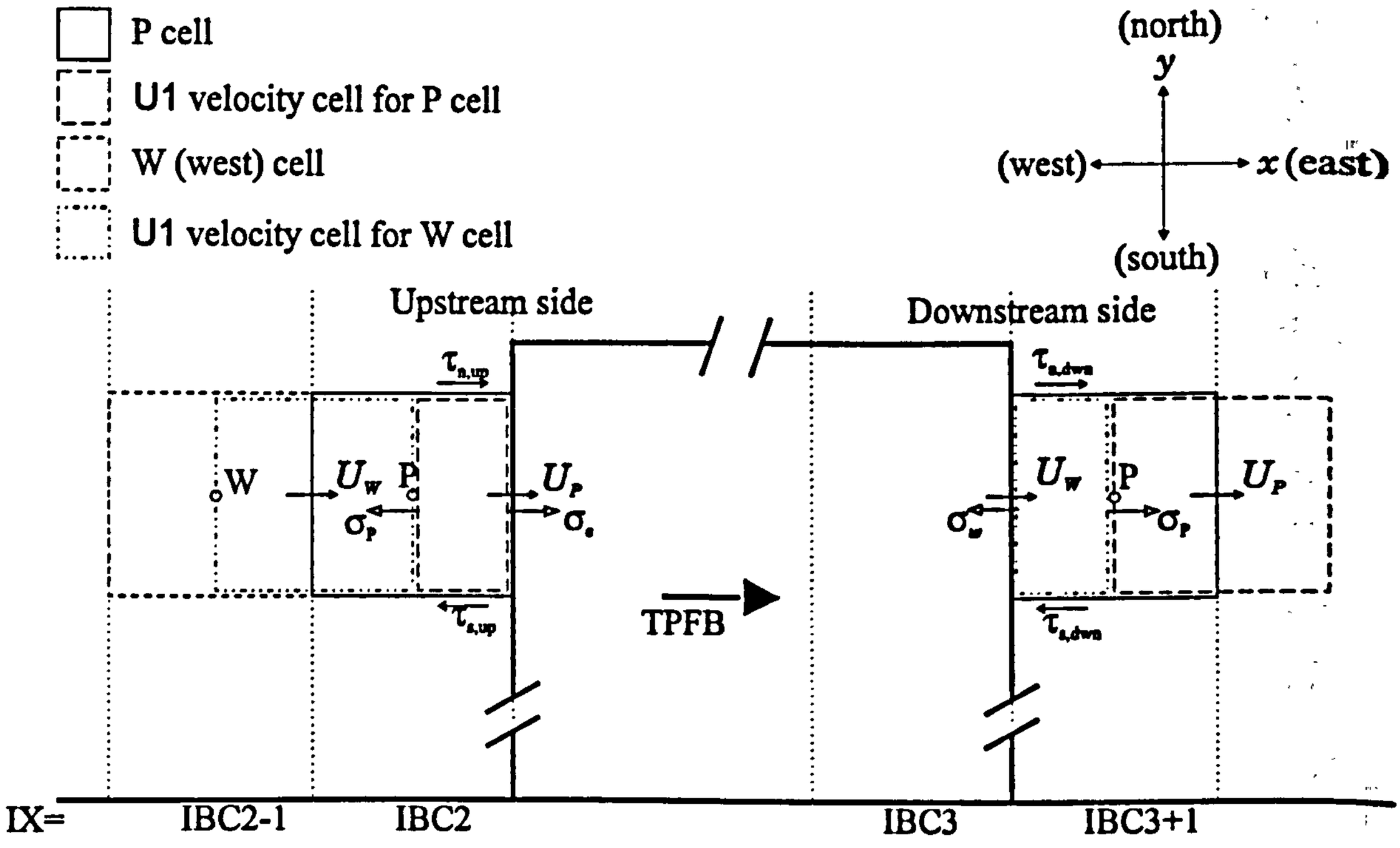


Figure 3-5: Pressure forces over a block

Replacing τ_s with $\tau_{s,up}$ and τ_n with $\tau_{n,up}$, Equation 3-26 gives the force balance for the half velocity cell upstream of the block in Figure 3-5.

Summing over the total number of cells in the vertical face of the block (BNY) gives:

$$\sum_{IY=1}^{BNY}\left[(\sigma_e-\sigma_P)\Delta y\right]+\left(\tau_{n(IY=BNY),up}-\tau_{s(IY=1),up}\right)\Delta x=0 \qquad \textit{Equation 3-27}$$

as $\tau_{n,up(IY=I)}=-\tau_{s,up(IY=I+1)}$.

Using the shear stress acting on adjacent cells as an estimate, the effect of $\left(\tau_{n(IY=BNY)}-\tau_{s(IY=1)}\right)\Delta x$ on the total block force was found to be of the order of 10^{-6} % and the second term in Equation 3-27 was therefore regarded as negligible. Therefore

$$\sum_{IY=1}^{BNY}\left(\sigma_p\Delta y\right)=\sum_{IY=1}^{BNY}\left(\sigma_e\Delta y\right) \qquad \textit{Equation 3-28}$$

and the total force exerted by the upstream face of the block on the fluid is

$$\sum_{IY=1}^{BNY}\left(\sigma_e\Delta y\right)=\sum_{IY=1}^{BNY}\left(\sigma_p\Delta y\right)\Big|_{up}=\mathrm{TPFU} \qquad \textit{Equation 3-29}$$

in the +ve x direction.

By the same method, the total force exerted by the downstream face of the block on the fluid in the +ve x direction is

$$-\sum_{IY=1}^{BNY}(\sigma_w \Delta y) = -\sum_{IY=1}^{BNY}(\sigma_P \Delta y)\Big|_{dwn} = \mathrm{TPFD} \qquad \textit{Equation 3-30}$$

and the total force on the fluid is

$$\mathrm{TPFB} = \mathrm{TPFU} + \mathrm{TPFD} \qquad \textit{Equation 3-31}$$

Evaluation of Direct Stress

The direct stresses σ_P in Equation 3-29 and Equation 3-30 were evaluated using the convection and diffusion coefficients used within the discretization equations in a similar manner to that used in finding the reduced shear stress in Section 3.3.1. Using the reduced direct stress (Equation 3-13) to be consistent with the equations employed by PHOENICS and

$$J_P = \int J_x . dy \qquad \textit{Equation 3-32}$$

where

$$J_x = \rho U^2 - \Gamma \frac{dU}{dx} \qquad \textit{Equation 3-33}$$

from Patankar (1980, Equation 5.49a), then

$$\sum_{IY=1}^{BNY}(-\sigma_P \Delta y)\Big|_{up} = \sum_{IY=1}^{BNY}\left[\left(P + \tfrac{2}{3}\rho k\right)\Delta y + J_P\right] \qquad \textit{Equation 3-34}$$

or

$$\sum_{IY=1}^{BNY}(\sigma_P \Delta y)\Big|_{up} = -\sum_{IY=1}^{BNY}[\mathrm{P1}.\Delta y + J_P] \qquad \textit{Equation 3-35}$$

as the PHOENICS variable $\mathrm{P1} = P + \tfrac{2}{3}\rho k$. Thus from Equation 3-29:

$$\mathrm{TPFU} = \sum_{IY=1}^{BNY}(-\mathrm{P1}.\Delta y - J_P)\Big|_{up} \qquad \textit{Equation 3-36}$$

Also

$$-\sum_{IY=1}^{BNY}(\sigma_P \Delta y)\Big|_{dwn} = \sum_{IY=1}^{BNY}(\mathrm{P1}.\Delta y + J_P)\Big|_{dwn} = \mathrm{TPFD} \qquad \textit{Equation 3-37}$$

Implementation into PHOENICS

The pressure variables $P1_{up}$ and $P1_{dwn}$ were taken directly from the cell values stored in the F-array. The flux $J_{P,up}$ was found from the convection and diffusion fluxes flowing through the east face of the U1 velocity cell associated with the cell IBC2-1. Following from Equation 3-16 then the integrated total flux for variable U1 over face P is

$$J_P = \tfrac{1}{2}(U_W + U_P)\rho\,\Delta y U_W + a_P[U_W - U_P] \qquad \textit{Equation 3-38}$$

where U_W is the velocity on face 'w' and U_P is the velocity on face 'e' in Figure (ii).

Also

$$U_P = 0$$

$$a_P = \max\left(0, d_P^U - \alpha\left|m_P^U\right|\right) + \max\left(0, -m_P^U\right)$$

$$m_P^U = \rho A_E \left(U_W + U_P\right)/2$$

Again, for the hybrid scheme, α=0.5. The convection flux m_P^U and the diffusion coefficient d_P^U were obtained from PHOENICS using FORTRAN coding in the GROUND module (Group 8, section 8 and 9), and were put into one-dimensional user arrays for each y cell in the roughness element. These were named MBU for the convection flux and DBU for the diffusion coefficient . Thus

d_P^U = DBU(IY) = LAE where IX = IBC2 - 1 and IY = 1 to BNY

where LAE is the F-array location for diffusion through an east cell face. PHOENICS stores the convection coefficient as an inflow to a cell only, so for a flow in the negative x direction, i.e. flowing west into the east face (i.e. face P) of the U1 velocity cell associated with cell IBC2-1 in Figure 3-5, the convection coefficient is stored in the F-array at location LD11. For a positive flow, the convection coefficient across face P is stored as an inflow to the west face of the $^1/_2$ wall cell (velocity cell of IBC2) which has an F-array location of LD12. Thus

m_P^U = MBU(IY) = $-$LD11$_{IX=P-1}$ + LD12$_{IX=P}$ where P = IBC2 and IY = 1 to BNY.

The downstream flux $J_{P,dwn}$ was based on flow through the east face of the U_W half wall cell, i.e. the velocity cell for IX=IBC3;

$$J_p = \tfrac{1}{2}(U_W + U_P)\rho\,\Delta y \times U_W + a_P[U_W - U_P] \qquad \textit{Equation 3-39}$$

where again

$$a_P = \max\left(0, d_P^U - \alpha\left|m_P^U\right|\right) + \max\left(0, -m_P^U\right) \qquad \textit{Equation 3-40}$$

and U_W= 0.

Using DBD and MBD as arrays containing the diffusion coefficient and convection flux respectively for this downstream side then

d_P^U = DBD(IY) = LAE where IX = IBC3 and IY = 1 to BNY

m_P^U = MBD(IY) = $-$LD11$_{IX=P-1}$ + LD12$_{IX=P}$ where P = IBC3 and IY = 1 to BNY

Having found d_P^U and m_P^U then J_e was calculated in the GROUND subroutine PBLK for both the upstream and downstream faces. The total pressure force acting on the fluid by the block was calculated from Equation 3-31 and Equations 3-36 and 3-37.

3.4 Obtaining k_s Values

The equivalent sand grain roughness, k_s, may be calculated from the universal velocity profile (Equation 2-30), or alternatively from the resistance equation which is derived from it (Equation 2-34). For flow over roughness elements which have a physical height k_h approaching the depth of flow h, it becomes increasingly difficult to accurately determine a region of flow which conforms to the logarithmic law and therefore the universal velocity distribution becomes impossible to apply. However, even for the lowest values of h/k_h a k_s value may be calculated from the resistance equation, although the assumption on which this equation is derived, i.e. a constant value of k_s, becomes uncertain for this flow condition. It is the manner in which k_s varies with decreasing flow depth which was of primary interest in this study.

3.4.1 Velocity Profile Analysis

Method 1: In his roughness experiments Schlichting (1936) obtained k_s values by comparing the universal velocity profile for flow over roughness elements of height k_h

$$U/U_\tau = A' + B' \log \frac{y - y_0}{k_h} \qquad \textit{Equation 3-41}$$

with the distribution specific to flow over sand grains

$$U/U_\tau = A + B \log \frac{y - y_0}{k_s} \qquad \textit{Equation 3-42}$$

Schlichting used the values for A and B suggested by Nikuradse (1933), i.e. $A = 8.48$ and $B = 5.75$. However in this study the values recommended by Jayatilleke (1969) were employed, whereby $A = 8.27$ and $B = 5.62$. Rearranging Equation 3-41 gives

$$A' = U/U_\tau - B' \log \frac{y - y_0}{k_h} \qquad \textit{Equation 3-43}$$

Assuming that $B' = B$, then a value for the coefficient A' may be obtained for each data point in the velocity profile and a mean value $(\overline{A'})$ calculated. Comparing Equation 3-42 and Equation 3-43 and substituting in the mean value of A' gives

$$B \log \frac{k_s}{k_h} = A - \overline{A'} \qquad \textit{Equation 3-44}$$

from which a value of k_s may be obtained for each experiment.

Method 2: A similar technique to that used by Schlichting may be performed graphically. Writing Equation 3-42 as

$$U/U_\tau = A - B \log k_s + B \log (y - y_0) \qquad \textit{Equation 3-45}$$

a plot of U/U_τ against log (y-y_0) will give a straight line for the logarithmic region of the boundary layer with an intercept $C = A - B \log k_s$. The equivalent sand roughness is then

$$k_s = 10^{(A - C/B)} \qquad \textit{Equation 3-46}$$

Method 3: Linear least squares regression may be performed on Equation 3-42 to obtain k_s. Writing Equation 3-42 as

$$y' = k_s \times 10^x \qquad \textit{Equation 47}$$

where $y' = y - y_0$, $U^+ = U/U_\tau$ and $x = \dfrac{U^+ - A}{B}$ then the error is

$$e = y' - k_s \times 10^x \qquad \textit{Equation 48}$$

Standard least squares regression (e.g. Crow *et al* 1960) then gives the k_s value as

$$k_s = \frac{\sum_{i=1}^{n} y' \times 10^x}{\sum_{i=1}^{n} 10^{2x}} \qquad \textit{Equation 3-49}$$

Method 4: The previous methods of calculating k_s have assumed that the position of the bed datum, y_0, and thus the shear velocity on this plane, are known. Various methods were discussed in Section 2.1.7 for obtaining the bed position. An alternative method is to optimise the values of both y_0 and k_s in Equation 3-42 to fit the known coefficients and the data set. The datum position is then calculated in a similar manner to the progressive origin shift method suggested by Clauser (1956) (Section 2.1.7). This may be performed by non-linear least squares regression using the MATLAB analysis program. The details of this method are given in the MATLAB manual (Optimisation Toolbox User Guide, Grace 1990).

Discussion

1. The above methods assume that the gradient of the plot of U/U_τ against $\log(y - y_0)$ does follow a standard law (B=5.75 or 5.62). A check on this may be made by evaluating the gradient given by the data set.

2. By plotting the velocity profile, the region in which the logarithmic law applies may be observed and the correct range of the data set chosen. As can be seen from Figure 5-16 the velocity distribution is not logarithmic at low values of y where it is disturbed by the roughness elements. Neither does the logarithmic law apply near the plane of maximum streamwise velocity, where the assumptions used in deriving the Prandtl universal velocity profile are not valid (Section 2.1.2). The

logarithmic region was also found to vary according to the value of y_0 used. In applying these techniques to the present analysis only the data which fitted the logarithmic law was used.

3.4.2 Resistance Equation

The fully rough turbulent resistance equation corresponding to the universal sand roughness velocity profile is

$$\frac{1}{\sqrt{f}} = a - b\log\left(\frac{k_s}{2R_h}\right) \qquad \textit{Equation 3-50}$$

where Nikuradse recommended the values a = 1.74 and b = 2.0. French (1986) suggested that the coefficient a should be 1.5563 for open channel flow but maintained the same value of b. For this study, the coefficients a and b were determined from simulations over two-dimensional roughness elements using a two-layer model for flows in which the roughness scale was as small as could be computed and for which the k_s value calculated from the velocity profile was approximately constant with depth. In addition alternative coefficients were determined from a series of tests performed using the standard k-ε model with k_s specified within a wall function (Section 5.5).

For shallow flows, in which the velocity profile is not logarithmic, the k_s value may then be found by rearranging Equation 3-50:

$$k_s = 2R_h \times 10^{\frac{1}{b}\left(a - \frac{1}{\sqrt{f}}\right)} \qquad \textit{Equation 3-51}$$

3.5 Datum Position of Rough Wall

The traditional method of analysing the resistance of a rough wall is to effectively treat the wall as a plain boundary but to apply a roughness parameter in order to compensate for the increased friction at the boundary. This is effectively what Nikuradse (1933) did when he derived the roughness length scale k_s based upon sand grain diameters, and what Schlichting (1936) did when he applied the same parameter to regularly distributed

roughness patterns on a flat plate. Moore (1951) effectively proposed that the position of the plane wall should be located at the hypothetical velocity origin, discussed in Section 2.1.7.

The position of the equivalent plane wall becomes increasingly significant as the relative roughness becomes larger. The dimension y_0 defining the datum level becomes a larger proportion of the flow depth thereby affecting not only the logarithmic velocity profile, but also the hydraulic radius of the equivalent plane duct itself. For this reason the present study gave attention to obtaining a y_0 which was dynamically significant, based on determining the line of action of the resulting streamwise force on the rough wall by using moments.

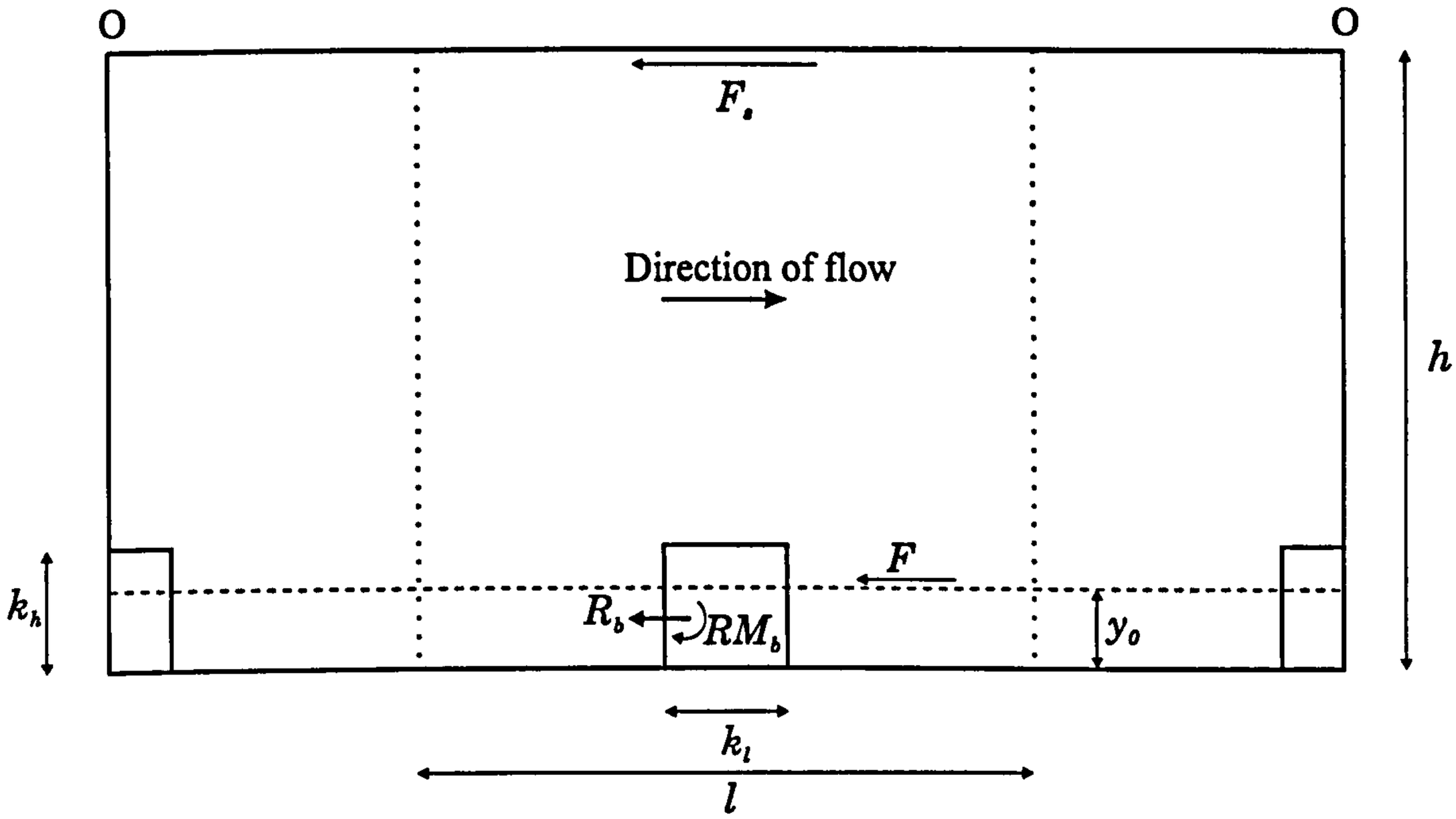


Figure 3-6: Moments of streamwise forces in a rough duct

Consider the duct shown in Figure 3-6 with one smooth wall and a bottom wall containing periodically spaced roughness elements. If the pressure drop is modelled by a source of linear momentum flux, then the driving force per unit width for one period is

$$\left(\frac{\mathrm{d}P}{\mathrm{d}x} lh\right) - \left(\frac{\mathrm{d}P}{\mathrm{d}x} k_l k_h\right) \qquad \textit{Equation 3-52}$$

The resistive force is equivalent to a sink of linear momentum flux. For the period length l the total resistive force (R_b) acting on the rough wall will be the sum of the bed shear forces on the upstream and downstream part of the period, the shear force acting on the top surface of the roughness element and the bluff body pressure force on the block. This total resistive force will have a moment about O-O denoted by RM_b.

Equilibrium of forces for this period gives

$$\left(\frac{\mathrm{d}P}{\mathrm{d}x}lh\right)-\left(\frac{\mathrm{d}P}{\mathrm{d}x}k_lk_h\right)=R_b+F_s \qquad \textit{Equation 3-53}$$

where F_s is the shear force on the smooth wall. Taking moments about O-O on the smooth wall gives the net moment of the system:

$$T_{O\text{-}O}=\frac{\mathrm{d}P}{\mathrm{d}x}\left[l\,h^2/2-k_lk_h\left(h-k_h/2\right)\right]-RM_b \qquad \textit{Equation 3-54}$$

If the rough bed is replaced by a smooth bed with displacement y_0 and a shear force F, the net moment of the new system is

$$T'_{O\text{-}O}=\frac{\mathrm{d}P}{\mathrm{d}x}\left[l\,(h-y_0)^{(h-y_0)}/2\right]-F(h-y_0) \qquad \textit{Equation 3-55}$$

Equilibrium of forces for the new system gives

$$F=\frac{\mathrm{d}P}{\mathrm{d}x}l(h-y_0)-F_s \qquad \textit{Equation 3-56}$$

Equating $T_{O\text{-}O}$ and $T'_{O\text{-}O}$ to conserve angular momentum flux, and substituting for F from Equation 3-56 gives;

$$\frac{\mathrm{d}P}{\mathrm{d}x}\left(l\,h^2/2-k_lk_h\left(h-k_h/2\right)\right)-RM_b=\frac{\mathrm{d}P}{\mathrm{d}x}\left(l.(h-y_0)^2/2\right)-(h-y_0)\left(\frac{\mathrm{d}P}{\mathrm{d}x}l(h-y_0)-F_s\right) \qquad \textit{Equation 3-57}$$

This quadratic in y_0, expressed as $a\,y_0^{\,2}+b\,y_0+c$, has the following coefficients:

$$a=-1/2\,l\frac{\mathrm{d}P}{\mathrm{d}x}$$

$$b=\frac{\mathrm{d}P}{\mathrm{d}x}hl-F_s$$

$$c=\frac{\mathrm{d}P}{\mathrm{d}x}\left[k_lk_h\left(h-1/2\,k_h\right)-lh^2\right]+RM_b+F_sh$$

and e may be found from

$$y_0 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \qquad \textit{Equation 3-58}$$

The positive root gives a value for y_0 between zero and the height of the roughness element. The negative root gives a mirror image of this position centred around the position of zero shear stress.

Substitution of y_0 into Equation 3-56 gives the equivalent plane wall shear force F acting on the plane given by the datum position. For the remainder of this thesis, the datum level obtained by equating streamwise force moments will be denoted by $y_{0\,e.s.m.}$.

Discussion

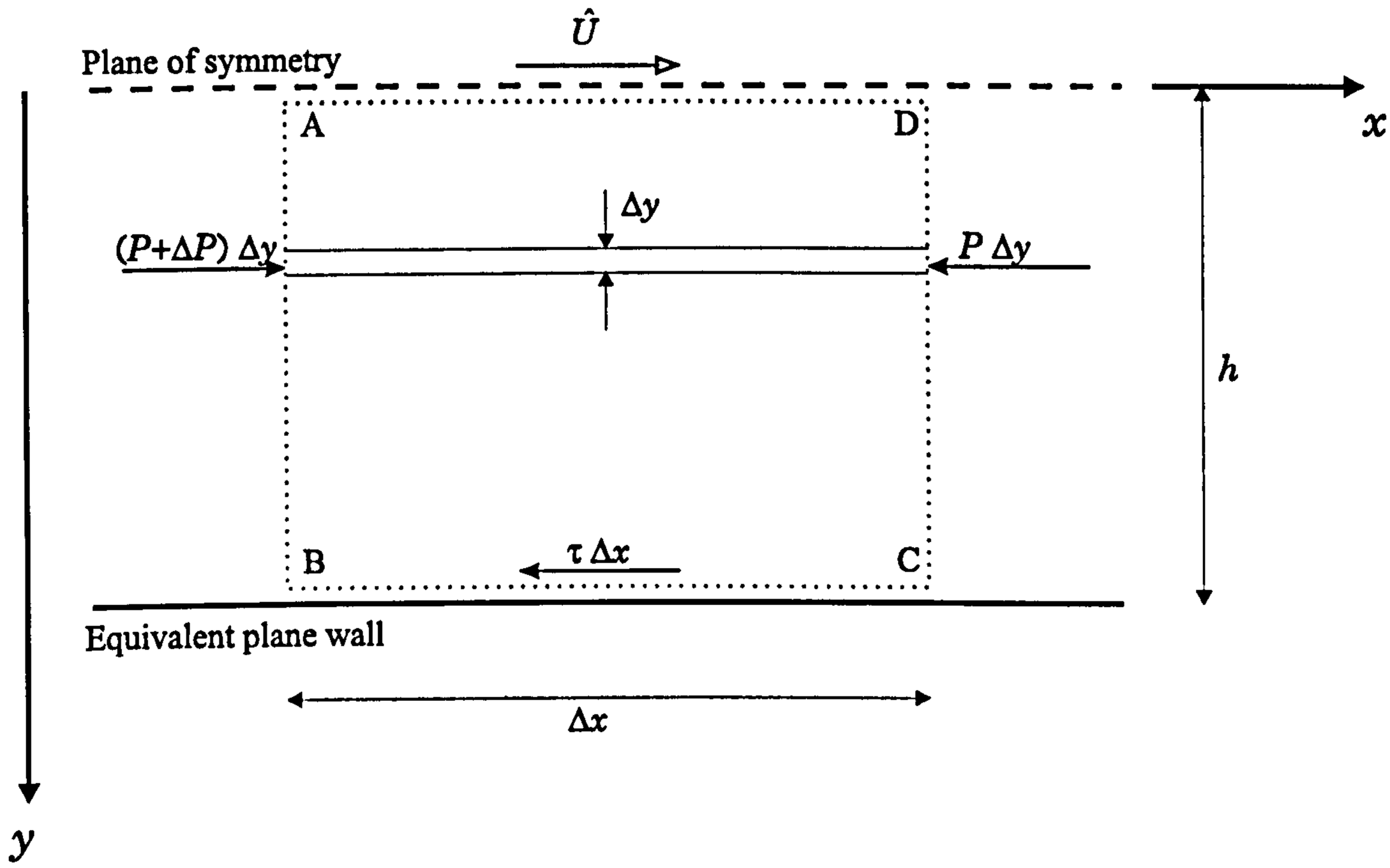


Figure 3-7: Force moment on an equivalent plane bed

Equation 3-57 ensures that the new, plane walled duct has the same net streamwise force moments as the original rough duct. Further insight may be gained by considering the force moments about the plane of symmetry on the control volume ABCD of the simplified duct shown in Figure 3-7.

The shear stress in the fluid will vary from zero at the plane of symmetry to a value of $\tau = h\,\mathrm{d}P/\mathrm{d}x$ at the position of the equivalent plane wall. Taking moments about the plane of symmetry gives

$$T = \int_{y=0}^{d} (\Delta P\, y\, \mathrm{d}y) - \tau\, \Delta x \times h$$
$$= \left[\tfrac{1}{2}\Delta P\, y^2\right]_0^h - \tau\, \Delta x \times h$$
$$= \tfrac{1}{2}\Delta P\, h^2 - \tau\, \Delta x \times h$$

where T is the moment (torque). Using $\tau = h\,\mathrm{d}P/\mathrm{d}x$ gives

$$T = \tfrac{1}{2}\Delta P h^2 - h\,\mathrm{d}P/\mathrm{d}x\, \Delta x\, h$$
$$= -\tfrac{1}{2} h^2 \Delta P$$

Equation 3-59

This approximate analysis shows that the moment due to the pressure gradient is half of that due to the bed shear force. The net effect of this shear force and the pressure gradient is to produce a circulation in the –ve (clockwise) direction, as indicated by $\oint U_s \mathrm{d}_s$ where s is measured along the line ABCD in Figure 3-7 and U_s is the velocity along the line. Starting from A the circulation is

$$\Gamma = 0.\Delta y + 0.\Delta x + 0.\Delta y - \hat{U}.\Delta x$$
$$= -\hat{U}.\Delta x$$

Equation 3-60

Figures relating to user controlled periodic boundary condition tests

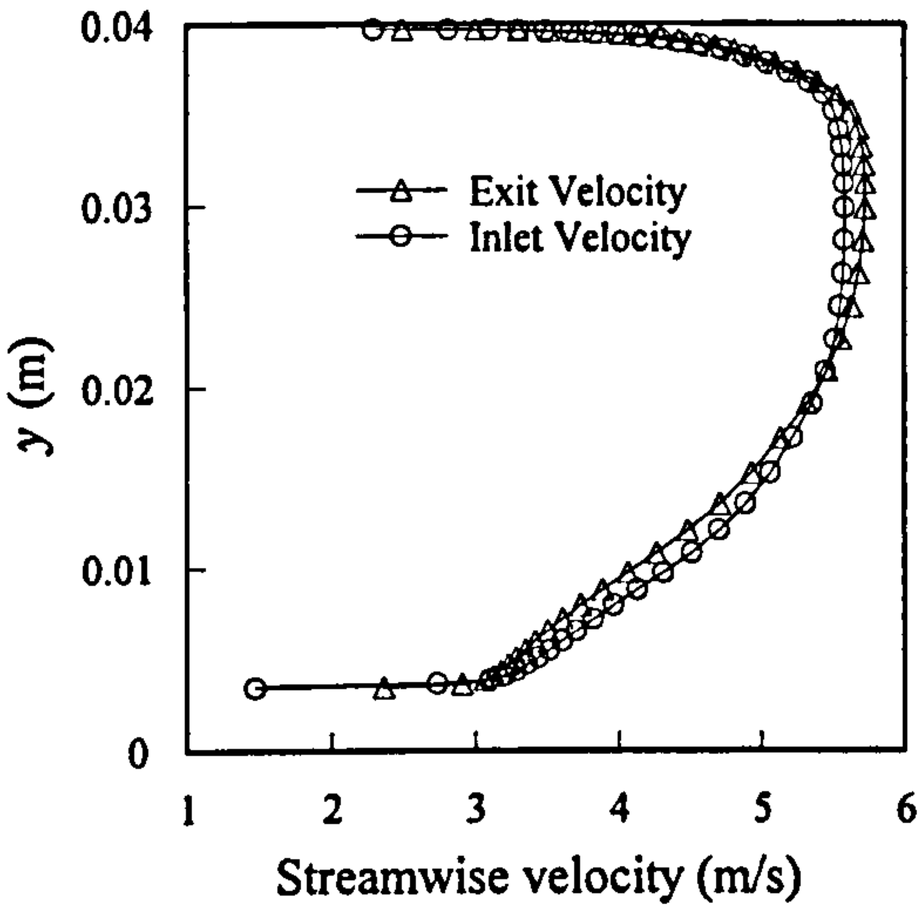


Figure 3-8: Streamwise velocity profile. Low exit coefficient.

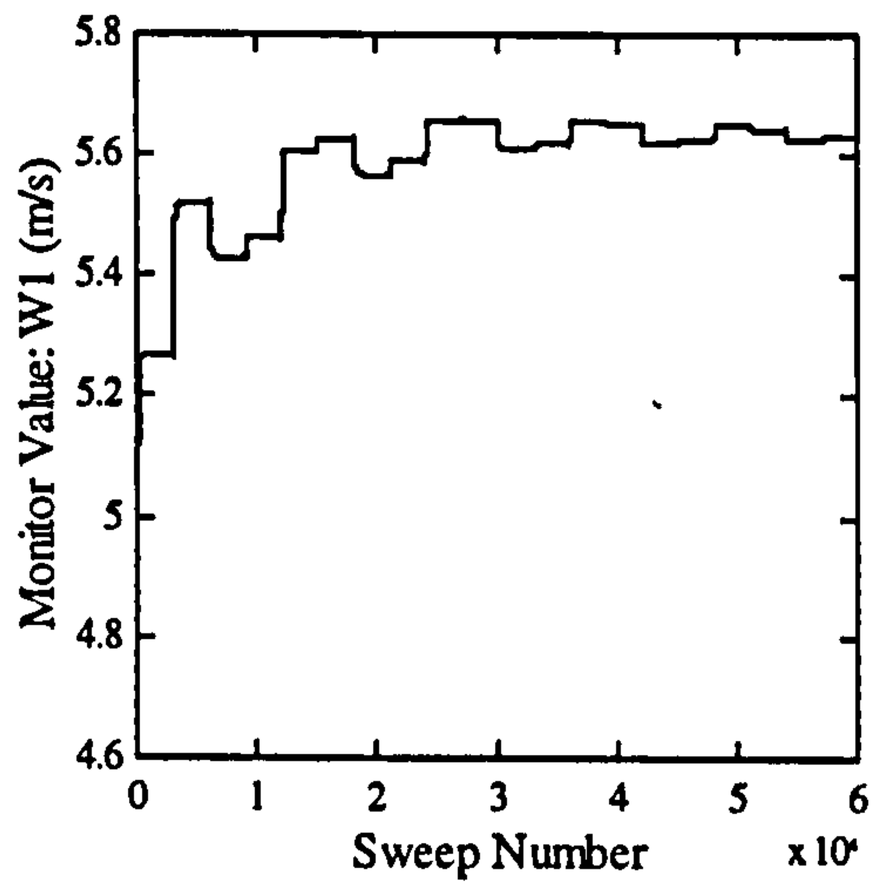


Figure 3-9: Convergence monitor of streamwise velocity. Low exit coefficient.

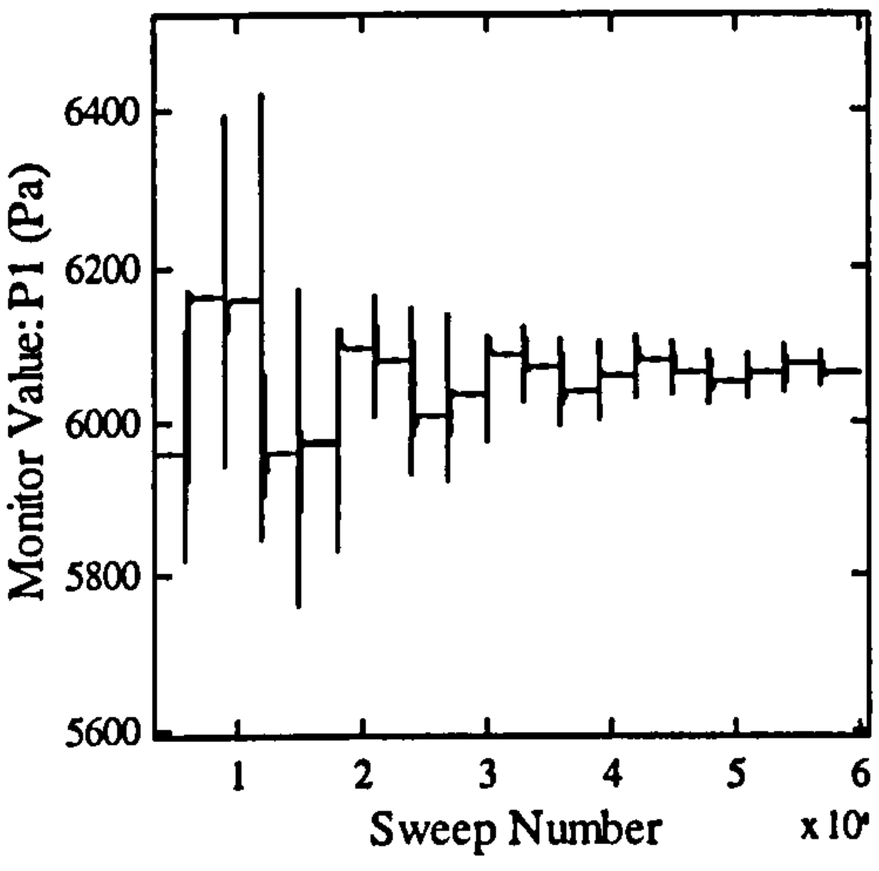


Figure 3-10: Convergence monitor of pressure. Low exit coefficient.

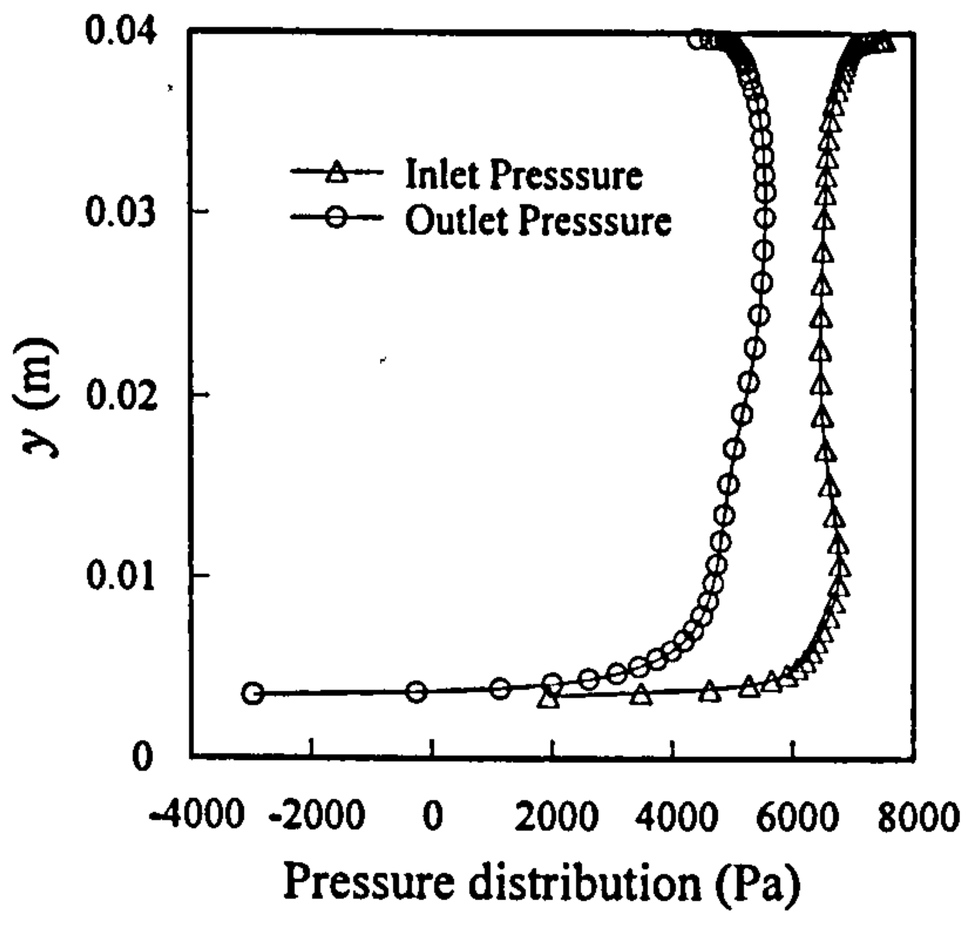


Figure 3-11: Transverse pressure distribution. Low exit coefficient.

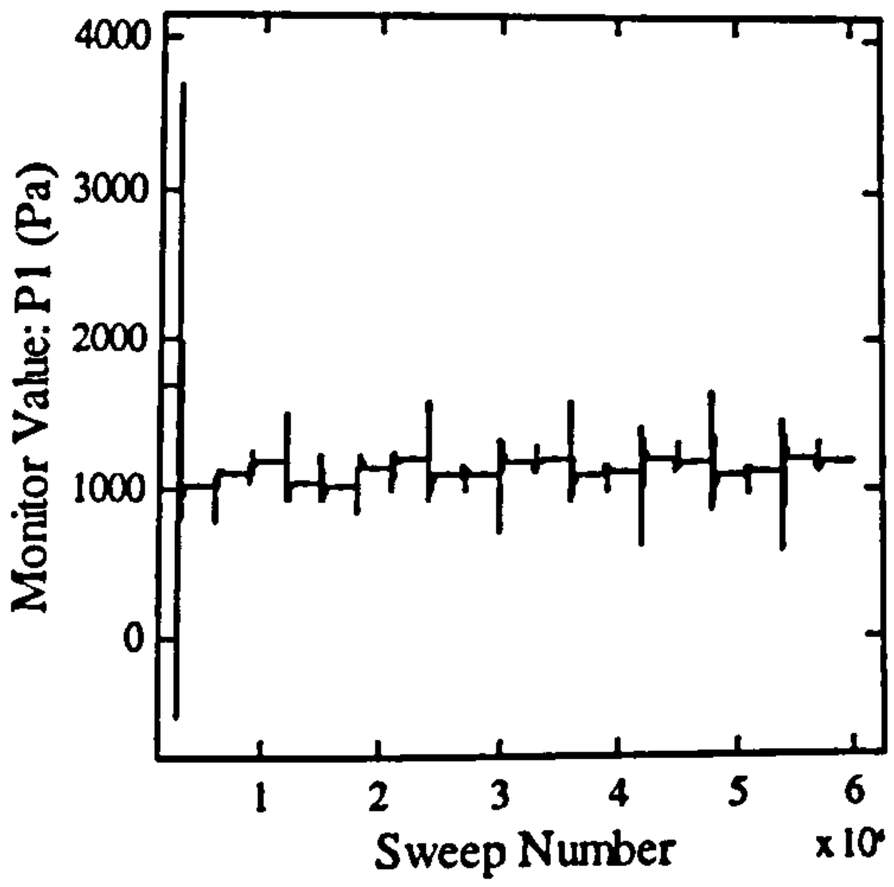


Figure 3-12: Convergence monitor of pressure. High exit coefficient.

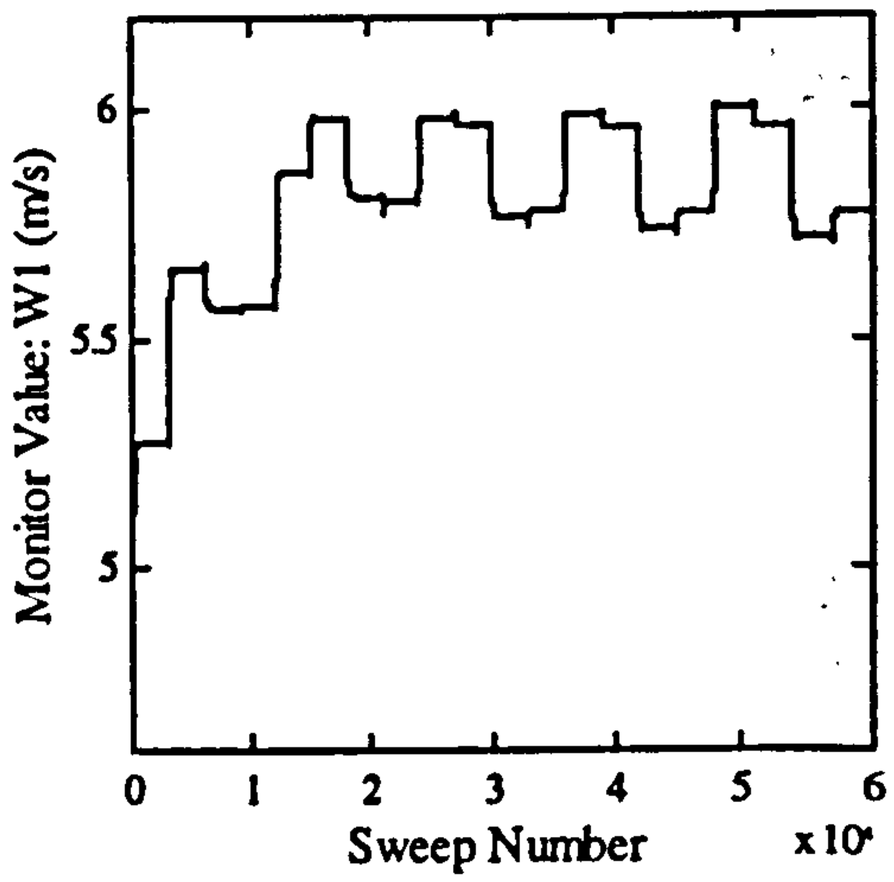


Figure 3-13: Convergence monitor of streamwise velocity. High exit coefficient.

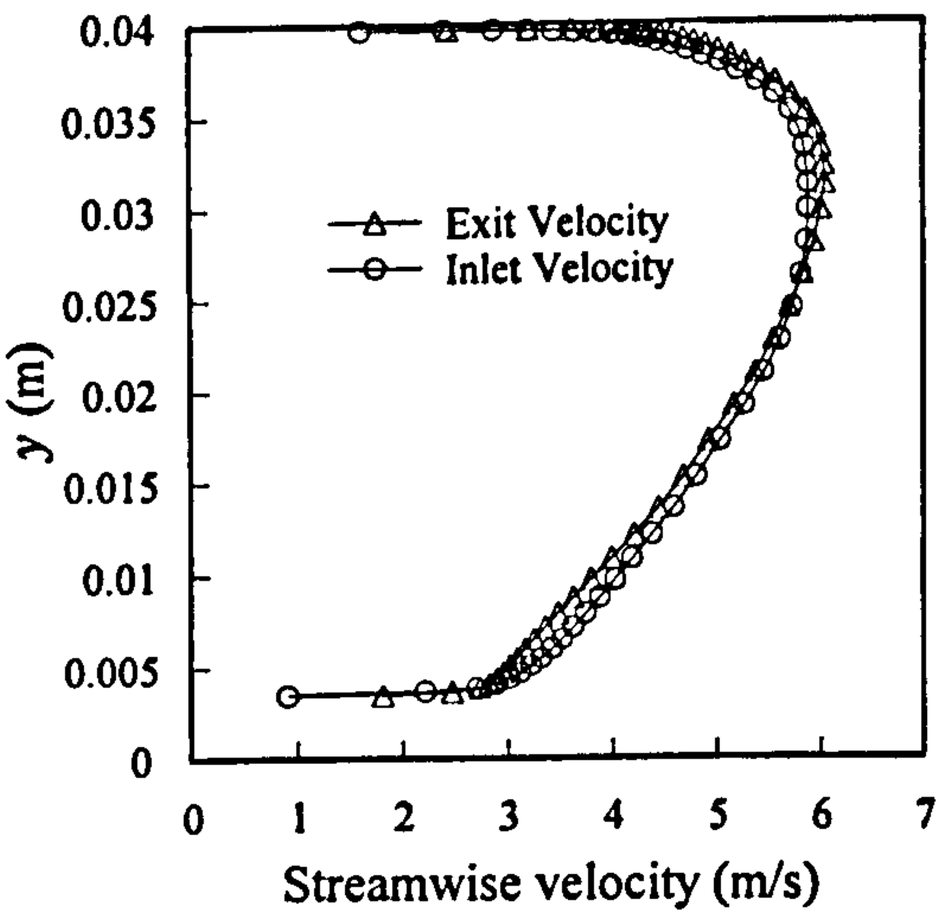


Figure 3-14: Streamwise velocity profile. High exit coefficient.

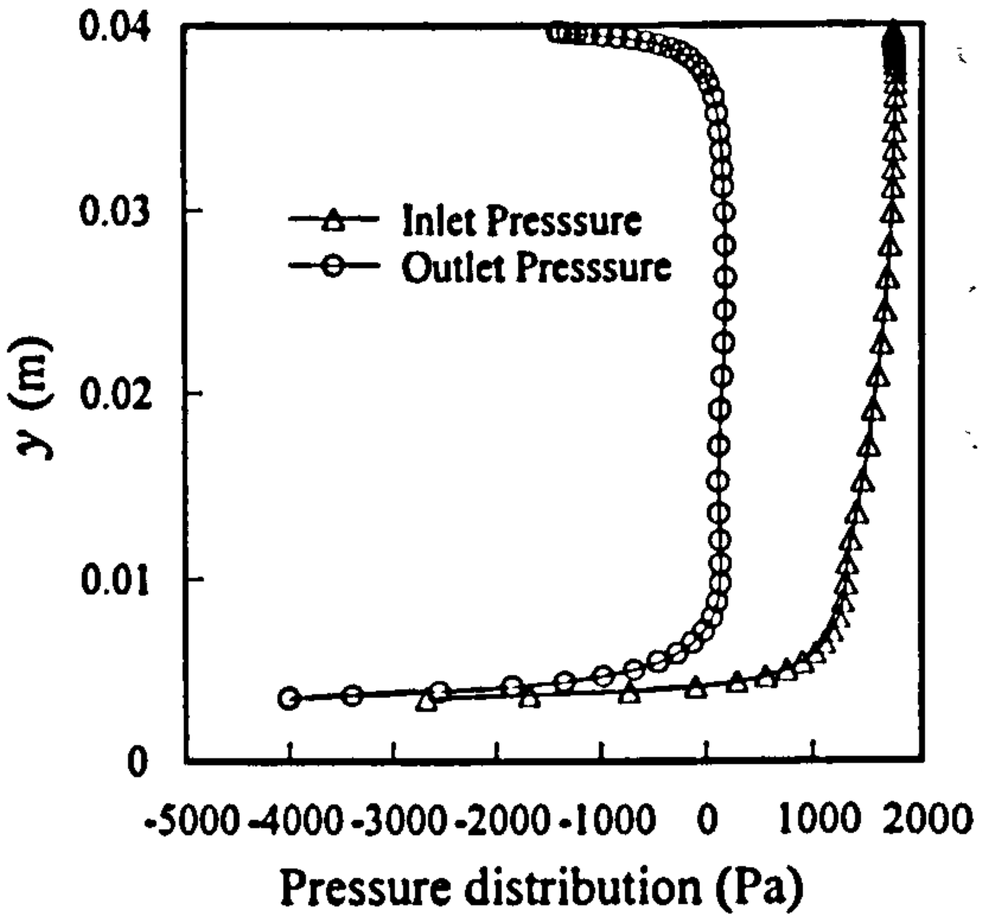


Figure 3-15: Transverse pressure distribution. High exit coefficient.

Figures relating to period averaged shear stress

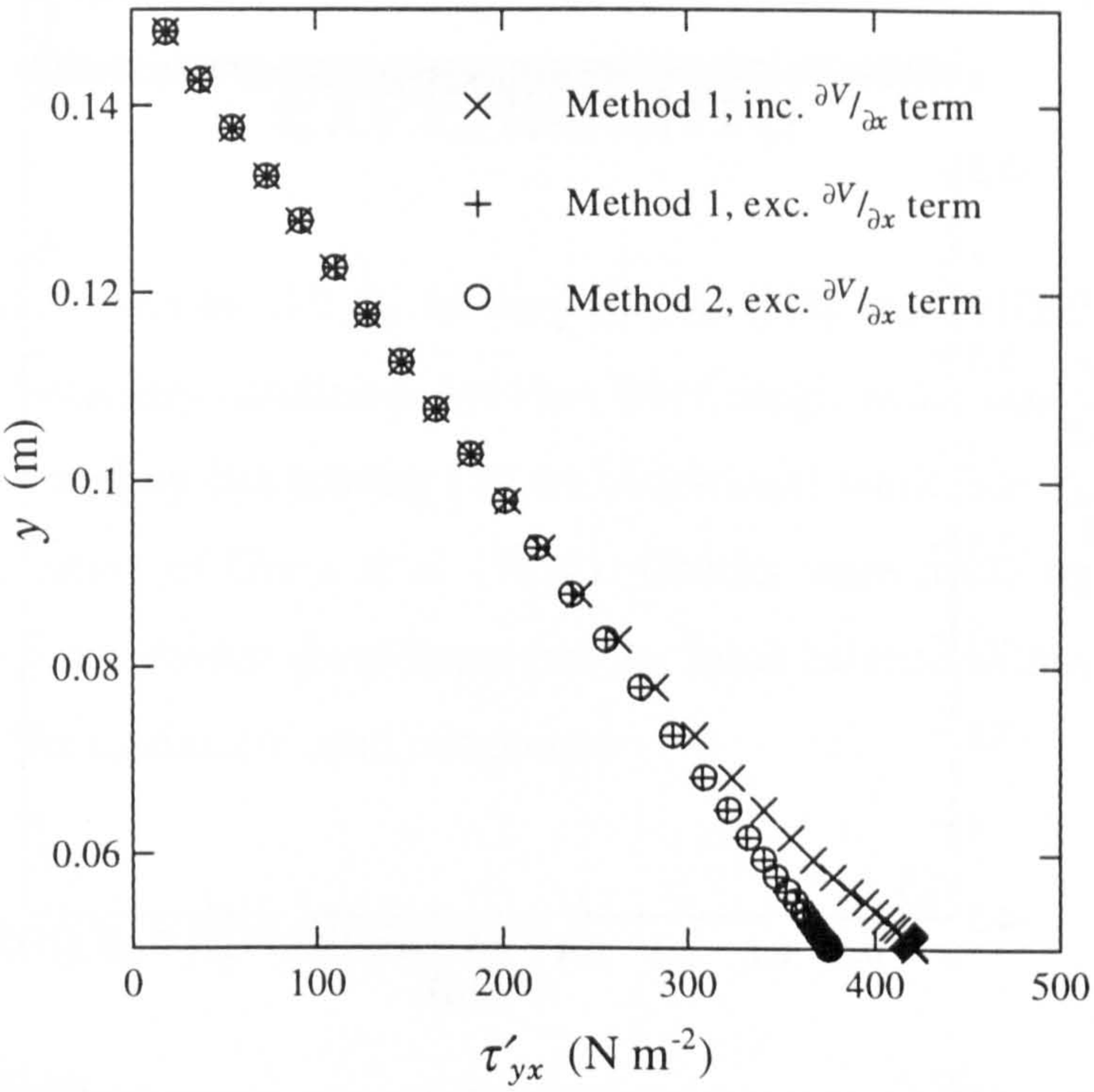


Figure 3-16: Plot of reduced shear stress by different methods. Method 1 calculates the shear stress using the values of field variables, both including and excluding the term $\Gamma \partial N/\partial x$. Method 2 utilises the convective and diffusive fluxes within the finite volume equations.

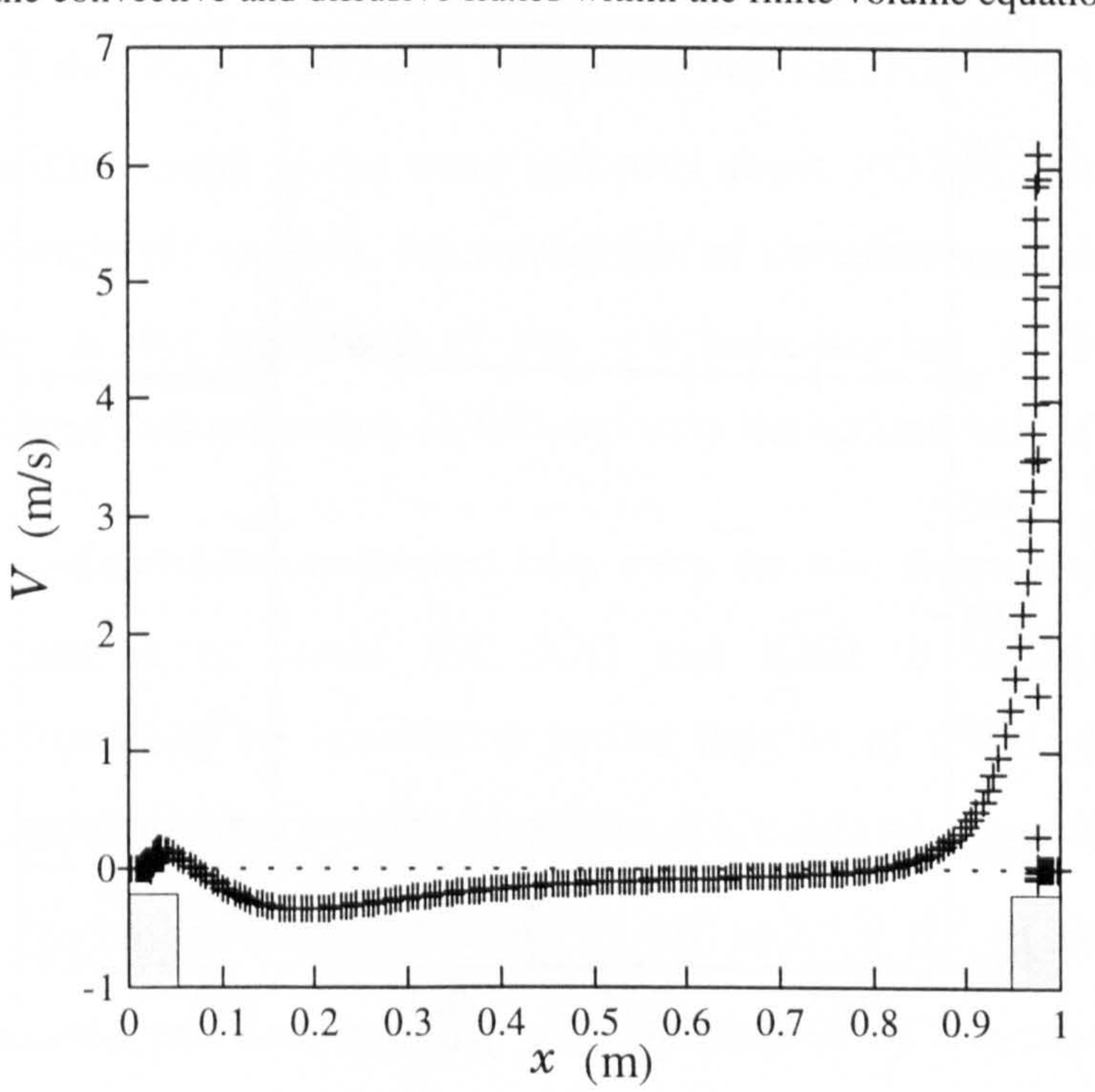


Figure 3-17: Plot of transverse (V) velocity over one period, located at a y plane immediately above the roughness elements.

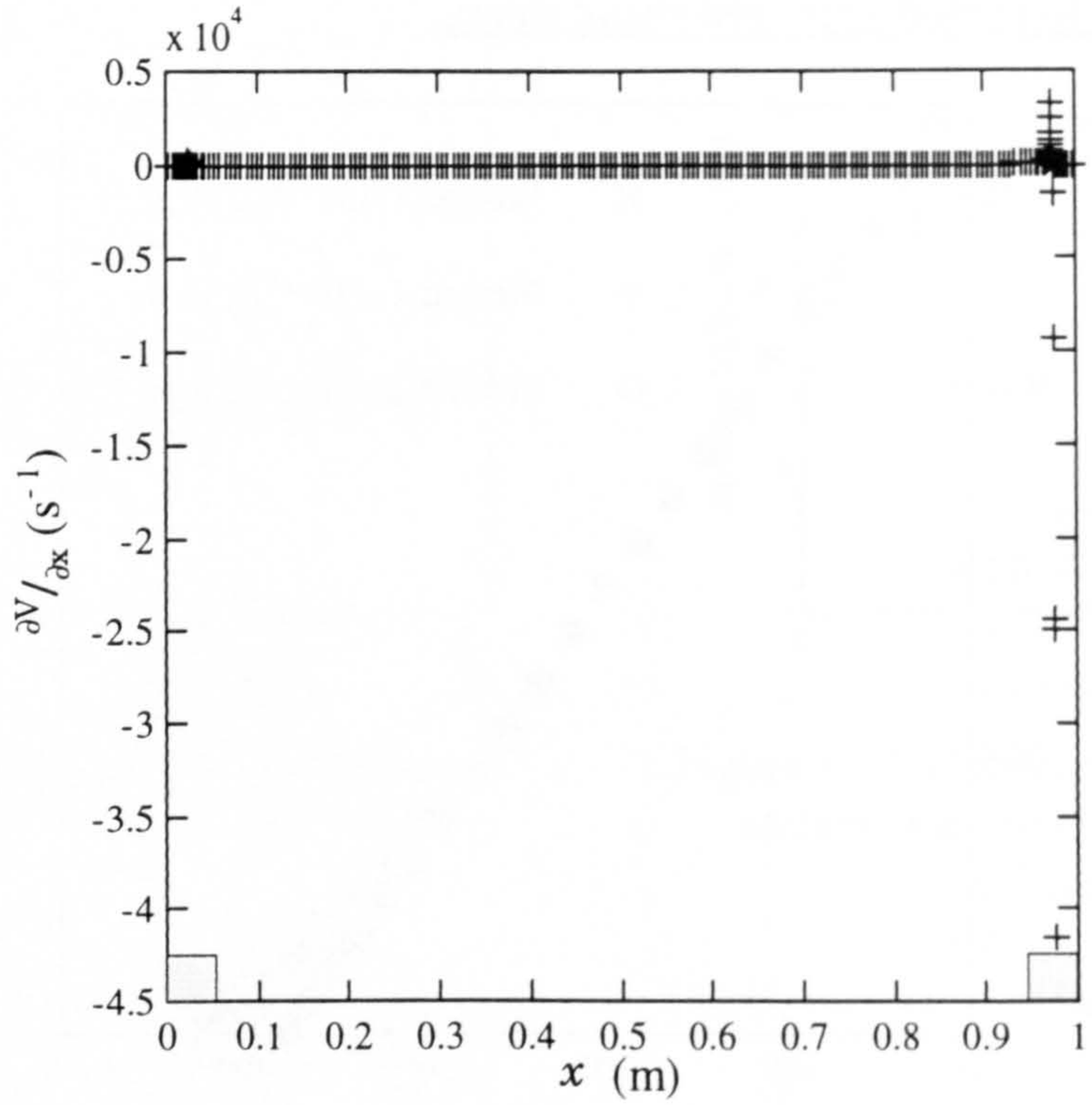


Figure 3-18: Plot of $\partial N / \partial x$ over one period, located at a y plane immediately above the roughness elements.

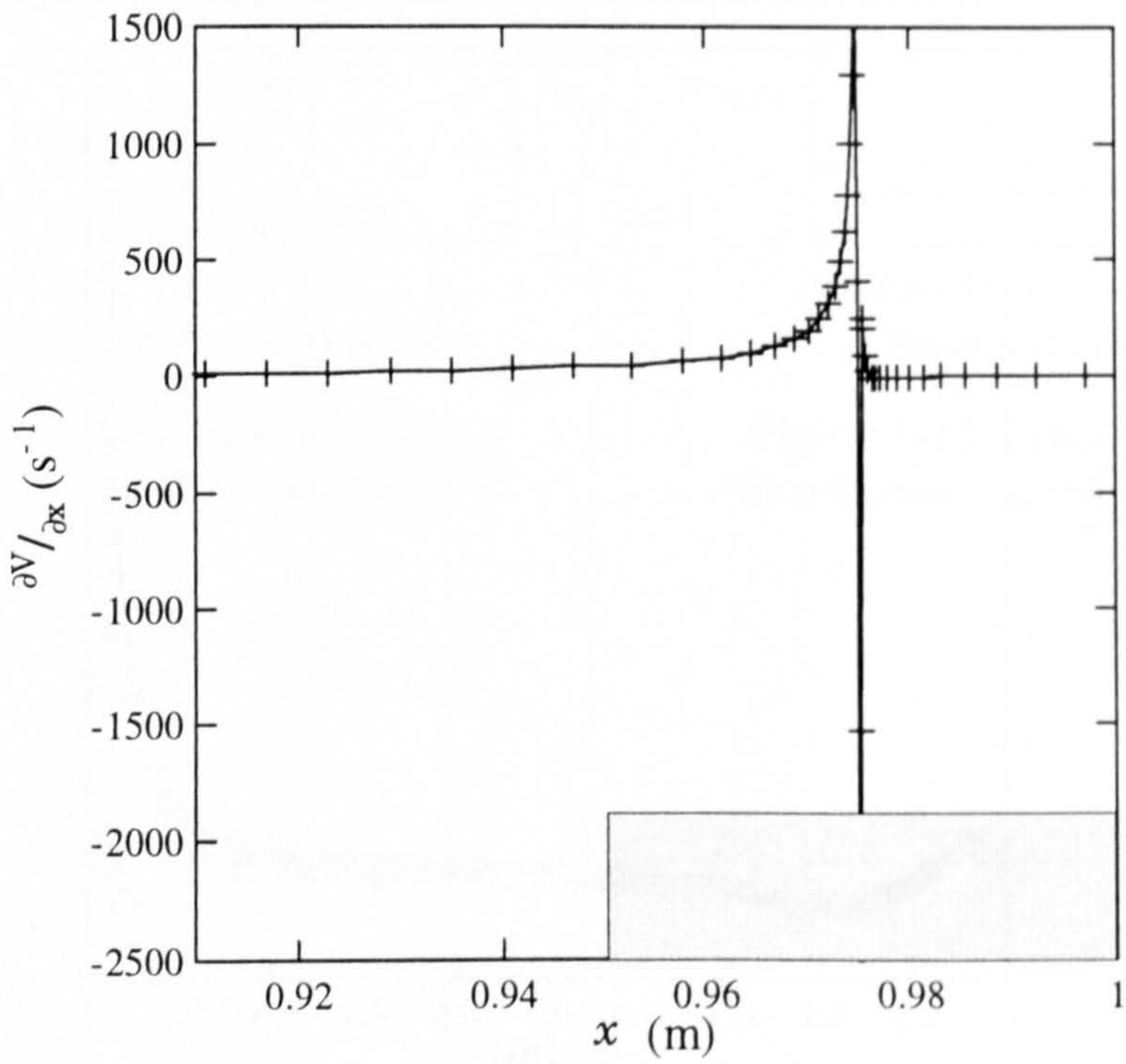


Figure 3-19: Expanded view of $\partial N / \partial x$ distribution.

4. COMPARISON WITH ROUGH DUCT PHYSICAL EXPERIMENTS

A two-layer model (Section 2.2.2), in conjunction with the PHOENICS controlled method of cyclic boundary conditions for flow over rough walls was validated against experiments performed by Schlichting (1936). Additional work was carried out on the roughness configuration of Grass *et al* (1991). Checks were made against theoretical predictions of the cross-stream shear stress profile, force balance within the domain and the calculation of the equivalent sand roughness.

4.1 Schlichting's Experiments

4.1.1 Introduction

Schlichting performed experiments in a rectangular duct of aspect ratio (width to height) $b/h = 4.25$. One of the longer walls was roughened and the remaining three boundaries were smooth. The total length of the water duct was about 100 hydraulic diameters, the first half being completely smooth. Measurements of streamwise velocity were taken using a pitot tube at the centreline of the exit cross-section. Static pressure was measured at eight locations uniformly distributed over the second half of the duct.

The roughnesses configurations examined here were the two-dimensional "long angle" strip roughness denoted as plates XX, XXI and XXII in Schlichting's original experiments, corresponding to streamwise period lengths of 6 cm, 4 cm and 2 cm respectively. For each roughened plate, 5 different Reynolds numbers (6 for case XXII) were simulated in the range of approximately 1×10^5 to 2.7×10^5. In each computation, Schlichting's measured pressure gradient was represented by a uniformly distributed source of momentum flux per unit volume. A complete listing of all the input parameters is given in Table 4-1a. Figure 4-1 gives a comparison of the geometry employed by Schlichting and that used in the computational model.

Schlichting assumed that the shear stresses on the side walls had negligible effect upon the centreline bed shear stress and the flow could therefore be treated as two-dimensional in this region. Coleman *et al* (1984) expressed a contrary opinion but based on the following argument, it is believed that Schlichting was correct in his assumption and consequently a two-dimensional solution was employed.

Side wall effect

It was assumed that the position of zero shear stress ($\tau_{yx} = 0$) on the $z = 0$ plane occurs at the location of the maximum primary velocity, i.e. at $y = h_2$ (Figure 4-1) and that the locus of $\tau_{yx} = 0$ across the duct may be approximated by the $y = h_2$ plane. The effect of the side wall on the centreline smooth bed shear stress is then approximately equal to the side wall effect in a smooth rectangular duct of aspect ratio b/h_1 where b = 170 mm. For all the long angle experiments $b/h_1 \approx 12$ and the smooth rectangular duct results presented in Figure 6-4 for $100/(1+b/h) \approx 8$ give a τ_t/τ_{2d} very nearly equal to one, i.e. the flow may be treated as two-dimensional on the centreline of the smooth bed. Analysis presented in Appendix 9 shows that the same conclusion can be drawn for the centreline rough bed shear stress.

Computational grid and convergence

The computational domain, together with the grid construction is shown in Figures 4-2 and 4-3. For the first tests undertaken (plate XX) a domain consisting of two periods was used. After fully developed periodic flow had been confirmed, the remainder of the tests used a single period domain to reduce the computational resources required.

A grid independence test (Grid 2) was performed for each roughness configuration at the highest Reynolds number. The construction of these grids is shown in Figure 4-4.

An example of a typical Q1 input file is given in Appendix 10. The ground coding used for these simulations was very similar to that used in the depth scale roughness tests, presented in Appendix 11.

4.1.2 Results

The results of the computational experiments are compared with Schlichting's original results in Table 4-1.

The grid independence tests show that the results varied only slightly with an increased cell density. The greatest change in the prediction of maximum velocity and the rough wall shear stress was 1%. The smooth wall shear stress however changed by up to 2.9%.

The numerical solution is shown to be periodically fully developed by examination of the cross-stream profiles of pressure and streamwise velocity (Figures 4-5 and 4-6). The figures show profiles taken at the start and end of the middle period for the plate XX case 5 (Grid 1). The streamwise pressure profile (Figure 4-7) taken over two periods at a fixed y co-ordinate also demonstrates periodic behaviour.

The cross-stream distribution of period averaged shear stress (Section 3.3.1) for plate XX case 5 is shown in Figure 4-8 and is seen to be perfectly linear. The slope, $\mathrm{d}\tau_{yx}/\mathrm{d}y = 1411.8$, differs from the pressure gradient by only 0.4%. Also shown is the resultant period averaged rough wall shear stress, calculated by streamwise force moments (Section 3.5) and the plane on which this acts. A linear period averaged shear stress distribution was confirmed for each test.

Schlichting measured the velocity profile mid-way between two roughness elements and immediately behind the trailing edge of a roughness element, but found little difference between the two profiles. Again using plate XX case 5 as an example, Figure 4-9 shows the velocity profiles obtained at these two locations and also the period averaged velocity profile. Schlichting's measured maximum velocity is also shown. In general the position and value of maximum velocity predicted by the computational model was in good agreement with Schlichting's measured values; the discrepancy in $\hat{U}$ was typically of the order of 1-2%, with case XXI profile 1 giving the highest error of 11.3% and the maximum error in the position of $\hat{U}$ was 2.5% of the total duct height.

Comparison of shear stresses

In the numerical model, the smooth wall shear stress was calculated from the velocity gradient within the laminar sublayer

$$\tau_s = \mu \frac{dU}{dy} \qquad \textit{Equation 4-1}$$

The shear stress on the rough wall was found by extrapolating the linear reduced shear stress to the rough wall. For comparison with Schlichting's results the plane on which the rough wall shear stress acted was determined from the geometric mean level of the roughness elements ($\tau_{g.m.l.}$).

Schlichting obtained the smooth wall shear stress from the Prandtl universal velocity distribution for a smooth wall, assuming that the roughness elements did not affect the smooth wall boundary layer. Writing the Prandtl universal velocity law (Equation 2.21) as

$$U = 5.75U_{\tau,s}\log y + 5.5U_{\tau,s} + 5.75U_{\tau,s}\log {U_{\tau,s}}/{\nu} \qquad \textit{Equation 4-2}$$

then the value of $U_{\tau,s}$ was obtained from the gradient of the graph of U against $\log y$ (where in this instance y was measured from the smooth wall).

Schlichting calculated the rough wall shear stress by two methods and took the average as his best estimate. In the first method the rough wall shear stress ($\tau_{r,1}$) was obtained by balancing the forces due to the smooth and rough wall boundary shear stresses and the force due to the pressure gradient. For two-dimensional flow

$$\tau_{r,1} + \tau_s = h\left(\frac{dP}{dx}\right) \qquad \textit{Equation 4-3}$$

where h is the height of the duct. Knowledge of the pressure gradient and of τ_s from $U_{\tau,s}$ led to a value for $\tau_{r,1}$. For the second method, Schlichting used the law of the wall in the relative roughness form (Equation 2.30);

$$\frac{U}{U_{\tau,r2}} = A + 5.75\log\left({(y-y_0)}/{k_h}\right) \qquad \textit{Equation 4-4}$$

By plotting U against log (y-y_0) and comparing the gradient (n_r) with that of Equation 4-4 , then

$$U_{\tau,r2} = {}^{n_r}/_{5.75} \qquad \textit{Equation 4-5}$$

This method gave the value of $\tau_{r,2}$ at the plane defined by y_0, whereas the first method gave $\tau_{r,1}$ at the plane through the base of the roughness elements. For the roughness configurations considered here, y_0 (calculated from the geometric mean level) is negligible and therefore the two planes were virtually coincident. Due to the sensitivity of the logarithmic velocity profile equation to y_0, Coleman *et al* (1984) recommended the first method of calculating τ_r.

When compared with the numerical predictions, the average values of rough wall shear stress quoted by Schlichting agreed to within 1.4% for plate XX (Table 4-1b). For plates XXI and XXII the maximum error increased to 16.6% and 14.4% respectively. The variation with Reynolds number of the rough wall friction coefficient defined by

$$f = \frac{8\tau_r}{\rho \hat{U}^2} \qquad \textit{Equation 4-6}$$

is shown in Figures 4-10 to 4-12 for both methods of calculating the shear stress used by Schlichting and for the value obtained numerically. For plate XX, the numerical prediction was mid-way between the two values calculated by Schlichting. For plates XXI and XXII, the two values calculated by Schlichting did not give constant values of f and followed opposite trends. The numerical predictions fell below the experimental values for both plates XXI and XXII, but gave constant values of f for all values of Reynolds number, indicating that the flow was in the fully rough regime. It should be noted however that the range of Reynolds numbers (1×10^5 to 2.7×10^5) was too narrow to be confident that f was truly Reynolds number independent.

The relative errors between experiment and numerical prediction were significantly higher for the shear stresses on the smooth walls, the average error for all tests being

approximately 130%. The variation of f with Reynolds number for the smooth walls is also shown on Figures 4-10 to 4-12. For all plates the numerical model showed that the value of f was approximately constant with Reynolds number, more characteristic of rough wall flows, though again the Reynolds number range was too narrow to draw any firm conclusions.

The solid line in Figure 4-13 shows the velocity profile on the smooth wall which would result from the Prandtl velocity distribution law (Equation 4-2) using the shear velocity found by Schlichting for plate XX case 5. It is evident that the profile given by Equation 4-2 under-predicted Schlichting's own value of maximum velocity by approximately 24%. This would imply that one of the following was in error in Schlichting's original experiments:

1. Value and position of maximum velocity.

2. Measurement of velocity profile used to obtain U_{τ_s}.

3. Use of the Prandtl velocity distribution law.

Value and position of maximum velocity

The value and position of maximum velocity could readily have been determined with reasonable accuracy using a pitot tube and therefore it was unlikely to have been the cause of the error.

Measurement of velocity profile used to obtain U_{τ_s}

It is more likely that experimental error was present in the measurement of the velocity distribution within the logarithmic region of the smooth wall boundary layer. This is a reasonable conclusion given that the profile was at maximum only 8 mm from the wall to the point of maximum velocity. In order that a sufficient number of points could be measured to an acceptable degree of accuracy, the traversing gear and pitot tube assembly would have needed special care. In flow regions of high shear it is now known that pitot tube measurements require an adjustment to account for the *effective centre* of the pitot tube. This procedure was demonstrated by Young and Maas in 1936, though it

was not mentioned by Schlichting in his experimental procedure, also performed in 1936.

Use of the Prandtl velocity distribution law

The last source of error concerns Schlichting's assumption that the presence of the rough wall did not influence the smooth wall boundary layer. In order to test this assumption additional computations were carried out in which the smooth wall was isolated from the influence of any roughness elements (Section 4.1.3). By extensive analysis it was concluded that the roughness elements did affect the distribution of flow variables in the smooth wall boundary layer, altering the value of smooth wall shear stress by 15.5%. This however does not account for the 130% discrepancy between the computed shear stress and that obtained by Schlichting, which is more readily explained in terms of inaccurate measurements of the smooth wall velocity distributions as previously discussed.

4.1.3 Simulations of Schlichting's Smooth Wall Region

A domain was chosen to replicate the smooth wall region of the computational tests of plate XX, case 5 in which a plane of symmetry, located at the position of maximum velocity computed from the rough walled duct, was used as the boundary condition. Three "smooth wall only" tests were performed, corresponding to the original grid in the model of the complete duct (Grid 1), the grid independence test for the complete domain (Grid 2) and a further grid independence test (Grid 3). The input conditions are summarised in Table 4-2. The pressure gradient was adjusted until the maximum velocity reached the value found in the computational test of the complete rough walled duct. Convergence was measured by the error between τ_w/ρ calculated from the laminar-sub layer velocity distribution and that calculated from

$$\tau_w/\rho = \frac{dP/dx \times h}{\rho} \qquad \textit{Equation 4-7}$$

An error of less than 0.1% indicated sufficient convergence.

Figure 4-14 shows the velocity distribution obtained from the smooth wall test (Grid 2) and the period averaged smooth wall velocity profile predicted by the complete computational model. The maximum velocities can be seen to be identical as specified. However the velocity distribution for the complete Schlichting duct shows a fuller profile, resulting in a higher depth averaged velocity and steeper velocity gradients near the wall with consequently higher shear stresses (by ≈ 15.5%). The difference between the two velocity profiles must have been due to the roughness elements present in the complete model affecting the smooth wall boundary layer. This was most likely to be caused by the roughness elements generating higher levels of turbulent mixing and also cross-stream mean flow convective fluxes. At $y = 0.033$, the location of maximum velocity, Figure 4-15 shows that the turbulent kinetic energy in the complete duct was about three times higher than for the smooth wall only. The position of minimum kinetic energy should also lie on this line of maximum velocity if Schlichting's assumption was correct. However the profile indicates that the minimum had been shifted towards the smooth wall, due to the influence of the rough wall. Figure 4-16 shows that the distribution of the rate of dissipation of kinetic energy was not significantly affected. The distribution of turbulent viscosity is shown in Figure 4-17. On the line of maximum velocity, the turbulent viscosity was nearly five times higher for the complete model. Also note that whereas the gradient was constrained to be zero on this line due to the plane of symmetry condition for the smooth wall only case, the complete model had a non-zero gradient. These results show that the roughness elements did affect the boundary layer on the smooth wall. It is also possible that the effect of the roughness elements was the cause of the constant value of f on the smooth wall for all Reynolds numbers (Figures 4-10 to 4-12).

As well as the effect of rough wall turbulence upon the smooth wall shear stress, in Schlichting's duct there must have been an influence from the periodic acceleration and deceleration of the fluid caused by the periodic variation in cross-sectional geometry. Figure 4-18 illustrates the distribution of shear stress on the smooth wall of the numerical model of Schlichting's duct, plate XX case 5. Over the two periods modelled, the distribution of shear stress was periodic, corresponding to the spacing of the

roughness elements. The variation of shear stress over one period was approximately 3.5%. This was due to the streamwise velocity increasing and decreasing as a consequence of the variation in flow area bounded by the continuous streamline over the roughness elements.

In the method by which Schlichting obtained the value of smooth wall shear stress, this variation of shear stress might have had two consequences:

1. The point values of shear stress obtained could have been in error by ± 1.75% from the period averaged value.

2. Diffusion of momentum deficit from the wall might not have occurred with sufficient speed to extend far enough into the boundary layer to achieve the universal law of the wall profile corresponding to the local value of wall shear. In this sense the velocity profile would always have been out of step with the wall shear stress.

From the latter it follows that in applying the law of the wall to the calculation of wall shear stress there would inevitably have been an error. In order to test this hypothesis a simple analysis of the flow was carried out using Bernoulli's equation, the conclusion of which was that the local wall shear stress and velocity profile mutually adjusted at a fast enough rate to preserve the logarithmic law. The analysis is as follows.

The variation of streamwise velocity over two periods is shown in Figure 4-19. The distribution illustrated was taken at the y location where the period averaged streamwise velocity was a maximum $\left(y_{\hat{U}}\right)$. The streamwise (x) location where U was a maximum is labelled 1 and location 2 refers to the position of minimum streamwise velocity. As the flow passes from location 1 to 2, the velocity profile on the smooth wall was slightly retarded (Figure 4-20). At the position $y = y_{\hat{U}}$, the vorticity was zero and therefore the change in velocity could be predicted by the Bernoulli equation rearranged to give

$$U_1 - U_2 = \frac{2}{\rho}\frac{(P_2 - P_1)}{(U_1 + U_2)} \qquad \textit{Equation 4-8}$$

where the pressure distributions are as shown in Figure 4-21. There is very good agreement between the velocity difference obtained by the numerical model and that obtained by Bernoulli at $y = y_{\hat{U}}$ as shown by Figure 4-22.

Moving towards the smooth wall the pressure difference P_2–P_1 decreased (Figure 4-21) and therefore the Bernoulli and numerical model predictions of U_1–U_2 also decreased. Nearer still to the wall the streamwise velocity (and therefore U_1+U_2) decreased rapidly and the pressure difference reached a constant value so that the Bernoulli and numerical model predictions of U_1–U_2 both increased. Between $y = y_{\hat{U}}$ and the wall the vorticity became increasingly significant limiting the validity of the Bernoulli equation to a streamline. The fact that lines of constant y were not streamlines is the most likely explanation for the discrepancy between U_1–U_2 predicted by the numerical model and the Bernoulli prediction over the range $1.2 \times 10^{-3} < y < y_{\hat{U}}$.

At approximately 1.2×10^{-3} m from the wall a point of inflection occured in the profile of U_1–U_2 given by the numerical model. This probably indicates the point reached by turbulent diffusion from the wall during the time fluid was convected between locations 1 and 2 by the primary flow. Between this point and the wall, turbulent viscosity would give rise to large energy losses which would invalidate the Bernoulli equation. Schlichting (1960, p38) gave the growth in a turbulent boundary layer over length l as

$$\delta = 0.3l\left(\frac{Ul}{\nu}\right)^{-1/5} \qquad \textit{Equation 4-9}$$

Taking U to be the streamwise velocity at a distance of 1.2×10^{-3} m from the wall, given by Figure 4-20 to be about 6 m/s, then the extent of turbulent diffusion between locations 1 and 2 is approximately given by:

$$\delta = 0.37 \times 0.031 \times \left(\frac{6 \times 0.031}{1.17 \times 10^{-6}}\right)^{1/5} \qquad \textit{Equation 4-10}$$

$$= 1.05 \times 10^{-3}\ \text{m}$$

This confirms that the extent of diffusion from the smooth wall was approximately 1.1×10^{-3} m, which corresponds to a $Y^+ \approx 300$. Figure 4-23 shows this to be well into the logarithmic region of the velocity profile and therefore the law of the wall would have been adequately developed for the local value of wall shear stress. This is also confirmed by the good agreement of the logarithmic velocity profiles at locations 1 and 2. The equation of both these lines is

$$\frac{U}{U_\tau} = 4.7 + 5.4 \log \frac{U_\tau y}{\nu} \qquad \textit{Equation 4-11}$$

which corresponds well to the profile obtained in the numerical simulation of a smooth wall only (Equation 4-13).

Comparison of two-layer model results on a smooth wall with the logarithmic law

The simulation on the isolated smooth wall region was also used as a validation of the two-layer model applied to a smooth wall and to investigate the effect of grid dependency. The computed velocity profile for the smooth wall only was fitted using linear regression by the equation

$$\frac{U}{U_\tau} = A + B \log \frac{U_\tau y}{\nu} \qquad \textit{Equation 4-12}$$

to give the coefficients A and B. These are compared with the coefficients used by Nikuradse (1933) and Jayatilleke (1969) in Table 4-2. Figure 4-24 and 4-25 gives the logarithmic velocity profiles for Grid 1 and Grid 3. The results for Grid 2 were nearly identical to those of Grid 3 and are omitted for clarity. From these results it was concluded that:

1. The first computation was not sufficiently grid independent. The two finer grids however did give substantially grid independent solutions.

2. The two-layer model predicted the smooth wall velocity profile to be

$$\frac{U}{U_\tau} = 5.06 + 5.54 \log \frac{U_\tau y}{\nu} \qquad \textit{Equation 4-13}$$

4.1.4 Evaluation of k_s

Schlichting proposed that the resistance due to any roughness configuration could be measured by comparing the developed logarithmic velocity profile with the standard law of the wall for Nikuradse's sand roughness, thus giving a value for the equivalent sand roughness k_s. Using Method 1 described in Section 3.4.1 and the velocity datum level y_0 found from the geometric mean level of the roughness elements, Schlichting obtained values of k_s for each of his roughness experiments.

Figure 4-26 (magnified in Figure 4-27) shows the logarithmic velocity profile over the rough wall obtained numerically for three of Schlichting's roughness experiments. For each roughness plate, the profiles at each Reynolds number were essentially the same so only the highest Reynolds number test for each plate is shown for clarity. The region over which the logarithmic law applied was determined by careful examination of the plot. The points where the profile was observed to deviate from the straight line gave the data range used. This subjective method was backed up by fitting a linear least squares regression line through the data points and confirming that the standard error in the least squares fit was a minimum. The resulting regression lines are also shown in Figures 4-26 and 4-27 by the solid lines. It can be seen that an extremely good fit is obtained from approximately $y - y_0 / h_2 - y_0 = 0.15$ to nearly the full depth of the boundary layer (h_2). According to current theory, this is in the "outer" layer which is not accurately modelled by the law of the wall.

The gradient of this line was found to vary slightly with each period length, (Table 4-1d), ranging from 6.58 to 6.8. These values are significantly higher than the universal gradient quoted by Prandtl and Nikuradse (5.75) and makes any evaluation of k_s based on Nikuradse's law inaccurate. For completeness however, the value of k_s obtained by Schlichting's method is given in Table 4-1d. Also given are the values of k_s obtained using Method 2 (Section 3.4.1). It is interesting to note that Schlichting's method gave consistently higher values of k_s than Method 2.

Further insight is gained by using an alternative method to determine the datum level y_0. Figure 4-28 shows the logarithmic velocity profiles obtained using y_0 found by equating streamwise force moments ($y_{0\ e.s.m.}$ Section 3.5). The profiles are seen to follow the accepted "inner" or logarithmic region and the "outer" or wake region. The two regions intersected at approximately $y-y_0/h_2-y_0 = 0.2$. The gradients obtained from linear least squares regression over the logarithmic region ranged from 5.50 to 4.86, depending on the roughness period length. These are consistently lower than the gradient quoted by Nikuradse, and closer to that found by Jayatilleke of 5.616. The k_s values obtained from this logarithmic velocity profile are also shown in Table 4-1d.

These results show that by altering the value of y_0, both the range and the gradient of the logarithmic region are changed. In order to define an equivalent sand roughness, a universal gradient (constant for all roughness patterns) needs to be established together with a method of determining the datum level which gives the correct gradient and a method of specifying the correct data range.

4.2 Roughness Configuration of Grass *et al.*

4.2.1 Introduction

Numerical simulations were performed on water flow over two-dimensional rib roughness (5 mm square rod) with a period length of 25 mm in a channel of depth 55.9 mm (Grass *et al* 1991, Experiment A). The Reynolds number of the original experiment was 8091 and a laser Doppler anemometer was used to measure the mean velocity profile and Reynolds stresses at two measuring locations, one above the centreline of the roughness elements and the second mid-way between the roughness elements.

In the numerical simulation, a plane of symmetry was used to model the free surface and the PHOENICS controlled periodic boundary conditions were employed to achieve fully developed flow. Turbulence was modelled by a two-layer model and the properties of

water adopted were the same as those used by Grass *et al* (ρ = 998.2 kg m^{-3}, $\nu = 9.7\times10^{-7}$ m^2/s). The computational domain and grid are shown in Figure 4-29.

The results quoted by Grass *et al* include the flow rate and the shear velocity at the bed, obtained by extrapolating the linear total shear stress distributions (Reynolds stress and viscous stress) to the boundary[1]. Two numerical simulations were therefore undertaken. The first used a pressure gradient calculated from

$$\frac{dP}{dx} = \frac{U_\tau^2 \rho}{h} \qquad \textit{Equation 4-14}$$

as the input value for the source of momentum flux per unit volume. In the second simulation this source of momentum flux was adjusted to achieve the flow rate obtained by Grass *et al.*

4.2.2 Results

The results of both the input shear stress and the input flow rate simulations are compared with the experimental results in Table 4-3.

For the simulation in which the shear stress was used as the input, the numerical model under-predicted the flow rate by 21.8% and the maximum velocity by approximately 20%. Similarly poor results were obtained when the flow rate was used as the input, the model over-predicted the bed shear stress by 59%. The error in the maximum velocity was approximately 4%, however inaccuracies of this order were possibly due to extracting the value of maximum velocity from the graph published by Grass *et al.*

The roughness length scale k_0 was obtained by plotting U^+ against $(y-y_0)$ on a logarithmic scale and evaluating $(y-y_0)$ for $U^+ = 0$ by linear least squares regression. The value of y_0 used was that quoted by Grass *et al* of 1.4 mm obtained using a method

---

[1] Note that only the period averaged shear stress distribution will be linear. Although undocumented, it is assumed that Grass *et al* used the average distribution obtained at the two measuring stations.

based on the progressive origin shift suggested by Clauser (1956). The resulting plot of U^+ against $(y-y_0)/k_0$ is shown in Figure 4-30 for the input shear stress method.

Following Grass *et al*, the k_s value may be calculated from (Section 2.1.1):

$$k_s = 30/k_0 \qquad \textit{Equation 4-15}$$

The roughness length scale (and therefore k_s) was over predicted in the numerical model by 24.8% for the input shear stress method and by 31.4% by the input flow rate method.

Convergence

The convergence criterion of an error of less than 0.1% between the driving force due to the pressure gradient and the resistance calculated from the sum of the boundary forces was not achieved. Final errors of 0.26% and 0.32% were obtained for the input shear stress and input flow rate methods respectively. Further iterations were found to have no effect on the solution and the history of residuals was constant. It should be noted that obtaining the same convergence criterion for similar flow regimes in Section 5.3.1 was also found to be problematical.

However, the transverse distribution of period averaged shear stress (Figure 4-31) was linear with a slope corresponding to the pressure gradient.

4.2.3 Discussion

The numerical solution (Figure 4-32[2]) shows that the streamline separating from the trailing edge of the upstream roughness element re-attaches below the tip of the next element downstream (at $y/k_h = 0.77$) resulting in a elongated vortex trapped between each roughness element. The distribution of the resulting normal stress acting on the upstream vertical face of the roughness element is shown in Figure 4-33, illustrating the effect of streamline impingement. The average value of normal stress acting on this face

---

[2] Figure 4-32 relates to the simulation in which the shear stress was used as the input, however similar streamlines were observed for the input flow rate simulation.

is governed by the curvature of the free-stream streamlines in this region. The low degree of curvature shown in Figure 4-32 indicates a relatively low form drag associated with the roughness element. A higher form drag would result if the point of re-attachment were to move lower down the block or onto the bed, requiring a larger curvature of the streamlines.

In order to match the total bed shear stress quoted by Grass *et al* for his given flow rate, a decrease in the numerical prediction of resistance is required, which could be achieved by the point of re-attachment moving further up the roughness element. This would result in a quasi-smooth or skimming type of flow regime, for which Perry *et al* (1969) suggested the dominant scaling parameter in the resistance function to be the flow depth, and which they denoted as 'd' type roughness. However this conclusion would be contrary to Grass *et al*'s description of their roughness as 'k' type. Unfortunately, time averaged streamlines were not presented by Grass *et al*, nor was the method by which the roughness was defined as 'k' type. There must therefore be some uncertainty about the roughness regime in this particular case.

Whether or not the roughness was 'k' type or 'd' type does not serve to explain the discrepancy between the physical measurement and the numerical model results. Given the much better agreement obtained in Schlichting's experiments (Section 4.1.2), where the model predicted re-attachment on the bed, it is tentatively concluded that the numerical model was inaccurate for the flow regime in which re-attachment occurred on the vertical face of a roughness element. There are at least two possible reasons for this:

1. Grid dependence. In particular a potential barrier to turbulent diffusion is created by the thin layer of cells of very wide aspect ratio across the top of the roughness elements on the plane $y = k_h$. Given the very small v-component limiting cross-stream convective flux in this case the possibility of grid dependence carried some weight. However, enlarging the cells in this region gave nearly identical streamlines. This is shown in Figure 4-32 for which the cross-stream grid consisted of 115 uniformly distributed cells. Although not exhaustively proved,

grid dependence seems an unlikely explanation for the large discrepancies between numerical model and experiment.

2. Inadequate representation of the flow physics by the two-layer model. The point of re-attachment on the vertical face of a roughness element is likely to be sensitive to the turbulence structure in this region, which in the model had been Reynolds averaged and its diffusive characteristics represented by an isotropic eddy viscosity. This is the more likely explanation for the discrepancies between model and experiment.

Figures relating to Schlichting's experiments

NB: Diagrams of grid and domains are not to scale.

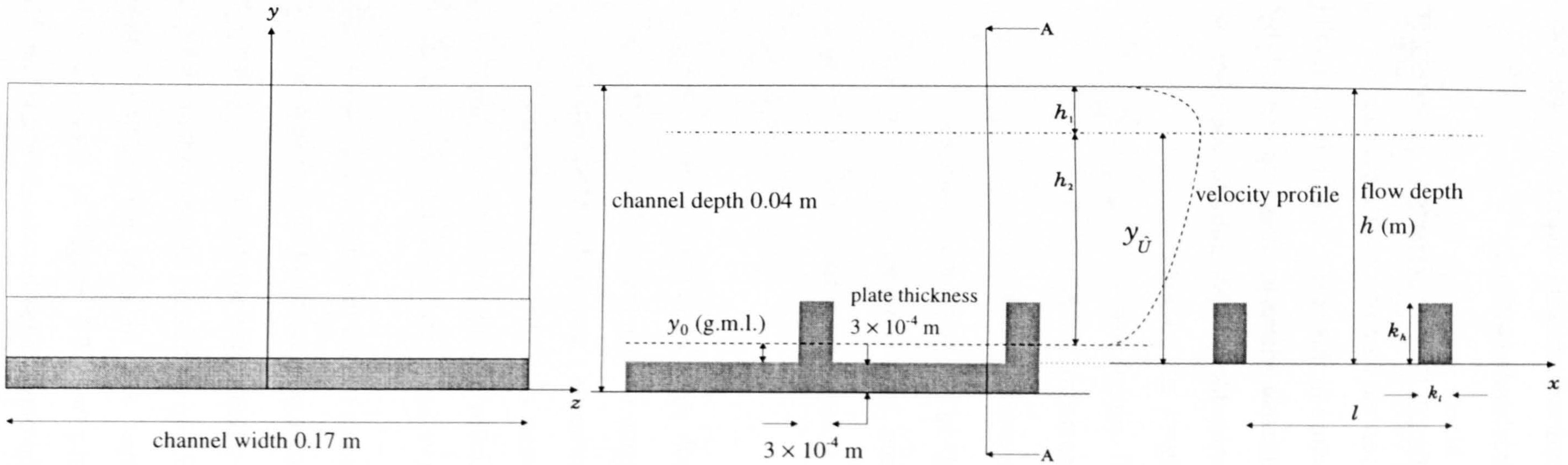


Figure 4-1: Comparison of experimental and numerical geometries for Schlichting's roughness experiments.

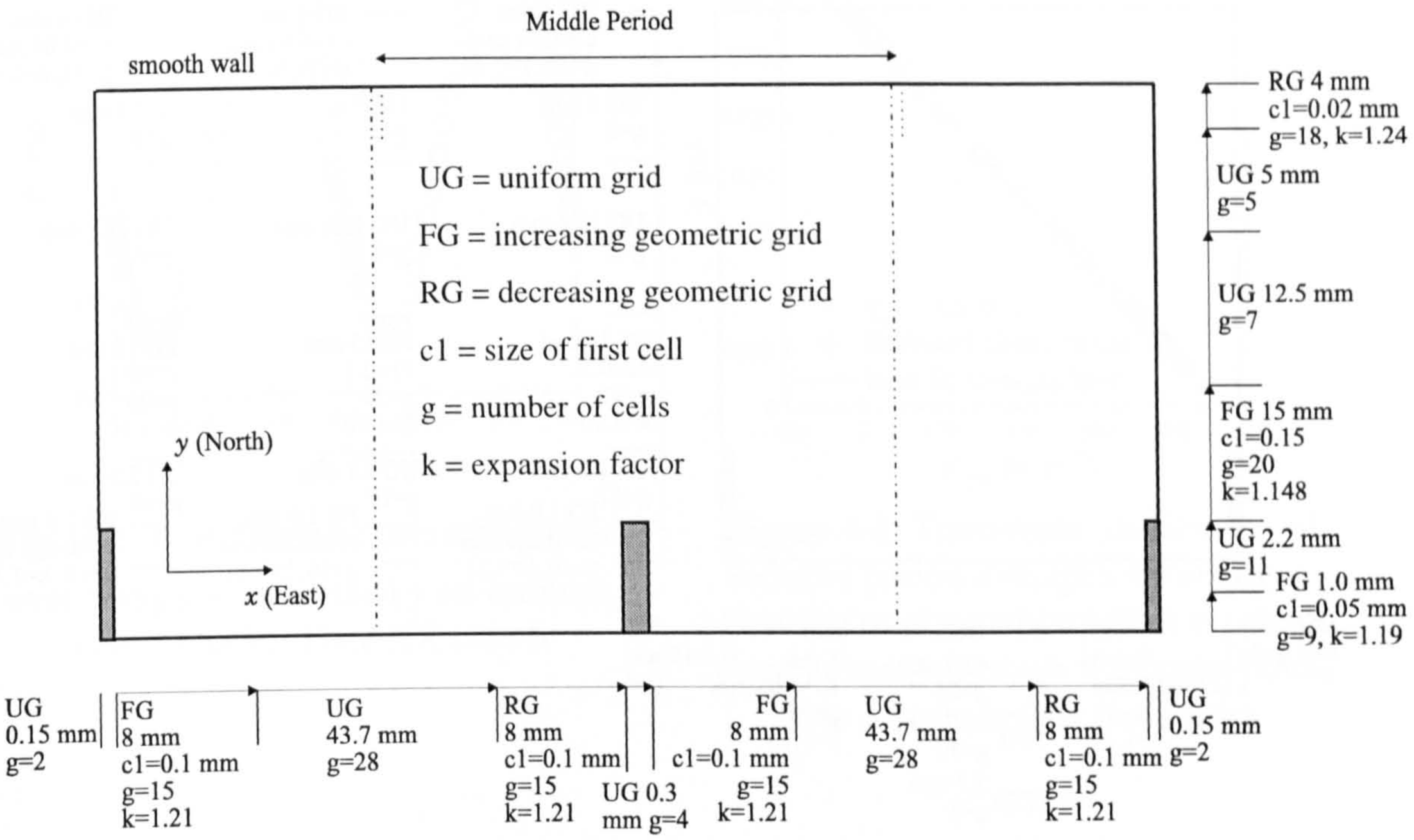


Figure 4-2: Computational grid and domain, plate XX (Grid 1)

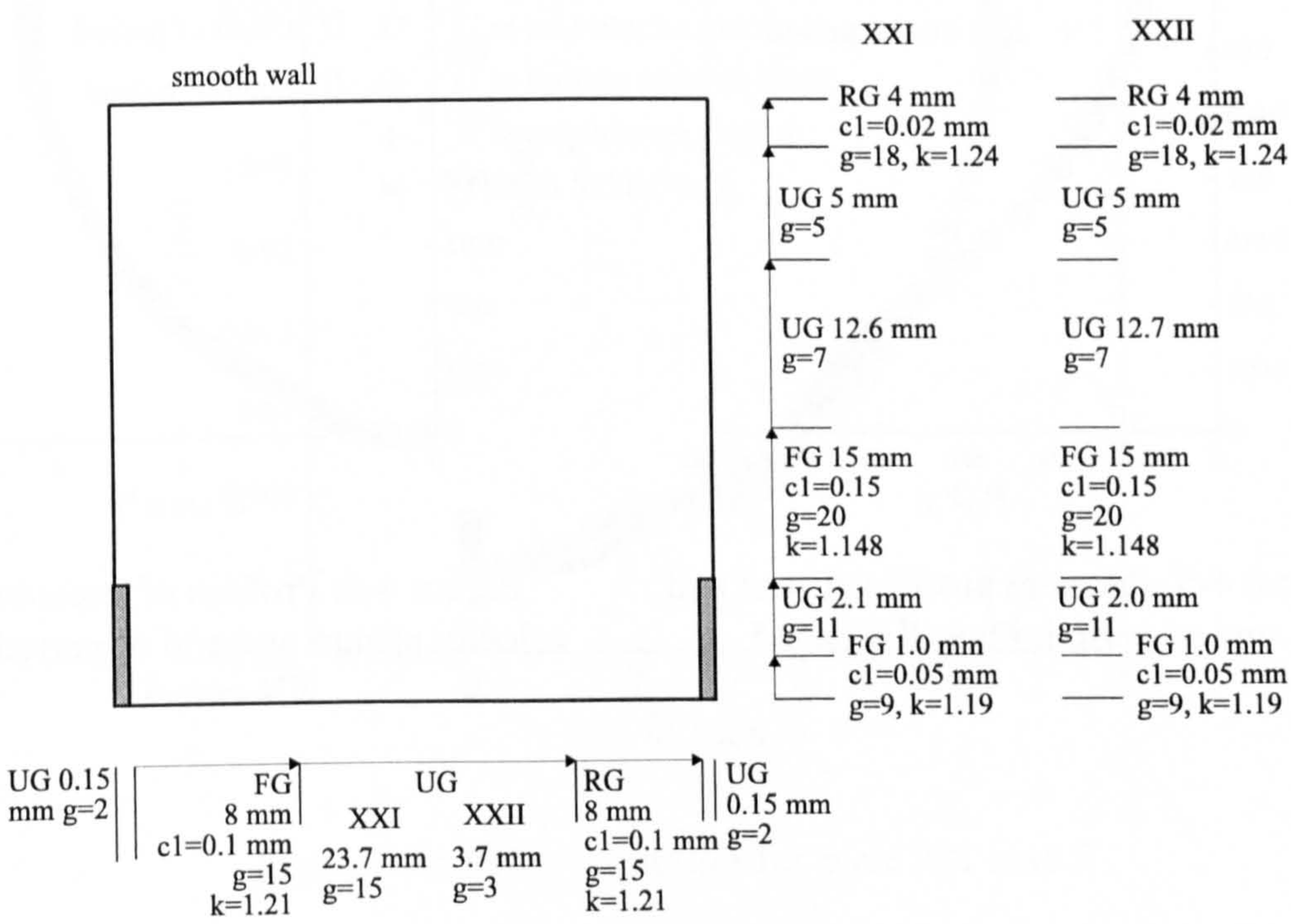


Figure 4-3: Computational grid and domain for plates XXI and XXII (Grid 1)

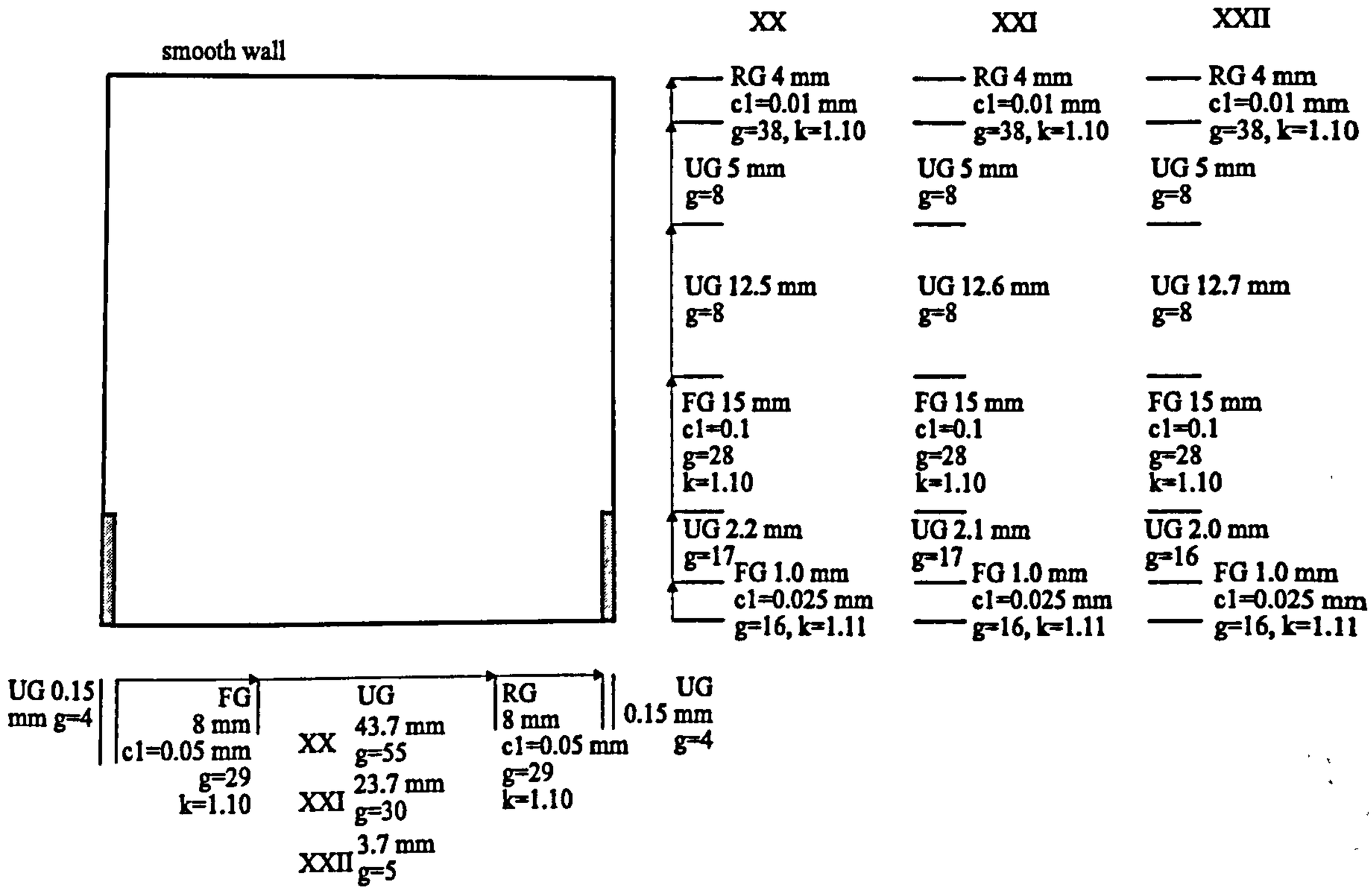


Figure 4-4: Computational grid and domain for plates XX, XXI and XXII (Grid 2)

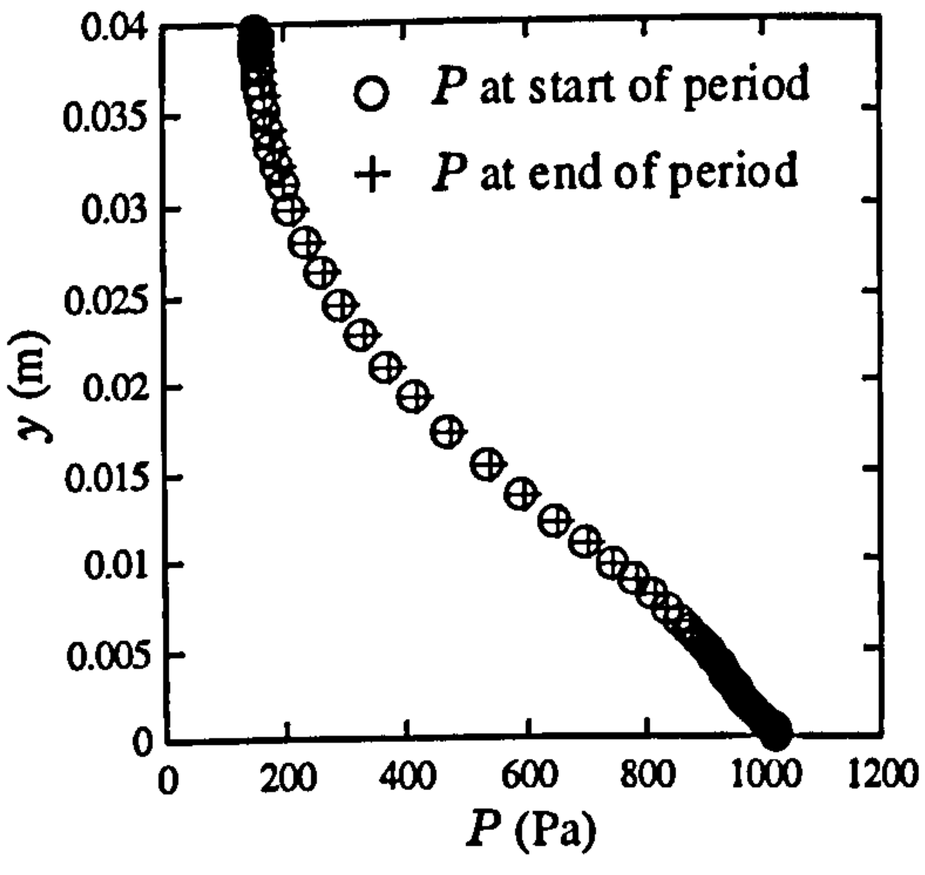


Figure 4-5: Pressure profiles at start and end of period. Plate XX case 5

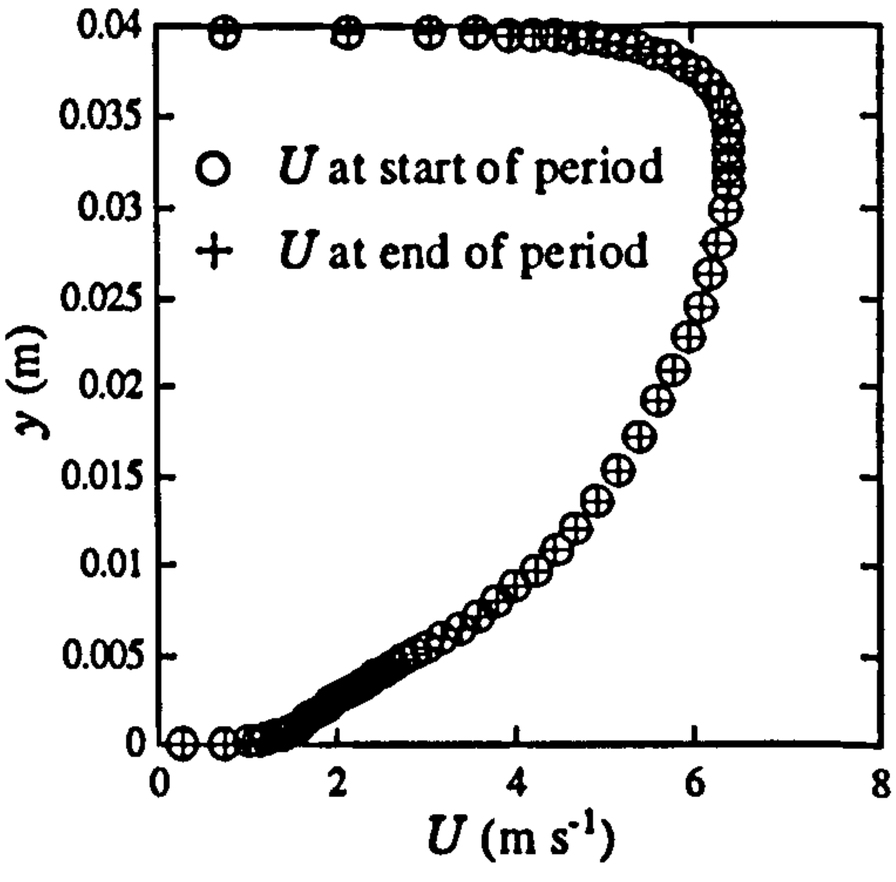


Figure 4-6: Profiles of streamwise velocity at start and end of period. Plate XX case 5

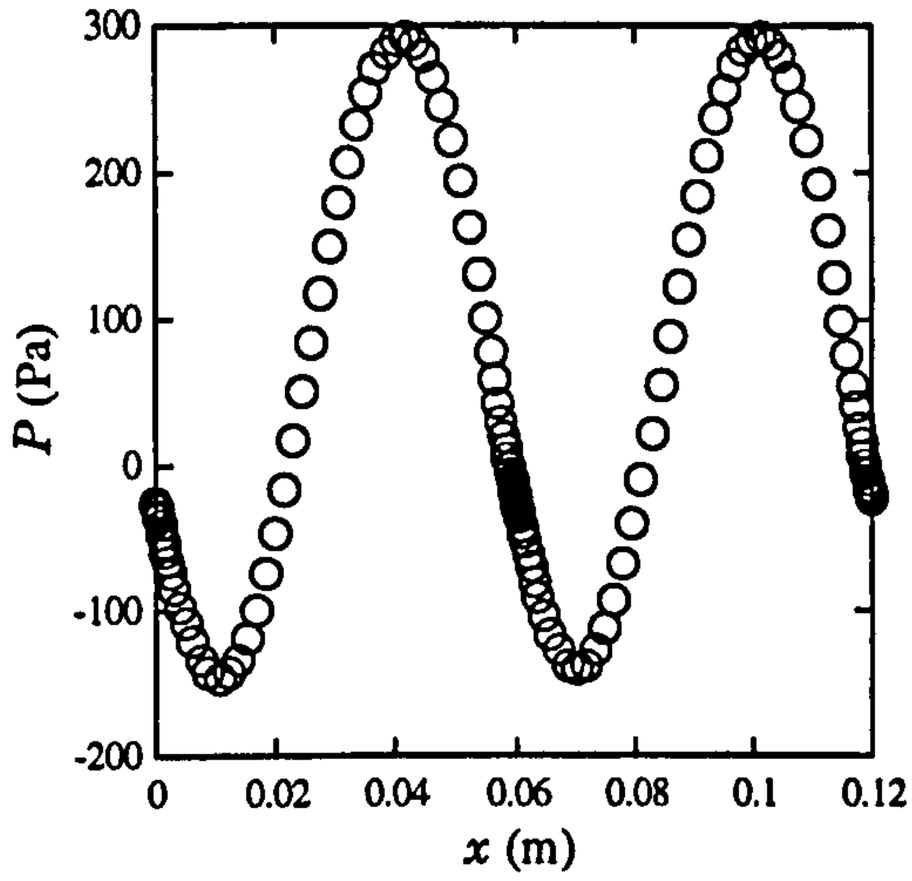


Figure 4-7: Streamwise pressure profile over two periods at fixed y co-ordinate above blocks. Plate XX case 5

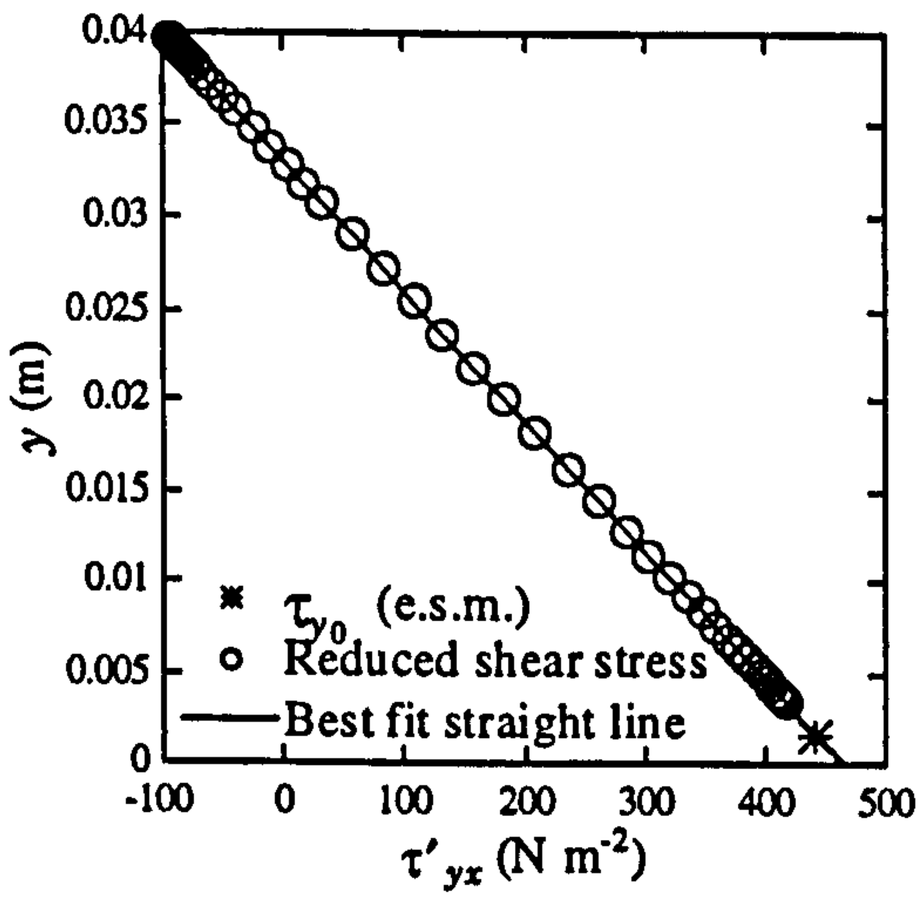


Figure 4-8: Transverse distribution of reduced period averaged shear stress. Position of plane upon which resultant period averaged rough wall shear stress acts is indicated by an asterisk

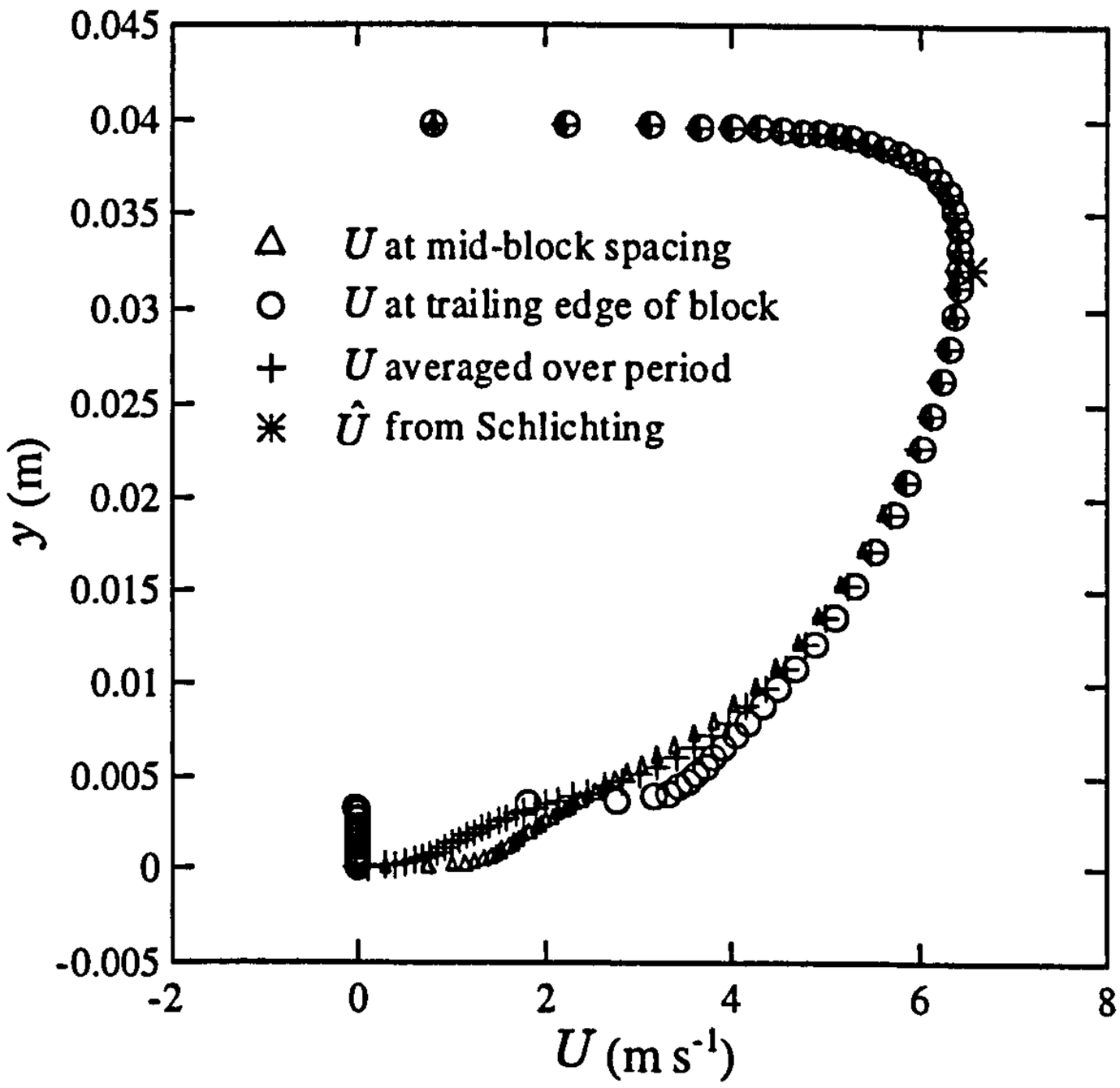


Figure 4-9: Velocity profiles for plate XX case 5

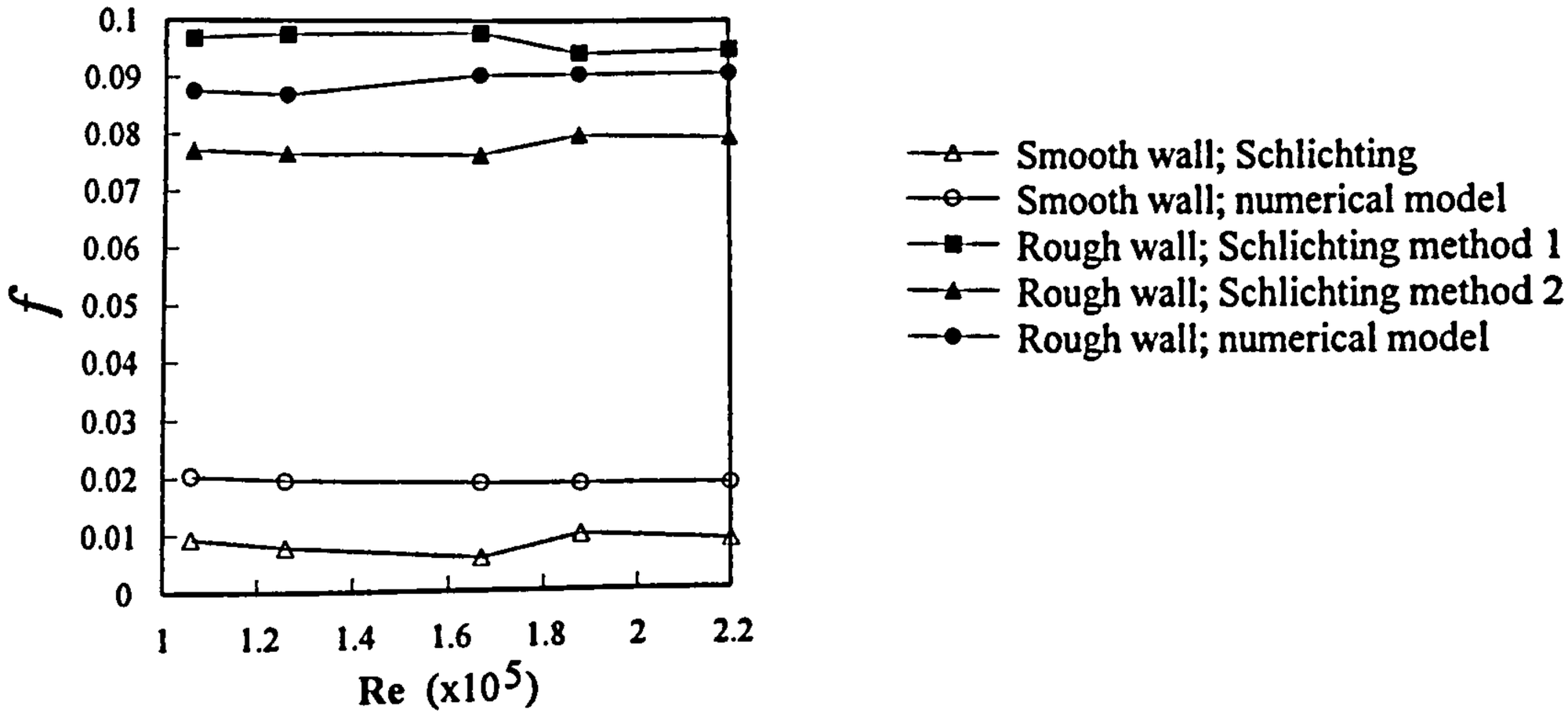


Figure 4-10: Friction factor for plate XX

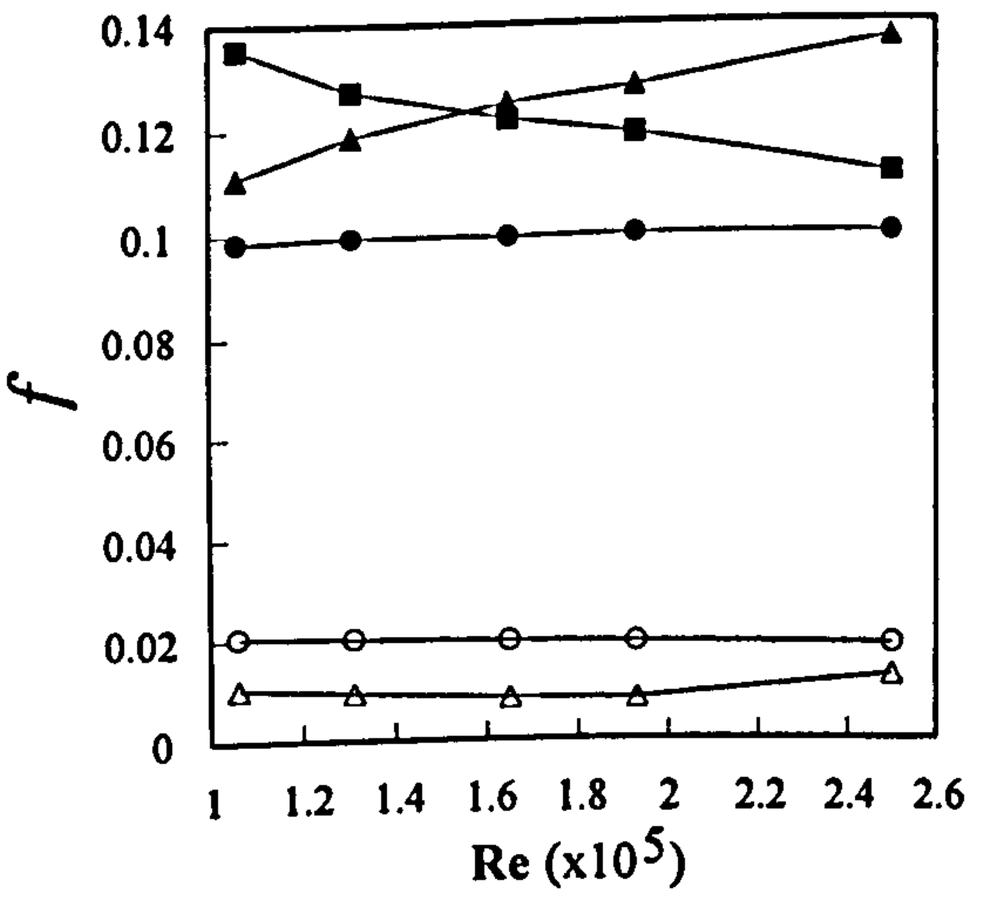


Figure 4-11: Friction factor for plate XXI

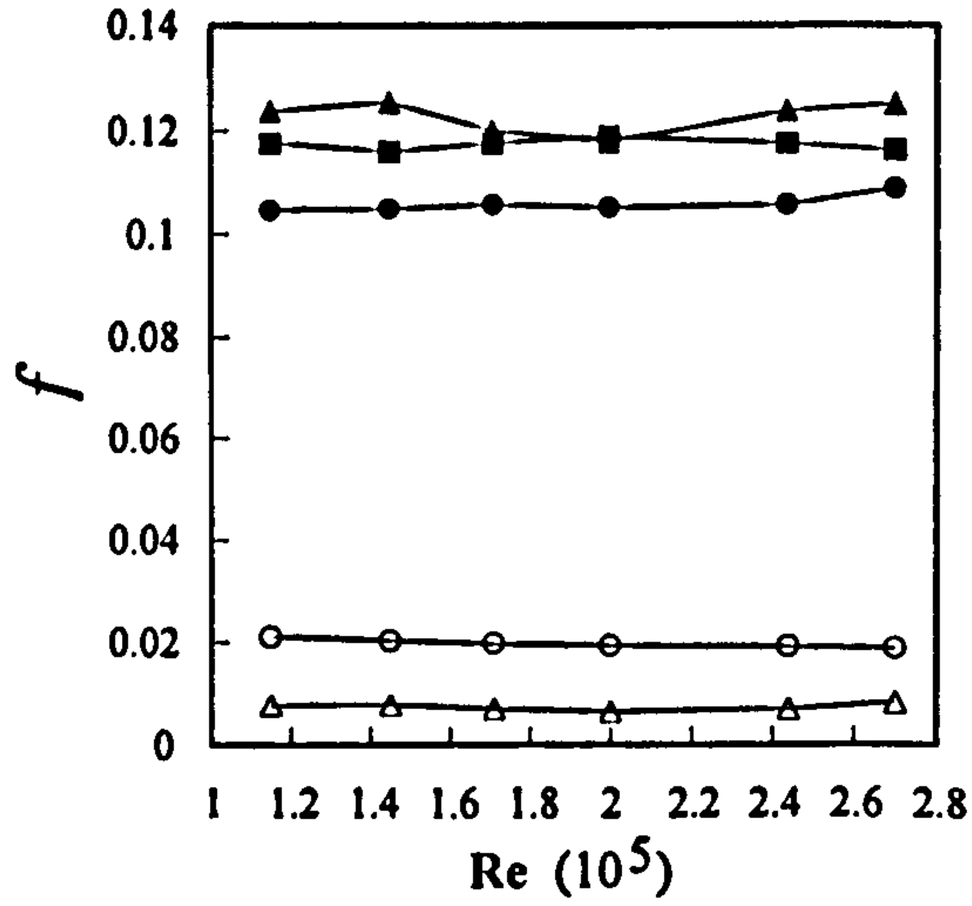


Figure 4-12: Friction factor for plate XXII

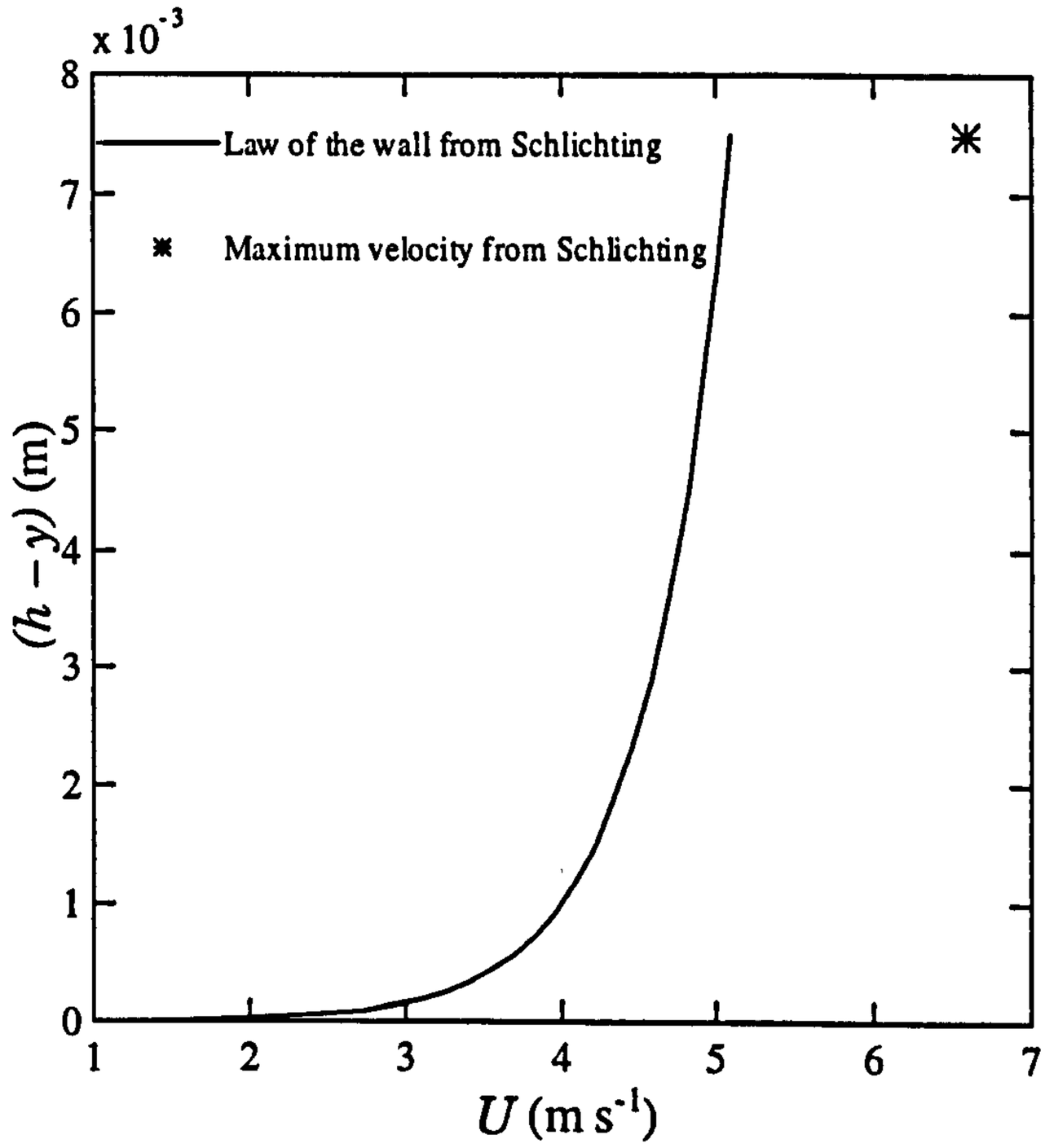


Figure 4-13: Velocity profile for the smooth wall predicted by the law of the wall using the value of U_τ quoted by Schlichting, compared with Schlichting's measured value of maximum velocity

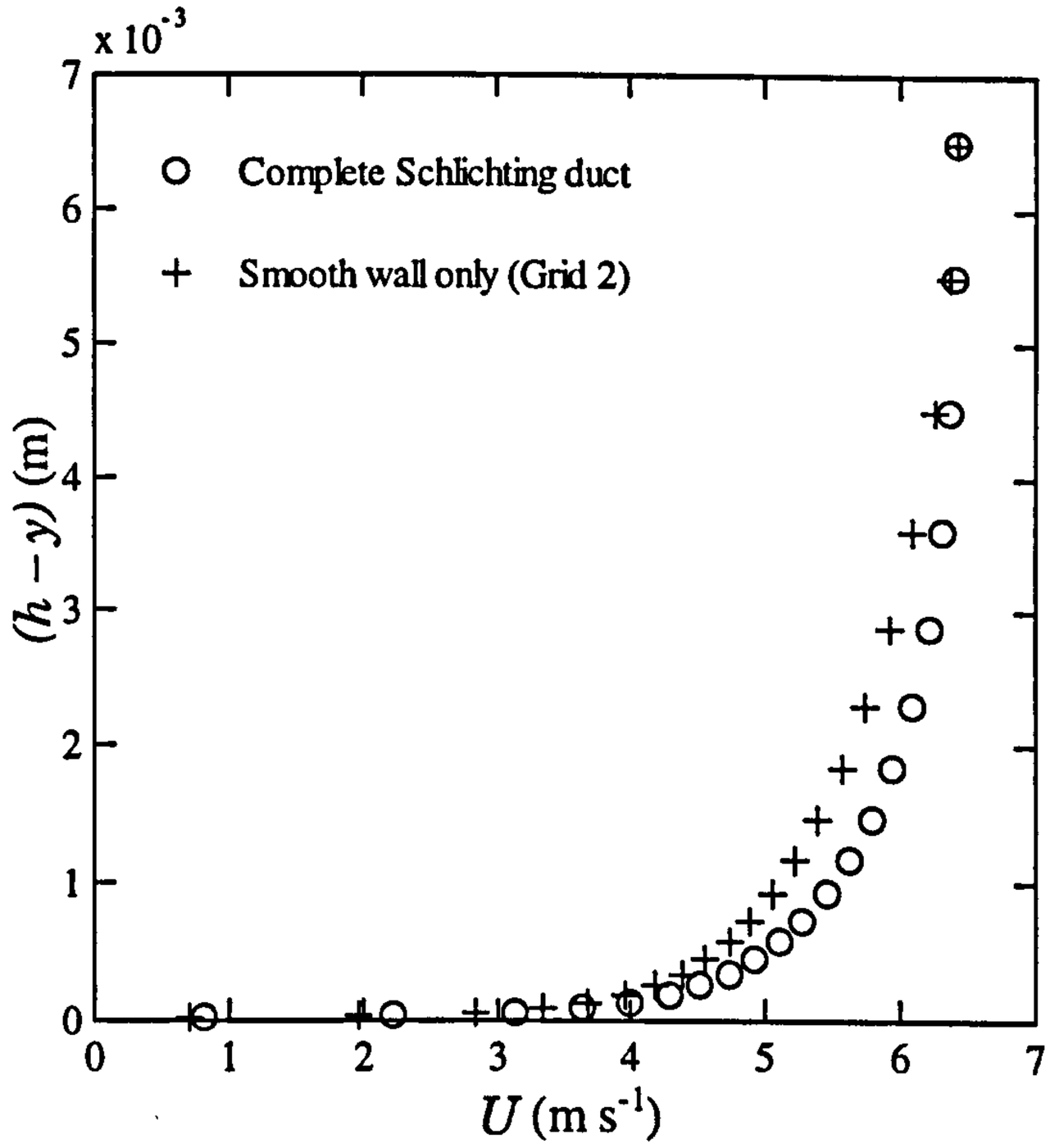


Figure 4-14: Velocity profiles over the smooth wall for the complete duct and the smooth wall only test

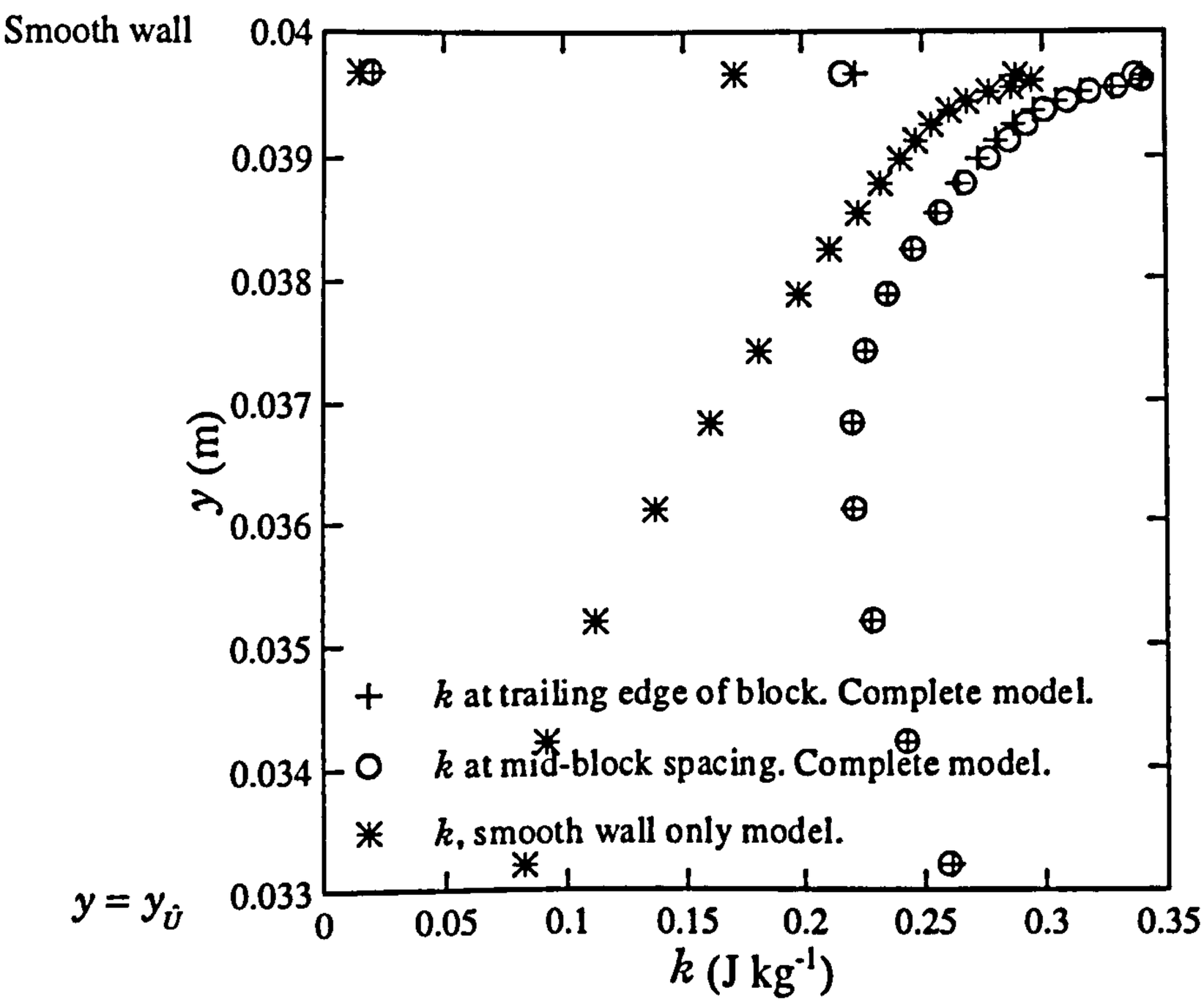


Figure 4-15: Distribution of turbulent kinetic energy near the smooth wall

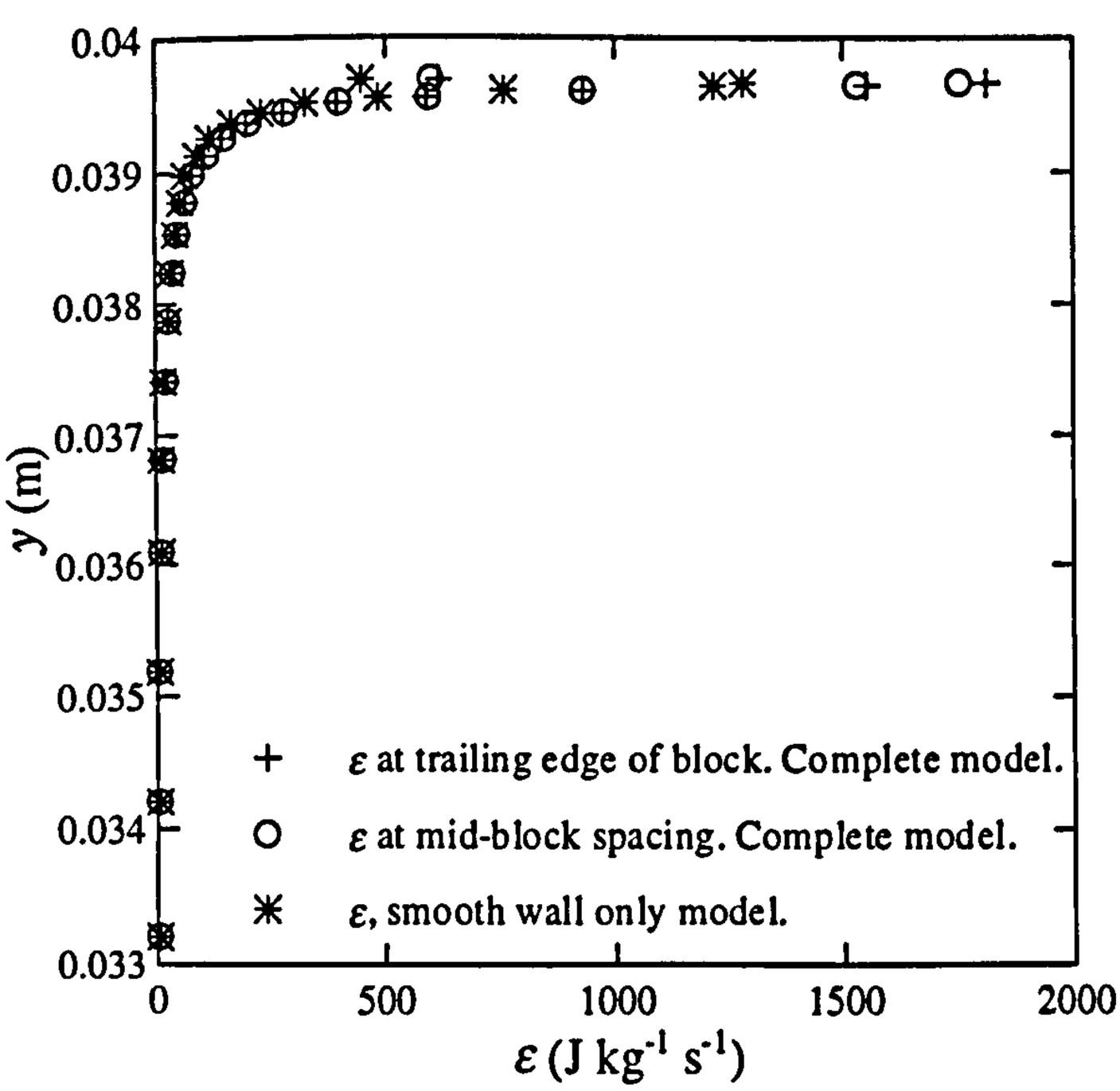


Figure 4-16: Distribution of the dissipation rate of turbulent kinetic energy over the smooth wall

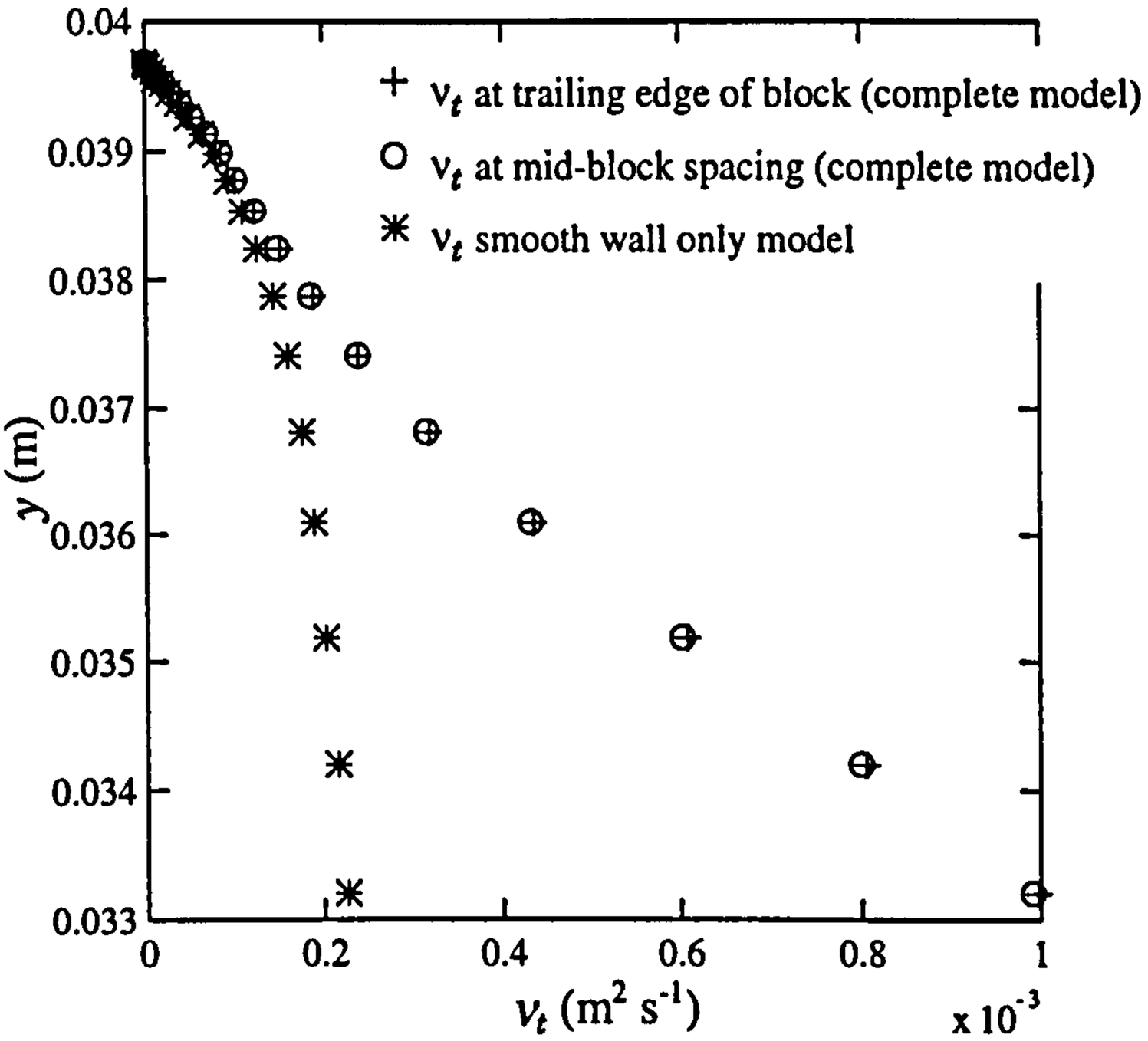


Figure 4-17: Distribution of turbulent kinematic viscosity over the smooth wall

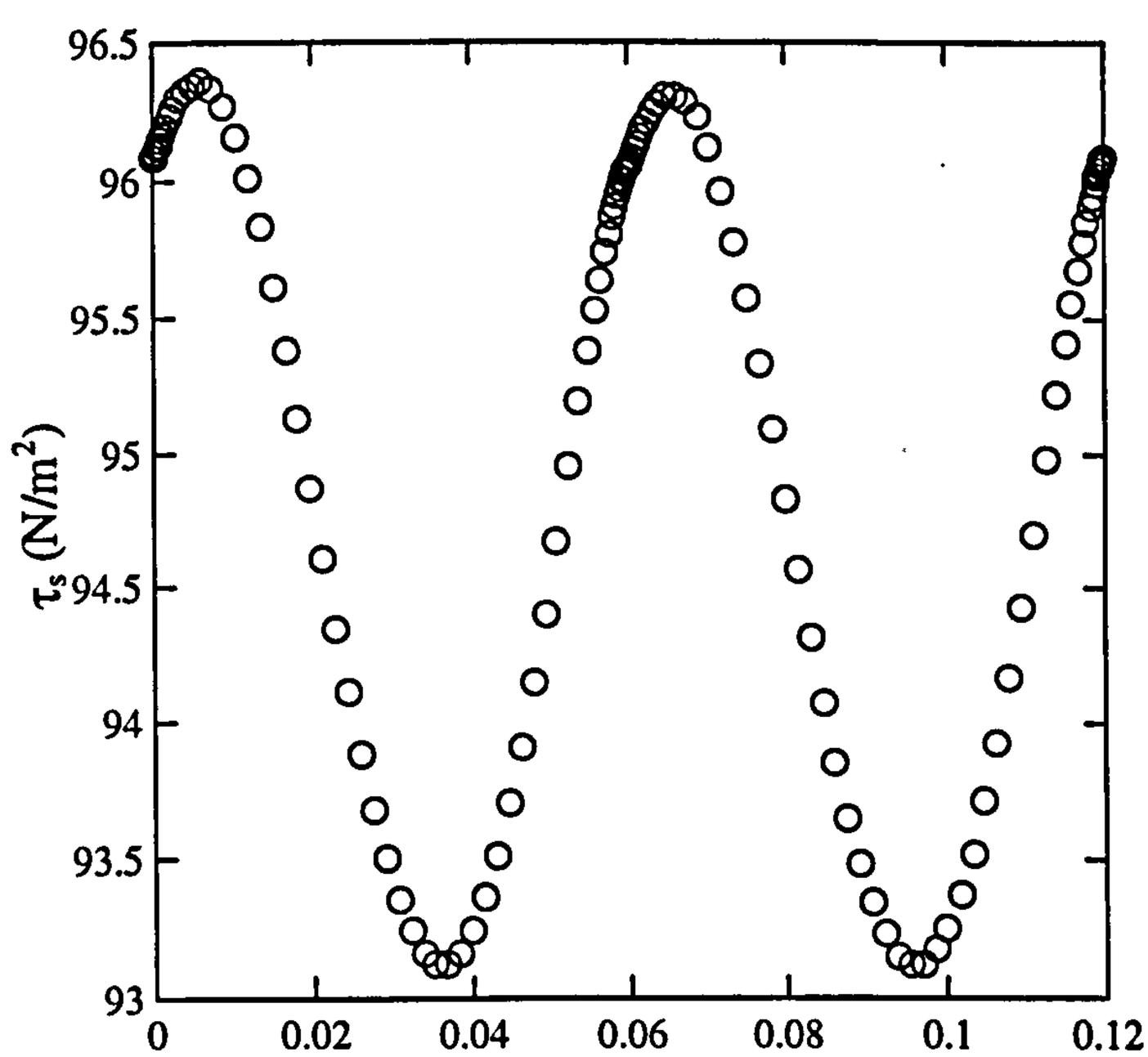


Figure 4-18: Streamwise distribution of smooth wall shear stress

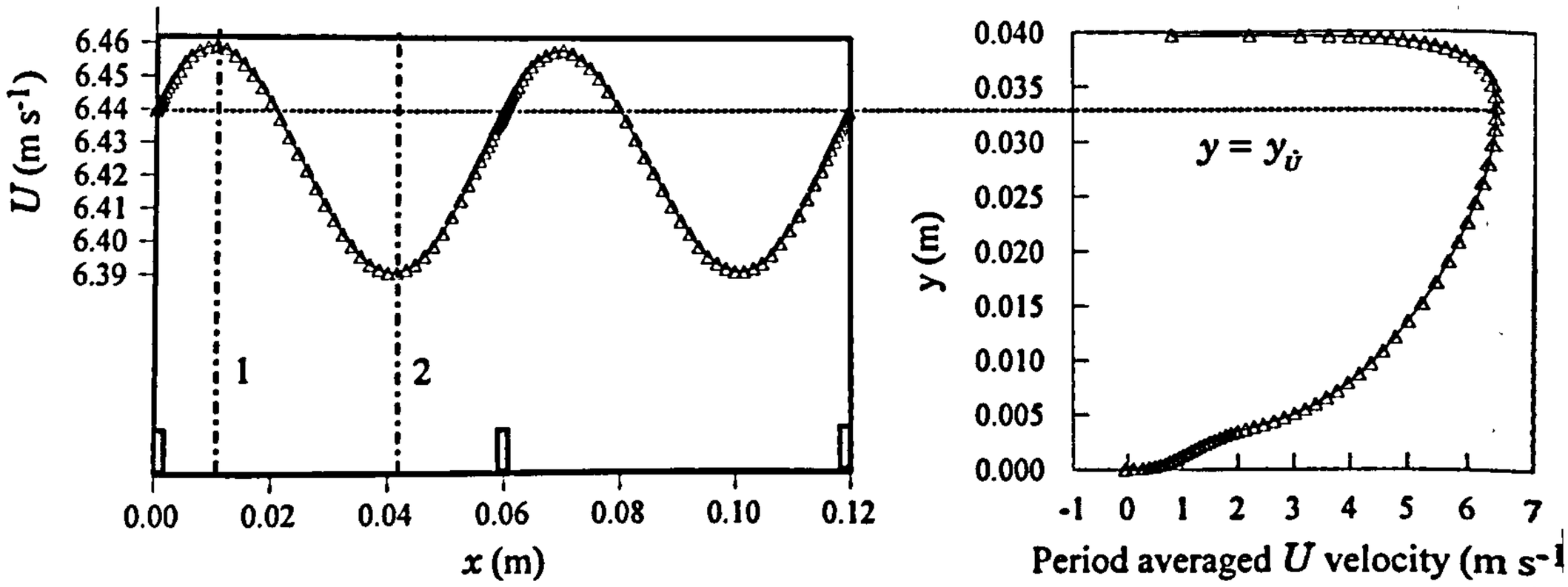


Figure 4-19: Distribution of streamwise velocity located on the plane $y = y_{\hat{U}}$

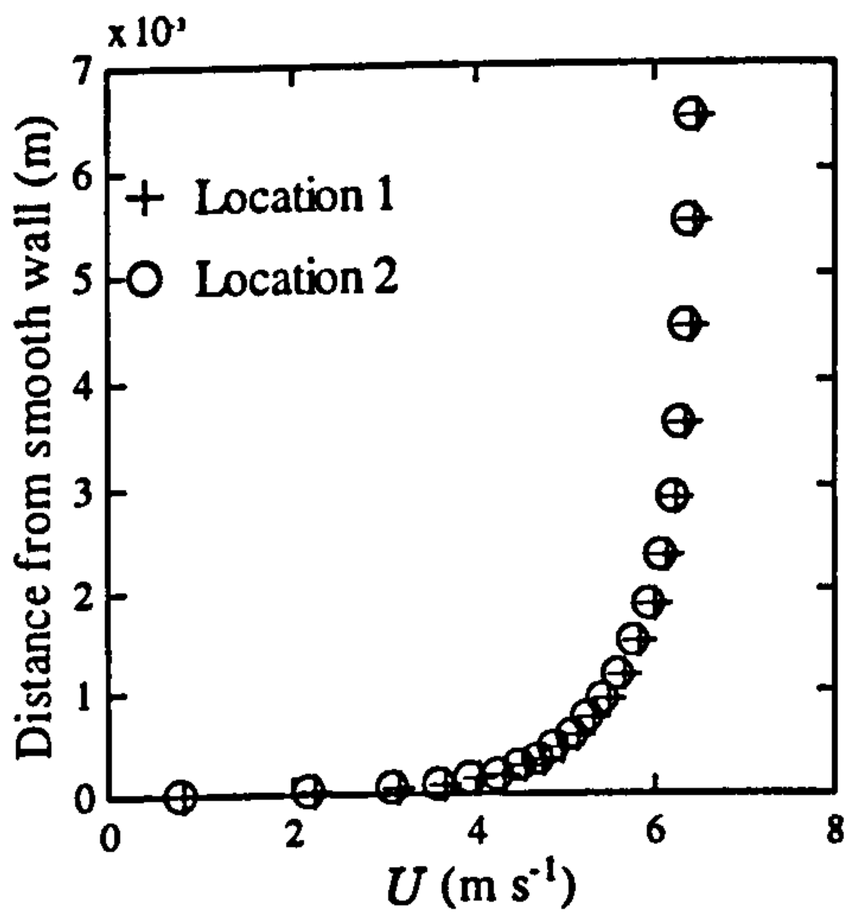


Figure 4-20: Smooth wall velocity profiles at locations of high (1) and low (2) streamwise velocity

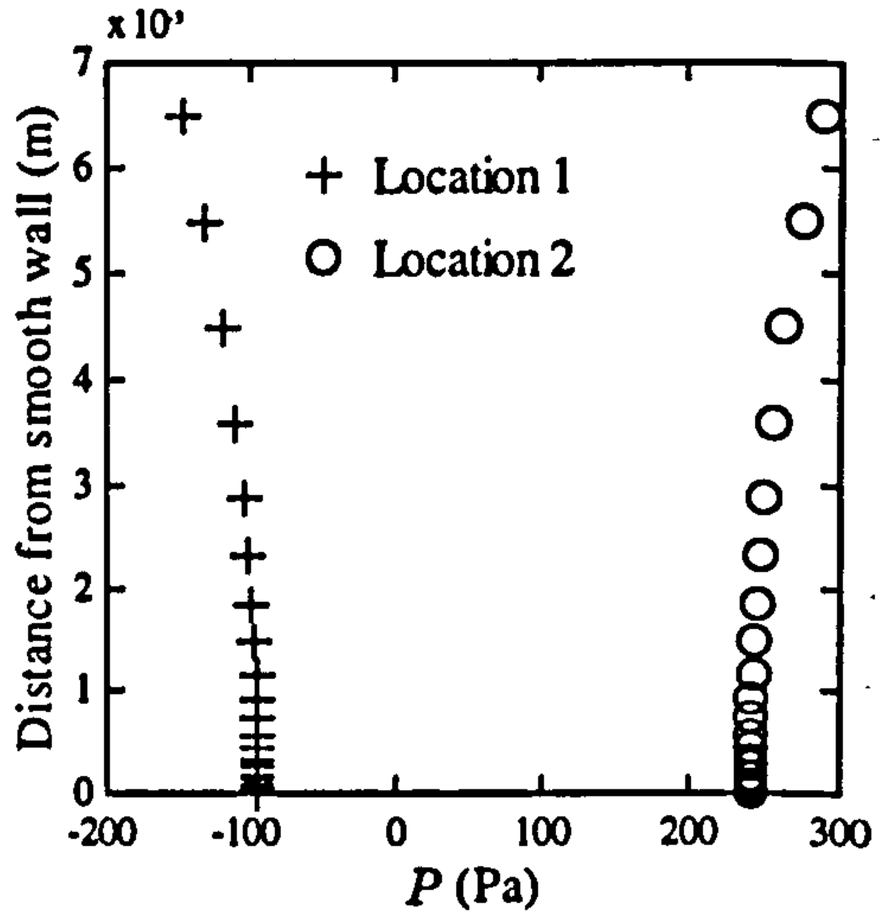


Figure 4-21: Profiles of pressure near the smooth wall at locations of high (1) and low (2) streamwise velocity

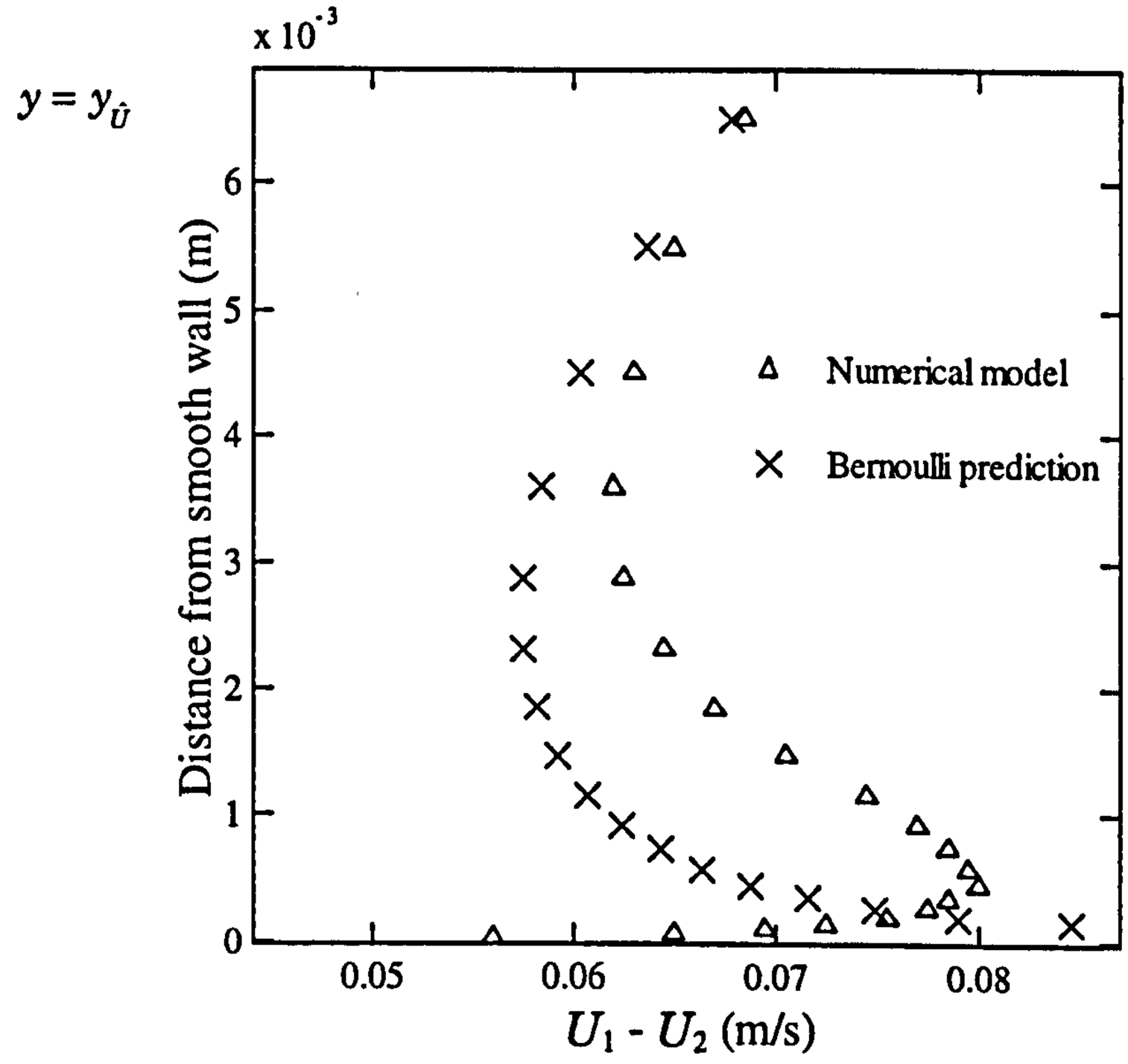


Figure 4-22: Comparison of the velocity difference between locations 1 and 2 with Bernoulli's prediction

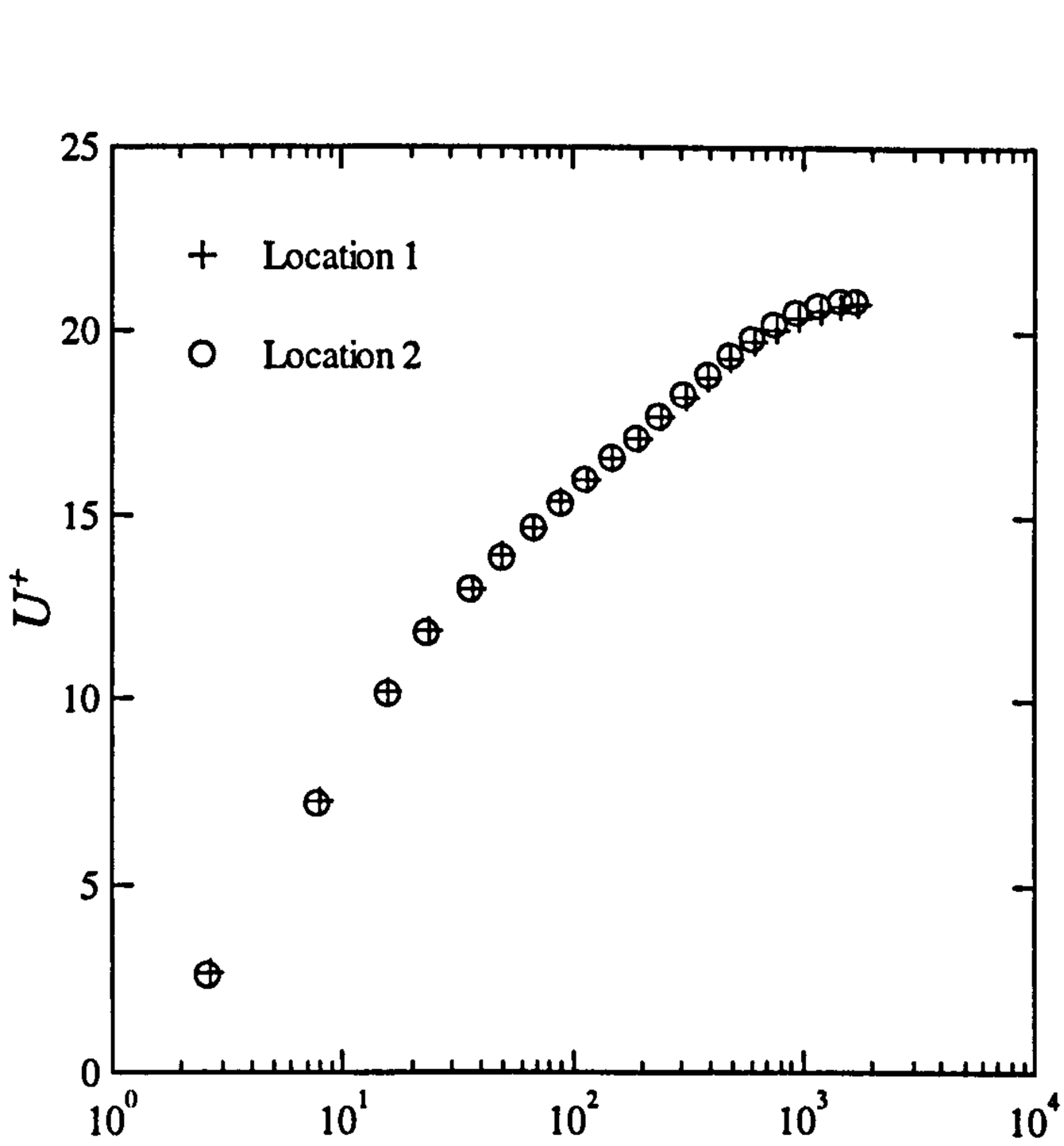


Figure 4-23: Logarithmic velocity profile of the smooth wall region at locations 1 and 2

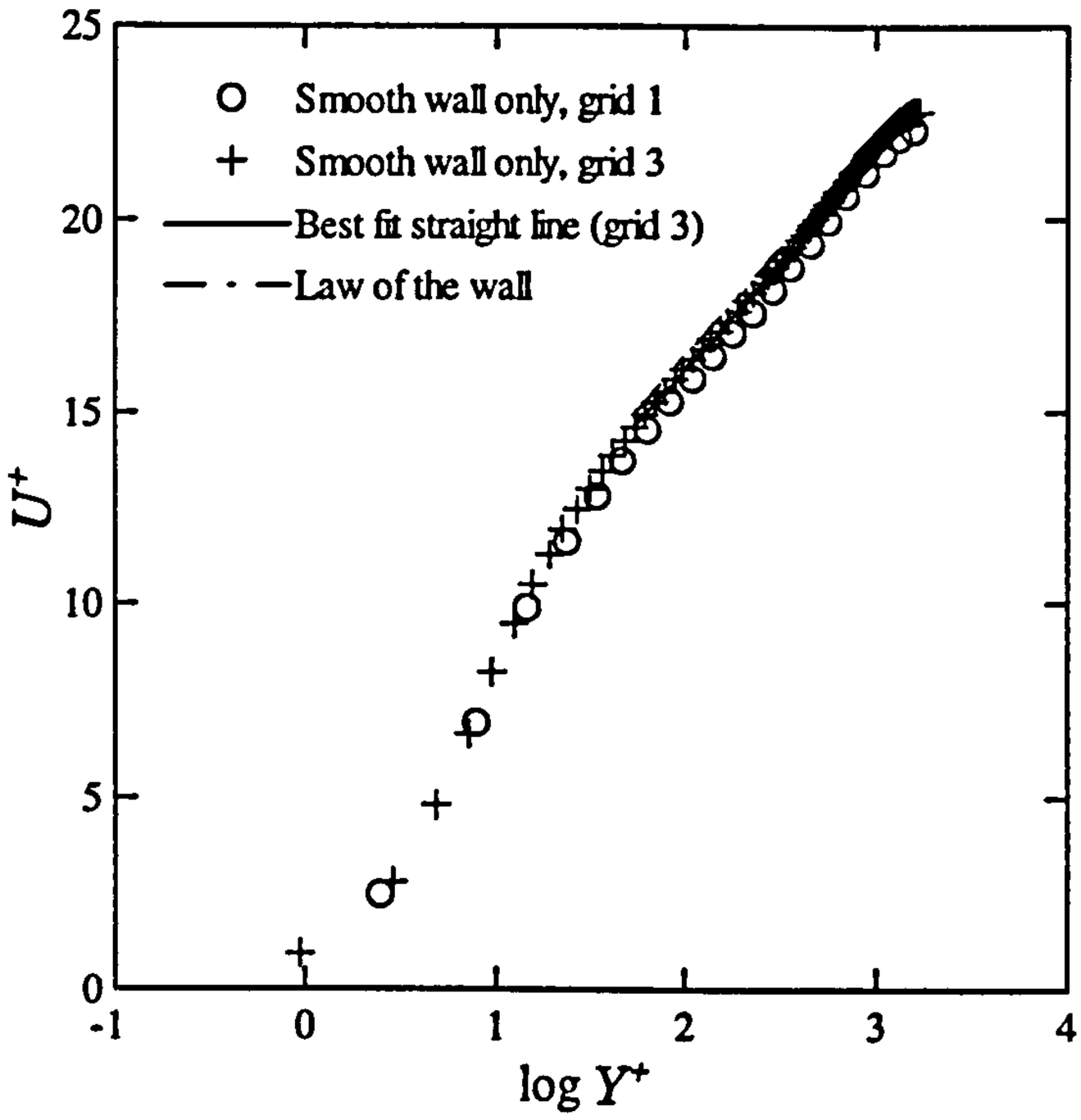


Figure 4-24: Logarithmic velocity profile on the smooth wall; comparison with the law of the wall after Jayatilleke (1969) and showing the effect of grid size

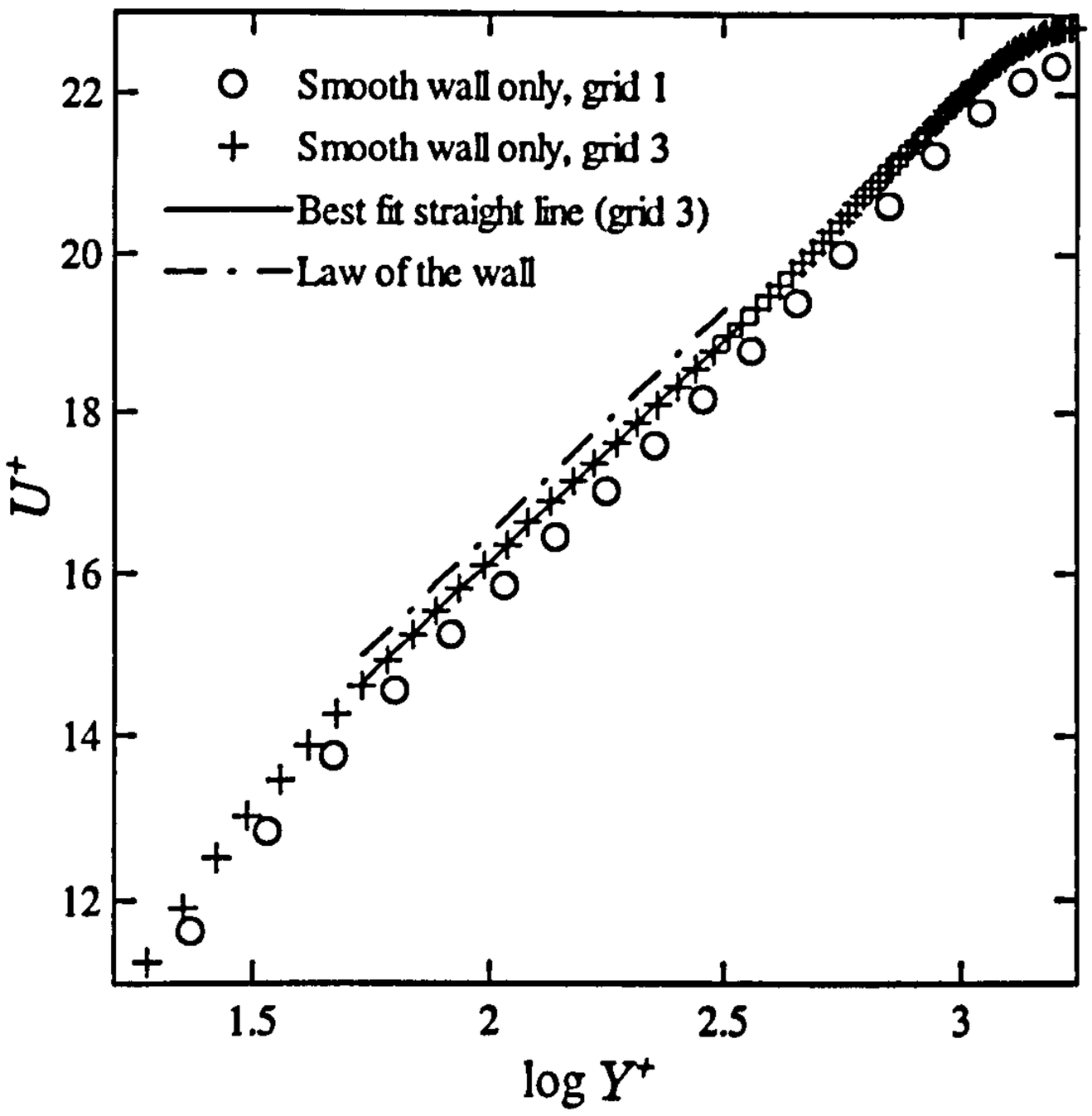


Figure 4-25: Expanded view of Figure 4-24

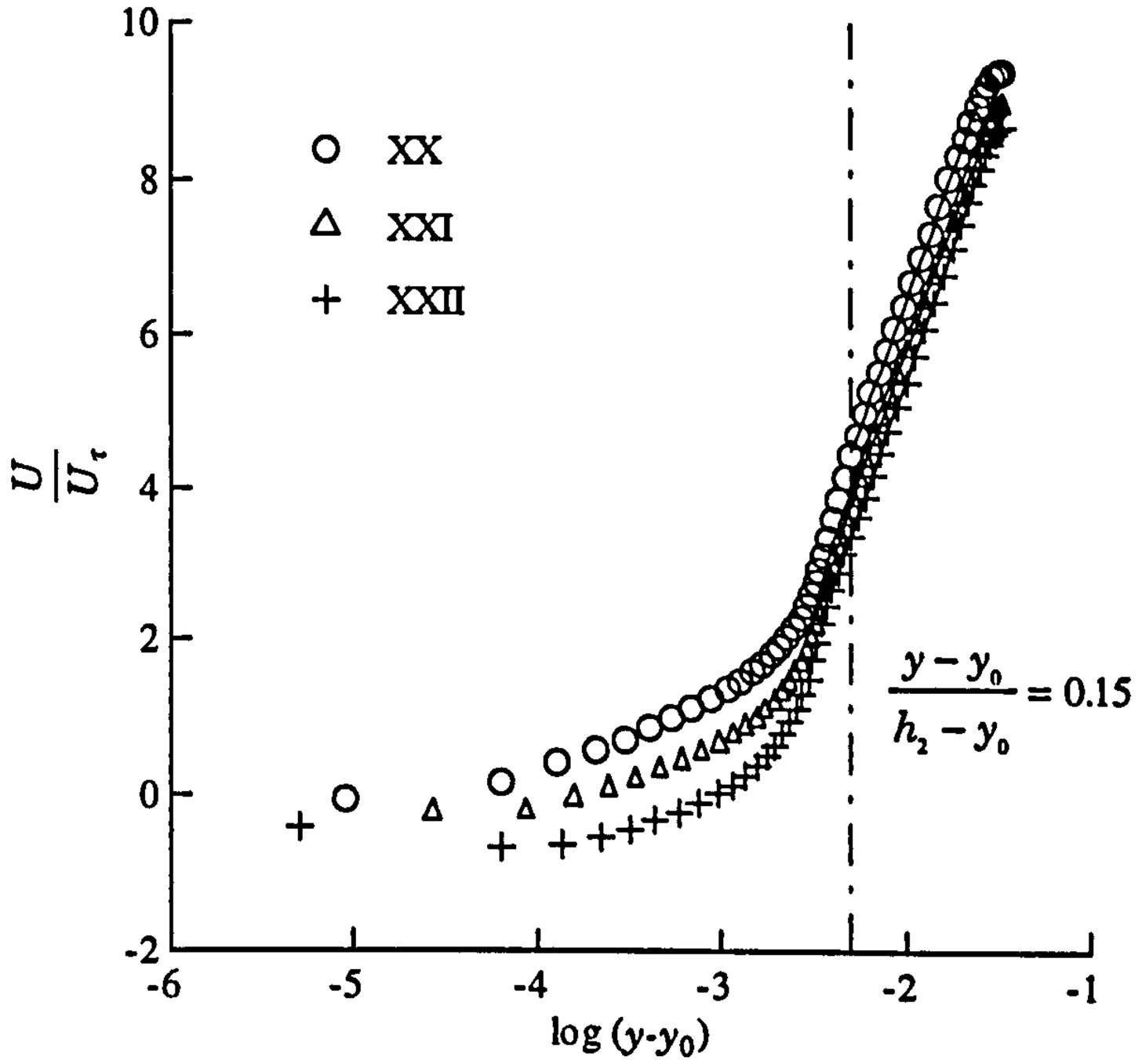


Figure 4-26: Logarithmic velocity profiles, with y_0 = g.m.l. The solid line in each case indicates the best fit straight line in the logarithmic region.

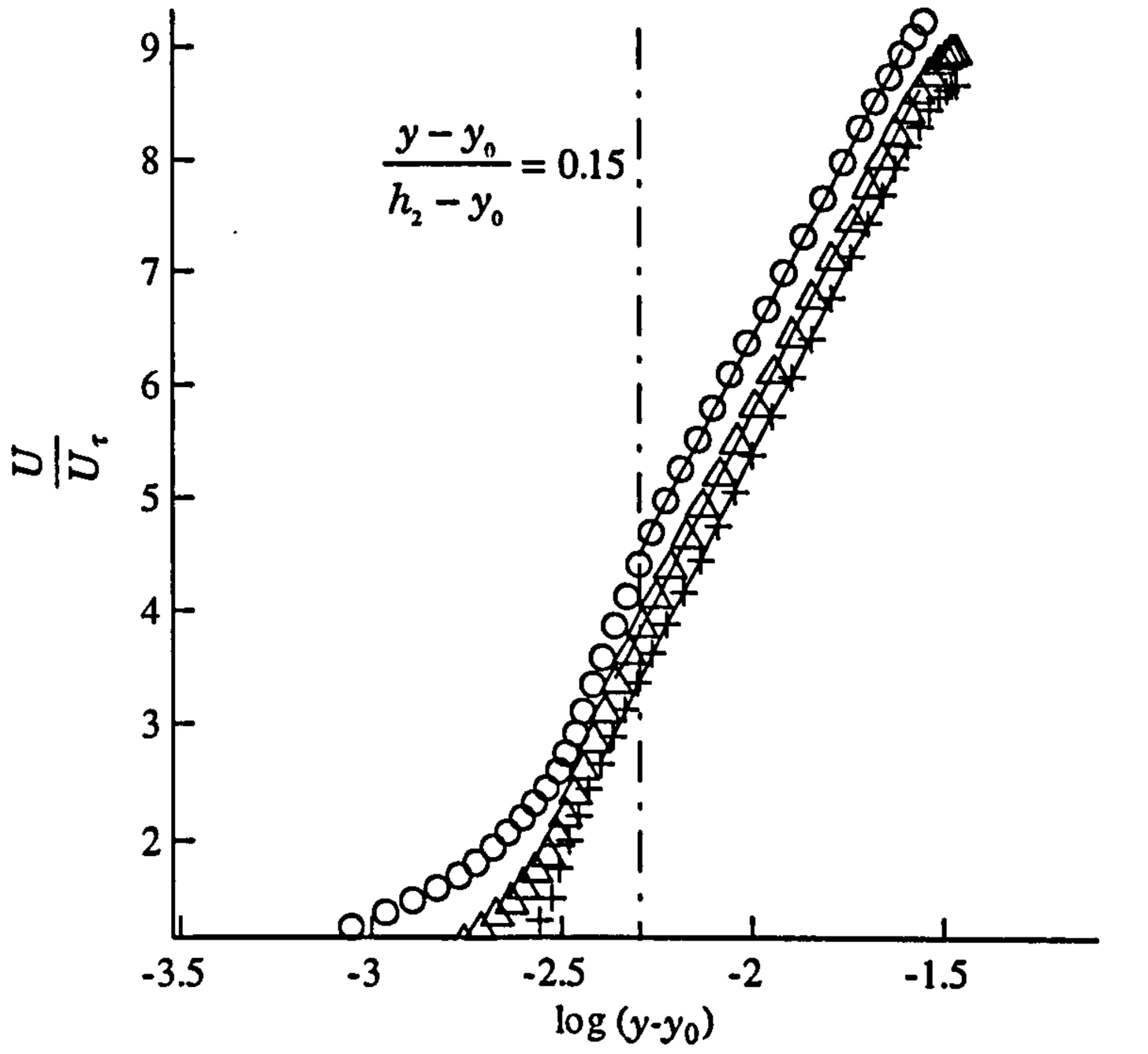


Figure 4-27: Expanded view of Figure 4-26

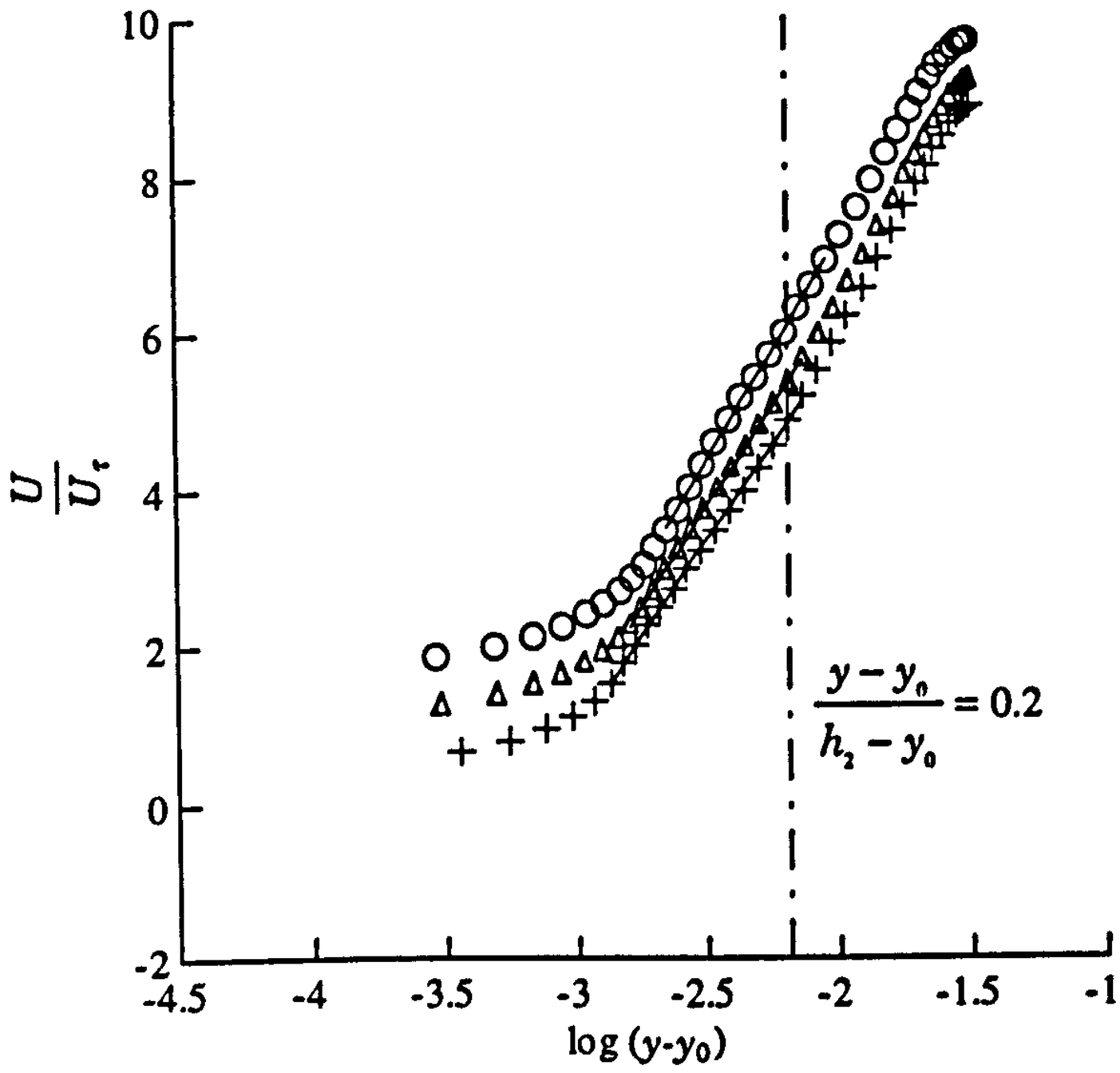


Figure 4-28: Logarithmic velocity profiles, with y_0 (e.s.m.). The solid line indicates the best fit straight line in the logarithmic region.

Figures relating to the roughness configuration of Grass *et al*

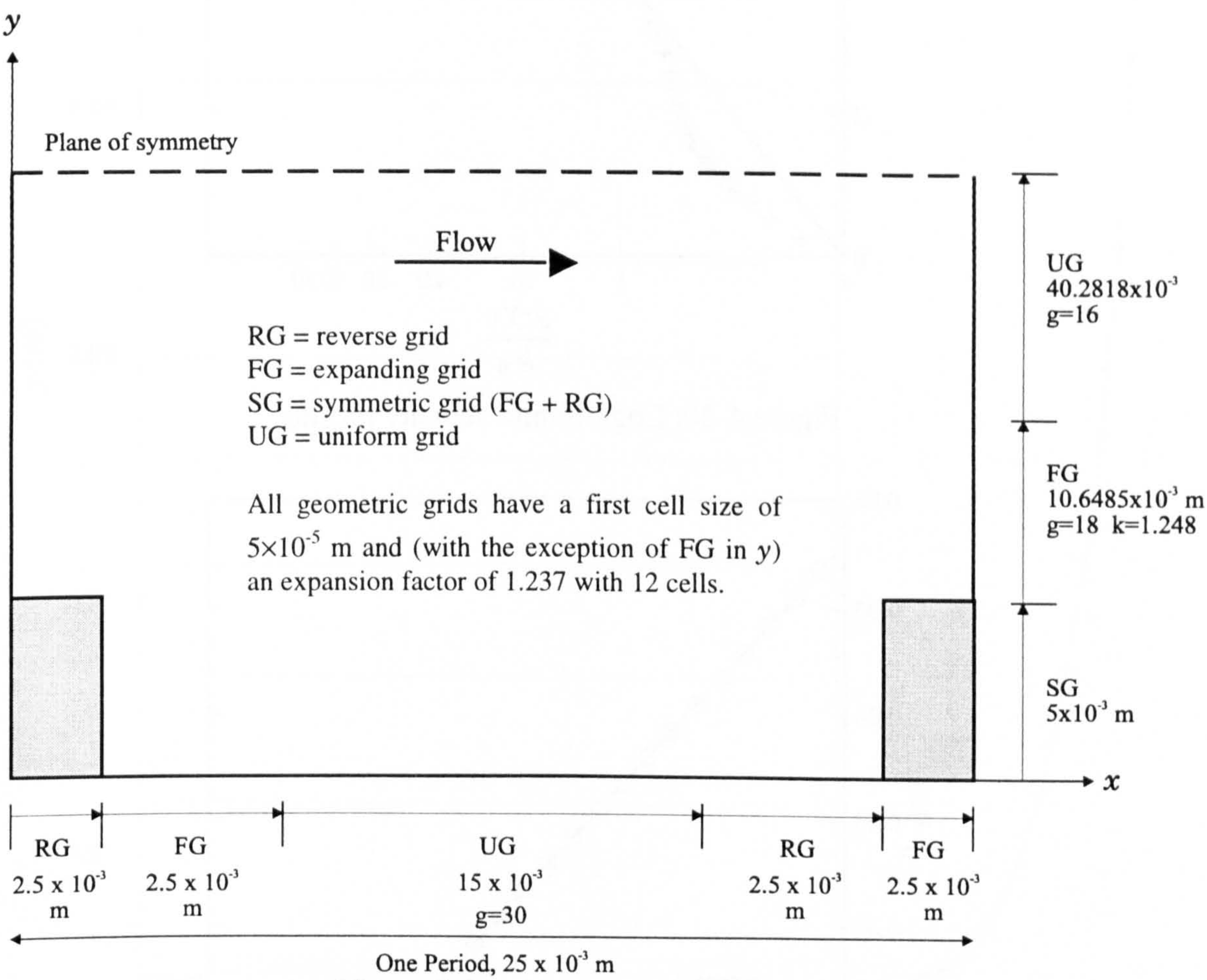


Figure 4-29: Domain and grid structure

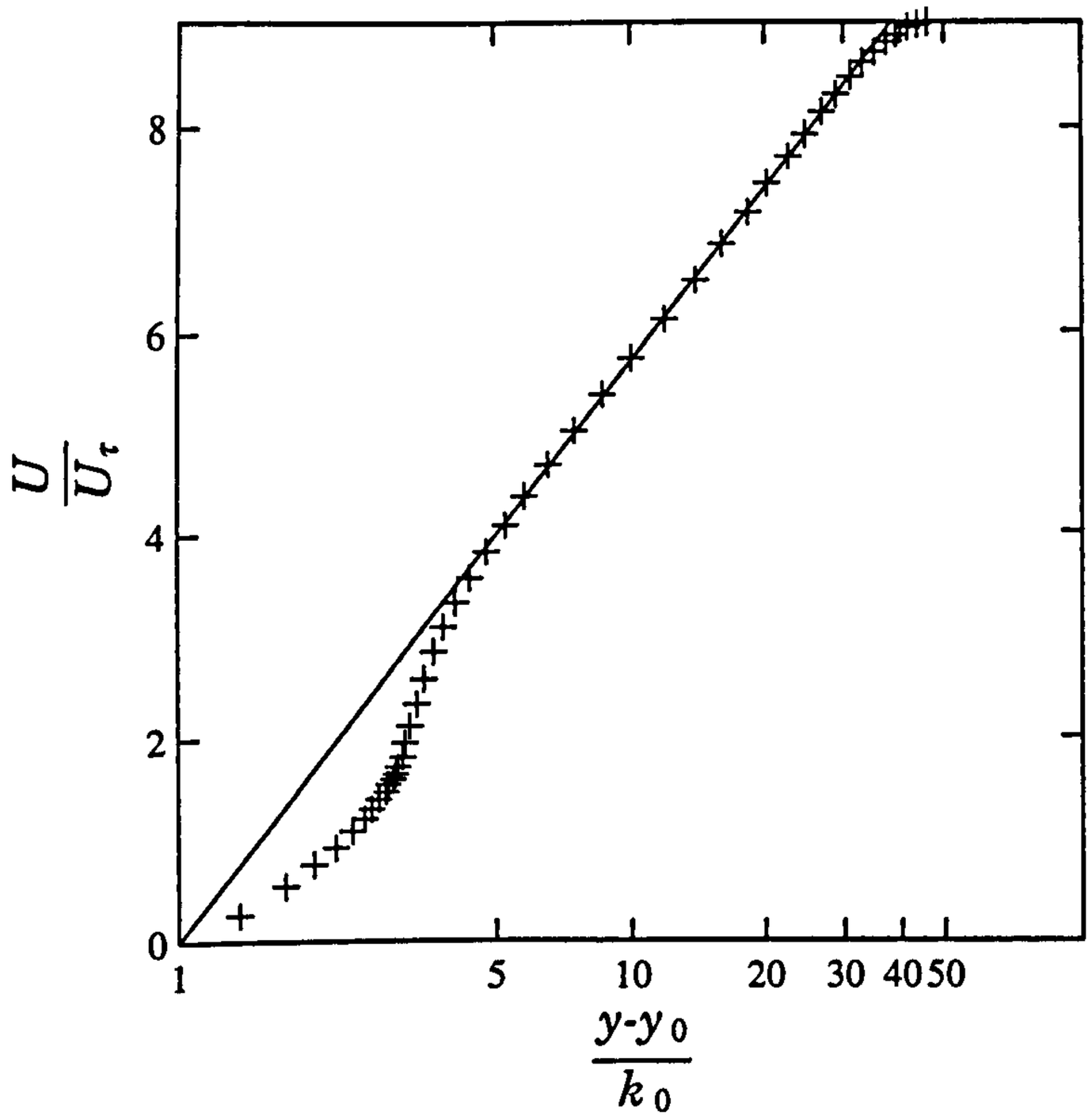


Figure 4-30: Logarithmic velocity profile

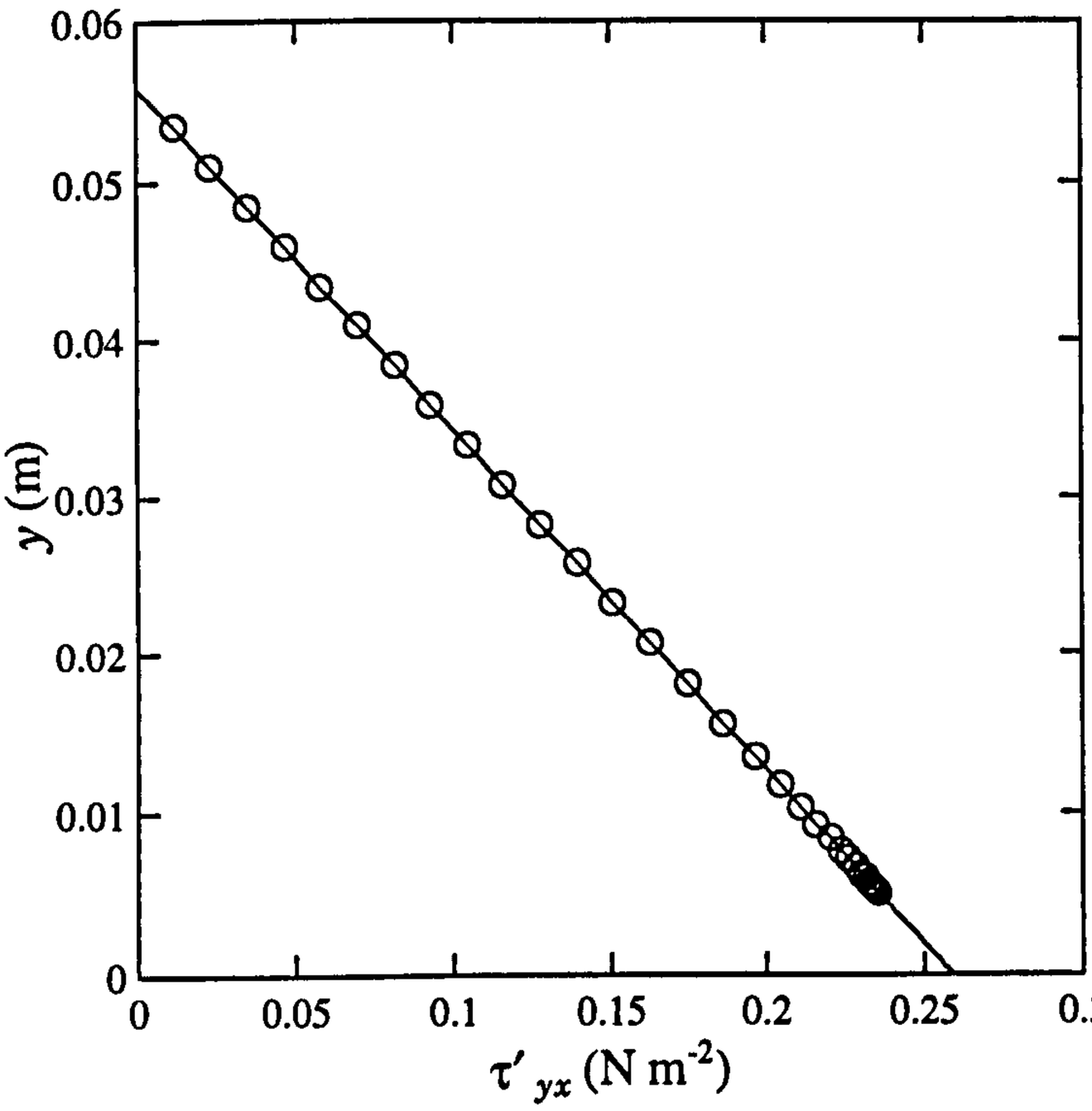


Figure 4-31: Period averaged reduced shear stress distribution

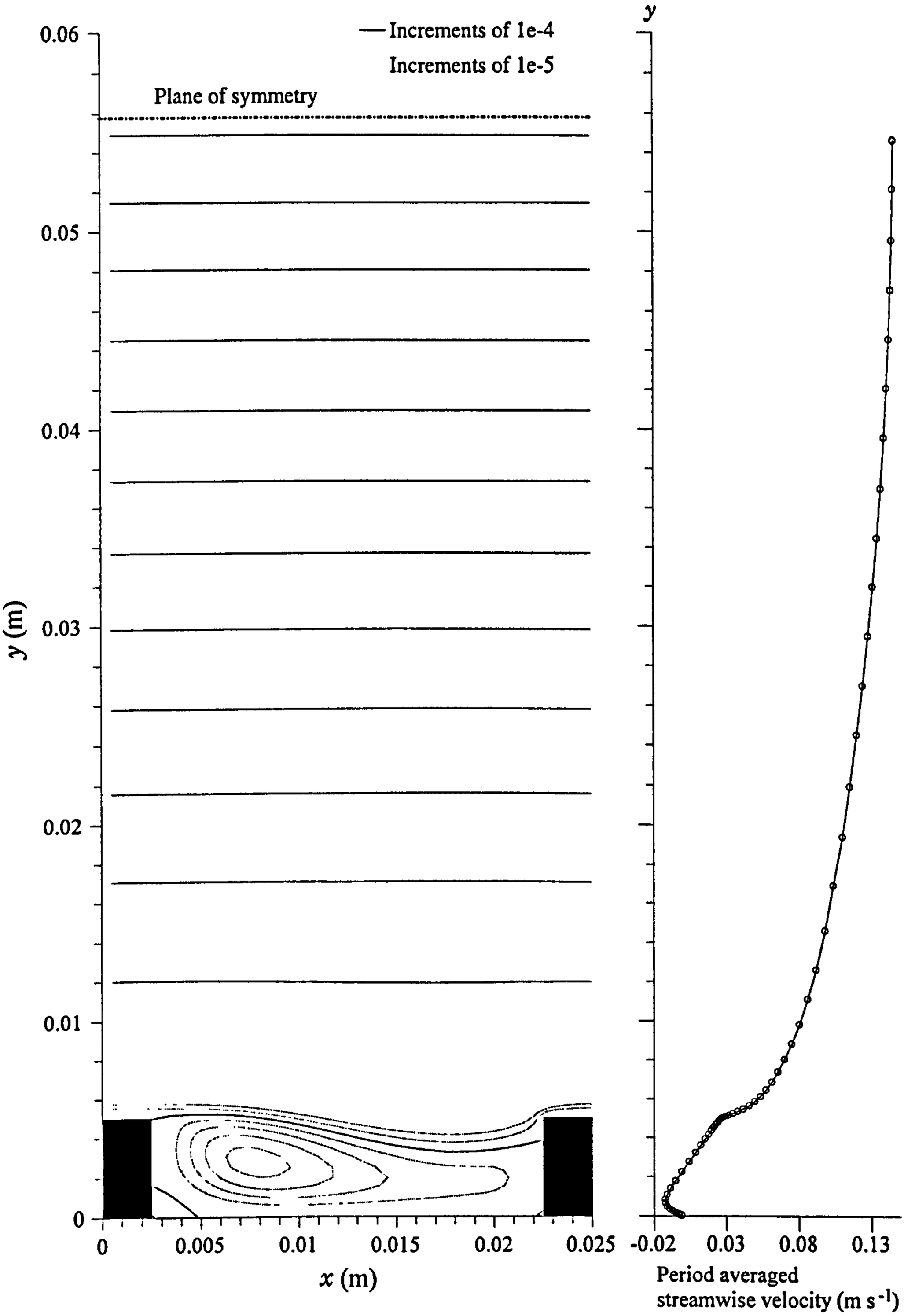


Figure 4-32: Streamlines and period averaged velocity profile

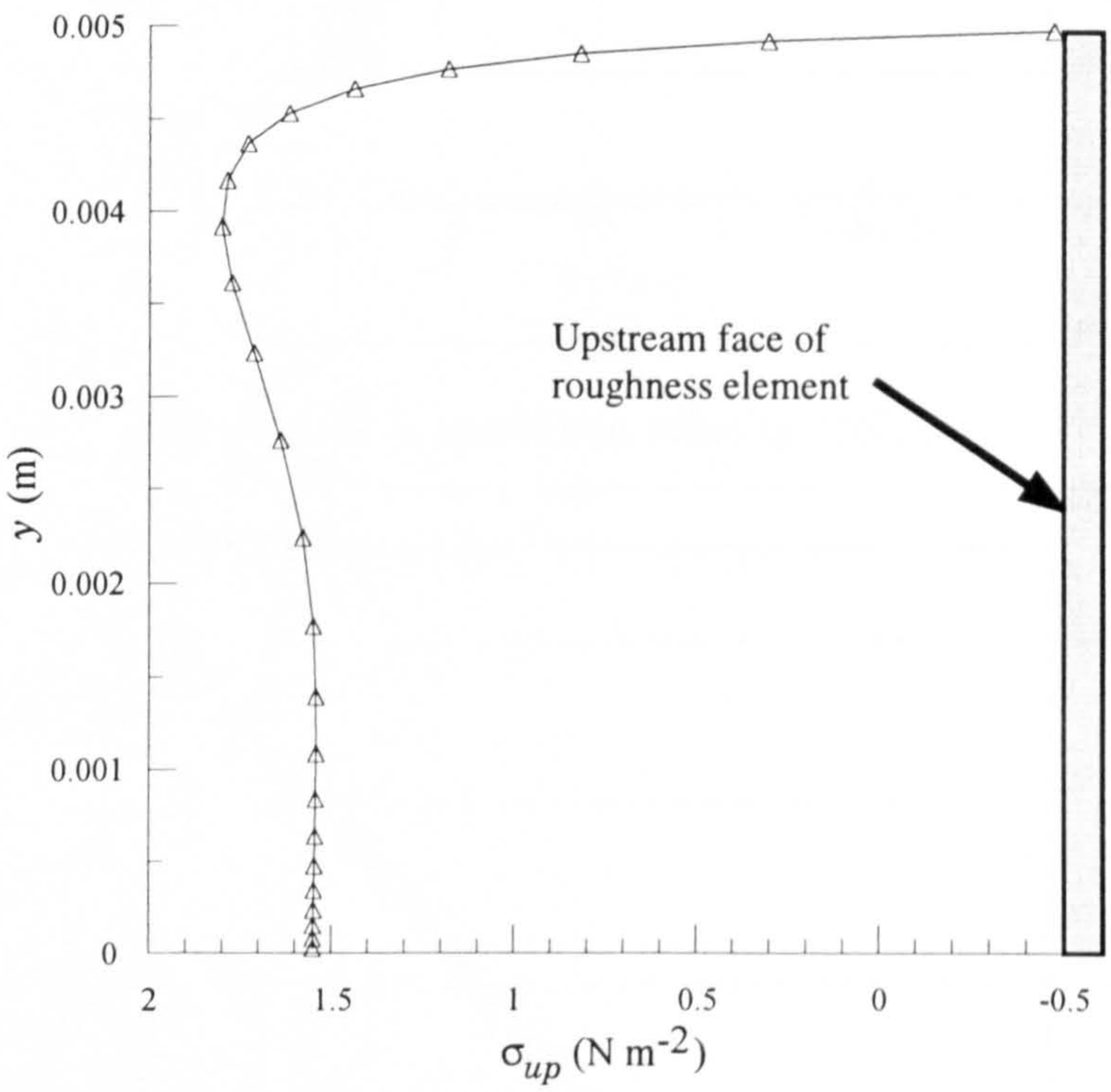


Figure 4-33: Distribution of normal stress on the upstream side of a roughness element showing the effect of impingement

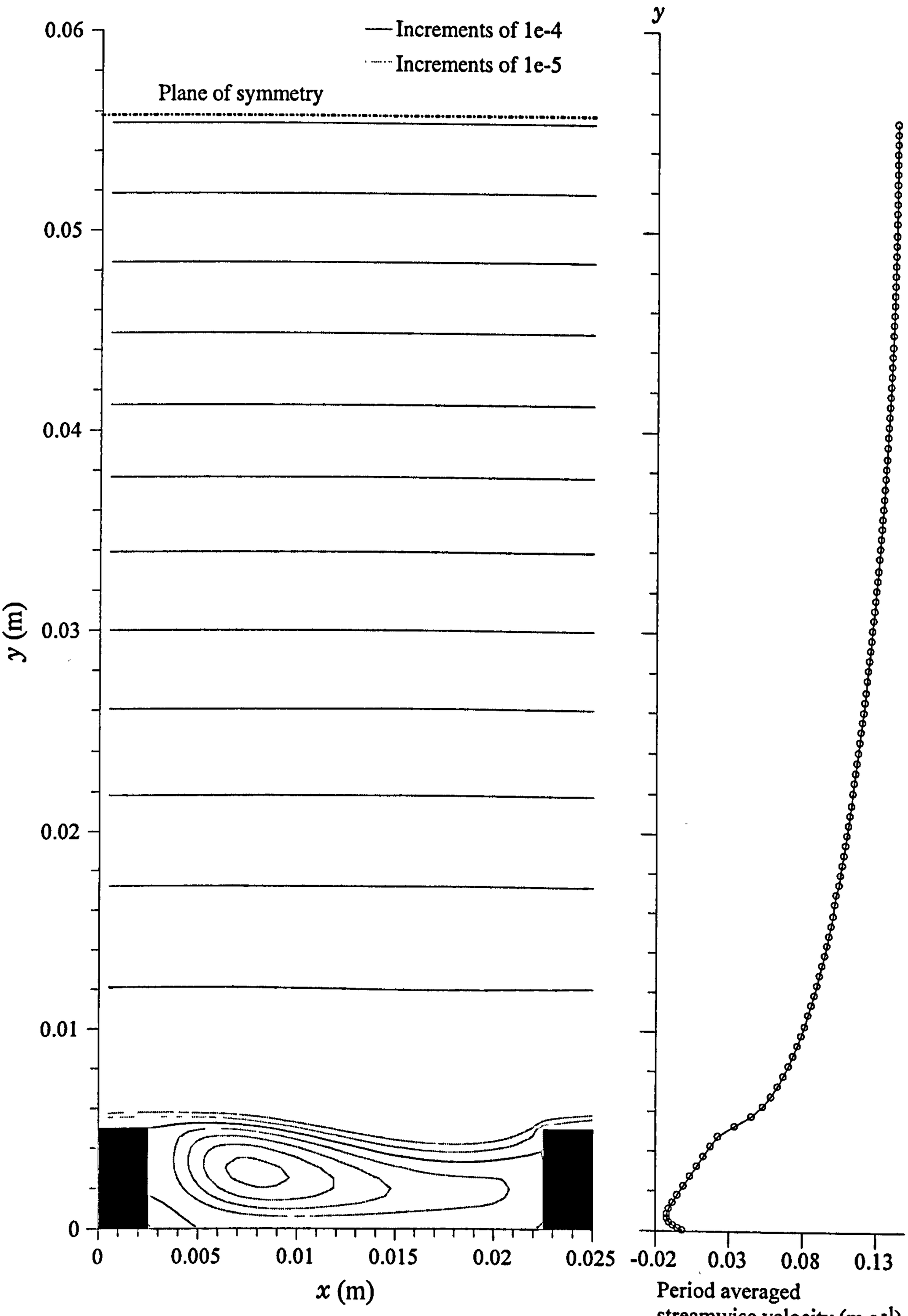


Figure 4-34: Streamlines and period averaged velocity profile for reduced grid (uniform y grid)

Tables relating to Schlichting's experiments

	Inputs			Results of streamwise velocity				
				Schlichting's results		Computational results		
Plate number	Period length (l) [m]	Roughness height (k_h) [m]	$\frac{dP}{dx}$ [kg s^{-2} m^{-2}]	$\hat{U}$ [m s^{-1}]	$y_{\hat{U}}$ [m]	$\hat{U}$ [m s^{-1}]	$y_{\hat{U}}$ [m]	$\overline{U}$ [m s^{-1}]
XX 1	0.06	3.20E-03	3333.988	3.16	3.25E-02	3.135	3.22E-02	2.467
XX 2	0.06	3.20E-03	4991.000	3.88	3.17E-02	3.849	3.22E-02	3.024
XX 3	0.06	3.20E-03	8634.430	5.15	3.25E-02	5.041	3.32E-02	3.953
XX 4	0.06	3.20E-03	10900.344	5.78	3.23E-02	5.660	3.32E-02	4.432
XX 5	0.06	3.20E-03	14054.656	6.58	3.22E-02	6.423	3.32E-02	5.025
XX 5 (G2)	0.06	3.20E-03	14054.656	6.58	3.22E-02	6.469	3.29E-02	5.056
XXI 1	0.04	3.10E-03	4491.900	3.13	3.31E-02	3.482	3.320E-02	2.688
XXI 2	0.04	3.10E-03	6468.336	3.88	3.35E-02	4.163	3.320E-02	3.207
XXI 3	0.04	3.10E-03	9732.450	4.88	3.39E-02	5.111	3.320E-02	3.932
XXI 4	0.04	3.10E-03	12786.942	5.66	3.34E-02	5.847	3.320E-02	4.491
XXI 5	0.04	3.10E-03	20183.604	7.22	3.29E-02	7.349	3.320E-02	5.635
XXI 5 (G2)	0.04	3.10E-03	20183.604	7.22	3.29E-02	7.375	3.351E-02	5.649
XXII 1	0.02	3.00E-03	3912.944	3.15	3.28E-02	3.152	3.320E-02	2.385
XXII 2	0.02	3.00E-03	6039.110	3.94	3.30E-02	3.911	3.320E-02	2.953
XXII 3	0.02	3.00E-03	9183.440	4.84	3.38E-02	4.806	3.320E-02	3.622
XXII 4	0.02	3.00E-03	12677.140	5.67	3.32E-02	5.661	3.320E-02	4.265
XXII 5	0.02	3.00E-03	17368.680	6.66	3.33E-02	6.610	3.320E-02	4.973
XXII 6	0.02	3.00E-03	21960.400	7.49	3.33E-02	7.433	3.420E-02	5.588
XXII 6 (G2)	0.02	3.00E-03	21960.400	7.49	3.33E-02	7.446	3.414E-02	5.589

Table 4-1a: Results of Schlichting's experiments

Notes:

duct height (h)	0.0397	[m]
density (ρ)	998.2	[kg m^{-3}]
kinematic viscosity (ν)	1.17×10^{-6}	[m^2 s^{-1}]
friction factor (f)	$8\tau_{g.m.l.} / \rho\hat{U}^2$	

G2 = grid 2 (grid independence test)

Subscripts:

s = smooth wall (with the exception of k_s)

r = rough wall

g.m.l. = geometric mean level

e.s.m. = equating streamwise force moments

1 = rough wall shear stress calculated from the pressure gradient (or method 1 (Section 3.4.1) in obtaining k_s)

2 = rough wall shear stress calculated from the velocity profile (or method 2 (Section 3.4.1) in obtaining k_s)

	Results of wall shear									
	Schlichting's results						Computational results			
Plate number	τ_s [N m^2]	f_s	$\tau_{r,1}$ [N m^2]	$f_{r,1}$	$\tau_{r,2}$ [N m^2]	$f_{r,2}$	τ_s [N m^{-2}]	f_s	τ_r [N m^{-2}]	f_r
XX 1	-11.643	9.34E-03	120.716	9.69E-02	96.078	7.71E-02	-24.939	2.03E-02	107.301	8.75E-02
XX 2	-14.857	7.91E-03	183.285	9.76E-02	143.758	7.65E-02	-36.310	1.96E-02	160.630	8.69E-02
XX 3	-19.286	5.83E-03	323.501	9.78E-02	252.269	7.62E-02	-60.099	1.90E-02	286.525	9.04E-02
XX 4	-40.328	9.67E-03	392.415	9.41E-02	332.335	7.97E-02	-74.451	1.86E-02	361.717	9.05E-02
XX 5	-46.572	8.62E-03	511.398	9.47E-02	427.253	7.91E-02	-94.687	1.84E-02	466.391	9.06E-02
XX 5 (G2)	-46.572	8.62E-03	511.398	9.47E-02	427.253	7.91E-02	-93.029	1.78E-02	461.998	8.85E-02
XXI 1	-12.746	1.04E-02	165.582	1.35E-01	135.706	1.11E-01	-30.901	2.04E-02	149.027	9.85E-02
XXI 2	-17.657	9.40E-03	239.136	1.27E-01	222.896	1.19E-01	-43.182	2.00E-02	214.598	9.93E-02
XXI 3	-24.292	8.18E-03	362.086	1.22E-01	371.092	1.25E-01	-63.197	1.94E-02	322.891	9.91E-02
XXI 4	-31.983	8.00E-03	475.658	1.19E-01	511.303	1.28E-01	-81.253	1.90E-02	424.229	9.94E-02
XXI 5	-80.511	1.24E-02	720.778	1.11E-01	890.000	1.37E-01	-124.694	1.85E-02	669.627	9.94E-02
XXI 5 (G2)	-80.511	1.24E-02	720.778	1.11E-01	890.000	1.37E-01	-121.589	1.79E-02	675.933	9.96E-02
XXII 1	-9.587	7.74E-03	145.757	1.18E-01	153.289	1.24E-01	-26.155	2.11E-02	129.734	1.05E-01
XXII 2	-15.102	7.80E-03	224.651	1.16E-01	243.202	1.26E-01	-38.956	2.04E-02	200.227	1.05E-01
XXII 3	-20.699	7.08E-03	343.884	1.18E-01	351.082	1.20E-01	-57.225	1.99E-02	304.477	1.06E-01
XXII 4	-25.874	6.45E-03	477.408	1.19E-01	473.083	1.18E-01	-77.554	1.94E-02	420.310	1.05E-01
XXII 5	-38.739	7.00E-03	650.797	1.18E-01	685.264	1.24E-01	-104.239	1.91E-02	575.858	1.06E-01
XXII 6	-57.496	8.21E-03	814.332	1.16E-01	875.984	1.25E-01	-129.500	1.88E-02	750.057	1.09E-01
XXII 6 (G2)	-57.496	8.21E-03	814.332	1.16E-01	875.984	1.25E-01	-125.872	1.82E-02	748.685	1.08E-01

Table 4-1b

	Force balance in numerical model						
Plate number	Driving force [N]	Shear force on top wall [N]	Shear force on bottom wall [N]	Shear force on block [N]	Pressure force on block [N]	Total resistive force [N]	% error in forces
XX 1	7.938	-1.499	0.073	-2.47E-03	-6.514	-7.939	-0.010
XX 2	11.884	-2.179	0.092	-2.92E-03	-9.801	-11.891	-0.058
XX 3	20.559	-3.606	0.105	-3.68E-03	-17.059	-20.564	-0.022
XX 4	25.954	-4.467	0.103	-4.08E-03	-21.605	-25.972	-0.070
XX 5	33.465	-5.681	0.110	-4.48E-03	-27.908	-33.484	-0.057
XX 5 (G2)	33.465	-5.582	0.153	-1.81E-03	-28.037	-33.468	-0.009
XXI 1	7.129	-1.236	0.140	-1.80E-03	-6.035	-7.133	-0.061
XXI 2	10.266	-1.727	0.177	-2.12E-03	-8.708	-10.261	0.050
XXI 3	15.446	-2.528	0.234	-2.59E-03	-13.139	-15.435	0.069
XXI 4	20.294	-3.250	0.271	-2.92E-03	-17.292	-20.275	0.094
XXI 5	32.033	-4.988	0.356	-3.68E-03	-27.368	-32.003	0.094
XXI 5 (G2)	32.033	-4.864	0.382	-3.10E-03	-27.554	-32.039	-0.018
XXII 1	3.103	-0.523	0.099	-1.58E-03	-2.681	-3.106	-0.095
XXII 2	4.790	-0.779	0.137	-1.97E-03	-4.142	-4.785	0.086
XXII 3	7.283	-1.145	0.183	-2.45E-03	-6.314	-7.278	0.075
XXII 4	10.054	-1.551	0.234	-2.86E-03	-8.732	-10.052	0.020
XXII 5	13.775	-2.085	0.286	-3.36E-03	-11.965	-13.767	0.060
XXII 6	17.417	-2.590	0.342	-3.78E-03	-15.147	-17.399	0.103
XXII 6 (G2)	17.417	-2.517	0.346	-4.92E-03	-15.237	17.413	0.020

Table 4-1c

	Datum positions		Results of equivalent sand roughness, k_s								
			Schlichting	numerical: y_0 = g.m.l.				numerical: y_0 = e.s.m.			
Plate number	y_0(g.m.l) [m]	y_0(e.s.m.) [m]	k_s [m]	cell range	gradient	k_s (1)	k_s (2)	cell range	gradient	k_s (1)	k_s (2)
XX 1	1.60E-05	1.65E-03	1.81E-02	28 - 44	6.584	1.05E-02	2.03E-02	24-36	5.444	2.05E-02	1.53E-02
XX 2	1.60E-05	1.62E-03	1.81E-02	28 - 44	6.626	1.02E-02	2.03E-02	24-37	5.548	1.90E-02	1.57E-02
XX 3	1.60E-05	1.63E-03	1.81E-02	28 - 44	6.539	1.17E-02	2.18E-02	24-38	5.500	2.06E-02	1.63E-02
XX 4	1.60E-05	1.61E-03	1.81E-02	28 - 44	6.551	1.17E-02	2.21E-02	24-39	5.529	2.06E-02	1.67E-02
XX 5	1.60E-05	1.61E-03	1.81E-02	28 - 44	6.566	1.17E-02	2.22E-02	24-40	5.539	2.06E-02	1.69E-02
XX 5 (G2)	1.60E-05	1.63E-03	1.81E-02	45 - 66	6.582	1.17E-02	2.23E-02	43-57	5.503	2.19E-02	1.75E-02
XXI 1	2.40E-05	1.60E-03	3.70E-02	26 - 45	6.566	1.45E-02	2.76E-02	21-34	5.121	3.73E-02	2.01E-02
XXI 2	2.40E-05	1.65E-03	3.70E-02	26 - 45	6.567	1.48E-02	2.82E-02	21-35	5.018	4.18E-02	2.03E-02
XXI 3	2.40E-05	1.65E-03	3.70E-02	26 - 45	6.582	1.47E-02	2.84E-02	21-36	5.020	4.23E-02	2.05E-02
XXI 4	2.40E-05	1.66E-03	3.70E-02	26 - 45	6.595	1.48E-02	2.88E-02	21-37	5.004	4.37E-02	2.09E-02
XXI 5	2.40E-05	1.65E-03	3.70E-02	26 - 45	6.627	1.46E-02	2.91E-02	21-38	5.059	4.24E-02	2.14E-02
XXI 5 (G2)	2.40E-05	1.60E-03	3.70E-02	41 - 66	6.620	1.46E-02	2.93E-02	36-54	5.209	3.69E-02	2.19E-02
XXII 1	4.80E-05	1.59E-03	3.56E-02	24 - 43	6.817	1.38E-02	3.28E-02	19-33	5.013	4.99E-02	2.39E-02
XXII 2	4.80E-05	1.67E-03	3.56E-02	24 - 43	6.815	1.40E-02	3.33E-02	19-34	4.845	5.86E-02	2.36E-02
XXII 3	4.80E-05	1.66E-03	3.56E-02	24 - 43	6.803	1.45E-02	3.40E-02	19-35	4.813	6.22E-02	2.43E-02
XXII 4	4.80E-05	1.63E-03	3.56E-02	24 - 43	6.825	1.42E-02	3.39E-02	19-36	4.877	5.92E-02	2.47E-02
XXII 5	4.80E-05	1.64E-03	3.56E-02	24 - 43	6.824	1.44E-02	3.44E-02	19-37	4.833	6.26E-02	2.49E-02
XXII 6	4.80E-05	1.66E-03	3.56E-02	24 - 43	6.726	1.62E-02	3.56E-02	19-38	4.789	6.54E-02	2.49E-02
XXII 6 (G2)	4.80E-05	1.61E-03	3.56E-02	35 - 64	6.784	1.53E-02	3.66E-02	33-50	4.808	6.50E-02	2.51E-02

Table 4-1d

depth	cells (NX x NY)	$Y^{+}_{\text{1st cell}}$	$\frac{dP}{dx}$	U_τ	A	B
7.00E-03	3×21	2.456	11768	0.2873	4.4589	5.5883
7.13E-03	3×43	1.212	11249	0.2835	4.9051	5.5801
7.13E-03	3×91	0.904	11113	0.2818	5.0564	5.5411
			Nikuradse:		5.5	5.75
			Jayatilleke:		5.25	5.62

Table 4-2: Results of simulations performed on the smooth wall region only of Schlichting's duct. A and B refer to the coefficients in the law of the wall for smooth walls and are compared with the coefficients suggested by Jayatilleke (1969) and Nikuradse (1933)

	Grass *et al* original expts	Input shear stress method	Input flow rate method
q ($m^2 s^{-1}$)	7.85E-03	6.14E-03	7.86E-03
Re	8091	6335	8108
$\hat{U}$ ($m s^{-1}$)	0.18	0.14	0.19
dP/dx ($Pa m^{-1}$)	/	4.63	7.37
τ_w ($N m^{-2}$)	0.26	0.26	0.41
k_0 (m)	9.60E-04	1.20E-03	1.26E-03
k_s (m)	0.0288	0.0359	0.0378

Table 4-3: Comparison of experimental and computational results

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

5. DEPTH SCALE ROUGHNESS CALCULATIONS

5.1 Introduction

The roughness problem would be solved if, for any given roughness shape, height and distribution, the rough wall resistance could be reduced by some dynamically consistent method to an equivalent wall shear stress located on a plane wall. In this case the rough wall resistance could be accounted for by three dynamically significant length scales. The first (y_0) would determine the position of the equivalent plane wall, the second would be the flow depth (h) and the third (k_s) would specify the equivalent wall shear stress. This concept was employed by Schlichting (1936) in which a sand roughness value (k_s) was applied to an equivalent plane wall by comparison with the sand roughness experiments performed by Nikuradse (1933).

If the k_s length scale determined by this means was small compared with the flow depth, it would be reasonable to suppose that the k_s value would be invariant with flow depth, an implicit assumption in the work of Schlichting and more generally in the fully rough resistance equation (Equation 3-50). If k_s was significant in scale compared with the depth of flow, then k_s would no longer be invariant with flow depth and the sand roughness concept would normally be invalid for this flow regime.

However, if every rough wall flow could be reduced to an equivalent plane wall, there is the prospect that the corresponding k_s value might vary with flow depth in a universal way. A possible relationship between k_s and flow depth that might be universally applicable is

$$\frac{k_s}{k_{s\infty}} = \mathrm{F}\left(\frac{k_s}{h - y_0}\right) \qquad \textit{Equation 5-1}$$

where $k_{s\infty}$ is the asymptotic value of k_s at very large flow depths and F denotes a functional relationship in which $(h-y_0)$ is the depth of flow in the equivalent plane walled conduit.

It was decided to test out this hypothesis using CFD over various two-dimensional roughness configurations initially, with the intention of extending the work to three-dimensional roughness distributions. However, given the difficulties experienced in successfully implementing the two-dimensional application over the wide range of flow depths required, it became impossible to simulate three-dimensional rough walled flow without unacceptably extending the project period.

5.2 Summary of Simulations Performed

The complete set of successful tests performed and the input parameters are presented in Table 5-1a. All simulations employed a two-layer turbulence model (Section 2.2.2) and the PHOENICS controlled method of cyclic boundary conditions. The source of momentum flux was adjusted so that the computed flow rate was within 5% of the required value. The criterion for convergence was that the percentage error between resistive forces on the boundary and the driving force provided by the source of momentum flux should be less than 0.1%. A sample Q1 input file and the GROUND coding used is presented in Appendix 11.

5.3 Block Roughness Simulations

Tests were initially performed to simulate turbulent flow over two-dimensional roughness elements of 50 mm square cross-section, periodically spaced in the primary flow direction. Period lengths of 1 m, 0.5 m and 0.1 m were used to provide flow regimes of isolated roughness, wake interference and quasi-smooth flow as defined by Morris (1954). The relative flow depth, defined as h/k_h, was varied from 1.1 to 20 by adjusting the position of a plane of symmetry. A constant Reynolds number (= 462,200), sufficiently high to ensure fully rough turbulent flow, was maintained for all tests. The

domain consisted of two periods which, when compared, confirmed that fully developed periodic flow was established.

At a defined frequency of sweeps[1], the source of momentum flux was adjusted in proportion to the ratio of the required flow rate and the integrated velocity profile (Section 3.2.3). It was found that all simulations required excessively large numbers of sweeps to reach the required flow rate and the stipulated convergence criterion.

For the isolated roughness regime (l = 1 m), labelled as the RCDA roughness configuration, 50,000 to 80,000 sweeps were typically required to achieve convergence at most depths of flow. Significantly more sweeps were required for the higher relative depths of 10.05 and 20.05 due to the larger distances over which turbulent diffusion had to take place in order to develop the correct wall profiles. Indeed the convergence criterion was not quite achieved for the deepest flow examined (relative depth of 20.05), the error between driving forces and resistive forces remaining constant at 0.112%.

The domain and grid structure for these tests are given in Figure 5-1a. Due to the lower wall shear stresses which occurred in the deeper flows, the size of the near wall cell could be increased while remaining within the viscosity affected layer[2]. This allowed the number of cells to be significantly reduced (Figure 5-1b) and was used for the relative depth of 10.05 as a grid independence test. Only one period was modelled to further reduce the computational resources required. The latter grid reduced the time per iteration, however a similar number of sweeps, in the order of 500,000, was again required. The effect on the solution of reducing the grid is shown in Table 5-2.

As the period length of the roughness elements was reduced below 1 m, a converged solution became more difficult to obtain. For the skimming flow regime, the

---

[1] A set of iterations over the domain (Section 2.3.3)

[2] As τ and therefore U_τ decrease, the value of y will increase for a fixed Y^+. Therefore the location of the near wall cell centre can be moved further away from the boundary.

convergence criterion was not achieved for any depth of flow and further iterations had no appreciable effect on the solution. The final percentage error between the driving and resistive forces varied from 0.7% for a relative depth of 1.1 to 200% for a relative depth of 20. Consequently these tests were not used for analysis.

5.3.1 Convergence Problems

Due to the possibility of computer operational interruptions, a complete simulation was designed to consist of several consecutive stages of approximately 10,000 sweeps each. For the relative depth of 10.05 each set of 10,000 sweeps required approximately 50 hours CPU time, relating to 150 hours real time when performed on an Axil SPARC 220 running SOLARIS. At the end of each stage, the history of spot values and of residuals in the FVE was examined to enable relaxation factors to be altered. Due to the large range of cell sizes employed (10^{-5} to 5×10^{-2} m) it was found difficult to estimate the optimum value of false time step relaxation required and large adjustments were required to have any effect.

One possible explanation for the long convergence times and the difficulty of obtaining full convergence for the shorter period lengths relates to the layer of very thin cells on the plane $y = k_h$. These large aspect ratio cells were required to model the shear over the top of the roughness elements. However it is possible that in-between the roughness elements they may have acted as a barrier to flow development. This would have been particularly significant for flows which did not have a large component of velocity across this layer of cells, such as the skimming flow regimes. Figure 5-2 shows the streamlines obtained for a relative depth of 4.05 and a period length of 0.5 m. The discrepancy between the driving force and resistive force for this case was 38%. The horizontal streamlines immediately above the roughness elements, typical of skimming flows, show that there was negligible flow across this layer of thin cells and thus no convective transport of momentum. Momentum exchange occurred only through the slower processes of turbulent and viscous diffusion.

In order to overcome this problem investigations were made by:

1. Employing multi-block gridding techniques to enable a fine grid to be located only in the region surrounding a roughness element. As discussed in Appendix 7 no adequate grid could be obtained.

2. Increasing the Reynolds number and therefore the velocity gradient across the layer of thin cells.

3. Using thin strip roughness elements, in which the shear stress on the top of the blocks could be neglected and therefore a much coarser grid could be used in this region.

None of these investigations provided a solution to the problem, which was similar to that encountered in simulating the roughness configuration of Grass *et al* (1991) (Section 4.2). Further studies involved increasing the number of iterations within the linear equation solver, which again showed no appreciable improvement. The use of alternative solvers, such as the conjugate linear equation solver, was not investigated.

Another reason for the excessive number of iterations, especially as the flow depth was increased, was the computational time required to fully develop the flow. The physically long duct length required for development of deeper flows necessitated more computational sweeps of the flow domain when periodic boundary conditions were employed.

Monitoring of the adjustments made to the source of momentum flux per unit volume showed oscillations which were also responsible for the large number of sweeps required. These oscillations were partially controlled by decreasing the sweep frequency at which the source of momentum flux was adjusted (an effect similar to relaxation). A new procedure for updating the source of momentum flux per unit volume was also developed (Section 3.2.3) which was employed for subsequent tests.

5.4 Thin Strip Roughness Simulations

Given the difficulty in obtaining converged solutions for block roughness flows for period lengths lower than 1 m, the roughness configurations of Schlichting examined in Section 4.1 were used in the knowledge that a converged solution was obtainable for a range of period lengths. The use of thin strip roughness elements also allowed a lower density of grid cells to be employed above the plane $y = k_h$. In these simulations of Schlichting's roughness geometries, only the rough wall region was modelled and the depth of flow was varied by altering the location of a plane of symmetry boundary condition. Relative depths of between 1.5 and 40 were examined. The Reynolds number was kept constant at approximately the same value as that used in Schlichting's original experiment ($\mathbf{Re} = 2 \times 10^5$). Period lengths of 6 cm and 4 cm were simulated, corresponding to Schlichting's roughness plates XX and XXI. For the current investigation, these were labelled RSA and RSB respectively.

The grid structures and the domain consisting of one period, are given in Figure 5-3. For the higher relative flow depths ($h/k_h > 7$) the density of cells in the vicinity of the walls was again reduced due to the low values of wall shear stress. A grid independence test was conducted for the relative depth of 7 to confirm that this did not affect the solution.

For these tests, the pressure gradient was adjusted using Equation 3-8. It was found that the number of sweeps required to achieve convergence was of the same order of magnitude as the RCDA tests but, due to the lower number of cells in the domain, run times were reduced to approximately 80 hours to achieve convergence for a specified pressure gradient. Longer run times were again required for the higher depths of flow.

5.5 Simulations Employing a Wall Function

For the purposes of comparison, simulations were also performed using the standard k-ε model with a wall function to determine the coefficients a and b in the fully rough turbulent flow resistance equation (Equation 3-50).

The two-dimensional duct consisted of a plane wall on which an equivalent sand roughness of 0.001 m was imposed and specified within the wall function. The depth of flow was varied by defining a plane of symmetry boundary condition at a distance of 0.1 to 1.0 m from the wall. In all 10 simulations the Reynolds number was maintained at 7×10^6 to ensure a fully rough flow regime ($\mathbf{Re_r} > 100$), by adjusting the source of momentum flux per unit volume according to Equation 3-7. A grid of 100 cells was employed in the cross-stream (y) direction, expanding from the wall according to a power law of factor 1.5. Although flow over a plane wall is parabolic, an elliptic equation solution procedure was employed using the PHOENICS controlled cyclic boundary conditions, to be consistent with other tests reported in this study. The streamwise (x) period dimension was chosen to be identical to the depth of flow (h) and uniformly distributed with 50 cells.

An estimate of f, and therefore also of τ_w, was obtained by using Equation 3-50 with the coefficients quoted by Nikuradse ($a = 1.74$, $b = 2.0$). The initial pressure gradient was then estimated from

$$\frac{dP}{dx} = \frac{\tau_w}{h} \qquad \textit{Equation 5-2}$$

The standard velocity profile of Jayatilleke (1969) (Equation 2-27) was used to establish a realistic starting condition for the velocity field.

The solution was deemed to have converged when the difference between the shear stress calculated from the velocity profile and the shear stress calculated from

$$\tau_w = \frac{dP}{dx} \times h \qquad \textit{Equation 5-3}$$

was less than 0.2%. A more stringent convergence criterion was found to require unacceptably high numbers of sweeps for negligible changes in the solution.

5.5.1 Correction to the Computational Code.

During these tests an error was found within the PHOENICS code in the calculation of the non-dimensional skin friction factor. The details of the error and the correction applied are given in Appendix 4.

5.5.2 Convergence

Despite the very simple nature of this simulation, large numbers of sweeps (≈ 50,000) were required to achieve convergence. This was due to two factors:

1. Oscillations in the adjustment of pressure gradient. This was controlled by the number of sweeps (typically 3,000) between each adjustment (Section 3.2.3). To achieve final convergence, the adjustment process was switched off.

2. Cyclic boundary conditions. It was concluded that the cyclic boundary conditions required a large number of sweeps to achieve convergence as a consequence of flow development.

5.5.3 Evaluation of Coefficients in the Resistance Equation

Table 5-2 shows the variation of friction factor with depth of flow. The wall shear stress used in the calculation of f was obtained from

$$\tau_w = \frac{dP}{dx} \times h \qquad \textit{Equation 5-4}$$

A plot of $1/\sqrt{f}$ against $\log \frac{2R_h}{k_s}$ (Figure 5-4) gave a near perfect straight line from which the coefficients $a = 1.888$ and $b = 1.985$ were determined by least squares regression.

5.5.4 Universal Velocity Profile

The opportunity was also taken to find the coefficients in the universal velocity profile law (Equation 2-30) predicted by the k-ε model by plotting $\frac{U}{U_\tau}$ against $\log \frac{y}{k_s}$ for each flow depth. The logarithmic region was taken to apply from the second cell away from the wall to $y/h = 0.15$ as suggested by Hama (1954), although even closer to the plane of symmetry the data was found to fit the logarithmic law. This is shown in Figure 5-5 for the depth of 0.5 m as an example. Least squares analysis gave the intercept A and the

gradient B, as shown in Table 5-3. It can be seen that the gradient was nearly constant for all depths, but was significantly higher than the gradients obtained by Nikuradse (1933) and Jayatilleke (1969). The coefficient A was found to decrease by 4.8% over the depths tested, the majority of the change occurring at the lower depths where the roughness might not have been small scale (Figure 5-6). The average value of A was however remarkably close to the value given by Nikuradse, with a difference of just 0.18%. The value used within the wall function (that of Jayatilleke) was 2.29% lower.

5.6 Discussion of Rib Roughness Simulation Results

The results of the numerical simulation of flow over the rib roughness configurations investigated using the two-layer model are presented in Tables 5-1, 5-2 and 5-3, and in Figures 5-7 to 5-22.

5.6.1 Fully Developed Periodic Flow

The flow periodicity is illustrated, by the RCDA roughness of a relative depth of 7.05 as an example, in Figures 5.7 and 5.8 which show transverse profiles of pressure and streamwise velocity one period apart.

5.6.2 Convergence

The convergence criterion was achieved for all of the cases presented with the exception of two high depth flows, RCDA relative depth of 20.05 and RSB relative depth of 40, for which the error between resistive and driving forces was 0.177% and 0.332% respectively (Table 5-1c). For all cases the distribution of the period averaged reduced shear stress was linear. The line of action of the equivalent bed shear stress, calculated by equating streamwise force moments ($y_{0\ e.s.m.}$), together with the value of the equivalent bed shear stress also fell on this line. This is shown in Figure 5-9 for the example RCDA, relative depth 7.05. The reciprocal of the gradient $\left(\frac{\partial \tau'_{yx}}{\partial y}\right)$ was within

0.1% of the pressure gradient for the majority of cases, the maximum error being 0.25% (Table 5-1c).

Monitoring of spot values and of the total residuals in the FVE for the variables P1, U1, V1, KE and EP during the solution procedure indicated that further iterations would have had negligible effect.

Additionally, a check on the conservation of mass was made by comparing the flow rate obtained by integrating the velocity profile above a roughness element with the integrated period averaged velocity distribution. In all cases the error was below 10^{-4}%.

5.6.3 Grid Independence

For the roughness configurations RSA and RSB, a more coarse grid was used for the higher depths of flow ($h/k_h > 7$). To confirm that the solution was independent of this change in the grid, the test RSA7 was simulated using both the fine and the coarse grid. The results are compared in Table 5-4 and show that, for the same pressure gradient, the discrepancy in flow rate is only 0.6%. The error in the position of the datum level $y_{0e.s.m.}$, the friction factor and the k_s value calculated from the resistance equation[3] (Equation 3-50) are all in the order of 1%. It was concluded that the solution was suitably grid independent and that the coarser grid could be safely used for the deeper flows.

An attempt to use a coarser grid was also made for the roughness configuration RCDA, but a grid independence test (Table 5-4) on the relative depth of 10.05 showed significant errors. Due to time constraints further grid independence tests were not made and the fine grid was used for all depths of flow.

---

[3] The coefficients used within the resistance equation were: $a = 1.49$ and $b = 1.97$ (Section 5.8.2). The hydraulic radius was calculated from $R_h = h - y_0$ where y_0 was obtained from equating streamwise force moments.

For all roughness configurations and all flow depths, checks were made to ensure that the first wall cell was located inside the laminar sublayer. For the vast majority of cases the Y^+ value at the centre of the cell adjacent to the bed was below 5. However for the extremely shallow depths, high velocity gradients resulted in high values of Y^+ in some regions of the bed. The maximum Y^+ values obtained were 25.1 and 17.1 for cases RCDA1.1 and RSA1.5 respectively. On the horizontal surface of the roughness elements large values of Y^+ were obtained, particularly on the leading edge due to the large shear in this region. Y^+ values of up to 77 were observed. To model the flow accurately in this region, without the aid of multi-block grid techniques, would have required an extremely large number of cells. The computational resources for such a grid were not available in this study. However, the rough wall resistance was predominantly form drag and therefore little affected by the error in skin friction.

5.6.4 Reynolds Number Dependency

Table 5-5 shows the effect of increasing Reynolds number upon the friction factor, the position of the datum plane $y_{0\ e.s.m.}$ and the k_s value obtained from the resistance equation. The case RSA, relative depth of 40 was used.

To increase the Reynolds number by 72.8% the pressure gradient was required to be increased by 208.4%. Note that

$$\frac{dP}{dx_{ii}} \approx \frac{dP}{dx_{i}} \times \left(\frac{q_{ii}}{q_{i}}\right)^2 \qquad \textit{Equation 5-5}$$

where the subscripts i and ii refer to the low Reynolds number and high Reynolds number tests.

The literature (e.g. Schlichting 1960) shows that for fully rough flows the friction factor is independent of Reynolds number, though a slight increase (3.2%) was observed in this test. The value of $y_{0\ e.s.m.}$ increased significantly with Reynolds number (19%) and the k_s value increased by 6.4%.

5.6.5 Variation in Streamlines and Velocity Profiles

Plots of streamlines and period averaged velocity profiles are presented in Figure 5-10 for the roughness configuration RSB for all relative depths between 1.5 and 15. For relative depths above 10, the streamlines have a similar appearance. The stream function was calculated within the GROUND subroutine by integrating the streamwise velocity distribution from each cell to the plane $y = 0$.

All the streamlines show a primary region of recirculation downstream of the roughness element, a characteristic of isolated roughness and wake interference flow regimes. The re-attachment length (l_{r1}) is constant with depth ($l_{r1}/k_h = 5.20$) for all but the shallowest of flow depths. Immediately upstream of each roughness element a secondary region of recirculation exists, with the distance (l_{r2}) between the separation point and the element increasing with flow depth from 0.55 to 3.73 block heights. The reason for this variation is not known at present. Within each of these regions of recirculation, an additional small eddy was found located in the lower corner region adjacent to the roughness element. The locations of the separation and re-attachment points were obtained by finding the cells adjacent to the wall in which the local velocity changed direction.

At low relative depths, the velocity above the roughness elements reached extremely high values (of the order of 160 m/s) for the Reynolds numbers employed. This prevented the recirculation region downstream of the roughness element from extending above the height of the roughness element and produced a shorter re-attachment length. Despite the physically unrealistic velocities obtained for these shallow flows, the Reynolds equations employed were still applicable given that the effects of compressibility could be neglected. Schlichting (1960, p9) assumed that compressibility could be neglected if

$$\tfrac{1}{2}M^2 \ll 1 \qquad \textit{Equation 5-6}$$

where M is the Mach number $\left(=U/c\right)$ and c is the speed of sound in water (1498 m/s) for the present application. Thus

$$U \ll \sqrt{2c^2} \qquad \textit{Equation 5-7}$$
$$\ll 2118$$

In all the numerical simulations performed, $U << 2118$ and therefore the effects of compressibility were ignored.

Figure 5-10 shows that as the relative depth increased the streamlines extended above the height of the roughness element and formed a typical vena contracta. The re-attachment length then rapidly assumed a constant value ($l_{r1}/k_h = 5.20$). For $h/k_h > 7$, the velocity profiles above the roughness elements took the shape of a typical rough wall distribution. Within the height of the roughness elements, the period averaged profile resembled a stretched 'S', with a slightly negative value of streamwise velocity adjacent to the boundary.

Streamlines for the other roughness configurations, RSA and RCDA, are shown for the relative depths of 4 in Figures 5-11 and 5-12 and the re-attachment and separation lengths are included in Table 5-1b. For an increase in period length, the roughness configuration RSA showed that the point of re-attachment moved slightly downstream. This was due to the turbulence intensity generated by roughness element having a greater length in which to dissipate before the next roughness element was reached, and was therefore a characteristic of the wake interference flow regime. As shown by Senior and Aroussi (1992), a greater turbulence intensity in the region of a backward facing step leads to a shorter re-attachment length. All the re-attachment lengths observed in this study were comparable with the range given in the literature for flow over a backward facing step (see for example the review by Senior 1992).

5.6.6 Boundary Stresses

The wall shear stress distribution between the roughness elements, plotted as a percentage of the period averaged rough wall shear stress[4], is shown for all depths of flow in Figures 5-13 and 5-14 for the roughness configurations RSA and RSB.

For all tests, the largest absolute shear stress was located in the primary reverse flow region, downstream of the roughness elements. Although the value of local shear stress

---

[4] Including form drag as well as viscous shear.

decreased with an increase in depth (due to the lower velocities), as a percentage of the period averaged bed shear stress, it significantly increased with deeper flows. This is because the form drag on the roughness elements is proportional to the square of the velocity, but the skin friction is proportional to the velocity.

As the period length was reduced (RSB, Figure 5-14), the region of positive shear stress also decreased. Consequently the integrated shear stress produced a resultant force on the fluid in the streamwise direction (Table 5-1c).

The integrated bed shear stress was negligibly small when compared with the driving force within the domain for all cases simulated. The resistance to flow was therefore almost entirely due to the direct stresses on the vertical faces of the roughness element (the shear on the top of the roughness elements being negligible). This is shown in Figure 5-15 for the configuration RSB. A breakdown of the constituent terms of σ' (Equation 3.13) showed that the only terms which had any significant effect were $P+\frac{2}{3}k$. The distribution of direct stress was almost uniform on the downstream face of the roughness element, which supports the Helmholtz hypothesis quoted by Haque and Mahmood (1983) of a constant pressure in the eddy region. On the upstream side of the roughness element the distribution of direct stress was again nearly uniform, falling slightly towards the top of the roughness element.

5.6.7 Roughness Datum Level

The following argument demonstrates that a rough estimate y_0' of the roughness datum level can be obtained from geometric considerations alone. Taking the distribution of direct stress on a roughness element to be approximately uniform, and the shear on all horizontal surfaces to be negligible, then the streamwise moments on the rough wall can be estimated by (Section 3.5)

$$RM_b' = \frac{dP}{dx}(l\,h - k_l\,k_h)\left(h - \tfrac{1}{2}k_h\right) \qquad \textit{Equation 5-8}$$

The estimated datum position (y_0') is then calculated by equating streamwise force moments (*c.f.* Equation 3-57, noting that $F_s = 0$):

$$y_0'^2 - 2h\,y_0' + k_h\,h = 0 \qquad \textit{Equation 5-9}$$

from which y_0' is given by

$$y_0' = \frac{2h - \sqrt{(2h)^2 - 4k_h\,h}}{2} \qquad \textit{Equation 5-10}$$

The estimated datum position is therefore a quadratic function dependent on the duct height and roughness height only. Table 5-1b compares these estimated datum positions with those obtained by equating the true streamwise force moments ($y_{0\ e.s.m.}$, Equation 3-57) and shows good agreement, the maximum error[5] being 10%, with the majority of the thin strip roughness configurations having an error of less that 5% of $y_{0\ e.s.m.}$. Also shown in Table 5-1c are the datum levels ($y_{0\ l.v.p.}$) obtained by non-linear regression on the logarithmic velocity profile (Method 4, Section 3.4.1) for $h/k_h \geq 7$. Reasonable agreement between $y_{0\ e.s.m.}$ and $y_{0\ l.v.p.}$ is shown, the maximum error being 18% of $y_{0e.s.m.}$ As the flow depth decreases a similar trend of increasing datum level position is shown for $y_{0\ e.s.m.}$, y_0' and $y_{0\ l.v.p.}$ For the roughness configurations examined[6], $y_{0\ e.s.m.}$ varied between $0.48 \leq y_0/k_h \leq 0.71$. This variation contrasts with the datum position predicted by the geometric mean level which varied with the roughness volume per period only. For the thin strip roughness elements (tests RSA and RSB) the geometric datum level was always negligible in comparison with the roughness height.

5.7 Velocity Profile Analysis

For flows with a high relative depth ($h/k_h \geq 7$) the period averaged velocity profiles each contained a region which had a logarithmic distribution. Figure 5-16 compares the

---

[5] This is with the exception of case RSB40, for which the value of $y_{0\ e.s.m.}$ was inaccurate due to lack of convergence.

[6] See footnote 5.

logarithmic velocity profiles for the bed datum positions $y_{0\ g.m.l.}$ and $y_{0\ e.s.m.}$ for the roughness configuration RSA.

Three regions were identified within each of the velocity profiles:

1. A "roughness" layer in which the velocity was controlled by the shape and distribution of the roughness elements. The velocity profile in this region varied with streamwise position, however the period averaged value (Figure 5-10) was approximately linear.

2. A logarithmic region (Equation 2-30).

3. A narrow region near the plane of symmetry in which the velocity decreased from that predicted by the logarithmic law. This was due to the boundary condition of zero gradient at the plane of symmetry.

The "outer" or wake region (e.g. Coles 1956) was not obtained for these simulations. The boundaries of the logarithmic region were identified by determining at which grid cells the logarithmic velocity profile deviated from a straight line. For a given roughness configuration and method of determining y_0, the lower boundary remained independent of the flow depth. For $y_{0\ e.s.m.}$ the logarithmic region extended from immediately above the roughness elements to approximately $y/h - y_0 = 0.72$, whereas a shorter region was obtained when y_0 was calculated from the geometric mean level. However when $y_{0\ e.s.m.}$ was employed a slightly wavy profile was observed within the logarithmic region (Figure 5-16f) for the deeper flows. This effect was quantified by calculating the standard error of estimate in the data set, defined as (e.g. Crow *et al* 1960):

$$s_\varepsilon = \sqrt{\frac{\sum (y_i - y_i')^2}{n-2}} \qquad \textit{Equation 5-11}$$

where y_i is the data value for point i in a total of n and y_i' is the value predicted by the equation derived by least squares linear regression. Values of the standard error of estimate are given in Table 5-6.

As shown in Table 5.6, the gradient of the logarithmic profile was found to be consistently higher when $y_{0\ g.m.l.}$ was employed and systematically decreased with increasing depth. The profiles are illustrated in Figure 5.16(a)-(f). Using $y_{0\ e.s.m.}$ the gradients were found to be much closer to the values (5.75 and 5.616) quoted by Nikuradse (1933) and Jayatilleke (1969) and no systematic variation of slope with depth was observed. A consequence of the two different velocity profiles, produced by the different wall datum values, is that the estimate of k_s is different in each case.

5.8 Evaluation of k_s

5.8.1 Comparison with Standard Velocity Profile

The k_s values, calculated from the logarithmic velocity profiles using each of the four methods described in Section 3.4.1, are shown in Table 5-6 for the relative depths greater than 7. The variation of k_s with relative depth is shown in Figure 5-17 for each roughness configuration. The alternative methods used to calculate k_s gave a larger variation in k_s when $y_{0\ g.m.l.}$ was used for the datum level as opposed to equating streamwise force moments. For $y_{0\ e.s.m.}$, each roughness configuration showed that Method 1 lay approximately mid-way between the other methods of calculating k_s, and this method was chosen for further analysis.

At the highest flow depths ($h/k_h > 20$), the value of k_s was found to be approximately constant with respect to flow depth (to within 4.5%) for the roughness configurations RSA and RSB. Indeed case RSB showed k_s to be within 3.2% over the range $7 \leq h/k_h \leq 40$. The maximum relative depth simulated for the RCDA cases was $h/k_h = 20$, and below this k_s varied with flow depth. Because the subsequent analysis required a constant value of k_s at very deep flows and this was not achieved for the RCDA roughness configuration, only the RSA and RSB results were used in the derivation of

the resistance equations described in the following sections. The RCDA simulations were however included in the final application of the resistance equations.

The deepest flows simulated for all of the roughness configurations had flow depths which were large compared with the height of the roughness elements, i.e. $h/k_h = 40$ ($h/k_h = 20$ for RCDA). However the k_s values obtained were significantly higher than the roughness heights, resulting in maximum h/k_s values of approximately 8.5, 6.0 and 5.8 for RSA, RSB and RCDA respectively. The roughness length scale (k_s) was therefore still of the same order of magnitude as the depth of flow.

The k_s values for RSA and RSB agreed well with the results of the closed duct simulation of Schlichting's original experiment (Section 4-1). The values of k_s were higher for the roughness configuration RSB than for RSA, i.e. the degree of roughness increased with a decrease in period length. This result agrees with Morris (1954) for the isolated roughness and wake interference flow regimes. However, for skimming flows Morris found the k_s value to decrease with a shorter period length.

The k_s values were not surprisingly an order of magnitude higher for the flow over the 50 mm block roughness elements, given the roughness height relative to that of Schlichting's long angle roughness.

5.8.2 Resistance Equation

The k_s value can also be obtained using a resistance equation of the form (Equation 3-50)

$$\frac{1}{\sqrt{f}} = a - b\log\frac{k_s}{2R_h} \qquad \textit{Equation 5-12}$$

provided that the coefficients a and b are known. The equation for k_s is

$$k_s = 2R_h \times 10^{\frac{1}{b}\left(a - \frac{1}{\sqrt{f}}\right)} \qquad \textit{Equation 5-13}$$

In the present study, the coefficients a and b were determined using the k_s values obtained from the velocity profile analysis of the deep flow thin strip rib roughness experiments RSA and RSB. The duct friction factor (Table 5-1b) was calculated from

$$f = \frac{8\tau_{y_0}}{\rho \overline{U}_{y_0}^{\,2}} \qquad \textit{Equation 5-14}$$

where τ_{y0} is the shear stress with line of action at $y = y_{0\ e.s.m.}$ [7] and $\overline{U}_{y_0}$ is the velocity averaged over the depth $h - y_0$, i.e. $\overline{U}_{y_0} = {}^{q}/_{h-y_0}$. This differed from the standard practice of calculating the friction factor relative to the base plane of the roughness elements:

$$f = \frac{8\tau_w}{\rho \overline{U}^2} \qquad \textit{Equation 5-15}$$

where τ_w is the shear stress with line of action through the base of the roughness elements and $\overline{U}$ is the velocity averaged over the maximum flow depth. The latter method was discarded because of its incompatibility with the rough wall datum level.

Figure 5-18 shows $1/\sqrt{f}$ plotted against $\log(2R_h/k_s)$ for the roughness configurations RSA and RSB ($h/k_h \geq 7$) with all points closely fitted by a straight line. Least squares regression gave the following resistance equation:

$$\frac{1}{\sqrt{f}} = 1.49 - 1.97 \log \frac{k_s}{2R_h} \qquad \textit{Equation 5-16}$$

Also shown in Figure 5-18 are the resistance equations by Nikuradse (1933) (a=1.74, b=2.0), French (1986) (a=1.556, b=2.0) and by simulations in which the k_s value was specified within a wall function using the standard k-ε model[8] (a=1.888, b=1.985).

---

[7] For all subsequent analyses, y_0 was calculated from equating streamwise force moments. For clarity of writing, the subscript e.s.m. will therefore be omitted for the remainder of this chapter.

[8] Described in Section 5.5.

All of the resistance lines shown in Figure 5-18 are parallel, with a coefficient b approximately equal to 2. As the intercept increases, the degree of roughness and therefore k_s also increases.

It should be noted that in the present application of the resistance equation (5-16) to an equivalent plane walled duct, the hydraulic radius $R_h = (h - y_0)$.

5.8.3 Non-dimensional Variation of k_s

Using the coefficients $a = 1.49$ and $b = 1.97$ (two-layer model) within the resistance equation, the k_s value was calculated for all of the relative depths using Equation 5-13. The distribution of k_s with depth is shown non-dimensionally in Figure 5-19 by plotting $k_s/h - y_0$ against $k_s/k_{s\infty}$, where $k_{s\infty}$ is the value of k_s obtained at the largest flow depth for each roughness configuration. A logarithmic scale was employed for the ordinate to resolve the data at higher flow depths. Also shown in Figure 5-19 are the k_s values obtained from the velocity profile analyses for relative depths greater than 7. The k_s values predicted by the resistance equation are in reasonable agreement with the k_s values from the velocity profiles.

The resistance equation predicted a very similar trend in k_s for the three roughness configurations tested. The k_s value decreased slightly as the depth of flow was reduced from the maximum, then at approximately $k_s/h - y_0 = 1$, the value of k_s fell rapidly. That $k_s/k_{s\infty}$ increased from unity as the depth of flow decreased from its maximum value highlights the fact that k_s was not entirely independent of flow depth, even for the deepest of the flows simulated. The variation was small enough however to deduce that the value of $k_{s\infty}$ would not have changed significantly even if simulations of deeper flows had been possible.

A curve of the form

$$\frac{k_s}{k_{s\infty}} = A\left(\frac{k_s}{h-y_0}\right)^n + B\left(\frac{k_s}{h-y_0}\right)^m + 1 \qquad \textit{Equation 5-17}$$

was fitted through the data of RSA and RSB, as shown in Figure 5-20. Non-linear least squares regression returned the coefficients as $A = -4.036$, $B = 4.087$, $n = 1.125$ and $m = 1.094$. This best fit line intercepts the x axis $\left({}^{k_s}/_{(h-y_0)} = 0\right)$ at ${}^{k_s}/_{k_{s\infty}} = 1$ with a vertical gradient.

Figure 5-21 shows the variation of k_s with flow depth obtained from Equation 5-17 using a Newton-Raphson solution, and the previously computed values of $k_{s\infty}$ and $(h-y_0)$. The agreement with the k_s values obtained from velocity profile analysis for $h/k_h > 7$, also shown in Figure 5-21 is quite good as it should be given that these k_s values were used to determine Equation 5-16.

Using Equation 5-14, the k_s values obtained from Equation 5-17 were also added to the resistance line of Figure 5-18, as shown in Figure 5-22. All of the roughness configurations (including RCDA which was not used in the derivation of either Equation 5-13 or Equation 5-17) are seen to lie close to the line predicted by Equation 5-13. This indicates that the resistance equation, Equation 5-16, in combination with Equation 5-17 defining k_s is valid for any depth of flow over the roughness configurations tested.

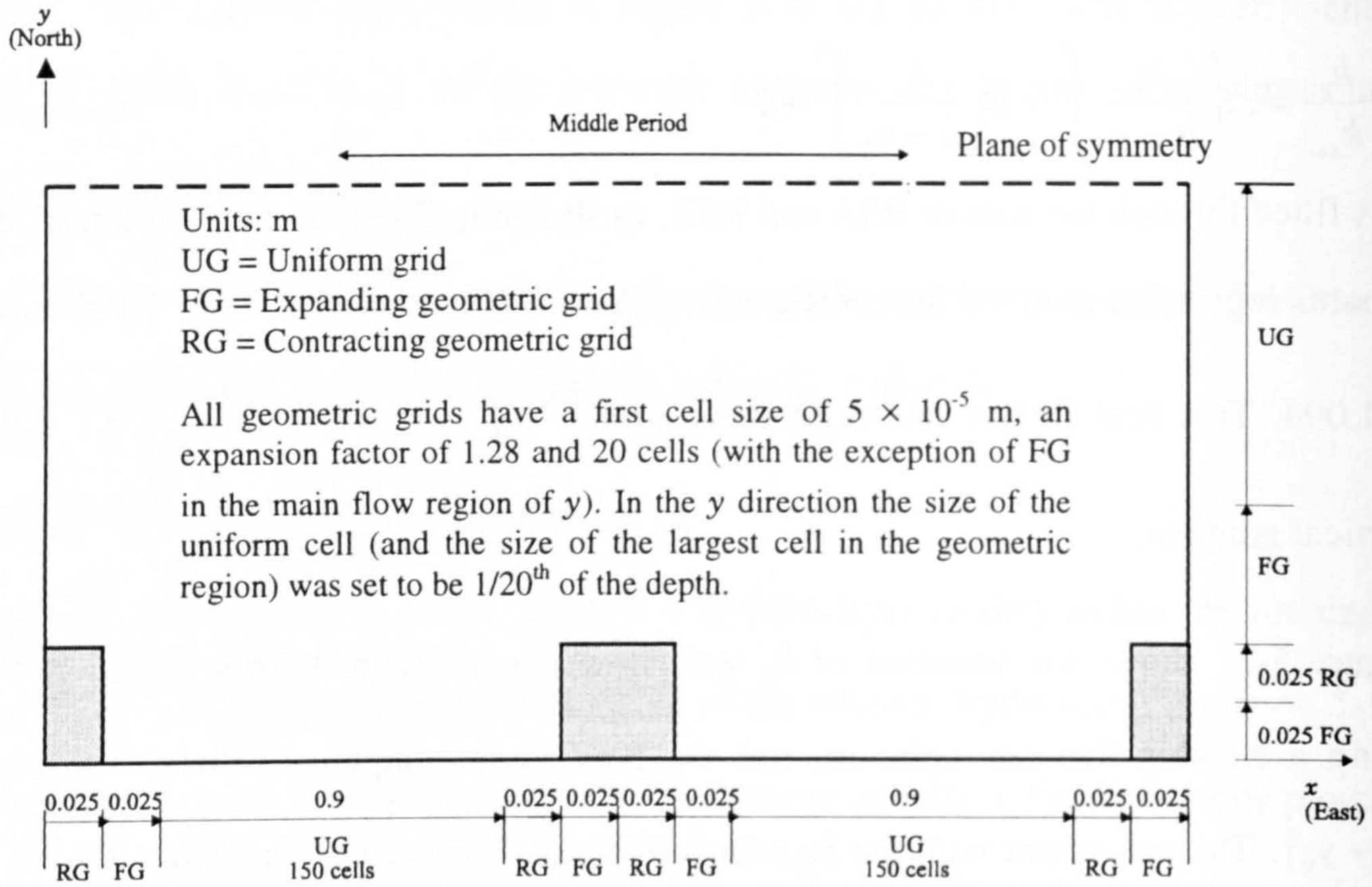


Figure 5-1a: Domain and grid structure for roughness configuration RCDA

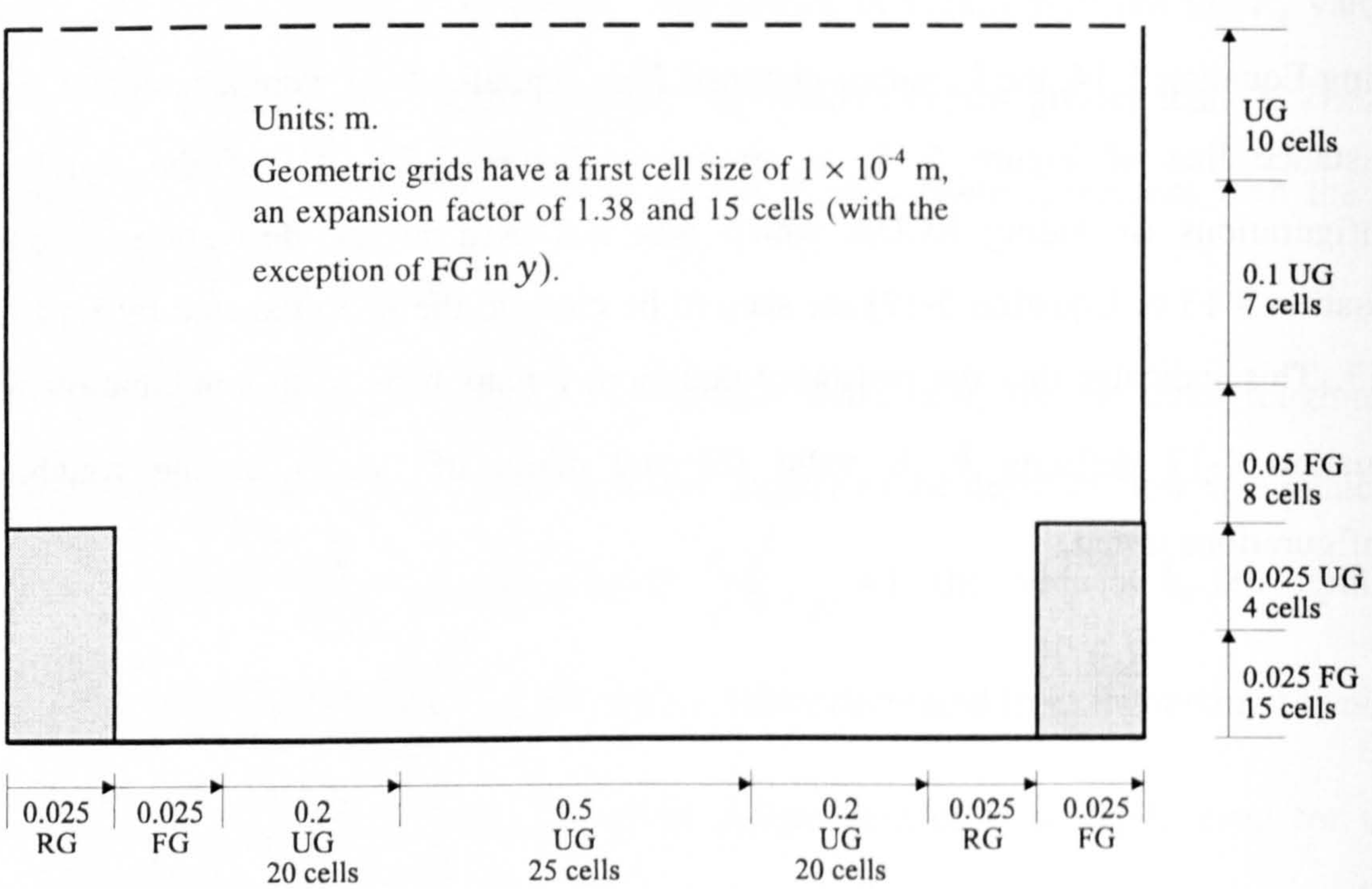


Figure 5-1b: Domain and grid structure for grid independence test, RCDA

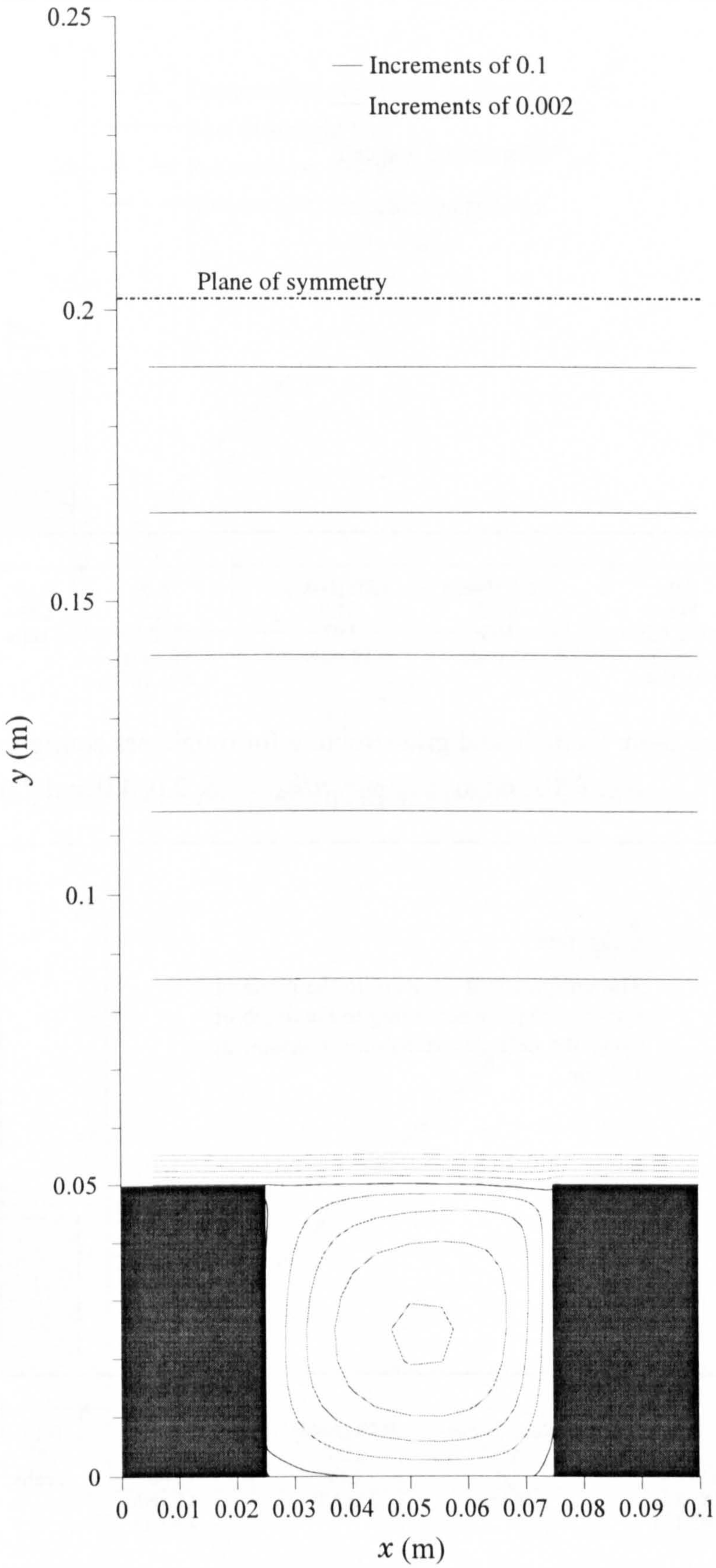


Figure 5-2: Streamlines for skimming flow regime

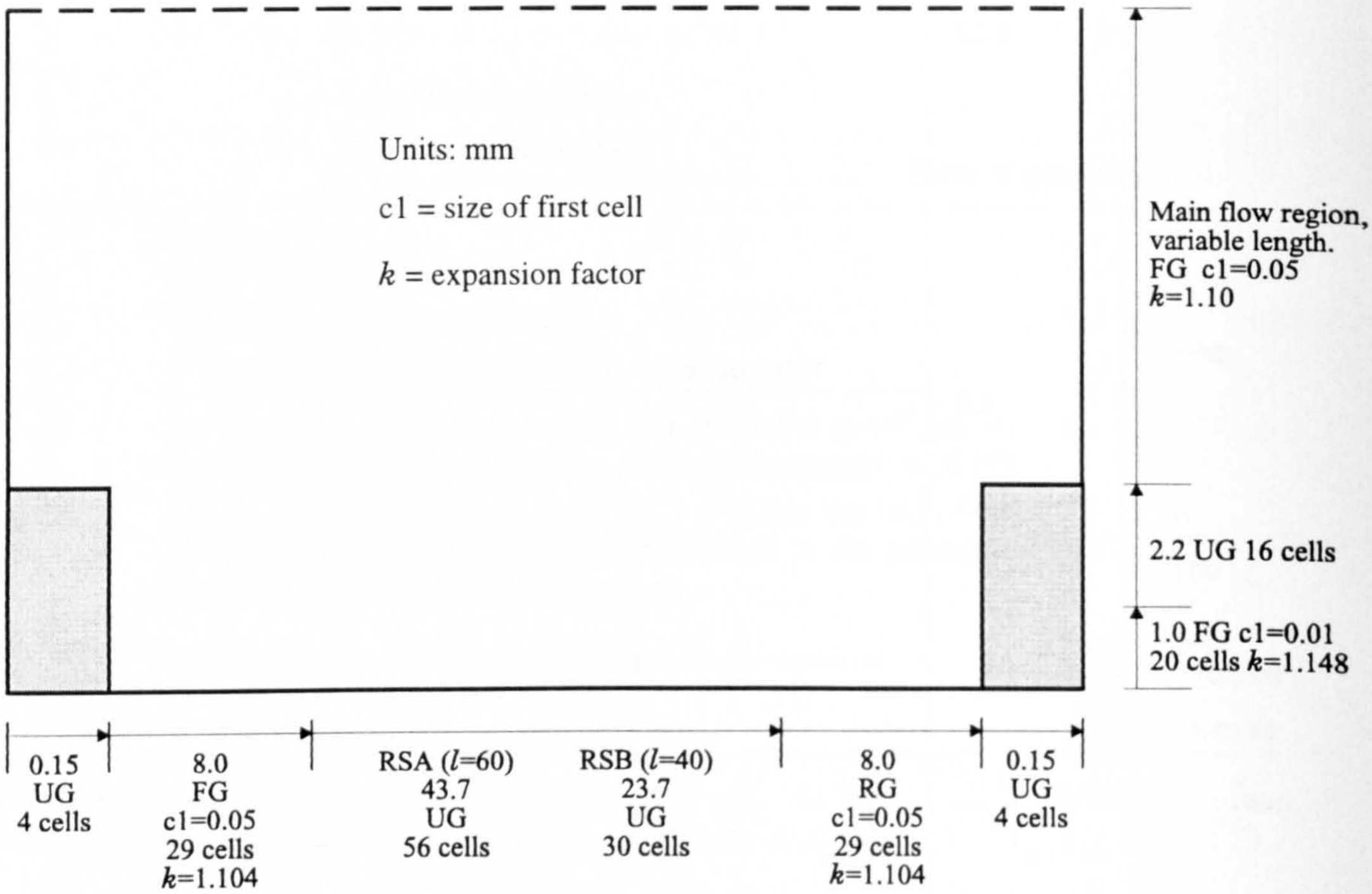


Figure 5-3a: Domain and grid structure for roughness configuration RSA and RSB, relative depths h/k_h = 1.5, 2.0, 4.0 and 7.0

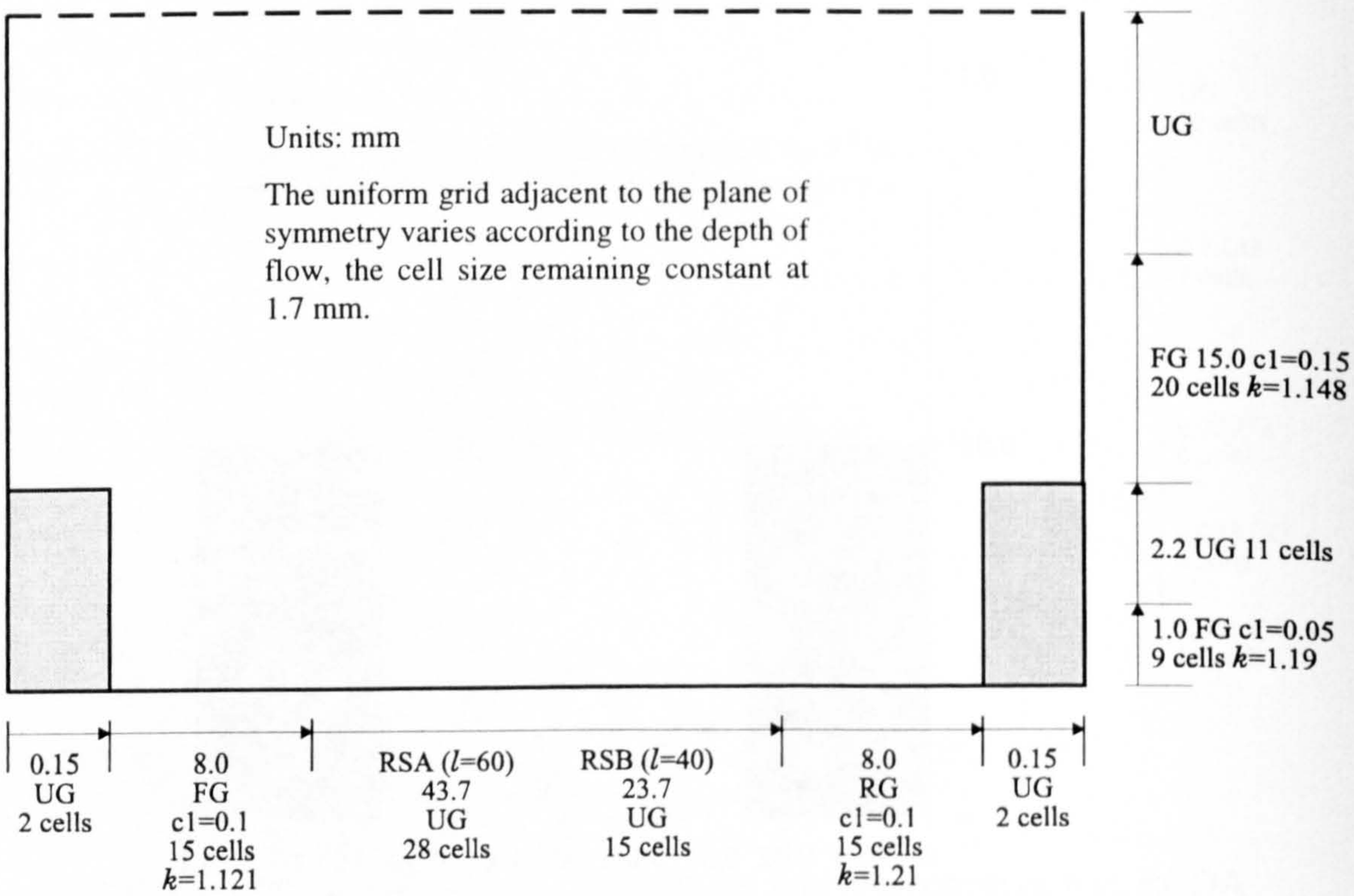


Figure 5-3b: Domain and grid structure for roughness configuration RSA and RSB, relative depths h/k_h = 7 to 40

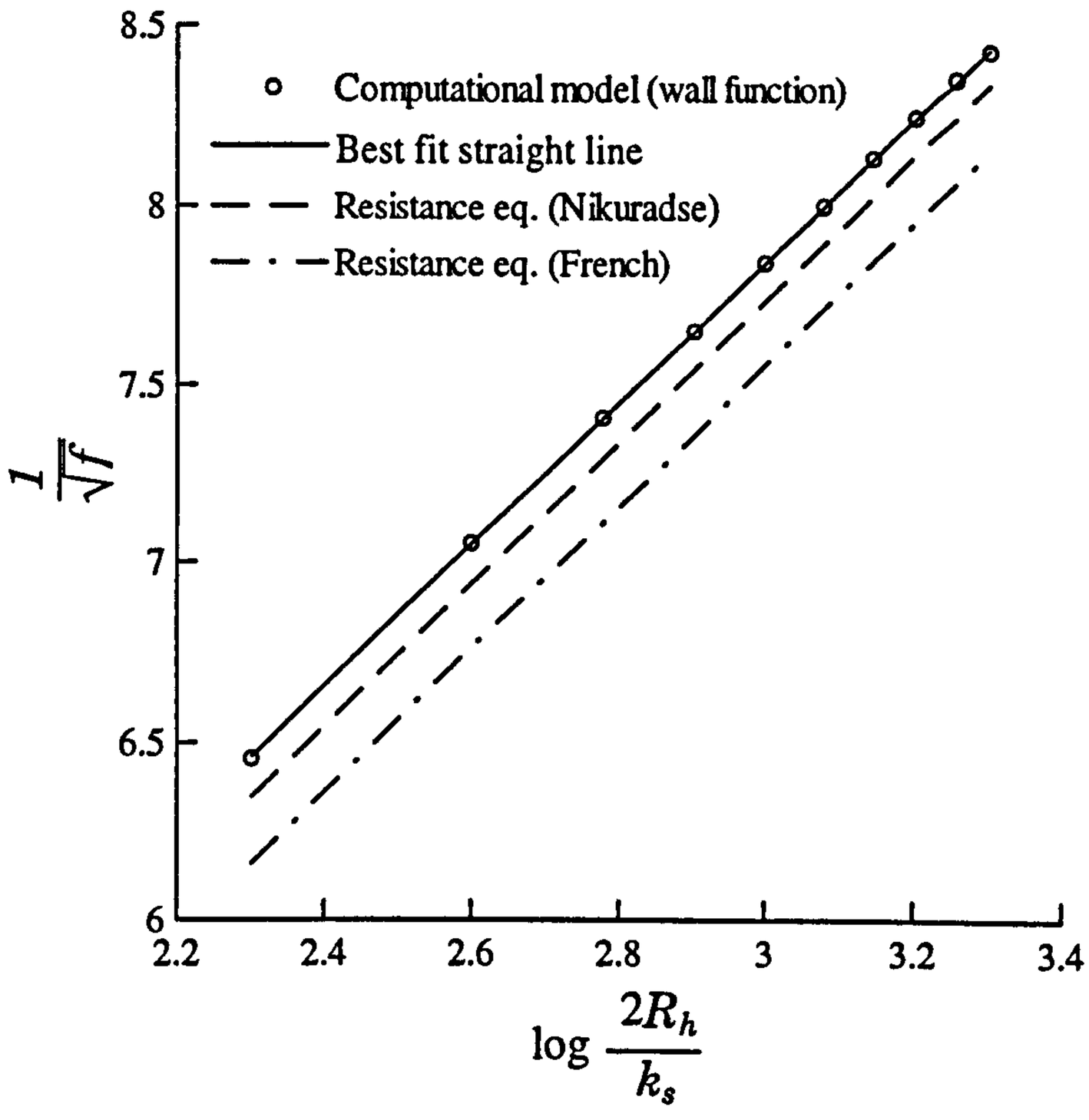


Figure 5-4: Resistance equation for wall function tests

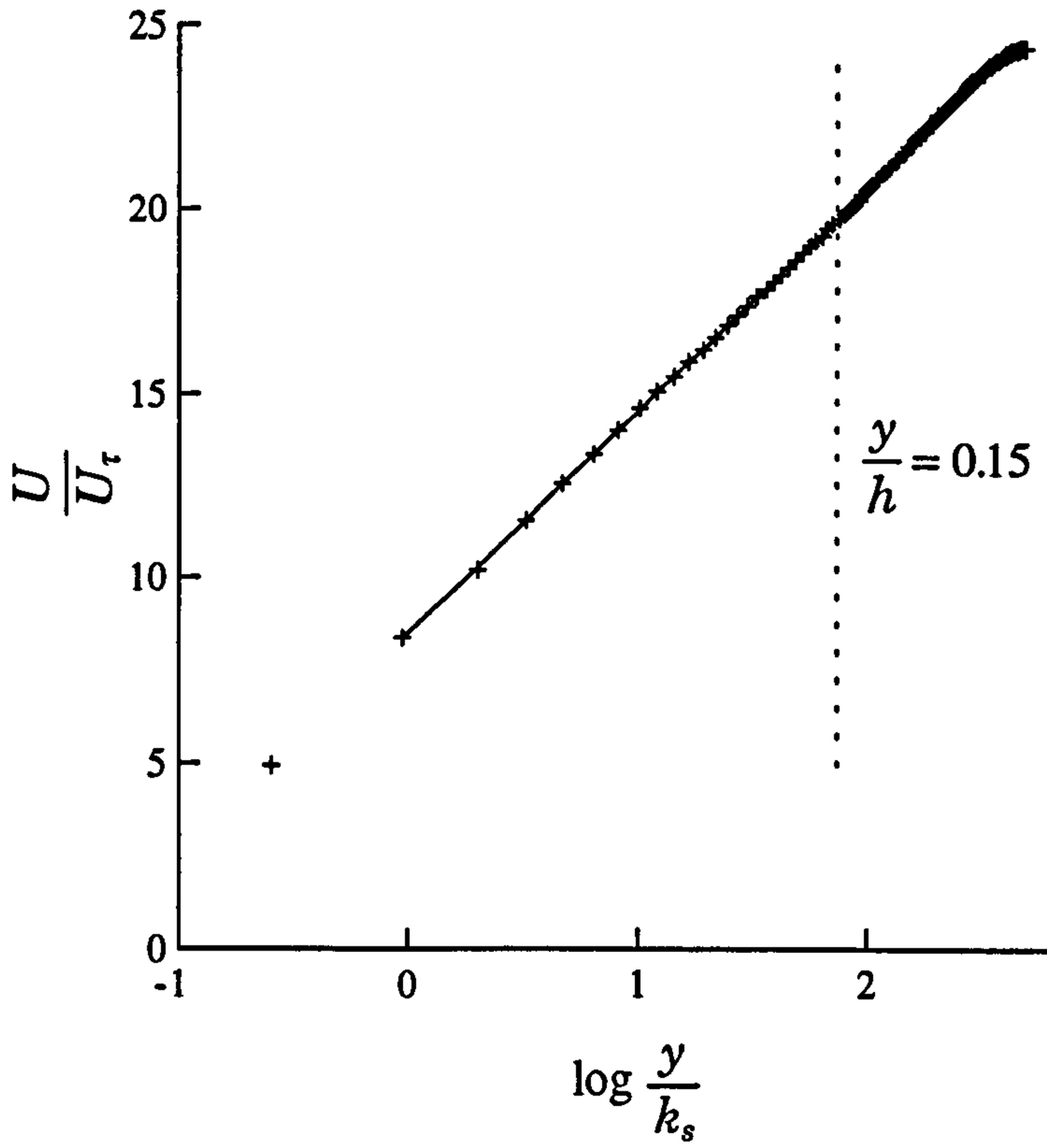


Figure 5-5: Logarithmic velocity profile for depth $h = 0.5$ m

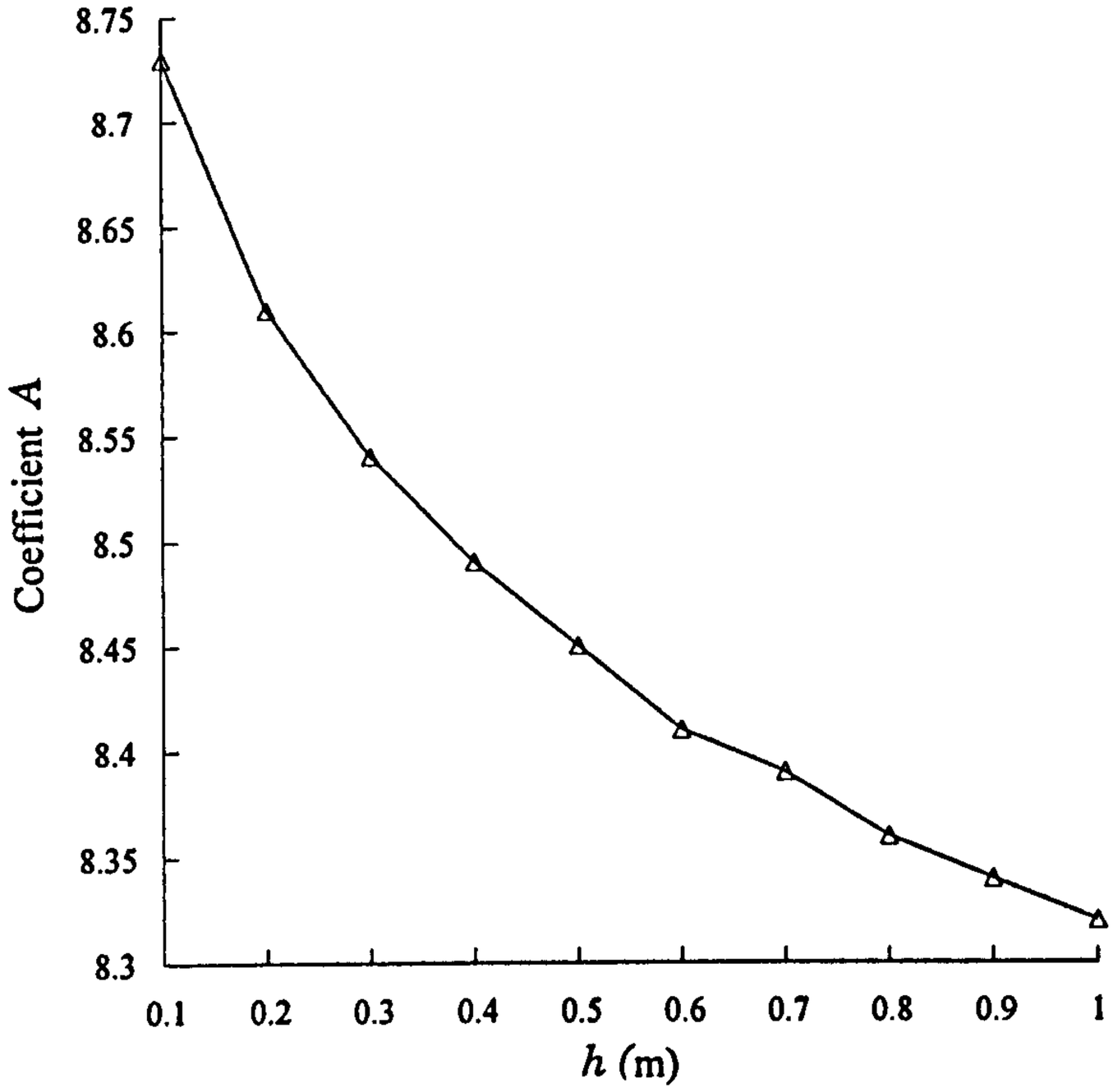


Figure 5-6: Variation of coefficient A in the velocity profile

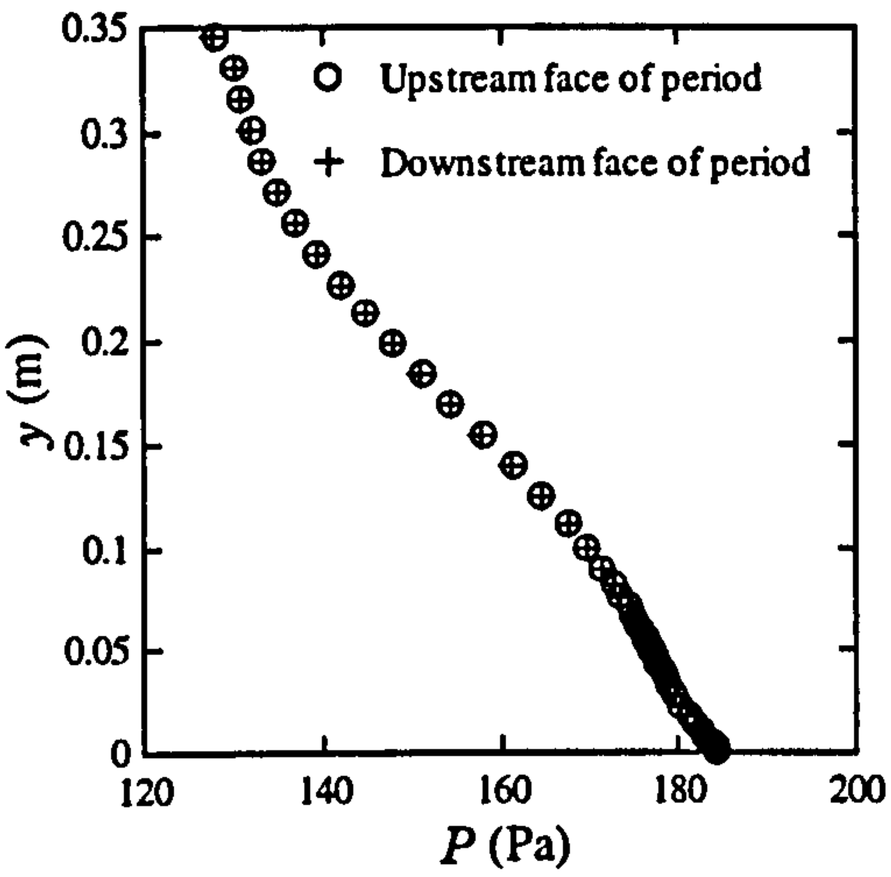


Figure 5-7: Profiles of pressure at the upstream and downstream faces of a period centred about a roughness element

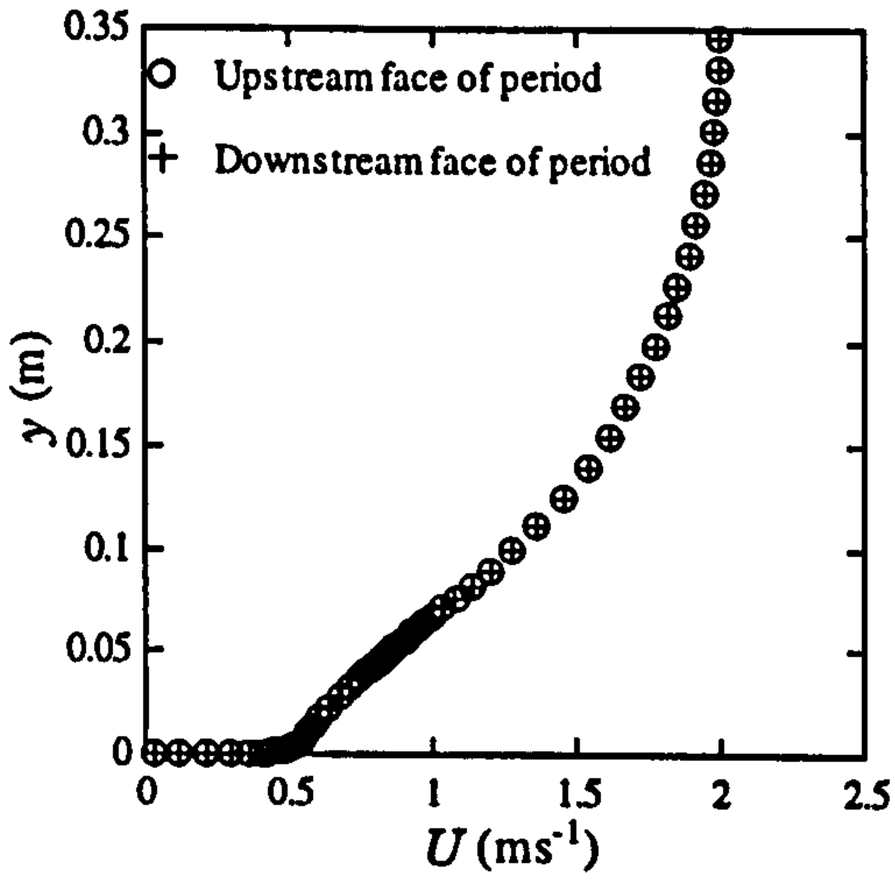


Figure 5-8: Profiles of streamwise velocity at the upstream and downstream faces of a period centred about a roughness element

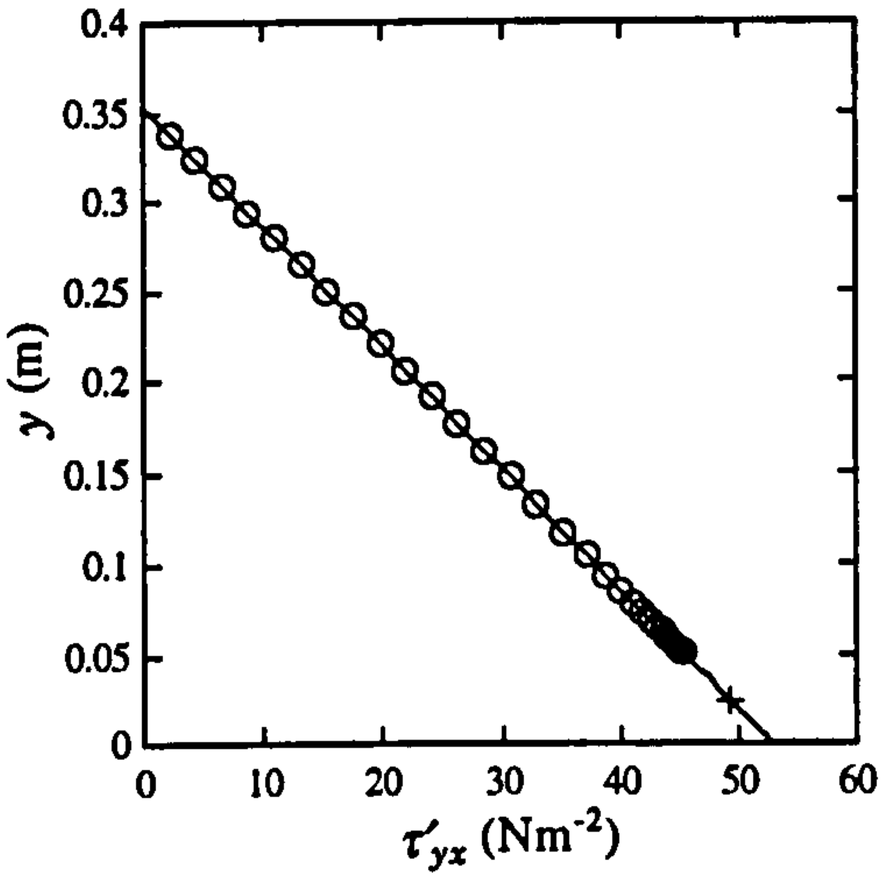


Figure 5-9: Transverse distribution of reduced shear stress. The '+' symbol denotes the plane and value of the equivalent bed shear stress obtained by equating streamwise force moments

Figures 5-7 to 5-9 refer to the roughness configuration RCDA, relative depth of 7.05

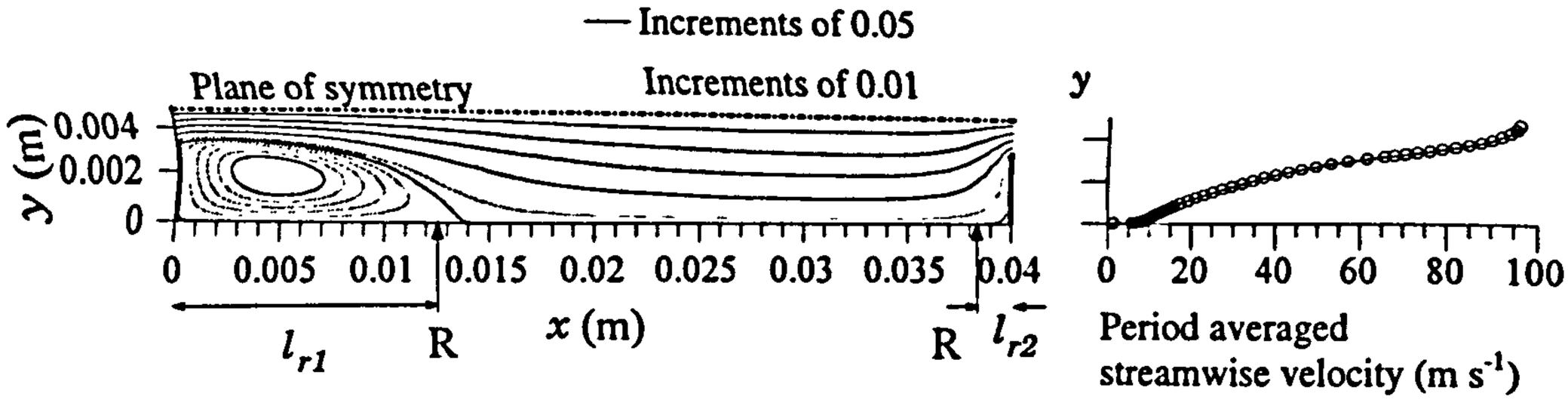


Figure 5-10a: relative depth of 1.5

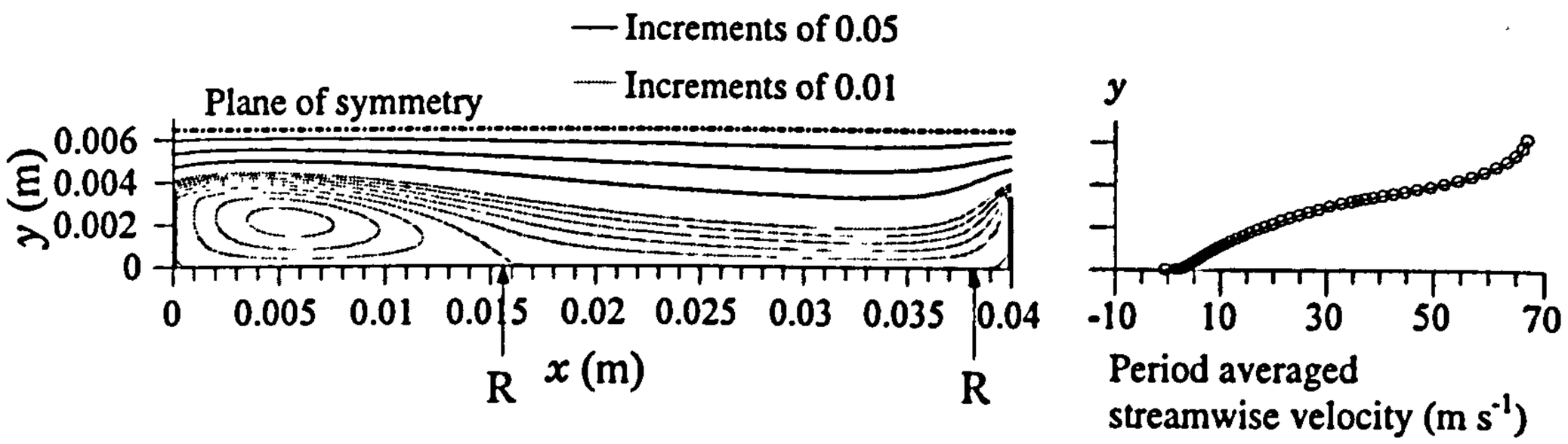


Figure 5-10b: relative depth of 2

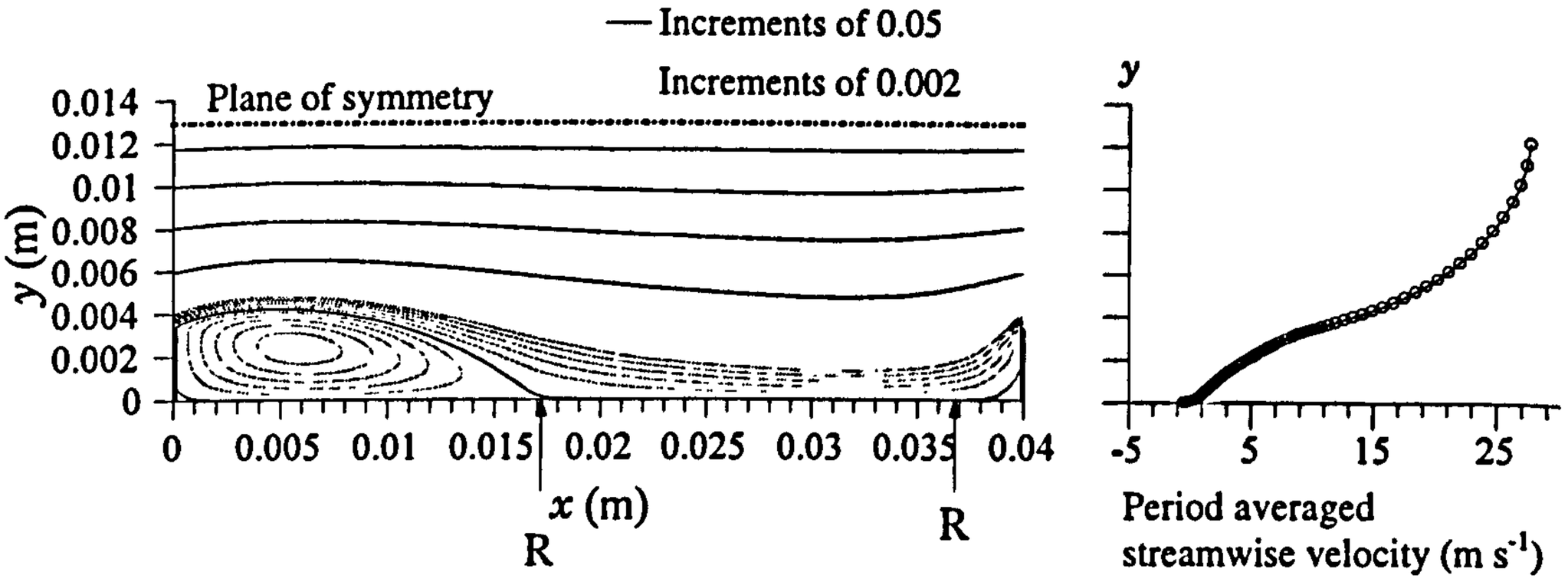


Figure 5-10c: relative depth of 4

Figure 5-10: Streamlines and period averaged velocity for roughness configuration RSB. 'R' denotes a point of flow re-attachment or separation.

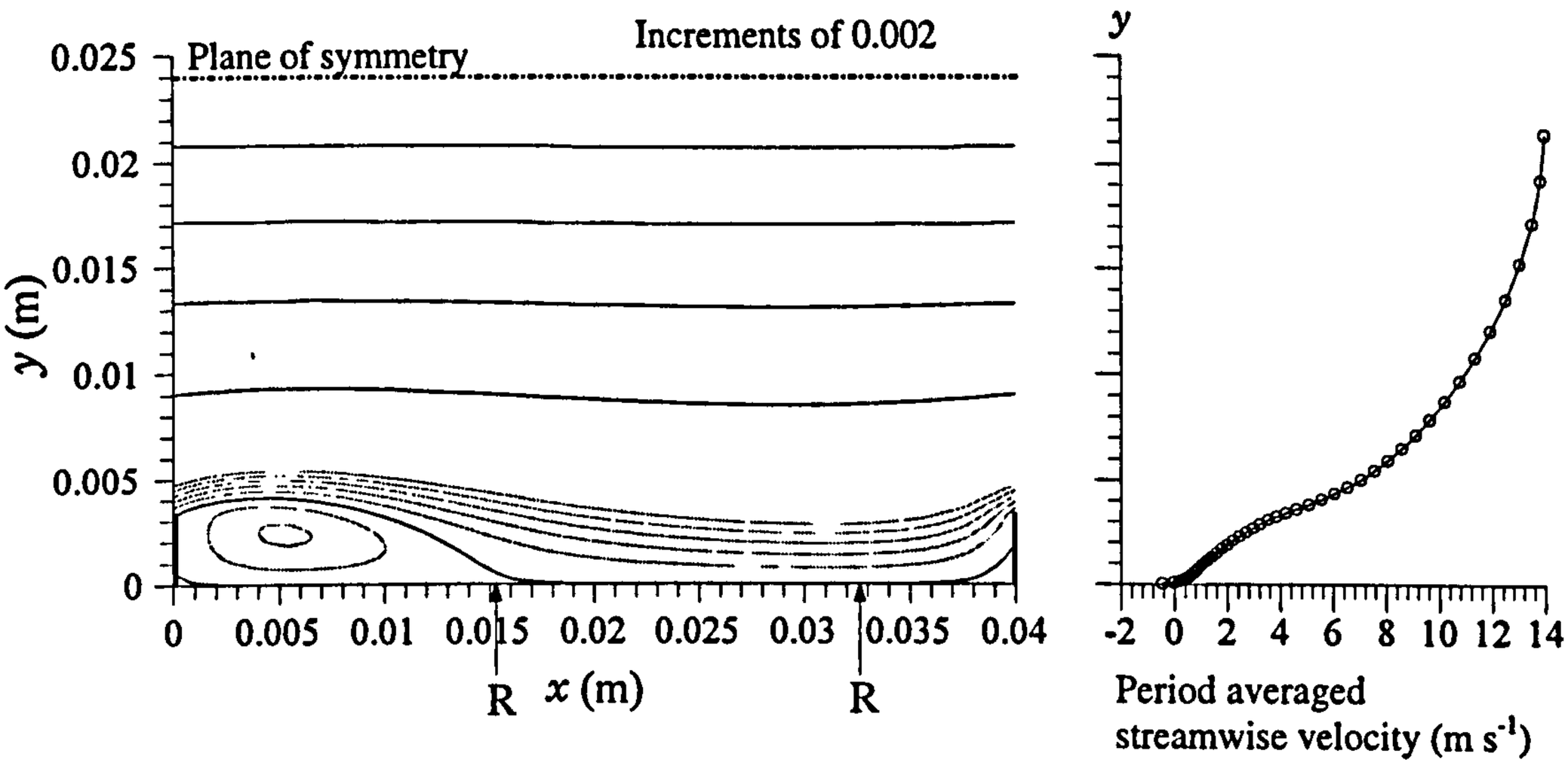


Figure 5-10d: relative depth of 7

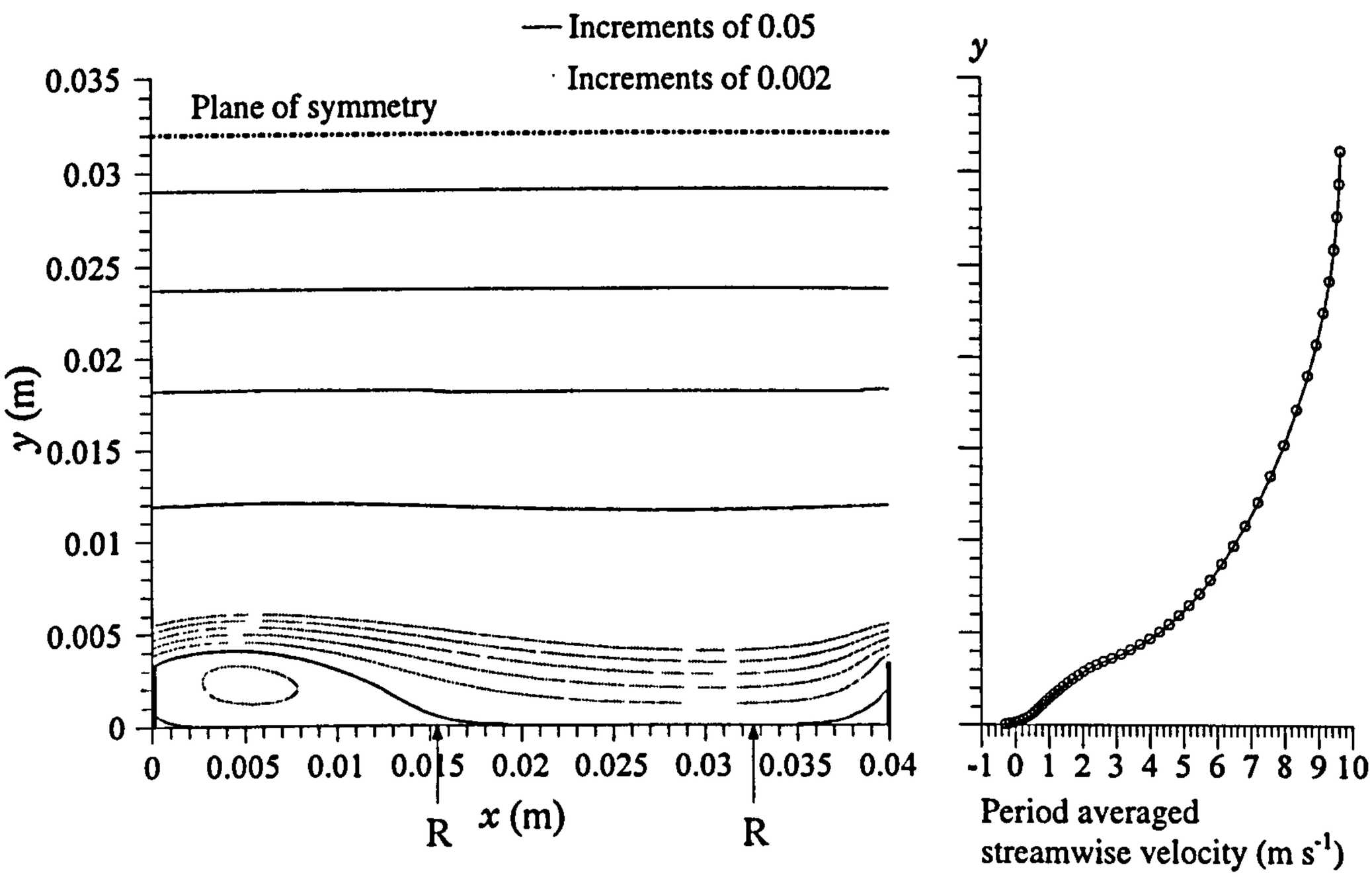


Figure 5-10e: relative depth of 10

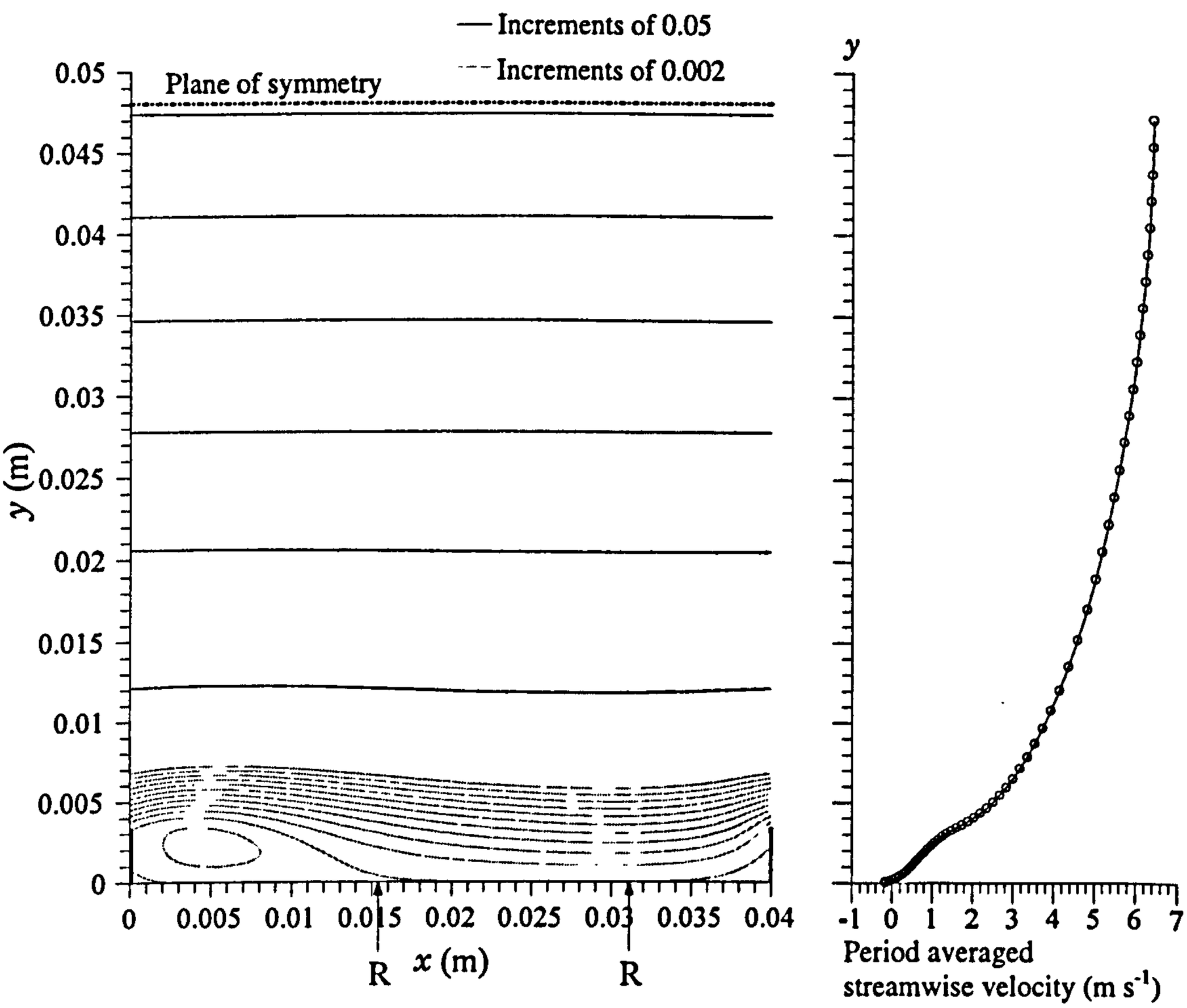


Figure 5-10f: relative depth of 15

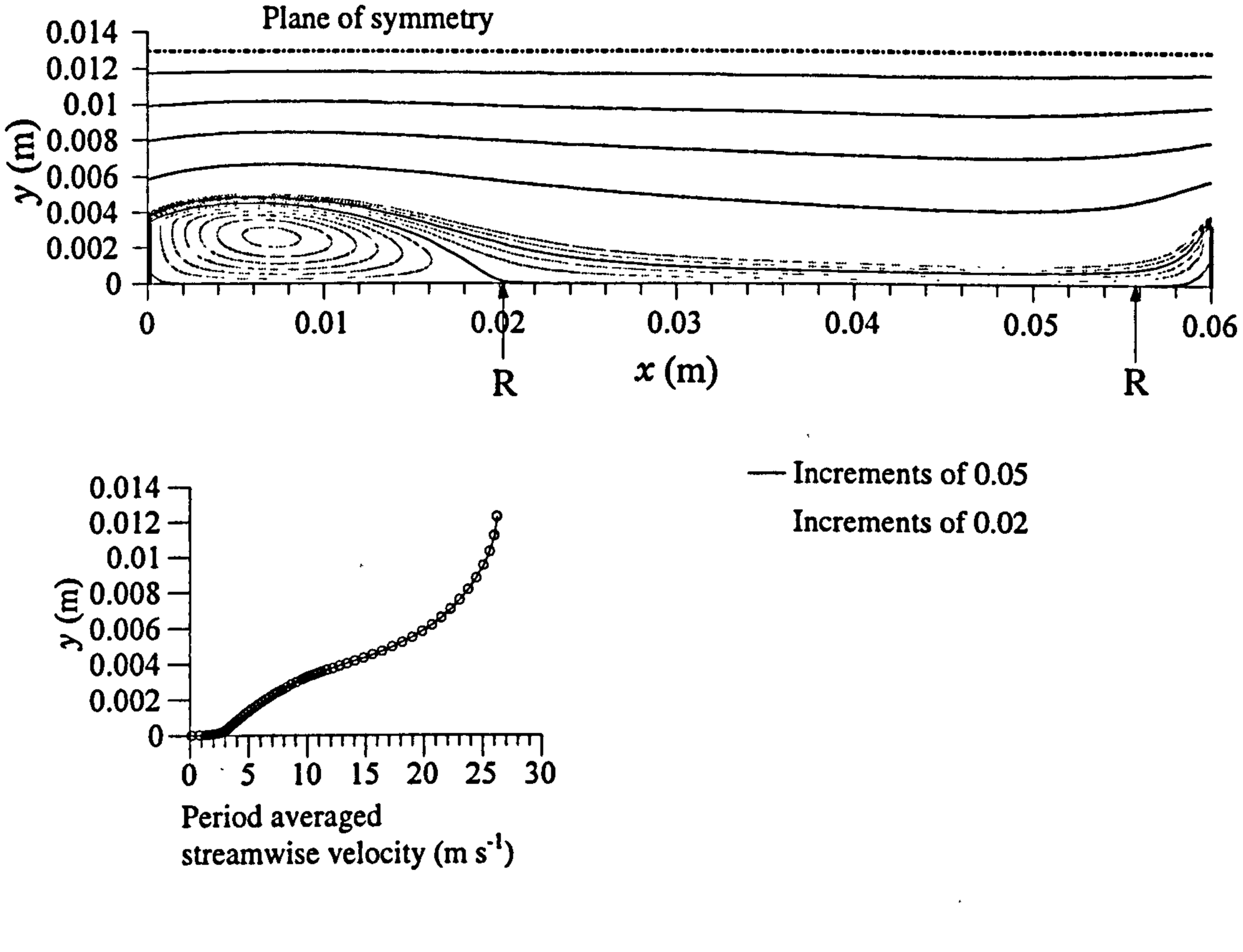


Figure 5-11: Streamlines and period averaged velocity for roughness configuration RSA, relative depth of 4

Figure 5-12: Streamlines and period averaged velocity for roughness configuration RCDA, relative depth of 4.05

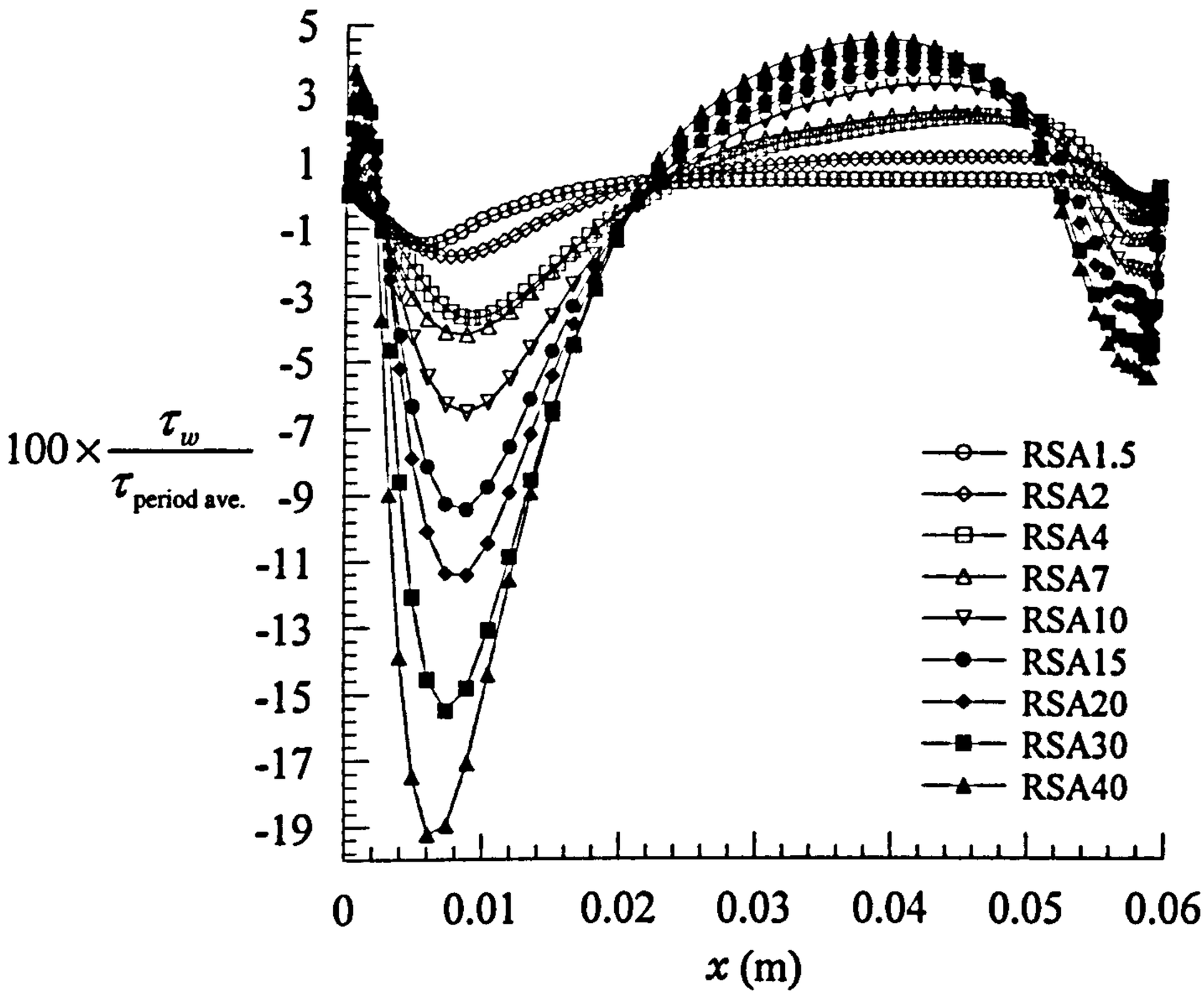


Figure 5-13: Local bed shear stress as a percentage of the period averaged bed stress including form drag on roughness elements. Roughness configuration, RSA.

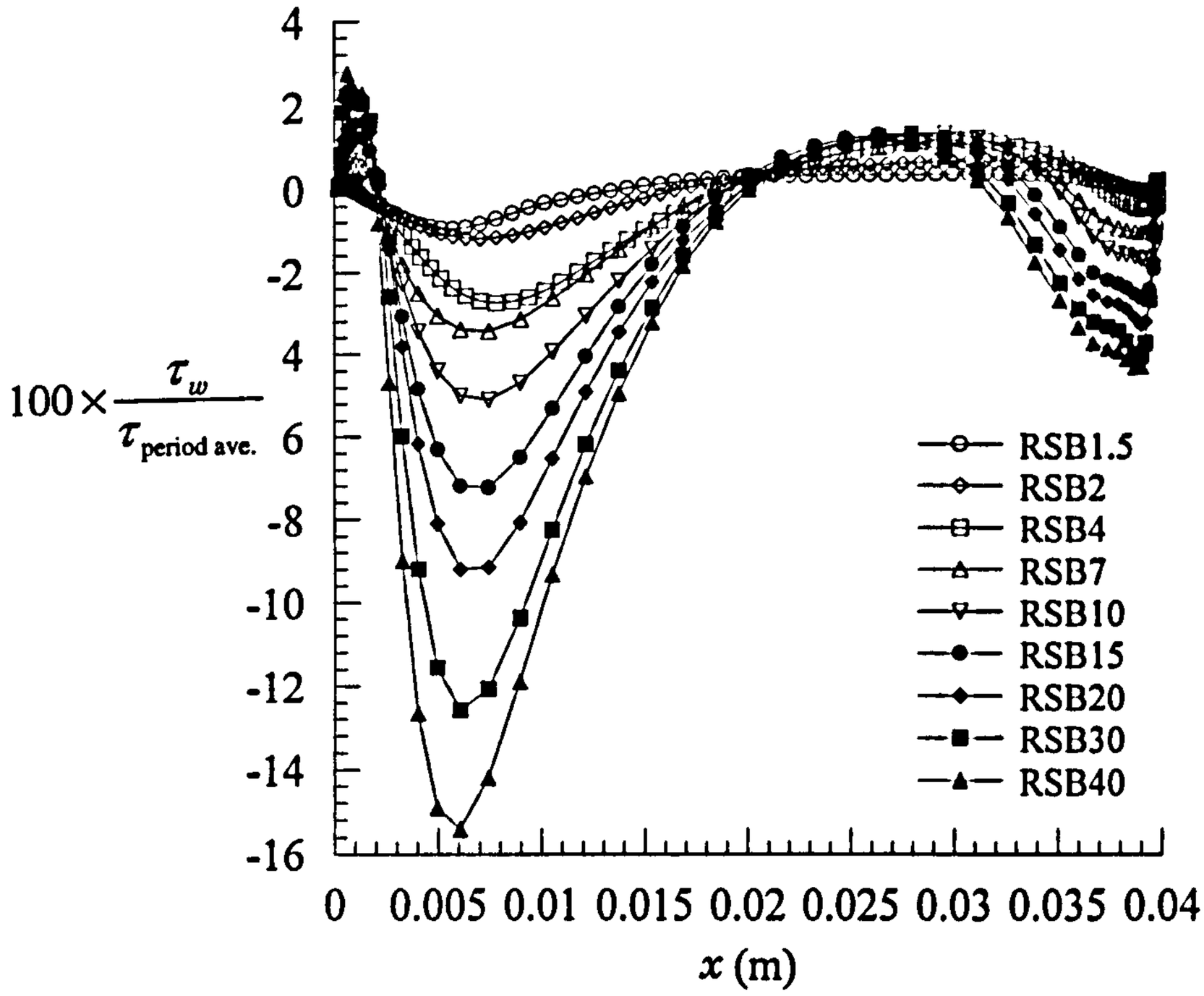


Figure 5-14: Local bed shear stress as a percentage of the period averaged bed stress including form drag on roughness elements. Roughness configuration, RSB.

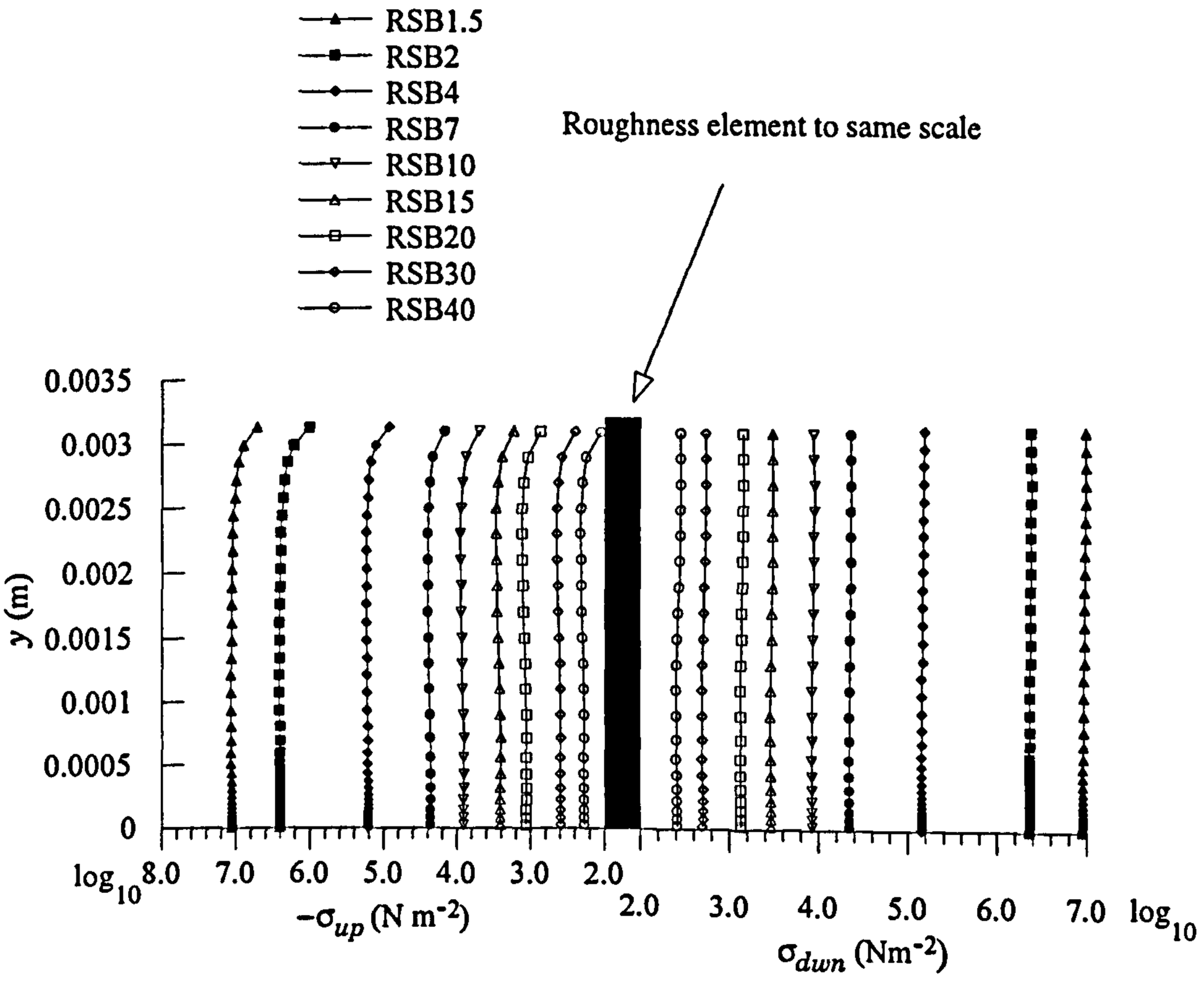


Figure 5-15: Direct stress on vertical faces of a roughness element. Roughness configuration RSB.

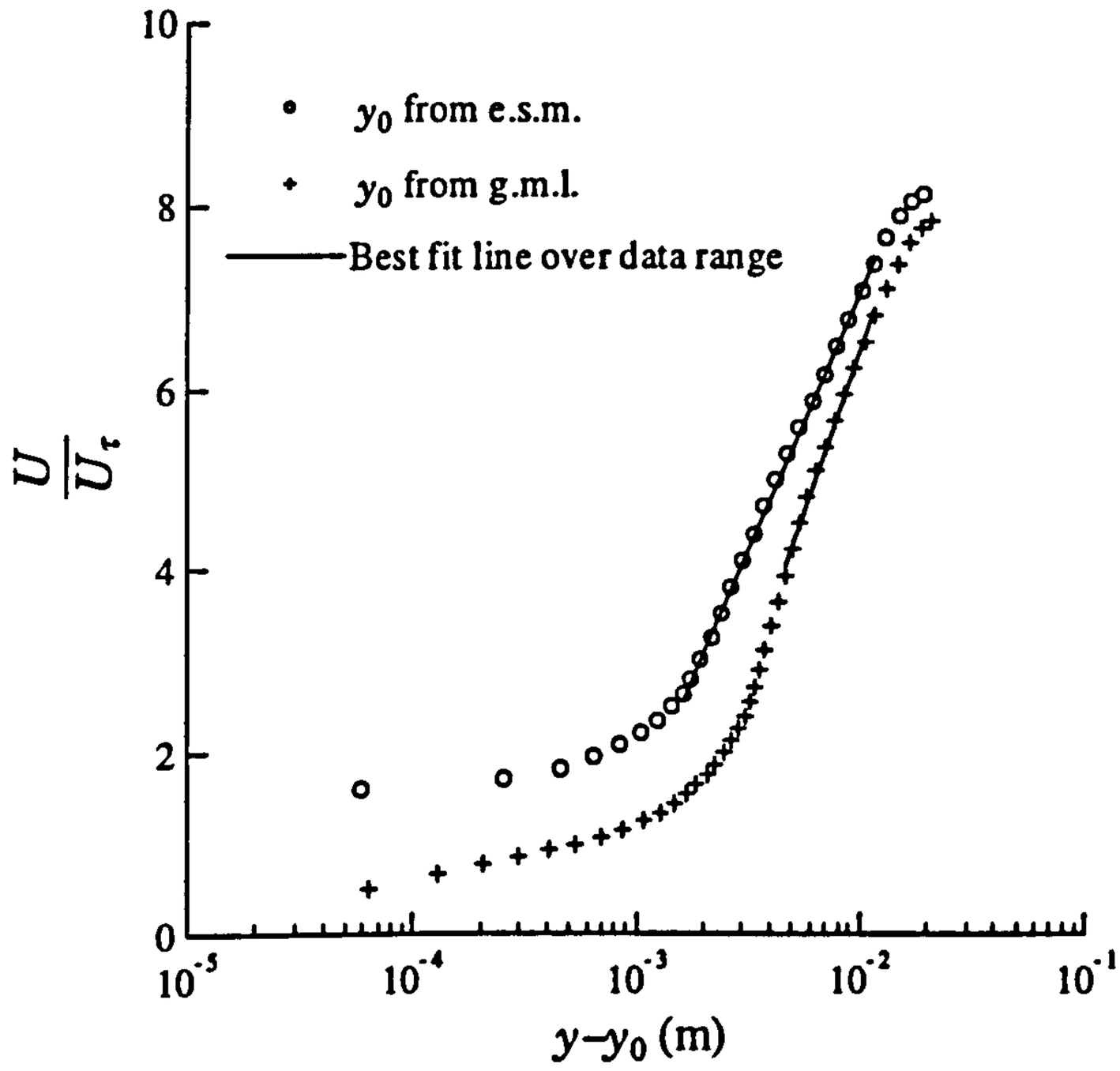


Figure 5-16a: Logarithmic velocity profile for RSA, relative depth of 7

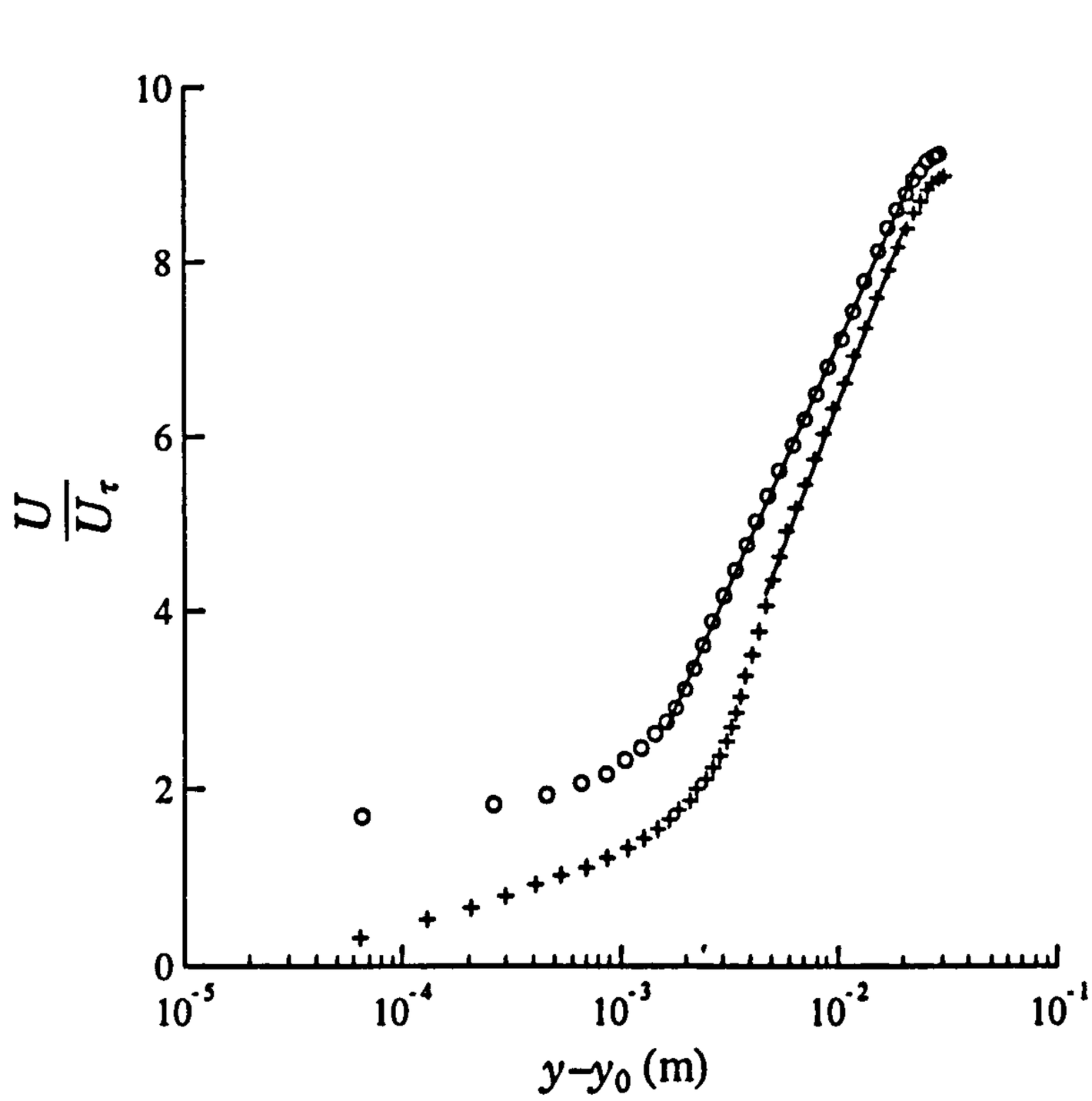


Figure 5-16b: Logarithmic velocity profile for RSA, relative depth of 10

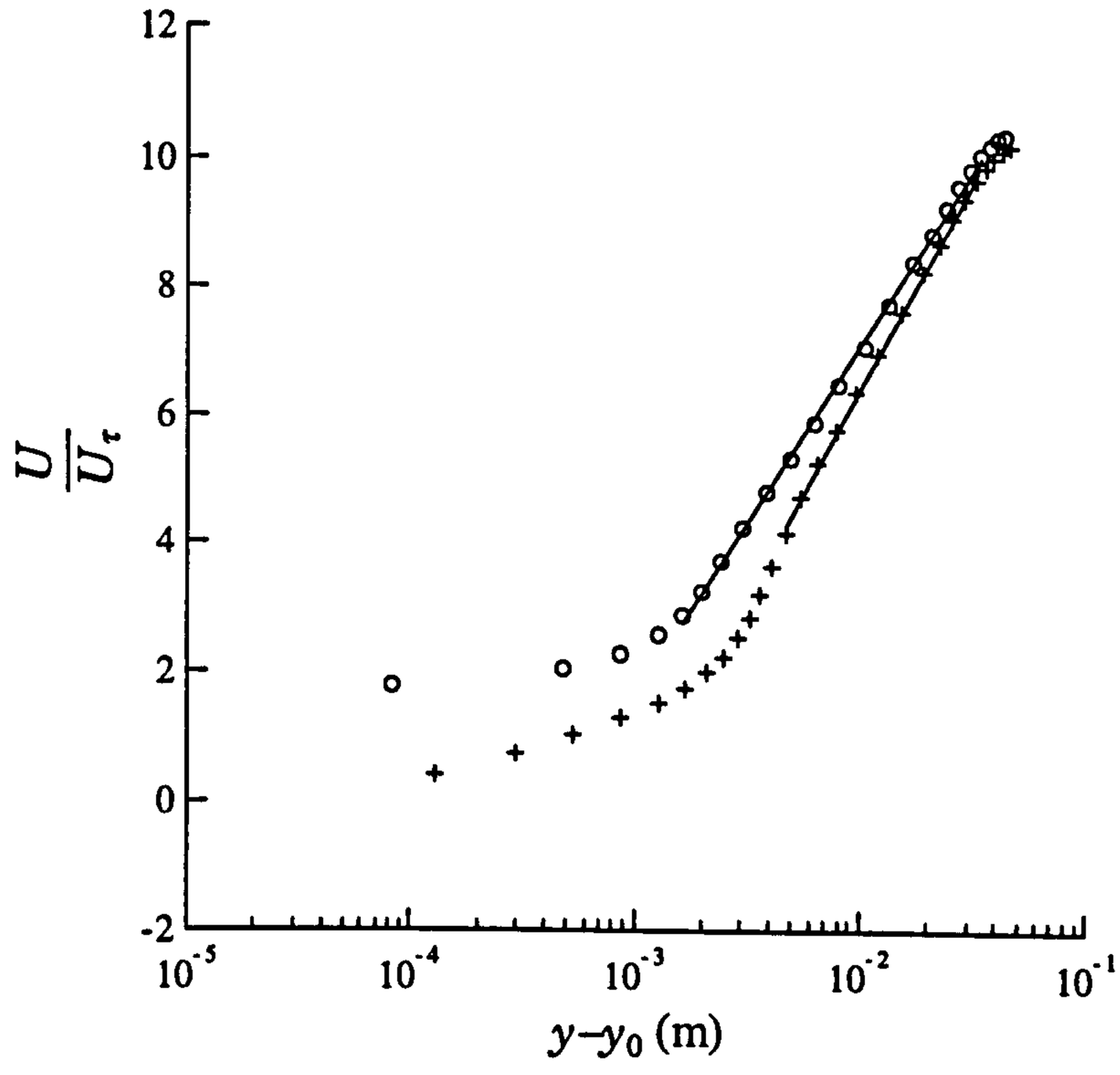


Figure 5-16c: Logarithmic velocity profile for RSA, relative depth of 15. (Data plotted every second cell for clarity).

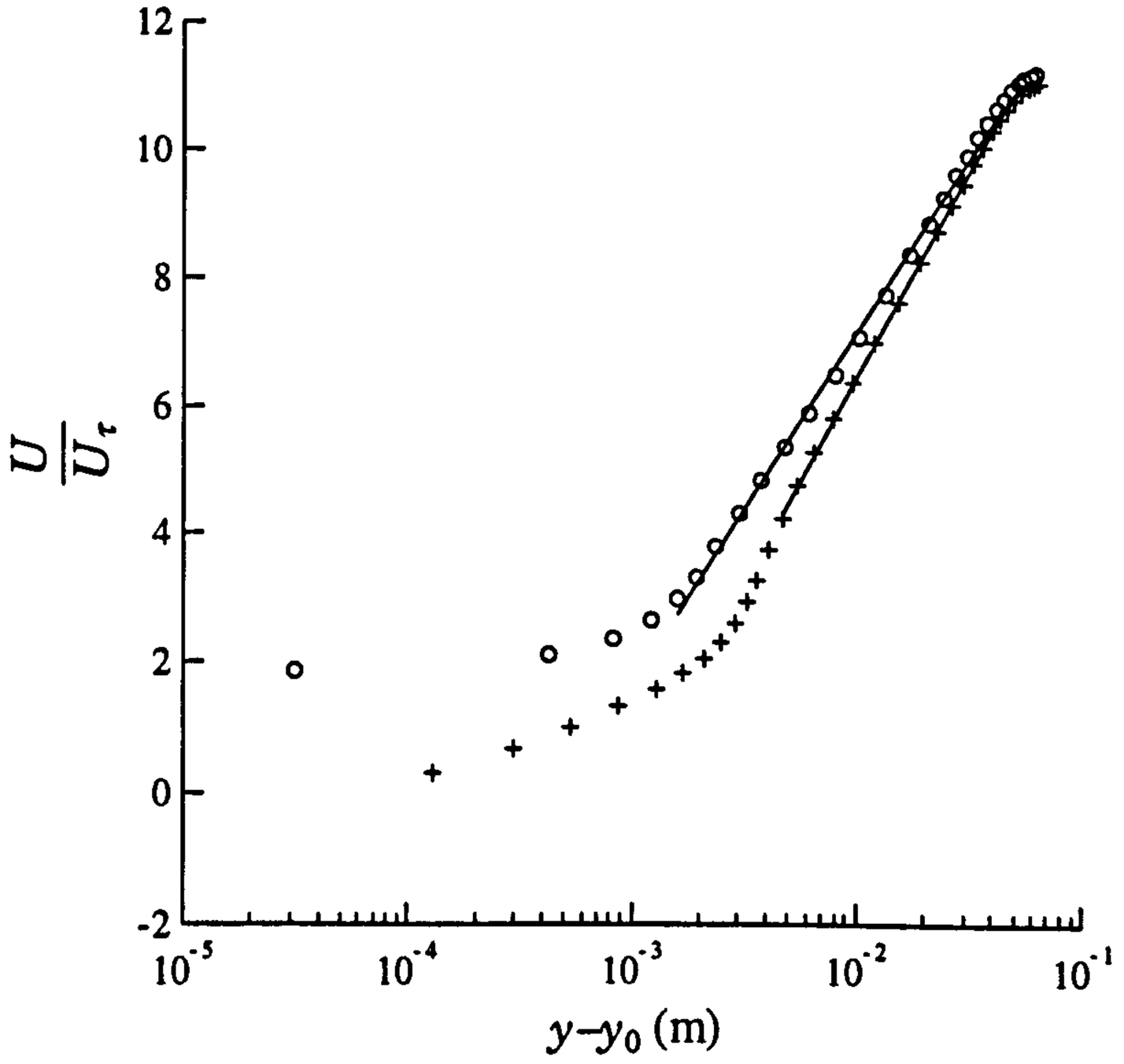


Figure 5-16d: Logarithmic velocity profile for RSA, relative depth of 20. (Data plotted every second cell for clarity).

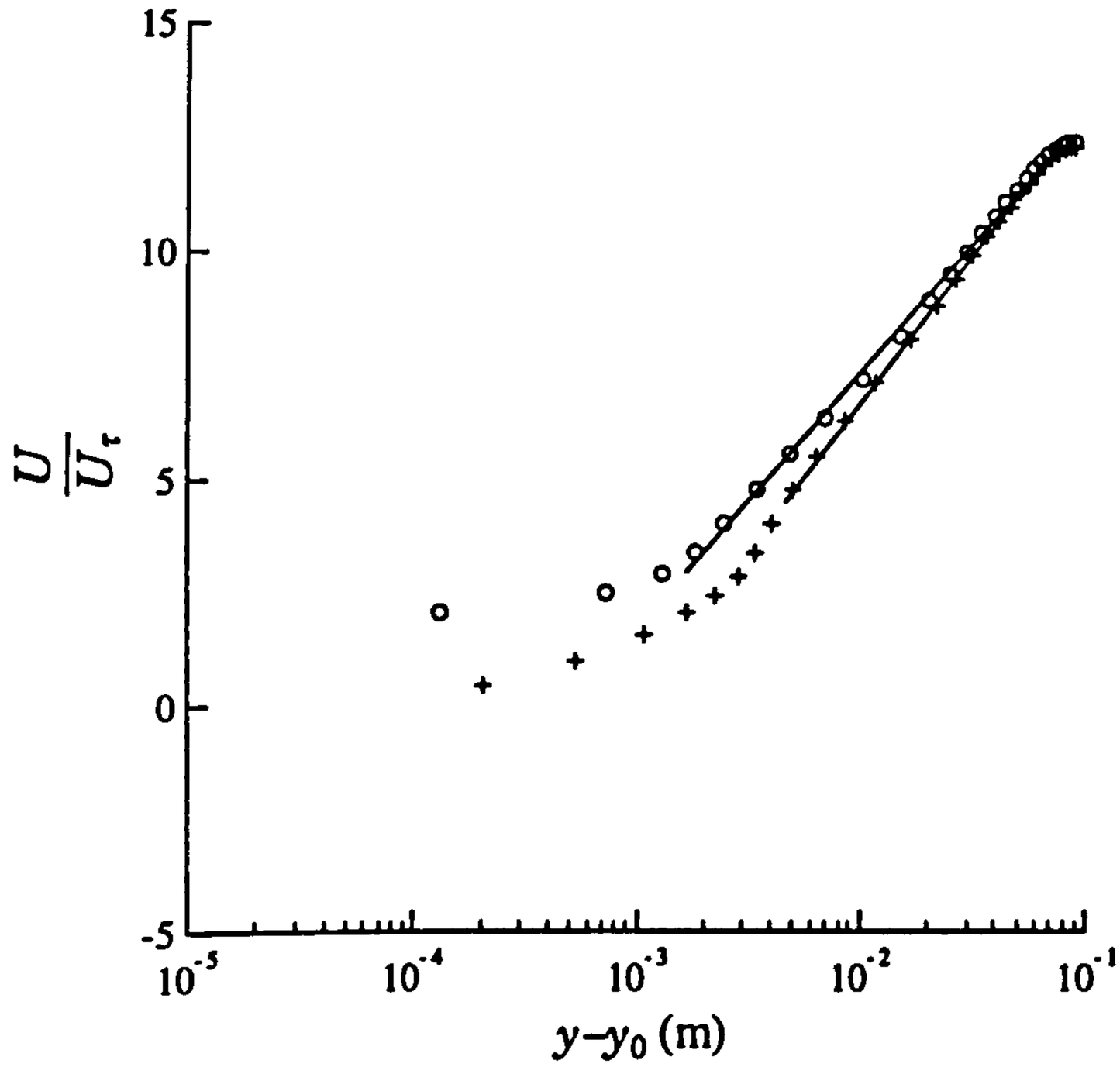


Figure 5-16e: Logarithmic velocity profile for RSA, relative depth of 30. (Data plotted every third cell for clarity).

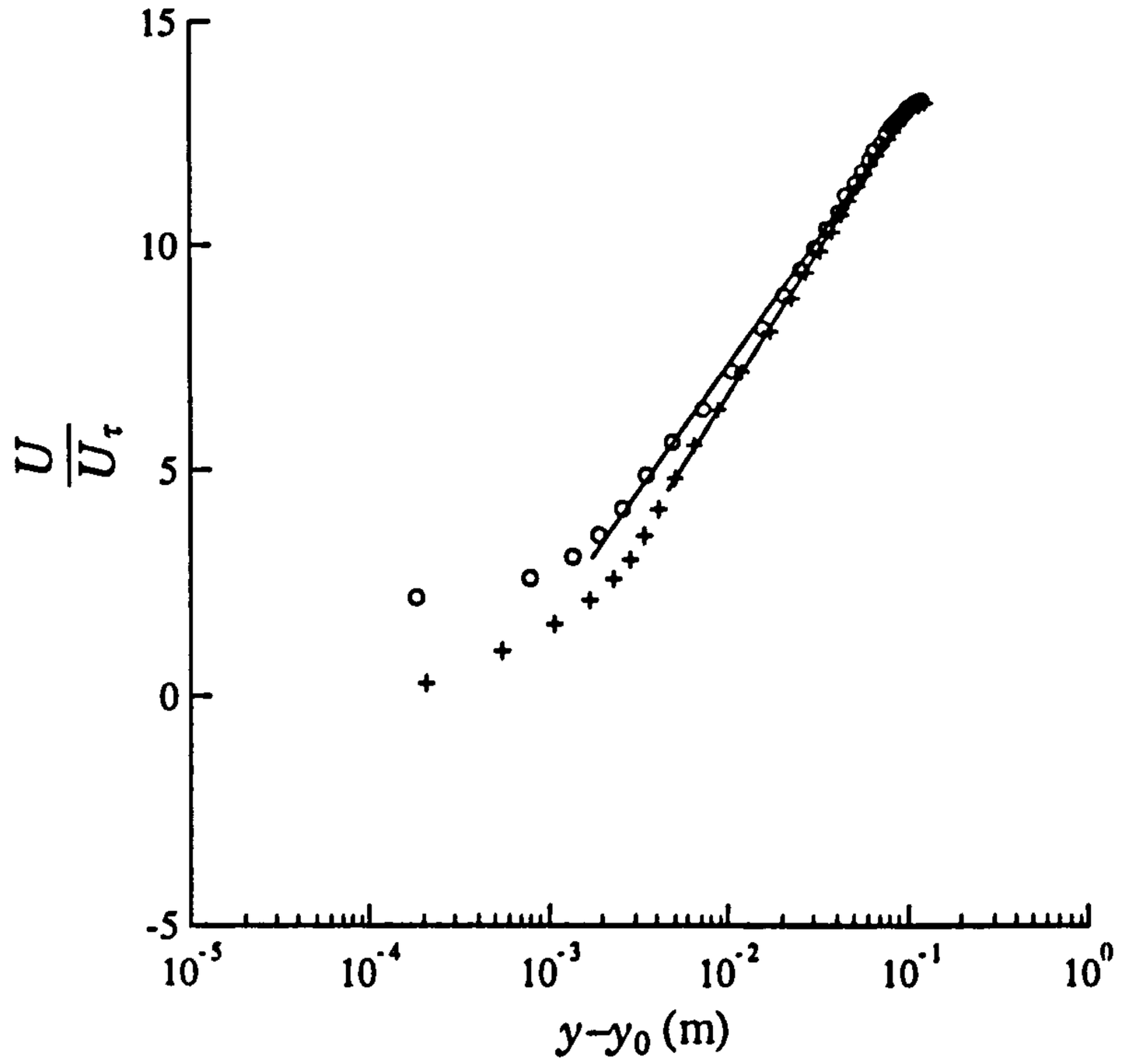


Figure 5-16f: Logarithmic velocity profile for RSA, relative depth of 40. (Data plotted every third cell for clarity).

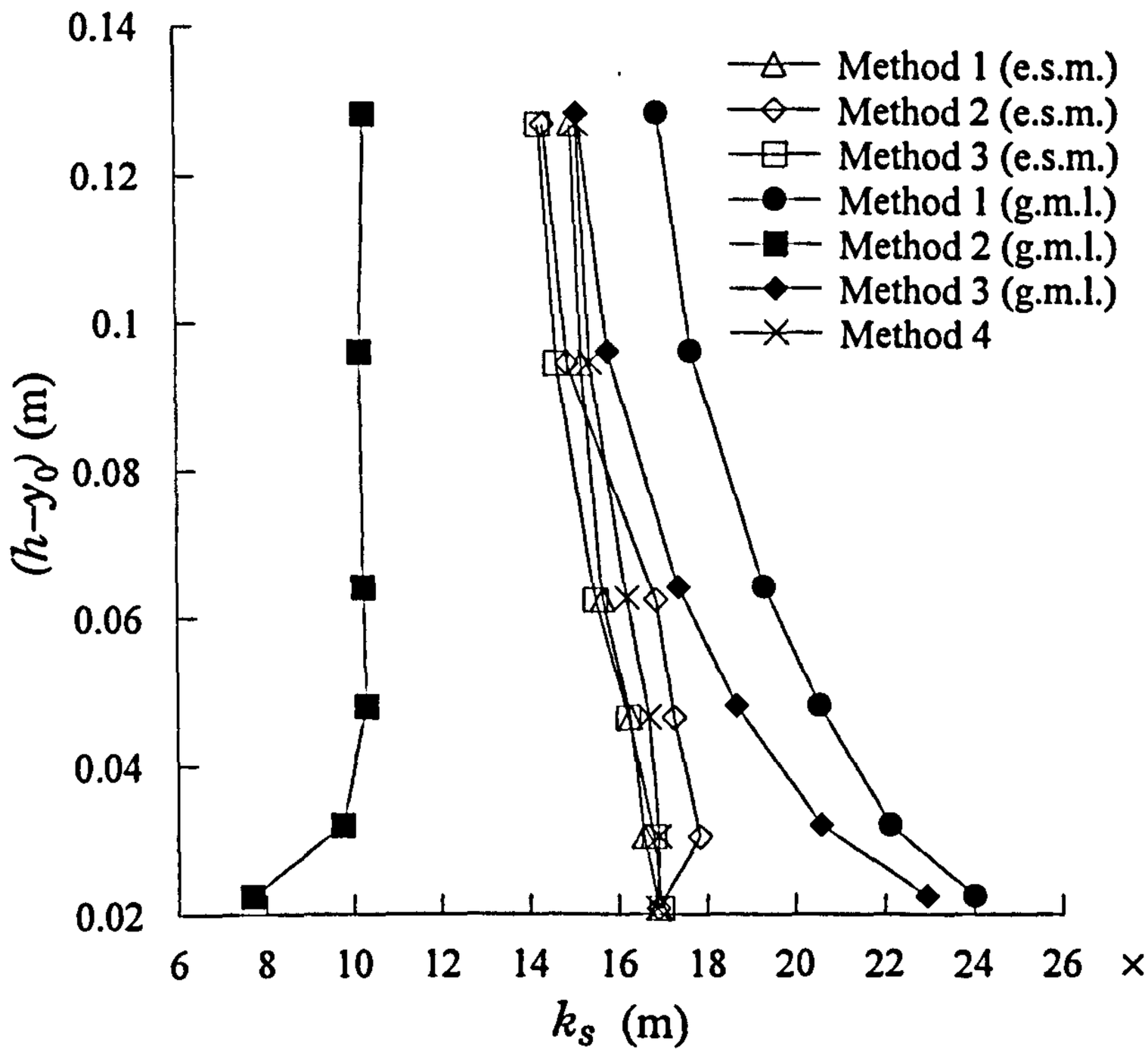


Figure 5-17a: k_s values obtained by velocity profile analysis for RSA

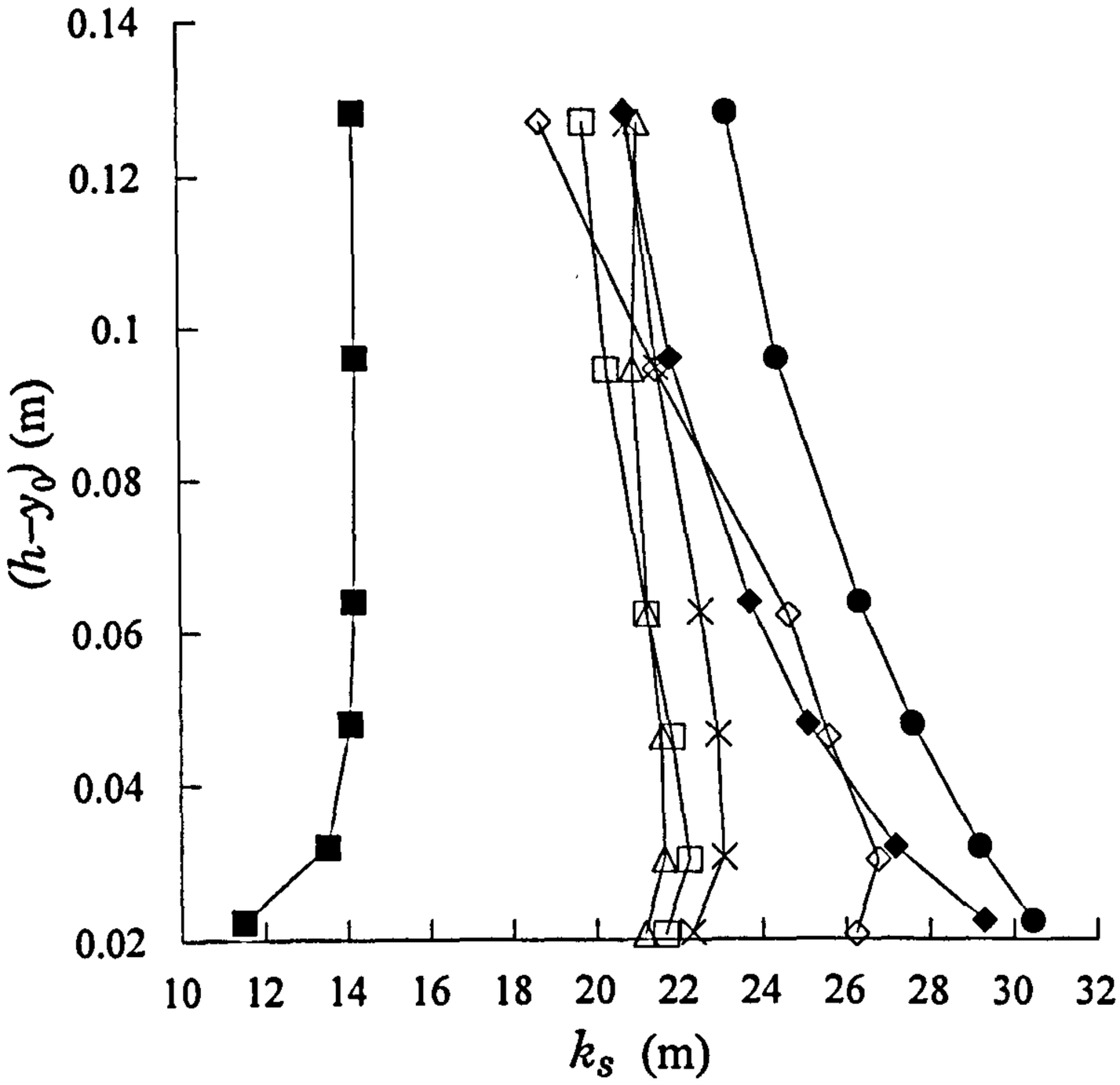


Figure 5-17b: k_s values obtained by velocity profile analysis for RSB

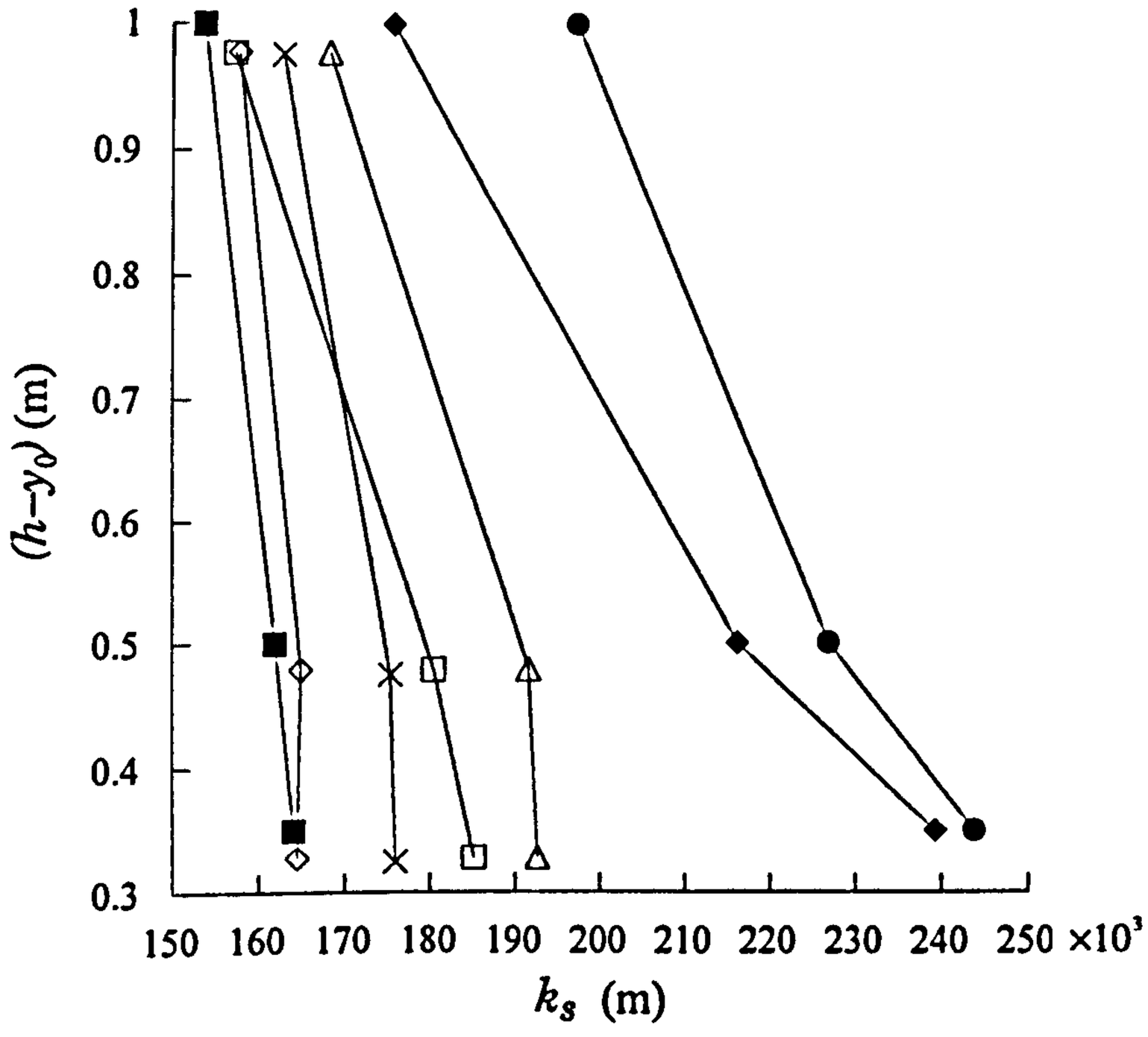


Figure 5-17c: k_s values obtained by velocity profile analysis for RCDA

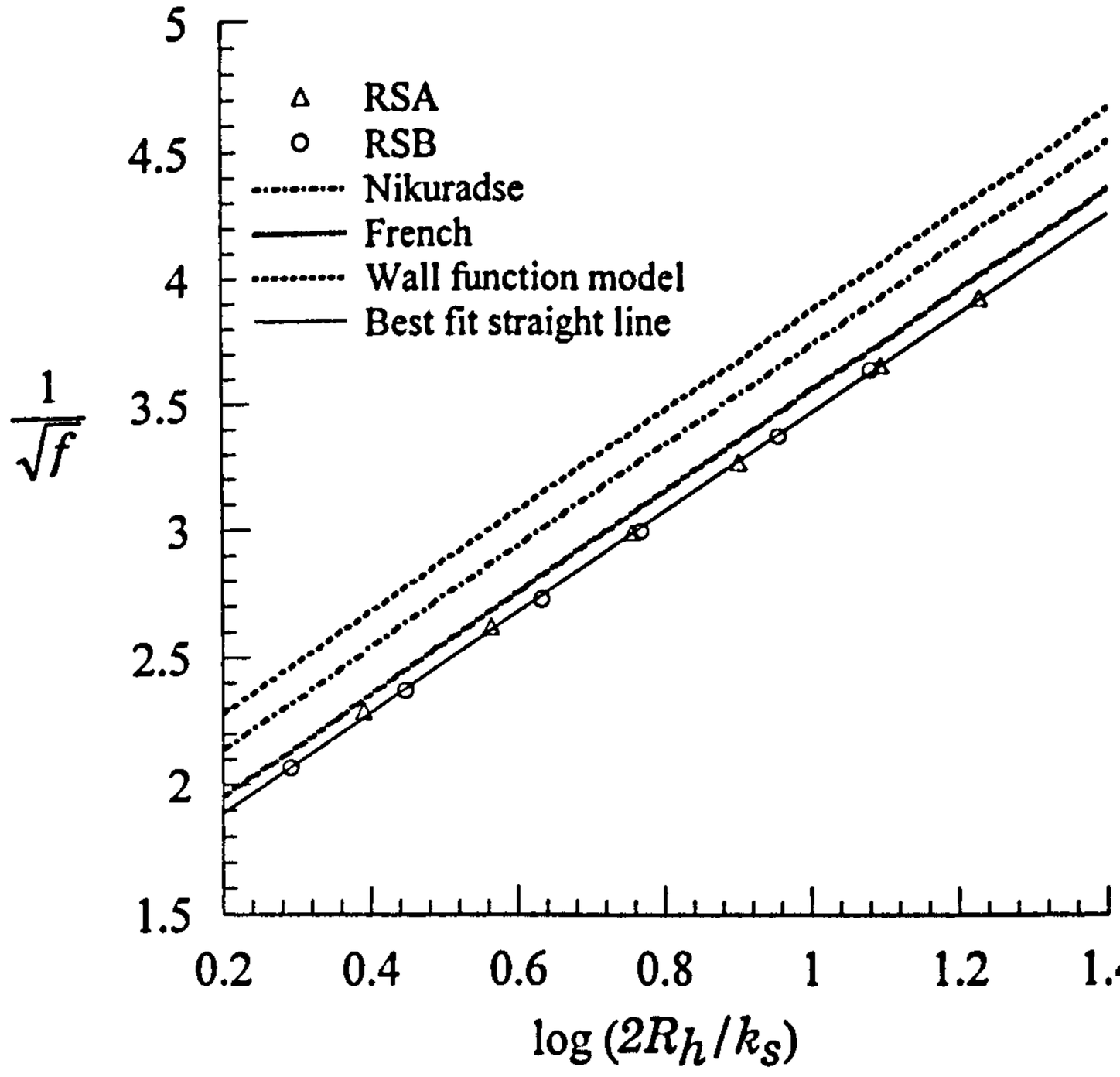


Figure 5-18: Resistance equations after Nikuradse (1933), French (1986), computational wall function model and two-layer model

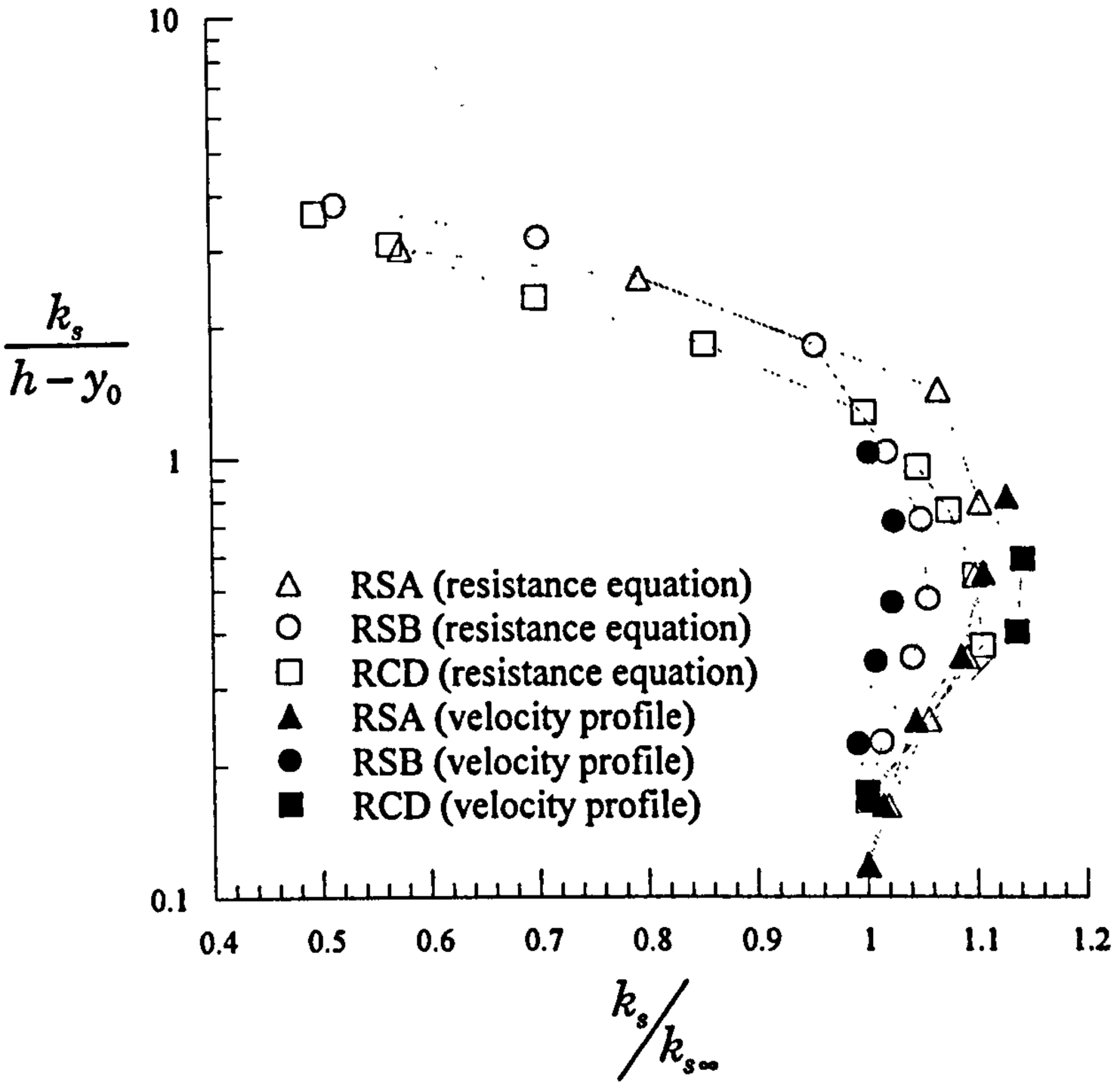


Figure 5-19: Non-dimensional variation of k_s obtained from velocity profile analysis and the resistance equation (Equation 5-16)

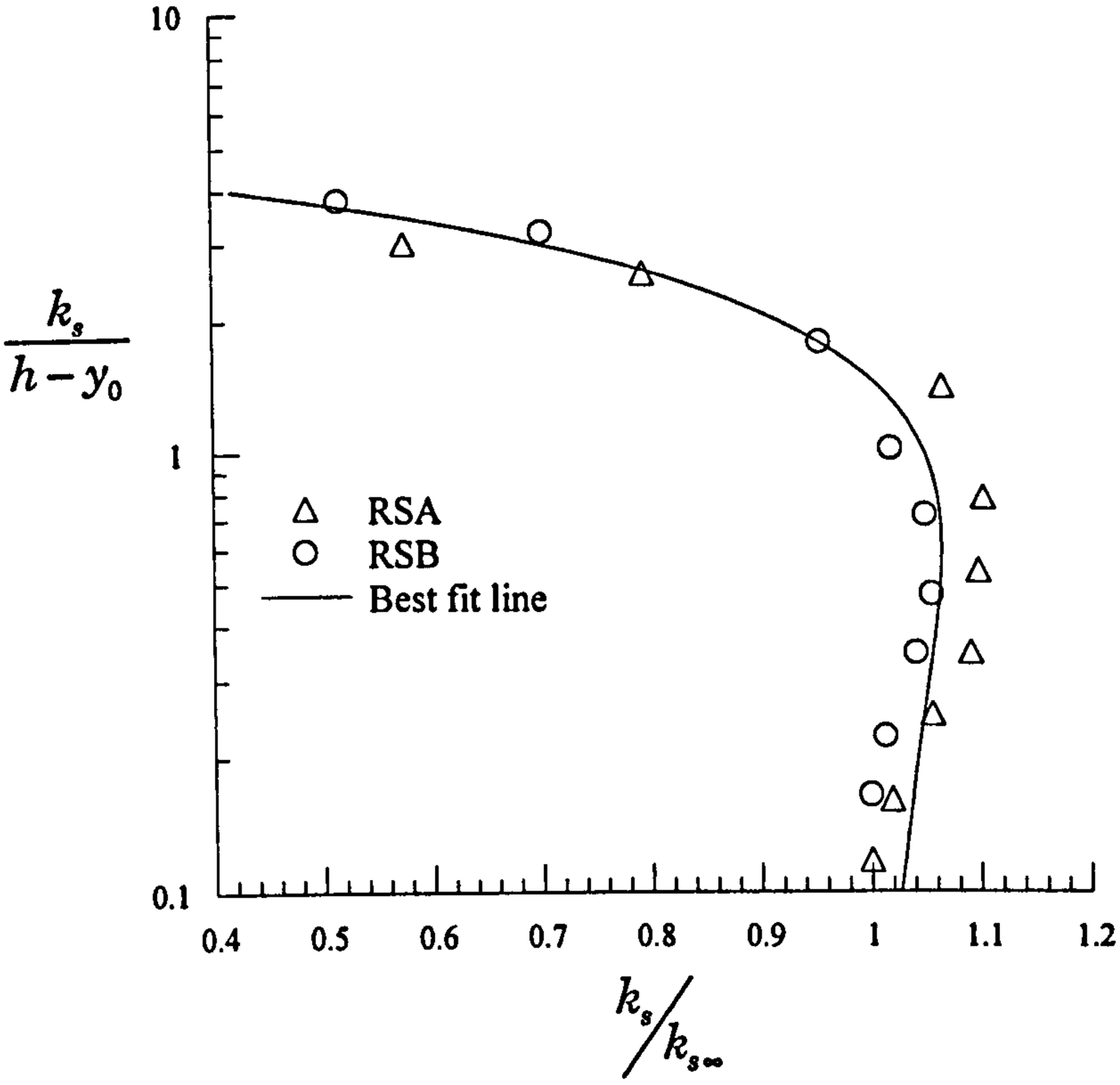


Figure 5-20: Non-dimensional variation of k_s for the roughness configurations RSA and RSB, and the best fit line (Equation 5-17) obtained by non-linear least squares regression

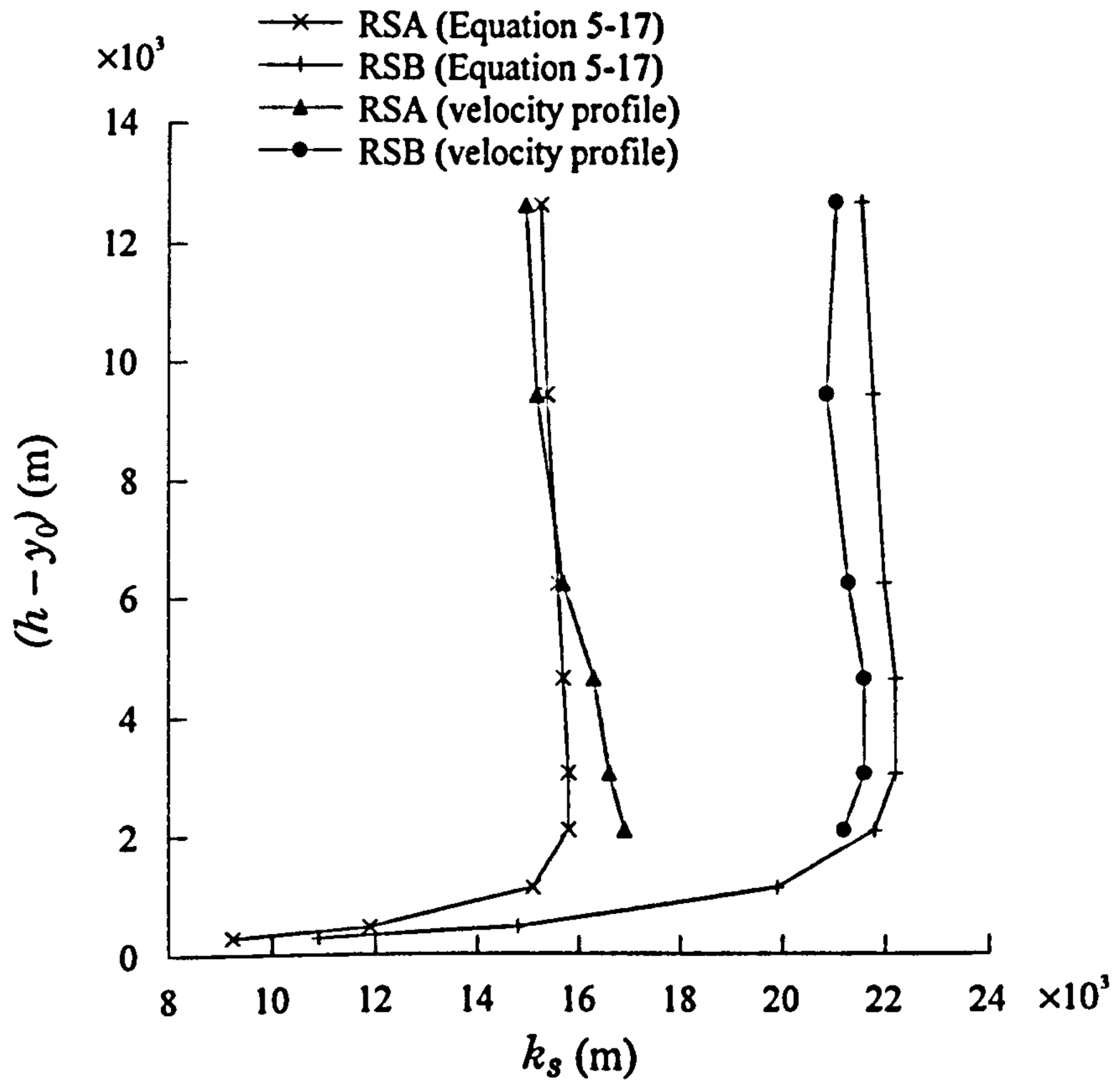


Figure 5-21a: Variation of k_s with depth of flow for roughness configurations RSA and RSB. k_s obtained from the non-dimensional k_s equation (Equation 5-17) and from velocity profile analysis

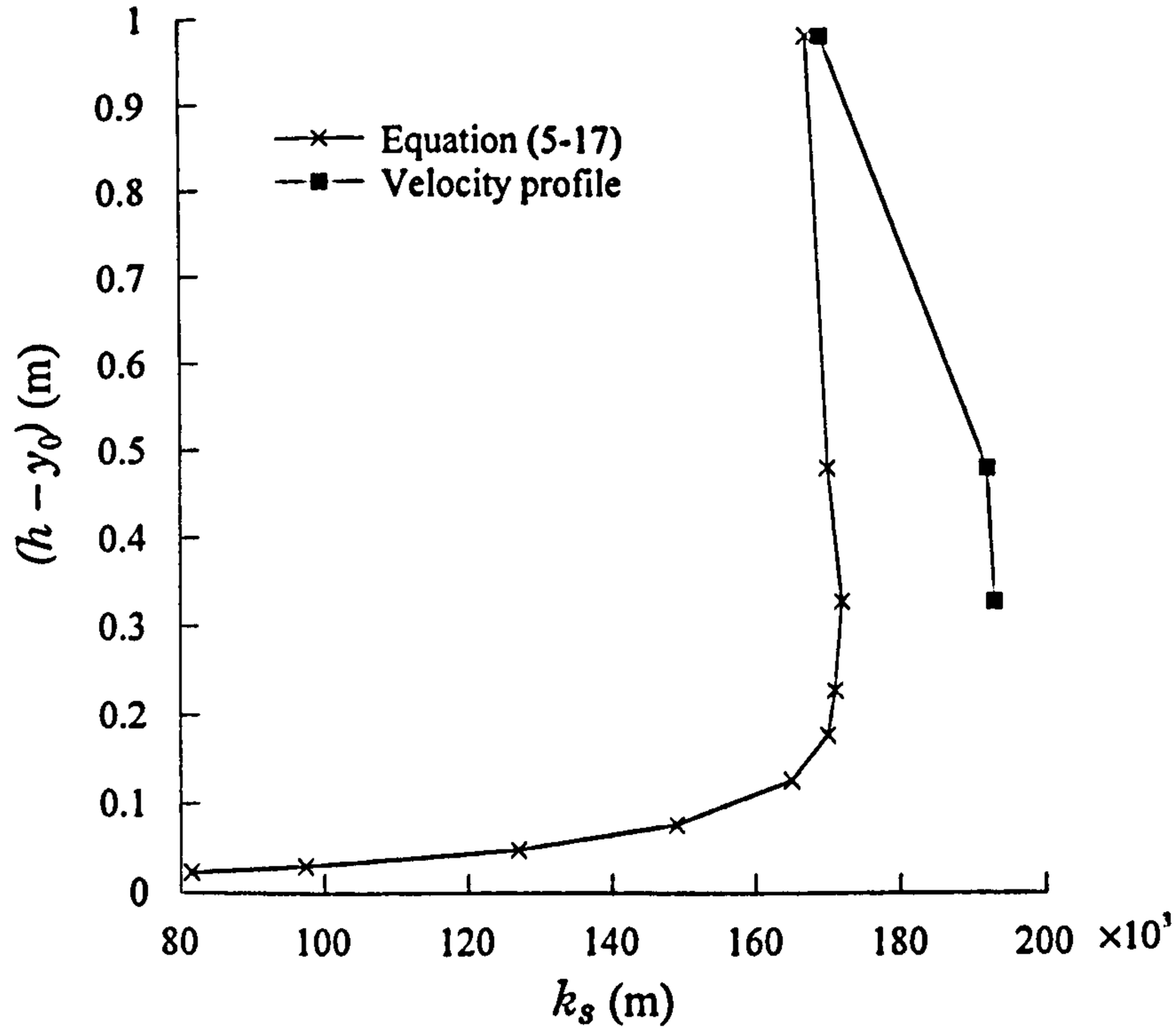


Figure 5-21b: Variation of k_s with depth of flow for roughness configuration RCDA

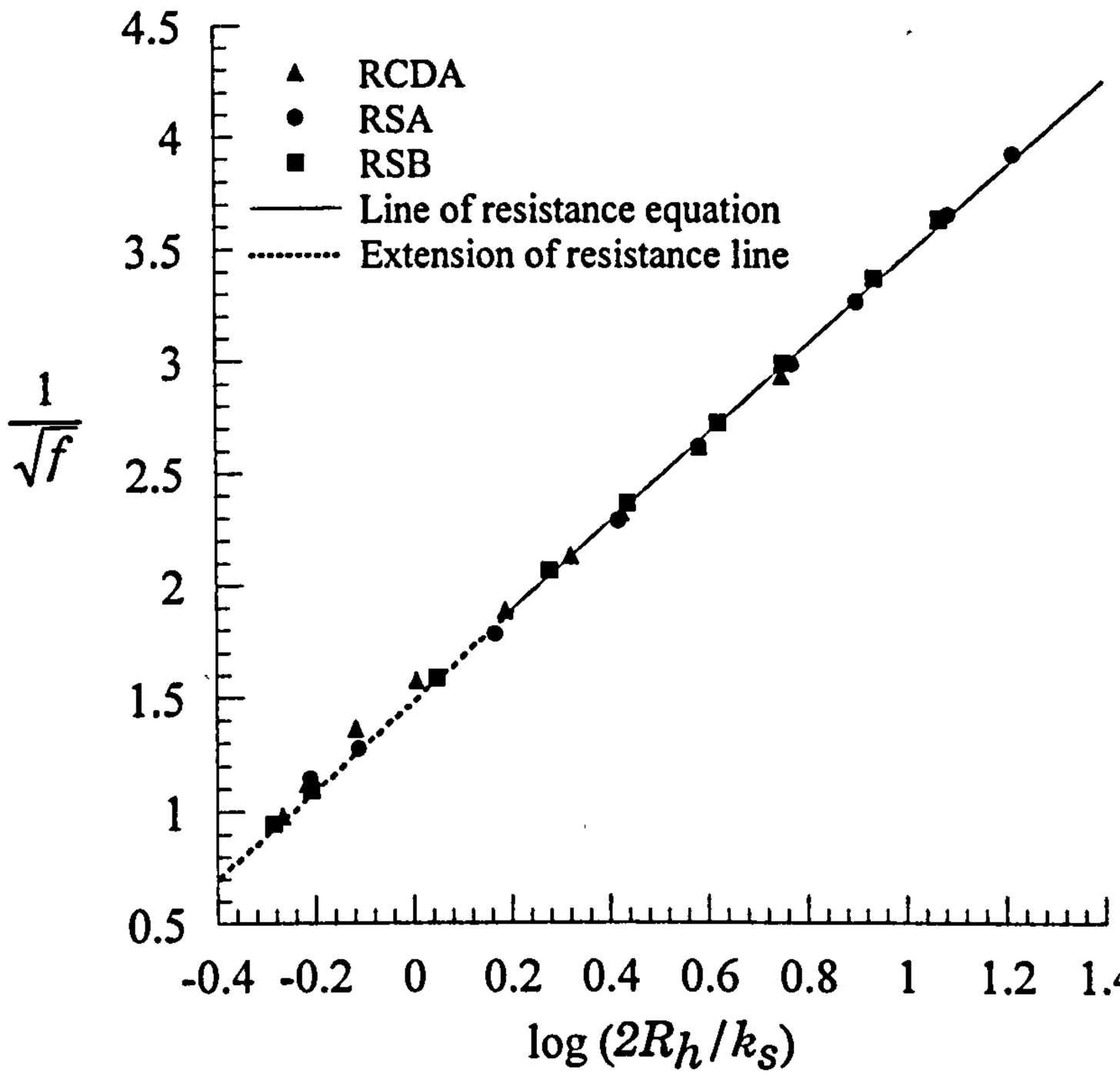


Figure 5-22: Resistance line (Equation 5-16) superimposed with values of k_s obtained from non-dimensional k_s equation (Equation 5-17)

Inputs									
case name	h/k_h	h	l	k_h	k_l	NX per period	NY	ν	$\frac{dP}{dx}$
		(m)	(m)	(m)	(m)			(m^2s^{-1})	(Nm^{-3})
RCDA	1.15	0.058	1.0	5.00E-02	5.00E-02	230	55	1.00E-06	2.64E+06
	1.25	0.063	1.0	5.00E-02	5.00E-02	230	57	1.00E-06	8.47E+05
	1.55	0.078	1.0	5.00E-02	5.00E-02	230	60	1.00E-06	1.28E+05
	2.05	0.103	1.0	5.00E-02	5.00E-02	230	74	1.00E-06	2.51E+04
	3.05	0.153	1.0	5.00E-02	5.00E-02	230	78	1.00E-06	3.67E+03
	4.05	0.203	1.0	5.00E-02	5.00E-02	230	79	1.00E-06	1.05E+03
	5.05	0.253	1.0	5.00E-02	5.00E-02	230	81	1.00E-06	4.32E+02
	7.05	0.353	1.0	5.00E-02	5.00E-02	230	83	1.00E-06	1.50E+02
	10.05	0.503	1.0	5.00E-02	5.00E-02	230	84	1.00E-06	3.03E+01
	20.05	1.003	1.0	5.00E-02	5.00E-02	230	88	1.00E-06	2.19E+00
RSA	1.5	0.005	0.06	3.20E-03	3.00E-04	122	50	1.17E-06	2.32E+08
	2	0.006	0.06	3.20E-03	3.00E-04	122	54	1.17E-06	4.10E+07
	4	0.013	0.06	3.20E-03	3.00E-04	122	64	1.17E-06	1.49E+06
	7	0.022	0.06	3.20E-03	3.00E-04	62	42	1.17E-06	1.60E+05
	10	0.032	0.06	3.20E-03	3.00E-04	62	48	1.17E-06	3.46E+04
	15	0.048	0.06	3.20E-03	3.00E-04	62	58	1.17E-06	7.00E+03
	20	0.064	0.06	3.20E-03	3.00E-04	62	67	1.17E-06	2.70E+03
	30	0.096	0.06	3.20E-03	3.00E-04	62	86	1.17E-06	6.44E+02
	40	0.128	0.06	3.20E-03	3.00E-04	62	105	1.17E-06	2.27E+02
RSB	1.5	0.005	0.04	3.20E-03	3.00E-04	96	50	1.17E-06	3.38E+08
	2	0.006	0.04	3.20E-03	3.00E-04	96	54	1.17E-06	5.96E+07
	4	0.013	0.04	3.20E-03	3.00E-04	96	64	1.17E-06	1.92E+06
	7	0.022	0.04	3.20E-03	3.00E-04	49	42	1.17E-06	1.63E+05
	10	0.032	0.04	3.20E-03	3.00E-04	49	48	1.17E-06	4.19E+04
	15	0.048	0.04	3.20E-03	3.00E-04	49	58	1.17E-06	9.25E+03
	20	0.064	0.04	3.20E-03	3.00E-04	49	67	1.17E-06	3.12E+03
	30	0.096	0.04	3.20E-03	3.00E-04	49	86	1.17E-06	7.41E+02
	40	0.128	0.04	3.20E-03	3.00E-04	49	105	1.17E-06	2.75E+02

Table 5-1a: Input parameters for simulating flow over roughness elements using a two-layer model

Results														
case name	h/k_h	q	Re	$\hat{U}$	$\left.\frac{l_r}{k_h}\right\|_1$	$\left.\frac{l_r}{k_h}\right\|_2$	y_0 (g.m.l.)	R_h (g.m.l.)	y_0 (e.s.m.)	y_0' (e.s.m.)	y_0 (l.v.p.)	R_h (e.s.m.)	τ_w	f
		(m^2s^{-1})		($m\,s^{-1}$)			(m)	(m)	(m)			(m)	(Nm^{-2})	
RCDA	1.15	0.464	4.62E+05	21.80	5.24	0.56	2.50E-03	0.055	3.55E-02	3.67E-02	/	0.022	1.52E+05	1.043
	1.25	0.466	4.64E+05	15.88	4.88	0.56	2.50E-03	0.060	3.30E-02	3.45E-02	/	0.029	5.29E+04	0.800
	1.55	0.463	4.61E+05	9.83	4.04	0.68	2.50E-03	0.075	2.92E-02	3.13E-02	/	0.048	9.95E+03	0.541
	2.05	0.464	4.62E+05	6.98	4.28	0.56	2.50E-03	0.100	2.68E-02	2.91E-02	/	0.076	2.57E+03	0.404
	3.05	0.463	4.61E+05	4.47	4.64	0.68	2.50E-03	0.150	2.53E-02	2.75E-02	/	0.127	5.59E+02	0.282
	4.05	0.464	4.62E+05	3.24	4.76	0.92	2.50E-03	0.200	2.47E-02	2.68E-02	/	0.178	2.13E+02	0.220
	5.05	0.471	4.69E+05	2.57	4.88	0.92	2.50E-03	0.250	2.42E-02	2.64E-02	/	0.228	1.09E+02	0.186
	7.05	0.539	5.37E+05	2.03	4.88	1.16	2.50E-03	0.350	2.43E-02	2.60E-02	2.81E-02	0.328	5.29E+01	0.146
	10.05	0.479	4.78E+05	1.23	5.12	1.40	2.50E-03	0.500	2.33E-02	2.57E-02	2.76E-02	0.479	1.52E+01	0.116
	20.05	0.464	4.62E+05	0.57	5.50	1.76	2.50E-03	1.000	2.52E-02	2.53E-02	2.71E-02	0.977	2.19E+00	0.076
RSA	1.5	0.236	2.02E+05	83.61	4.08	0.80	1.60E-05	4.78E-03	1.96E-03	2.03E-03	/	0.003	1.11E+06	0.767
	2	0.228	1.95E+05	58.67	5.30	0.80	1.60E-05	6.38E-03	1.80E-03	1.87E-03	/	0.005	2.62E+05	0.613
	4	0.228	1.95E+05	26.24	6.27	1.23	1.60E-05	1.28E-02	1.68E-03	1.71E-03	/	0.011	1.91E+04	0.315
	7	0.245	2.09E+05	14.85	5.67	2.27	1.60E-05	2.24E-02	1.64E-03	1.66E-03	1.65E-03	0.021	3.58E+03	0.191
	10	0.230	1.97E+05	9.46	5.67	2.73	1.60E-05	3.20E-02	1.64E-03	1.64E-03	1.56E-03	0.030	1.11E+03	0.146
	15	0.223	1.91E+05	5.93	5.67	2.73	1.60E-05	4.80E-02	1.62E-03	1.63E-03	1.49E-03	0.046	3.36E+02	0.112
	20	0.236	2.02E+05	4.62	5.67	2.73	1.60E-05	6.40E-02	1.67E-03	1.62E-03	1.47E-03	0.062	1.73E+02	0.094
	30	0.240	2.06E+05	3.06	5.67	3.23	1.60E-05	9.60E-02	1.58E-03	1.61E-03	1.46E-03	0.094	6.18E+01	0.075
	40	0.238	2.04E+05	2.24	5.67	3.23	1.60E-05	1.28E-01	1.52E-03	1.61E-03	1.42E-03	0.127	2.91E+01	0.065
RSB	1.5	0.234	2.00E+05	96.16	3.86	0.55	2.40E-05	4.78E-03	1.97E-03	2.03E-03	/	0.003	1.62E+06	1.124
	2	0.233	2.00E+05	66.93	4.86	0.48	2.40E-05	6.38E-03	1.82E-03	1.87E-03	/	0.005	3.82E+05	0.841
	4	0.230	1.96E+05	27.62	5.33	0.98	2.40E-05	1.28E-02	1.71E-03	1.71E-03	/	0.011	2.46E+04	0.398
	7	0.223	1.90E+05	13.94	4.73	2.27	2.40E-05	2.24E-02	1.67E-03	1.66E-03	1.51E-03	0.021	3.66E+03	0.235
	10	0.229	1.96E+05	9.65	4.73	2.27	2.40E-05	3.20E-02	1.66E-03	1.64E-03	1.41E-03	0.030	1.34E+03	0.179
	15	0.234	2.00E+05	6.34	4.73	2.73	2.40E-05	4.80E-02	1.67E-03	1.63E-03	1.36E-03	0.046	4.44E+02	0.135
	20	0.232	1.99E+05	4.63	5.20	3.23	2.40E-05	6.40E-02	1.69E-03	1.62E-03	1.34E-03	0.062	2.00E+02	0.112
	30	0.238	2.03E+05	3.08	5.20	3.23	2.40E-05	9.60E-02	1.60E-03	1.61E-03	1.35E-03	0.094	7.11E+01	0.088
	40	0.244	2.09E+05	2.33	5.20	3.73	2.40E-05	1.28E-01	1.26E-03	1.61E-03	1.35E-03	0.127	3.53E+01	0.076

Table 5-1b: Results for simulating flow over roughness elements using a two-layer model

Results (continued)								
case name	h/k_h	Driving force	Resistive force	% error in forces	Viscous drag (% of total drag)	Form drag (% of total drag)	$\frac{\partial \tau'_{yx}}{\partial y}$	% error in $\frac{\partial \tau'_{yx}}{\partial y}$ and $\frac{dP}{dx}$
		(N)	(N)				(Nm^{-3})	
RCDA	1.15	1.45E+05	-1.45E+05	0.007	-0.03	99.79	2.63E+06	0.03
	1.25	5.08E+04	-5.08E+04	0.006	0.03	99.76	8.47E+05	0.02
	1.55	9.63E+03	-9.64E+03	-0.053	0.38	99.57	1.28E+05	-0.03
	2.05	2.51E+03	-2.51E+03	-0.035	0.79	99.28	2.51E+04	-0.06
	3.05	5.50E+02	-5.51E+02	-0.080	1.02	99.16	3.67E+03	-0.04
	4.05	2.11E+02	-2.11E+02	-0.082	1.11	99.15	1.05E+03	-0.14
	5.05	1.08E+02	-1.08E+02	-0.109	1.21	99.10	4.32E+02	-0.06
	7.05	5.25E+01	-5.25E+01	0.059	1.27	99.11	1.50E+02	-0.01
	10.05	1.52E+01	-1.52E+01	-0.102	1.23	99.32	3.04E+01	-0.16
	20.05	2.19E+00	-2.18E+00	0.177	0.88	100.06	2.19E+00	0.02
RSA	1.5	6.65E+04	-6.65E+04	0.058	0.14	99.86	2.32E+08	0.01
	2	1.57E+04	-1.57E+04	0.040	0.28	99.72	4.10E+07	-0.02
	4	1.14E+03	-1.14E+03	0.018	0.24	99.76	1.49E+06	-0.01
	7	2.15E+02	-2.15E+02	0.022	0.16	99.83	1.60E+05	0.00
	10	6.64E+01	-6.64E+01	0.021	-0.07	100.07	3.46E+04	0.01
	15	2.02E+01	-2.02E+01	-0.023	-0.44	100.42	7.00E+03	-0.04
	20	1.04E+01	-1.04E+01	0.067	-0.79	100.77	2.70E+03	0.05
	30	3.71E+00	-3.71E+00	-0.055	-1.38	101.34	6.44E+02	-0.05
	40	1.75E+00	-1.75E+00	-0.098	-1.85	101.78	2.27E+02	-0.10
RSB	1.5	6.46E+04	-6.45E+04	0.095	-0.01	100.00	3.38E+08	0.01
	2	1.52E+04	-1.52E+04	0.062	-0.03	100.02	5.96E+07	0.02
	4	9.80E+02	-9.80E+02	0.028	-0.35	100.35	1.92E+06	0.00
	7	1.46E+02	-1.46E+02	0.035	-0.59	100.58	1.63E+05	-0.01
	10	5.36E+01	-5.36E+01	0.029	-0.99	100.98	4.19E+04	-0.02
	15	1.78E+01	-1.78E+01	0.029	-1.55	101.54	9.25E+03	0.04
	20	7.99E+00	-7.99E+00	0.042	-2.10	102.07	3.12E+03	0.02
	30	2.84E+00	-2.85E+00	-0.079	-2.93	102.89	7.42E+02	-0.11
	40	1.41E+00	-1.42E+00	-0.332	-3.61	103.51	2.76E+02	-0.25

Table 5-1c: Results for simulating flow over roughness elements using a two-layer model (continued)

Constants:

ρ	[kg m^{-3}]	998.2
ν	[m^2 s^{-1}]	1.17E-06
Re		7.00E+06
k_s	[m]	0.001

Notes:

$$\tau_w/\rho\big|_1 = \frac{dP}{dx} \times h \Big/ \rho$$

$$\tau_w/\rho\big|_2 = \left(U_{IY=1} \Big/ 8.271 + 5.612 \log y/k_s \right)$$

Inputs:				Initial estimates:					Results:							
h [m]	h/k_s	$\overline{U}$ [m s^{-1}]	$y_{1st\ cell}$ [m]	f	τ_w [N m^{-2}]	Y^+	Re_r	$\frac{dP}{dx}$ [N m^{-3}]	$\frac{dP}{dx}$ [N m^{-3}]	$\tau_w/\rho\big\|_1$ [m s^{-1}]	$\tau_w/\rho\big\|_2$ [m s^{-1}]	% error in τ_w/ρ	Y^+	Re_r	**Re**	f
0.1	100	40.95	5.00E-05	2.49E-02	5202	97.56	1951	5.20E+04	5.17E+04	5.1841	5.1792	-0.10	88.38	1946	7.10E+06	2.40E-02
0.2	200	20.48	1.00E-04	2.07E-02	1085	89.10	891	5.42E+03	4.85E+03	0.9712	0.9724	0.12	84.48	842	6.72E+06	2.01E-02
0.3	300	13.65	1.50E-04	1.88E-02	436.7	84.80	565	1.46E+03	1.36E+03	0.4084	0.4088	0.10	82.10	546	6.86E+06	1.82E-02
0.4	400	10.24	2.00E-04	1.76E-02	229.6	81.99	410	5.74E+02	5.50E+02	0.2205	0.2208	0.13	80.12	401	6.95E+06	1.71E-02
0.5	500	8.19	2.50E-04	1.67E-02	139.7	79.94	320	2.79E+02	2.73E+02	0.1366	0.1368	0.14	78.78	316	7.01E+06	1.62E-02
0.6	600	6.83	3.00E-04	1.60E-02	93.17	78.34	261	1.55E+02	1.54E+02	0.0923	0.0924	0.10	77.64	260	7.05E+06	1.56E-02
0.7	700	5.85	3.50E-04	1.55E-02	66.19	77.03	220	9.46E+01	9.45E+01	0.0662	0.0663	0.13	76.65	220	7.09E+06	1.51E-02
0.8	800	5.12	4.00E-04	1.51E-02	49.24	75.93	190	6.16E+01	6.20E+01	0.0497	0.0498	0.14	75.80	191	7.11E+06	1.47E-02
0.9	900	4.55	4.50E-04	1.47E-02	37.95	74.99	167	4.22E+01	4.27E+01	0.0385	0.0386	0.16	75.03	168	7.13E+06	1.43E-02
1	1000	4.10	5.00E-04	1.44E-02	30.07	74.17	148	3.01E+01	3.06E+01	0.0307	0.0307	-0.06	74.35	150	7.15E+06	1.41E-02

Table 5-2: Input parameters and results for simulations using a wall function

h	A	B
0.1	8.735	6.016
0.2	8.609	6.013
0.3	8.537	6.014
0.4	8.486	6.014
0.5	8.447	6.014
0.6	8.415	6.014
0.7	8.388	6.014
0.8	8.364	6.013
0.9	8.343	6.013
1.0	8.325	6.013
Average	8.465	6.014
Nikuradse	8.480	5.750
Jayatilleke	8.271	5.616

Table 5-3: Coefficients in the logarithmic velocity profile obtained using a wall function. Comparisons against Nikuradse (1933) and Jayatilleke (1969)

		NX per period	NY	dP/dx	q	$\hat{U}$	y_0 (e.s.m.)	f	k_s
RCDA10	original grid	230	84	30.3	0.479	1.232	2.33E-02	0.134	0.189
	coarse grid	125	44	30.3	0.468	1.205	2.53E-02	0.141	0.202
% error					2.5	2.2	-8.4	-5.0	-6.8
RSA7	original grid	62	42	1.60E-05	0.245	14.849	1.64E-03	0.240	1.71E-02
	fine grid	122	73	1.60E-05	0.243	14.813	1.62E-03	0.243	1.73E-02
% error					-0.6	-0.2	-1.0	1.2	1.3

Table 5-4: Grid independence test for two-layer model

	dP/dx	q	Re	f	y_0 (e.s.m.)	k_s
RSA40 (i)	2.27E+02	0.238	203648	0.0675	1.47E-03	0.0152
RSA40 (ii)	7.00E+02	0.412	351823	0.0697	1.75E-03	0.0162
% change	208.4	72.8	72.8	3.2	19.0	6.4

Table 5-5: Reynolds number independence test for two-layer model

		y_0 = g.m.l.							y_0 = e.s.m.							
case name	h/k_h	$\frac{y_{l,1}}{k_h}$	$\frac{y_{l,2}}{h-y_0}$	S_e	Grad	$k_s(1)$	$k_s(2)$	$k_s(3)$	$\frac{y_{lr1}}{k_h}$	$\frac{y_{lr2}}{h-y_0}$	S_e	Grad	$k_s(1)$	$k_s(2)$	$k_s(3)$	$k_s(4)$
RCDA	7.05	1.984	0.525	0.013	6.723	2.44E-01	1.64E-01	2.39E-01	1.046	0.694	0.031	5.936	1.93E-01	1.64E-01	1.85E-01	1.76E-01
	10.1	1.968	0.588	0.011	6.693	2.27E-01	1.62E-01	2.16E-01	1.044	0.705	0.025	5.945	1.92E-01	1.65E-01	1.80E-01	1.75E-01
	20.1	1.908	0.636	0.016	6.625	1.98E-01	1.54E-01	1.76E-01	1.044	0.751	0.079	5.779	1.69E-01	1.58E-01	1.57E-01	1.63E-01
RSA	7	1.462	0.540	0.055	6.913	2.40E-02	7.69E-03	2.29E-02	1.023	0.653	0.036	5.617	1.69E-02	1.69E-02	1.70E-02	1.69E-02
	10	1.462	0.650	0.051	6.599	2.21E-02	9.76E-03	2.06E-02	1.023	0.798	0.027	5.536	1.66E-02	1.78E-02	1.68E-02	1.69E-02
	15	1.462	0.673	0.029	6.503	2.05E-02	1.03E-02	1.87E-02	1.023	0.767	0.069	5.547	1.63E-02	1.73E-02	1.62E-02	1.67E-02
	20	1.462	0.669	0.024	6.473	1.93E-02	1.02E-02	1.74E-02	1.023	0.714	0.111	5.523	1.57E-02	1.69E-02	1.55E-02	1.62E-02
	30	1.462	0.674	0.032	6.434	1.77E-02	1.02E-02	1.58E-02	1.023	0.721	0.138	5.643	1.52E-02	1.49E-02	1.46E-02	1.54E-02
	40	1.462	0.637	0.040	6.400	1.70E-02	1.03E-02	1.51E-02	1.023	0.671	0.157	5.674	1.50E-02	1.44E-02	1.43E-02	1.51E-02
RSB	7	1.462	0.540	0.030	6.726	3.05E-02	1.15E-02	2.93E-02	1.023	0.654	0.013	5.395	2.12E-02	2.62E-02	2.17E-02	2.23E-02
	10	1.462	0.650	0.023	6.541	2.92E-02	1.35E-02	2.72E-02	1.023	0.799	0.052	5.381	2.16E-02	2.68E-02	2.22E-02	2.31E-02
	15	1.462	0.673	0.017	6.481	2.76E-02	1.41E-02	2.51E-02	1.023	0.768	0.110	5.414	2.16E-02	2.56E-02	2.18E-02	2.29E-02
	20	1.462	0.669	0.025	6.454	2.64E-02	1.42E-02	2.37E-02	1.023	0.714	0.145	5.431	2.13E-02	2.47E-02	2.13E-02	2.25E-02
	30	1.462	0.674	0.037	6.412	2.44E-02	1.42E-02	2.18E-02	1.023	0.721	0.163	5.577	2.09E-02	2.15E-02	2.03E-02	2.15E-02
	40	1.462	0.637	0.044	6.386	2.32E-02	1.42E-02	2.07E-02	1.023	0.670	0.140	5.787	2.11E-02	1.87E-02	1.98E-02	2.08E-02

Notes:

$y_{l,1/2} = y$ boundaries of logarithmic region

S_e = Standard error of estimate

Grad = Gradient of the plot U/U_τ against $\log(h-y_0)$

k_s = Equivalent sand roughness. The number in brackets denotes the method of calculation (Section 3.4.1)

Table 5-6: Velocity profile analysis for two-layer model

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

6. OTHER APPLICATIONS

6.1 Introduction

During the course of this study some of the techniques developed in Chapter 3 were utilised in two applications of common engineering interest, but which were not directly related to the study of flow over rough boundaries. The two applications were three-dimensional flow in a smooth rectangular duct and the heat transfer in a counterflow heat exchanger. The former application was performed in the early stages of this project primarily as a means of validating the computational code using a simple duct geometry. Calculations on a plate-fin heat exchanger were carried out as part of a feasibility study into the application of CFD for heat flow calculations for Serck Heat Transfer Ltd. (Birmingham).

6.2 Simulations of Smooth Rectangular Ducts

Computational experiments were performed on three-dimensional flows in a smooth walled rectangular duct in order to:

1. Confirm that PHOENICS controlled cyclic boundary conditions could be used to achieve fully developed flow.

2. Validate the two-layer turbulence model against theoretical and experimental results.

3. Investigate the distribution of the boundary shear stress between the wall and bed elements.

4. Investigate the side wall effect upon centreline bed shear stress.

Items 3 and 4 were opportunistic experiments unconnected with the primary investigation into rough ducts.

6.2.1 Description of Computational Experiments

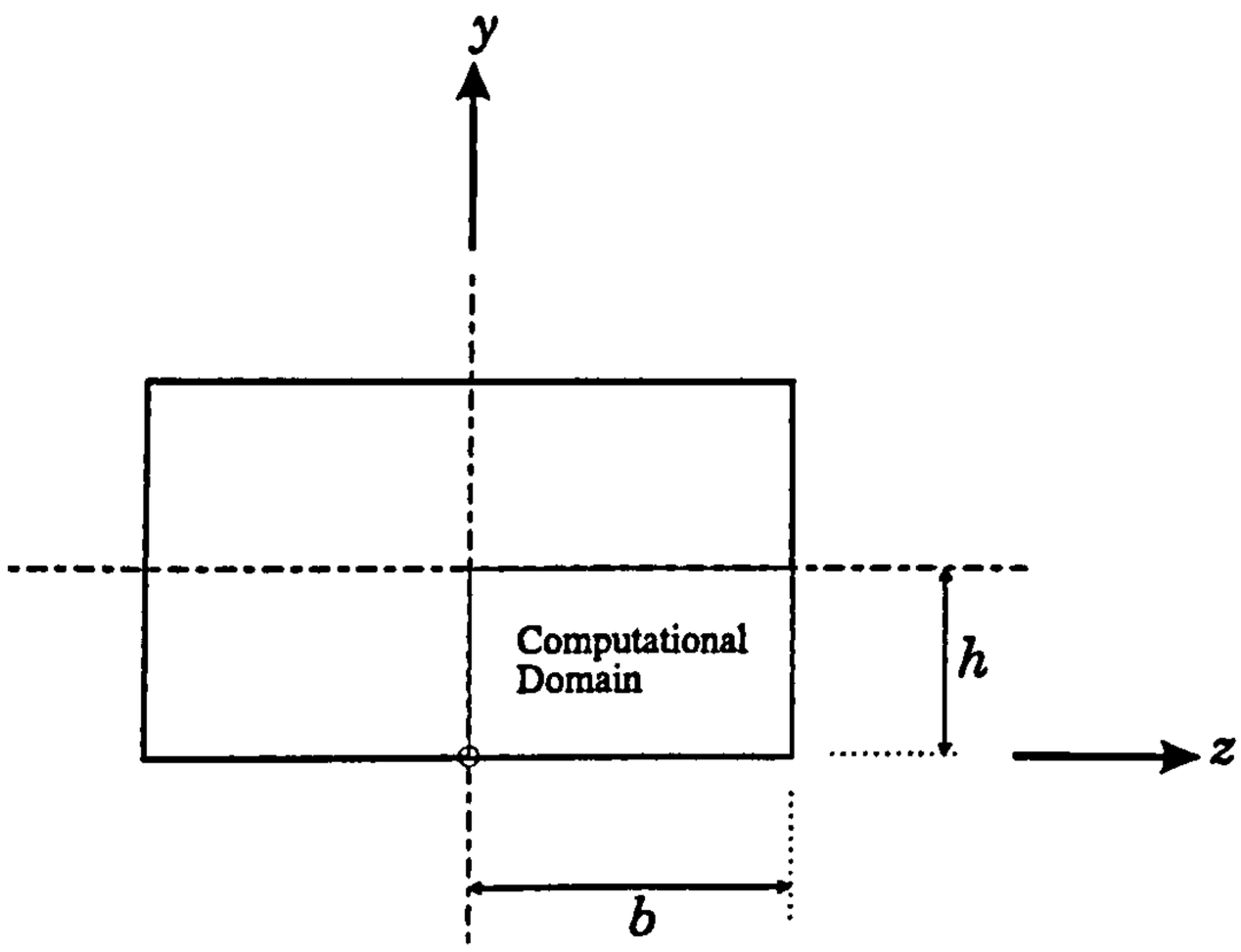


Figure 6-1: Cross-section of smooth duct

Calculations were performed on a ¼ segment of a three-dimensional duct as shown in Figure 6-1, utilising the horizontal and vertical planes of symmetry. Seventeen aspect ratios were computed from $1.0 \leq {}^{b}/_{h} \leq 50.008$, chosen to correspond to the physical experiments conducted by Knight and Patel (1985) and Rhodes and Knight (1994). All of the tests modelled turbulent air flow at a Reynolds number $\left({}^{4R_h\overline{U}}/_{\nu}\right)$ of between 2.95×10^4 and 1.07×10^5. The test parameters for each case are shown in Table 6-1. Fully developed flow was achieved in all cases by using PHOENICS controlled cyclic boundary conditions (Section 3.2.3) in which the streamwise pressure gradient was modelled by a source of momentum flux per unit volume within each cell of the domain. The correct Reynolds number was achieved by adjusting the momentum source after each sweep, so that the calculated cross-section averaged velocity matched the required average velocity. This was performed using GROUND coding (Appendix 12) which was also used for preliminary analysis to calculate the shear stresses and force balance.

The rate of convergence was monitored by the examination of residuals within the finite volume equations (Section 2.3.2) and of spot values of the flow variables at specific points in the domain. The criterion for terminating the solution procedure was that the driving force due to the momentum source and the resistive force due to boundary shear should balance to an acceptable degree of accuracy. The maximum discrepancy in all of the computational tests performed was 0.078%. Additionally it was confirmed that

further iterations did not produce a significant change in the value of percentage shear force on the side wall ($\%SF_w$).

6.2.2 Computational Grid

In the two transverse directions (y and z) a geometrically expanding grid of factor ≈ 1.24 was used to model the high gradients adjacent to the duct walls. The core of the duct was covered by a uniform mesh of cell size equal to $1/20$ of the duct dimension. The procedure used to calculate the grid is described below.

1. The size of the uniform cells in the y direction was calculated from $h/20$.

2. The size of the first cell adjacent to the wall was set to be 1.56×10^{-4} m for all tests and was assumed to have its cell centre located at approximately $Y^+ = 1$. The exact values of Y^+ for the wall cell on the bed centre line for each case are shown in Table 6-1. Note that $Y^+ \propto \sqrt{U_{\text{wall cell}}}$ and therefore the maximum value of Y^+ occurred at the position of maximum wall velocity, i.e. the bed centreline.

3. A FORTRAN program was used to calculate the geometric expansion factor and the number of cells required in the geometric portion of the grid. This was performed by choosing the number of cells which would be required if the complete duct height were to be filled using a geometric grid with an expansion factor of ≈1.24. Manual inspection of the resulting cell sizes showed where the *geometrically expanded* cell size exceeded the required *uniform cell* size and thus the number of geometric cells required, together with the distance over which the geometric grid was applied. The results of this procedure are given in Table 6-2.

4. Using PIL commands in the Q1 file (Appendix 12), the grid for the y direction was then generated.

5. The z direction grid was calculated within the Q1 file by utilising the same geometric expansion, but extending (or truncating) the geometric region of the grid to the point where the last geometric cell was only just smaller than $1/20 \times b$.

While developing the coding, ten slabs were used in the streamwise (x direction) to confirm fully developed flow. The x dimension of each slab was varied with each simulation, the total "period" length of the duct being made equal to the full duct height. For the tests described here however, only two streamwise slabs were used, both having the same x dimension as the size of the uniform cells employed in the y direction.

Grid independence test

A grid independence test was performed on the duct with aspect ratio b/h = 1.94. The transverse grid in both the x and y directions was calculated using the same procedure as described above. However the thickness of the first cell next to each wall was reduced to 0.8×10^{-4} m and the maximum cell size (uniform grid size) was reduced to $1/40$ of the domain dimension. The results shown in Table 6-1, indicate a slight grid dependency. The greatest effect was on the centreline bed shear stress which fell by 5.76% , while the ratio τ_ℓ / τ_{2D} decreased by only 0.238%.

6.2.3 Results and Discussion

Fully developed flow

Confirmation of fully developed flow is most easily seen by viewing profiles of flow variables at various streamwise locations. For the domain used in these tests the domain length was only $1/10$ of the semi height and therefore any further development within this domain length was indetectable. However initial tests on a longer duct in which the streamwise direction was equal to the full duct height and contained a grid of 10 cells showed identical profiles for all variables.

Boundary shear stress

Laminar flow solution

The computational results were compared with the laminar flow solution calculated from the velocity distribution for fully developed flow in smooth rectangular ducts given by Cornish (1928). In current notation this is

$$u = \frac{16}{\pi^3}\frac{h^2}{\mu}\frac{\mathrm{d}p}{\mathrm{d}x}\left\{\sum_{n=1,3\ldots}^{\infty}\frac{1}{n^3}(-1)^{n-1/2}\frac{\cosh\left(n\frac{\pi}{2}\frac{z}{h}\right)}{\cosh\left(n\frac{\pi}{2}\frac{b}{h}\right)}\cosh\left(n\frac{\pi}{2}\frac{y+h}{h}\right)\right\}+\frac{1}{2\mu}\frac{\mathrm{d}p}{\mathrm{d}x}(y^2-2hy)$$

Equation 6-1

The viscous shear stress on the side wall is then

$$\tau_w(y) = \tau_{zx}\big|_{z=b} = \mu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)$$

$$= -\frac{16}{\pi^3}h\frac{\mathrm{d}p}{\mathrm{d}x}\sum_{n=1,3\ldots}^{\infty}\frac{1}{n^2}(-1)^{n-1/2}\frac{\pi}{2}\tanh\left(n\frac{\pi}{2}\frac{b}{h}\right)\cos\left(n\frac{\pi}{2}\frac{y+h}{h}\right)$$

Equation 6-2

because $w = 0$, and the bed shear stress is

$$\tau_b(z) = \tau_{yx}\big|_{y=0} = \mu\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)$$

$$= \frac{16}{\pi^3}h\frac{\mathrm{d}p}{\mathrm{d}x}\left\{\sum_{n=1,3\ldots}^{\infty}\frac{1}{n^2}\frac{\pi}{2}\frac{\cosh\left(n\frac{\pi}{2}\frac{z}{h}\right)}{\cosh\left(n\frac{\pi}{2}\frac{b}{h}\right)}\right\}-h\frac{\mathrm{d}p}{\mathrm{d}x}$$

Equation 6-3

because $v = 0$. The total force per unit axial length over the bed b is

$$\int_0^b \tau_b \mathrm{d}z = \frac{16}{\pi^3}h^2\frac{\mathrm{d}p}{\mathrm{d}x}\left\{\sum_{n=1,3\ldots}^{\infty}\frac{1}{n^3}\tanh\left(n\frac{\pi}{2}\frac{b}{h}\right)\right\}-bh\frac{\mathrm{d}p}{\mathrm{d}x}$$

Equation 6-4

and over the side wall h

$$\int_0^h \tau_w \mathrm{d}y = -\frac{16}{\pi^3}h^2\frac{\mathrm{d}p}{\mathrm{d}x}\sum_{n=1,3\ldots}^{\infty}\frac{1}{n^3}\tanh\left(n\frac{\pi}{2}\frac{b}{h}\right)$$

Equation 6-5

Adding Equation 6-4 and Equation 6-5 (for each quarter of the duct) and multiplying by 4 gives the total perimeter shear force per unit axial length as

$$-4bh\frac{\mathrm{d}p}{\mathrm{d}x}$$ *Equation 6-6*

i.e. equal to the duct cross-sectional area × pressure gradient. The shear force on the two side walls as a percentage of the total boundary shear force ($\%SF_w$) is obtained by multiplying Equation 6-5 by 4, dividing by Equation 6-6 and multiplying by 100:

$$\%SF_w = 100\frac{16}{\pi^3}\frac{h}{b}\sum_{n=1,3..}^{\infty}\frac{1}{n^3}\tanh\left(n\frac{\pi}{2}\frac{b}{h}\right)$$ *Equation 6-7*

Boundary shear stress distribution

The distribution of local boundary shear stress, non-dimensionalised by cross-sectional mean boundary shear stress is given in Figure 6-2 for the aspect ratio b/h = 10. The results of the two-layer model are compared with the laminar flow solution and the experimental results of Rhodes and Knight (1994) for the same aspect ratio.

In the vicinity of the corner the boundary shear stress is reduced for the laminar and turbulent flow, however for the latter the increase in mixing maintains a higher shear stress deep into the corner. The experimental results show the effect of turbulence driven secondary flow cells; there is a further increase of shear stress deep into the corner region, and a decrease at the mid-height of the shorter side wall and at an approximately equal distance along the bed. The two-layer model is incapable of generating turbulence driven secondary flow and does not show these effects.

Percent shear force on side wall

The relationship between the shear force on the side wall (as a percentage of the total perimeter shear force) and the aspect ratio b/h was examined by Knight and Patel (1985) for 11 smooth ducts in which the aspect ratio was varied within the range $1 \le b/h \le 10$. The work was extended by Rhodes and Knight (1994), using 6 aspect ratios in the range $10 \le b/h \le 50$. These experimental results are plotted in Figure 6-3 with the current computational results. Rather than $\%SF_w$, the more sensitive dependent variable Γ was used, defined by

$$\Gamma = \%SF_w - \frac{100}{\left(1+\frac{b}{h}\right)}$$ *Equation 6-8*

The term $100/\left(1+b/h\right)$ represents the geometric estimate of $\%SF_w$ made by assuming a uniform boundary shear stress and by taking the side wall length as a proportion of the total perimeter. The abscissa used was also $100/\left(1+b/h\right)$ rather than b/h because it gave a better resolution at low aspect ratios.

Rhodes and Knight applied a non-linear least squares fit to the experimental data to give

$$\%SF_w = \frac{100}{1+\left(\frac{1+1.345\,h/b}{1+1.345\,b/h}\right)^{-1.057}}$$ *Equation 6-9*

shown by the solid line in Figure 6-3.

Figure 6-3 utilises a useful property of rectangular closed ducts, in that when the duct is rotated by 90°, the wall and bed elements are transposed but there is no effect at all upon the fluid mechanics. However, the aspect ratio defined as the semi-width divided by the semi-height is now h/b. Therefore it may be concluded that a set of boundary shear stress measurements for a particular range of aspect ratios $b/h \geq 1$, also applies to another range $b/h \leq 1$ in which each aspect ratio is the reciprocal of the corresponding value in the first range.

The empirical model, laminar flow solution and the computational results all show maxima and minima in Γ. The reason for this is that as b/h tends to infinity both $\%SF_w$ and $100/\left(1+b/h\right)$ tend to 0%, and as b/h tends to zero both $\%SF_w$ and $100/\left(1+b/h\right)$ tend to 100%. Thus at the extremes Γ is zero. At b/h =1, symmetry dictates that both $\%SF_w$ and $100/\left(1+b/h\right)$ equal 50% and therefore Γ is again zero. As can be seen from Figure 6-2, within the ranges $0 < b/h < 1$ and $1 < b/h < \infty$ the wall shear stress is

depressed in the vicinity of the duct corners. This will significantly affect the average values for the shorter side but will have less effect on the longer boundary. The effect is to cause $\%SF_w$ to deviate from the value predicted from purely geometrical considerations, creating the maximum and minimum values in Γ.

The deviation is greatest for laminar flow and least for the experimental results represented by Equation 6-9, the latter reflecting the greater uniformity of the boundary shear stress distribution caused by turbulent mixing. The results of the computational model lie between the two, indicating an intermediate mixing process. The momentum transfer to the side wall is underpredicted by the two-layer model for two possible reasons: either the level of turbulent mixing is underestimated or the absence of secondary flow in the computational solution means that this enhanced mixing mechanism is unrepresented and the effect is significant. Given that the two-layer model has been found to satisfactorily predict two-dimensional boundary layers (Rodi 1991), it seems most likely that the discrepancy between the computational model and the experimental results is due to the inability of the two-layer model to generate turbulence driven secondary flows.

Centre-line bed shear stress

The ratio of centre-line bed shear stress to the two-dimensional value, calculated from the pressure gradient, is shown in Figure 6-4 plotted against $100/\left(1+{}^{b}/_{h}\right)$. The experimental results of Leutheusser (1963) and Knight and Patel (1985) are shown together with the laminar flow solution and the computational results. Given the good agreement between the experimental results of Leutheusser and Knight and Patel, it is surprising that at the three highest aspect ratios Knight and Patel give non-dimensional centre-line bed shear stress values greater than 1.0. These results could be caused by secondary flow convecting high momentum fluid towards the wall in the vicinity of the centre-line, though at the aspect ratios concerned (${}^{b}/_{h} > 7.5$) the more likely explanation is that of experimental error.

For $b/h \leq 6$ the results of the computational model lie systematically between the experimental measurements and the laminar flow solution. This shows that the mixing mechanism that transfers momentum deficit from the side wall towards the centre-line of the channel is underestimated by the two-layer model. For the reason previously suggested this is likely to be due to the absence of turbulence driven secondary flow in the computational model, rather than inadequate representation of turbulent mixing. This conclusion is reinforced by the fact that at low aspect ratios, especially for the square duct, secondary flow convects low momentum fluid away from the centre of the bed thus depressing the measured local shear stress.

6.3 Simulation of a Plate-Fin Heat Exchanger

6.3.1 Periodic Distribution of Temperature

There are many applications in all disciplines of engineering for which periodically developed flow over rough boundaries is significant. Several of these flows involve the transfer of heat (e.g. Beale 1989, Prakash 1985, Patankar *et al* 1977), for which the application of periodic boundary conditions is problematical due to the fact that the distribution of temperature is not periodic. This may be overcome in an manner similar to the treatment of the pressure variable (Section 3.2) for the idealistic situation in which the temperature at any point may be described by the sum of two components, a "local" temperature T^* which is periodic and a linear "reduced" streamwise temperature gradient $\mathrm{d}T'/\mathrm{d}x$. Just as the pressure gradient may be replaced by a source of momentum flux per unit volume, a source of heat flux per unit volume will account for a change in T' over a period length.

6.3.2 Simulation of a Prototype Heat Exchanger

This technique was tested on a plate-fin counterflow heat exchanger using the PHOENICS controlled cyclic boundary conditions in which a source of momentum flux per unit volume was used to simulate the fluid pressure gradient. The simulation was based on a prototype oil/water laminar flow heat exchanger designed for use in a diesel

engine and manufactured by Serck Heat Transfer Ltd. (Figure 6-5). The physical prototype consisted of multiple contiguous water and oil filled ducts which allowed the centreline of a typical water duct to be modelled by a plane of symmetry. The flow domain therefore consisted of a single oil duct sandwiched between two half-ducts conveying water. Symmetry conditions were also employed in the z direction to model a single period width and the x direction consisted of one streamwise period (Figure 6-6). Uniformly distributed cells were used in each region of the grid, though varying in size from region to region. Details of the grid are given in the Q1 data file (Appendix 13).

The simulation was performed in two stages. Initially only the flow field was calculated, using a pressure gradient of 500 Pa/m and 10^6 Pa/m for the water and oil ducts respectively. On obtaining a converged solution the calculation of T^* was activated and the solution of the velocity field was switched off. The temperature[1] of the oil at the start of the period, on the centreline of the oil duct was specified as a datum value of 100°C. An estimated value of the heat flux source per unit volume was added to the oil and enclosed fins and a sink of heat flux per unit volume was applied to the water ducts so that the net source of heat flux was zero. For the dimensions simulated, the volume of water and oil happened to be identical and therefore the magnitude of the source and sinks per unit volume were equal. After each sweep the estimate of heat flux source (S) was adjusted by comparing the temperature at a reference point (T^*_{calc}, located at the start of the period on the water duct centreline) with a specified reference temperature ($T^*_{ref} = 70°C$) using the formula

$$S_{new} = S_{old} \times \frac{T^*_{calc}}{T^*_{ref}} \qquad \textit{Equation 6-10}$$

The simulation was continued until the required adjustment of the heat flux source was less than 0.1% and the histories of spot values and residuals were constant.

---

[1] For the remainder of this chapter references to temperature will mean T^*.

6.3.3 Discussion of Results

The input conditions and the results of the simulation are presented in Table 6-3. The essential output is that for a prescribed temperature difference between the oil (100°C) and water (70°C) the heat flux from the working fluid to the coolant has been obtained. This could readily be used to compare the heat transfer efficiencies of different plate-fin heat exchanger designs. In addition the calculated temperature field helps in the diagnostic process, when seeking to remedy inefficiencies in the design.

Source of heat flux per unit volume

The following analysis of the heat flux over one period through the copper/steel plate separating the oil and water filled ducts confirms that the source of heat flux corresponded to the resultant temperature field. The heat flux through the plate equals the sink of heat flux in one of the two water half-channels

$$\dot{q}_{\text{plate}} = S_w \times {}^{V_w}\!/_{2} \qquad \textit{Equation 6-11}$$
$$= 0.3168 \text{ W}$$

where S_w is the sink of heat flux per unit volume in the water channel and V_w is the volume of one period of a single complete water channel. The heat flux through the copper/steel plate can also be found from the temperature gradient across either the copper or steel plate, as the heat flux through each was identical. Accounting for the fact that this temperature gradient varied with streamwise position then

$$\dot{q} = \sum_{IX=1}^{IX=NX} K_s \, {}^{dT^*}\!/_{dy} \times A_y \qquad \textit{Equation 6-12}$$

where K_s is the conductivity of steel and A_y is the cell area in the plane of the copper/steel plate. This gave a value for the heat flux through the copper/steel plate of 0.3182 W, a discrepancy of 0.7%.

Distribution of temperature

Combined with the heat flux across a heat exchanger, a useful aid to the designer is the distribution of temperature throughout the domain (Figure 6-7). The temperature

gradients in the oil flow were significantly higher on the faces of the vertical fins than on the base and top plates, indicating higher rates of heat transfer to the fins. This was due to the flow structure through the fin configuration (Figure 6-8) which convected high temperature fluid towards the fins. There was virtually no convection in the vertical direction (Figure 6-9) in either the oil or water flows and heat transfer was due to viscous diffusion only.

Figures relating to simulations of smooth rectangular ducts

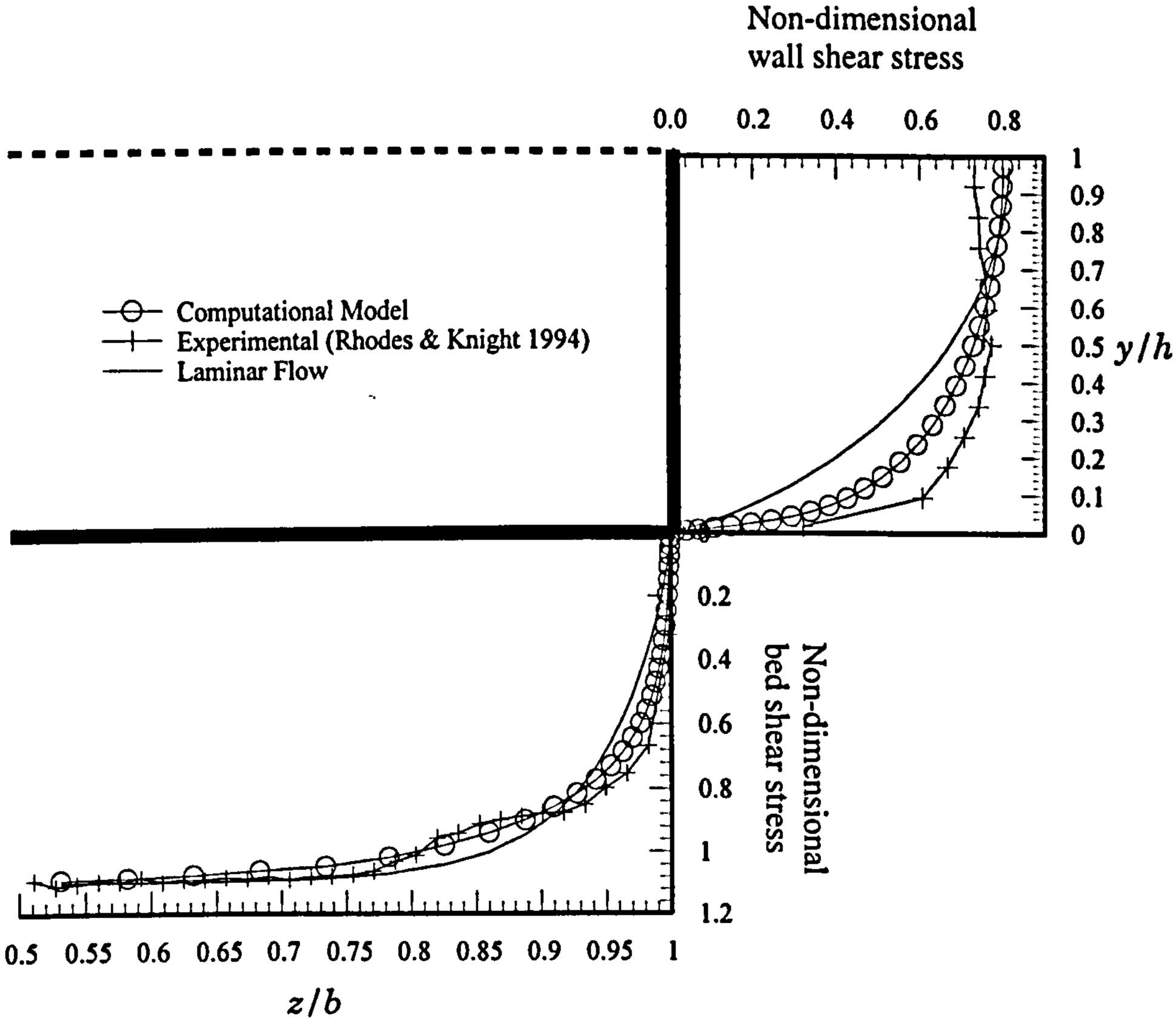


Figure 6-2: Distribution of local boundary shear stress for aspect ratio $b/h = 10$

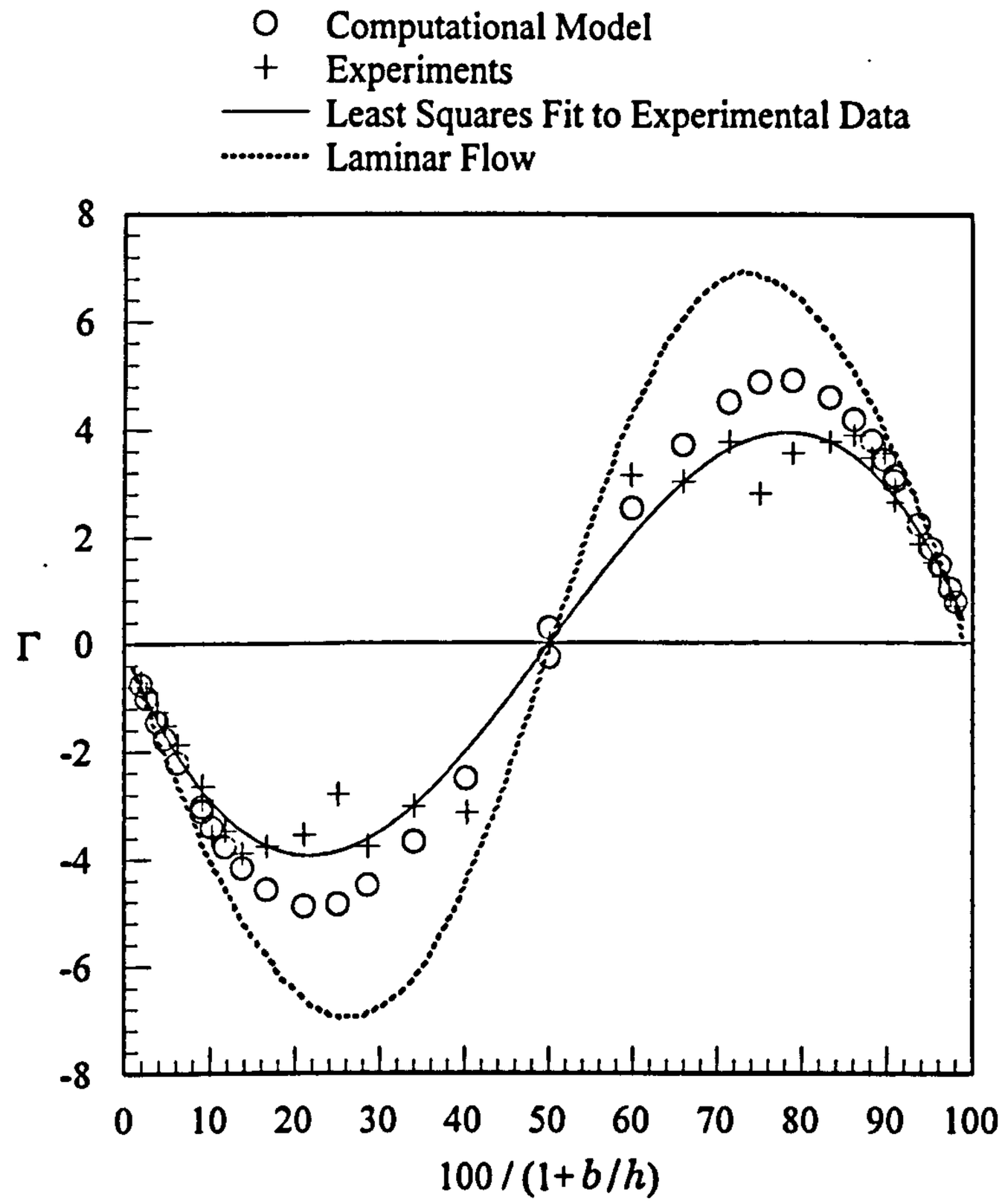


Figure 6-3: Relationship between aspect ratio and the shear force on the side wall as a percentage of the total perimeter shear force

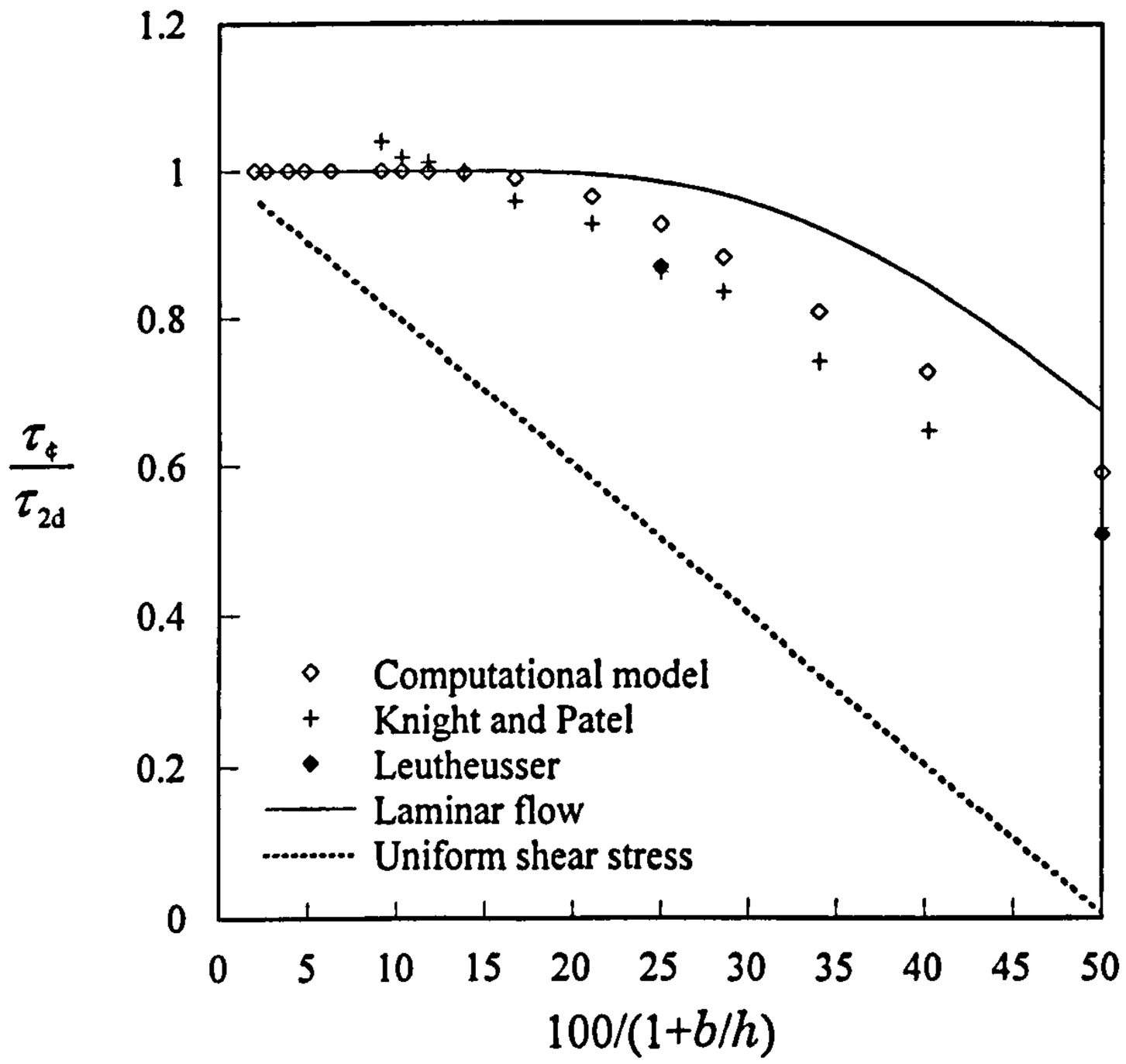


Figure 6-4: Ratio of centre-line bed shear stress to the two-dimensional value

Figures relating to simulations of a plate-fin heat exchanger

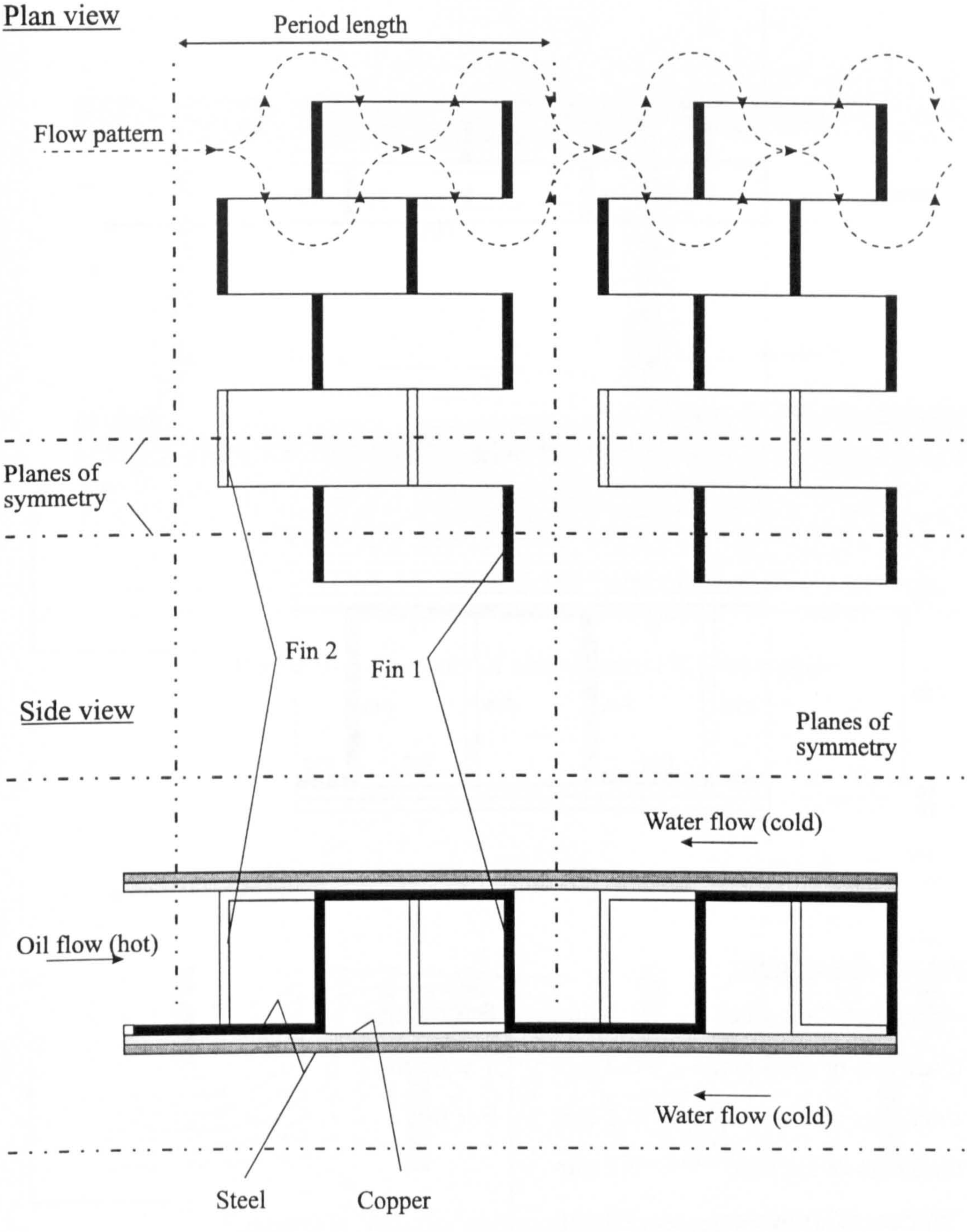
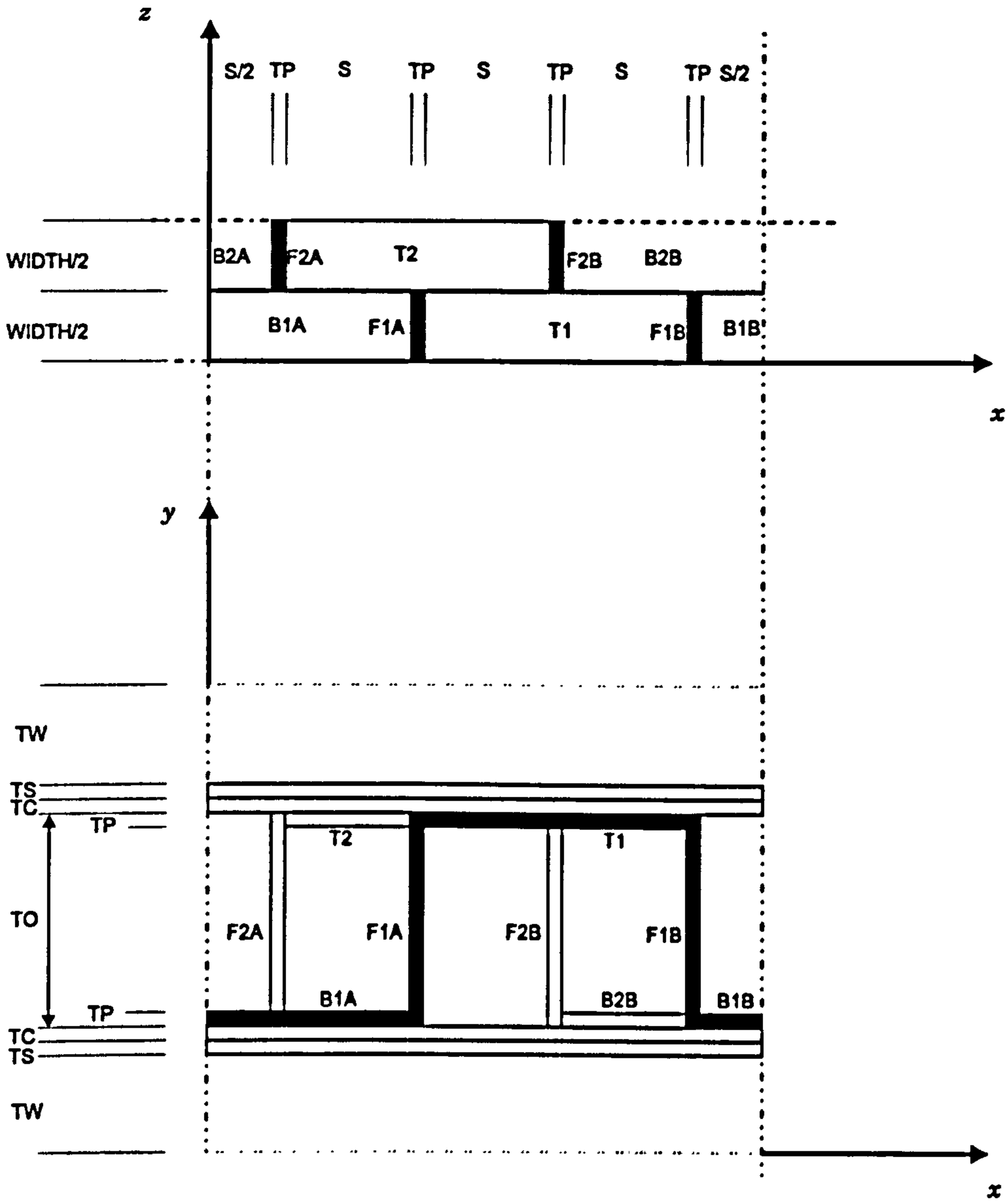


Figure 6-5: Heat exchanger design (idealised)



Notation and dimensions:

TW = "thickness" of water	1.5 mm	B = fin base
TS = thickness of steel plate	0.2 mm	T = fin top
TC = thickness of copper plate	0.1 mm	F = fin
TP = thickness of fin base	0.2 mm	
TO = "thickness" of water	3.0 mm	
WIDTH = width of fins	2.0 mm	
S = spacing between fins	2.0 mm	

Figure 6-6: Heat exchanger domain

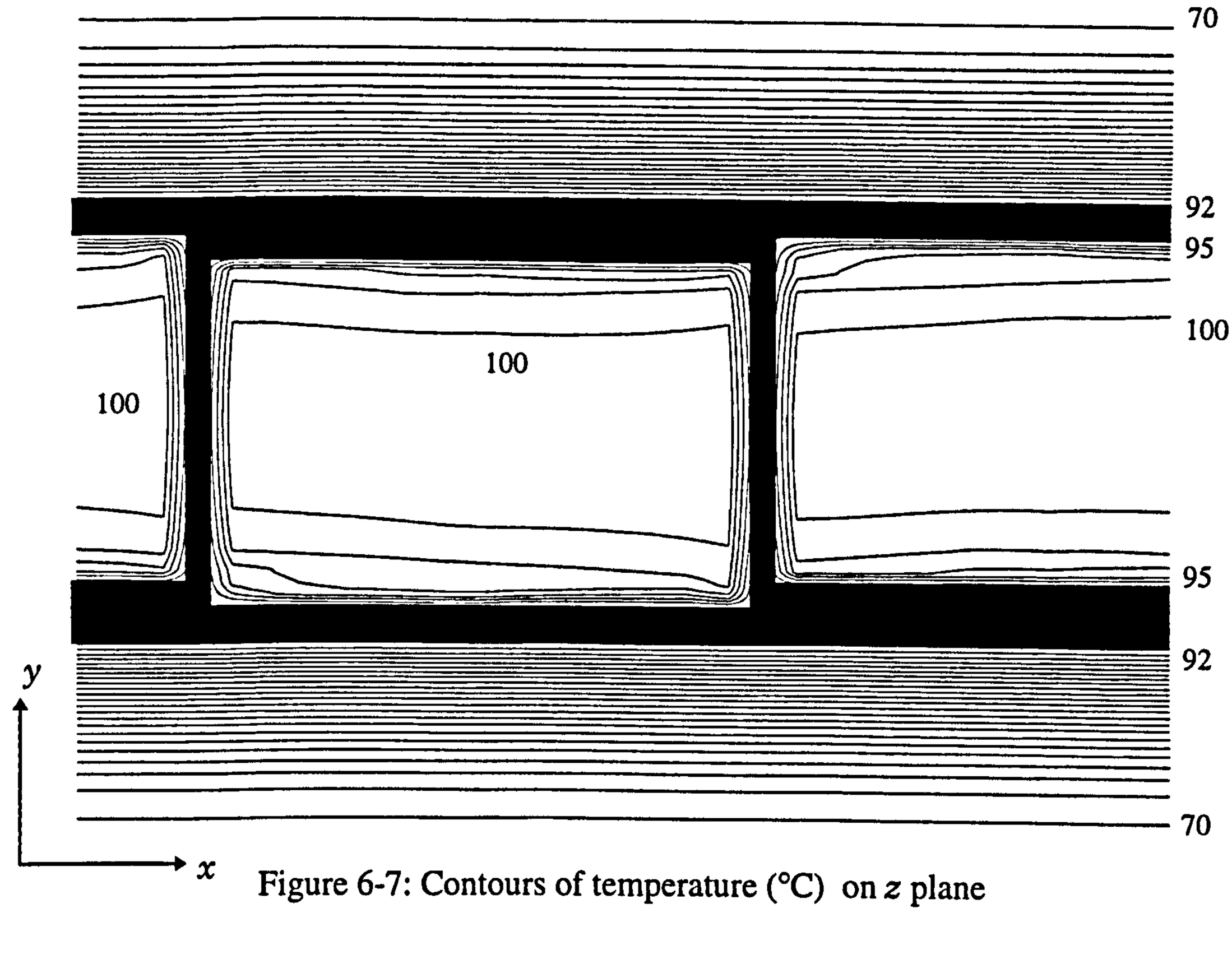


Figure 6-7: Contours of temperature (°C) on z plane

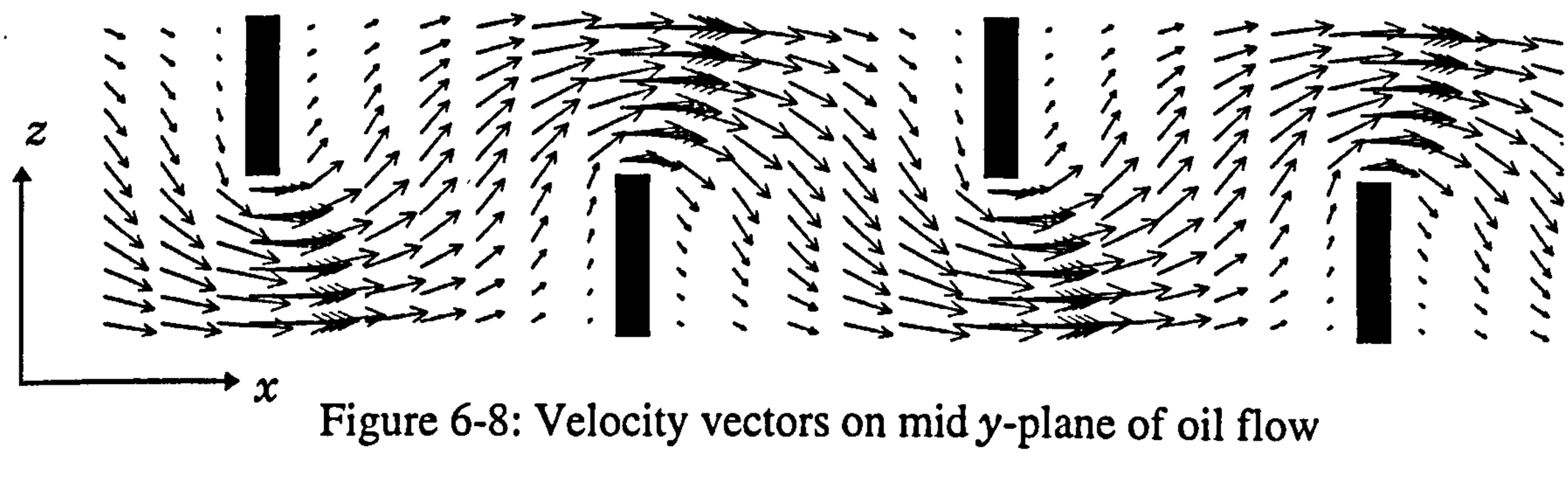


Figure 6-8: Velocity vectors on mid y-plane of oil flow

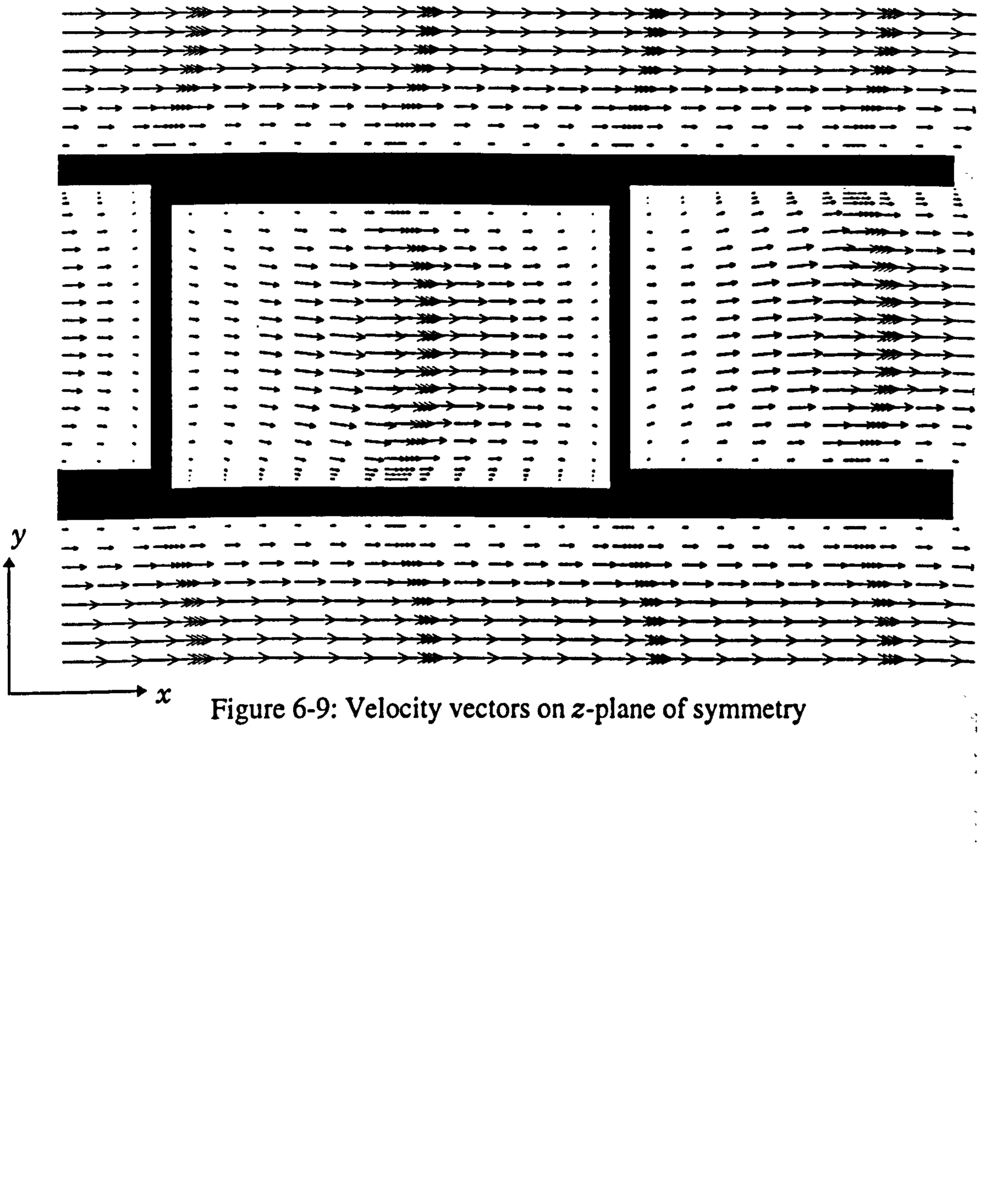


Figure 6-9: Velocity vectors on z-plane of symmetry

Tables relating to simulations of smooth rectangular ducts

b/h	b [m]	h [m]	R_h [m]	$\overline{U}$ [m s^{-1}]	ρ [kg m^{-3}]	ν [m^2 s^{-1}]	Re	Y^+	dP/dx [Pa m^{-1}]	F_{sw} [N m^{-1}]	F_{bed} [N m^{-1}]	$\overline{\tau}$ [N m^{-2}]	%SF	τ_ℓ [N m^{-2}]	τ_ℓ/τ_{2D}	%E
1.000	0.083	8.250E-02	0.0413	9.796	1.183	1.513E-05	1.068E+05	2.63	6.281	2.124E-02	2.147E-02	0.259	49.734	0.307	0.593	7.813E-02
1.490	0.123	8.221E-02	0.0492	7.557	1.183	1.513E-05	9.830E+04	2.06	3.163	1.200E-02	1.988E-02	0.156	37.644	0.190	0.729	-6.045E-02
1.940	0.160	8.247E-02	0.0544	6.318	1.183	1.519E-05	9.053E+04	1.74	2.042	8.167E-03	1.877E-02	0.111	30.316	0.136	0.809	1.608E-02
2.500	0.200	8.000E-02	0.0571	5.205	1.183	1.513E-05	7.862E+04	1.48	1.371	5.283E-03	1.667E-02	0.078	24.065	0.097	0.883	-5.145E-02
3.000	0.200	6.667E-02	0.0500	6.064	1.183	1.531E-05	7.923E+04	1.71	2.158	5.792E-03	2.297E-02	0.108	20.134	0.133	0.928	4.236E-02
3.750	0.200	5.333E-02	0.0421	7.175	1.183	1.513E-05	7.987E+04	2.05	3.642	6.275E-03	3.258E-02	0.153	16.152	0.188	0.965	-1.222E-02
5.000	0.200	4.000E-02	0.0333	8.781	1.183	1.512E-05	7.746E+04	2.52	7.133	6.892E-03	5.017E-02	0.238	12.079	0.283	0.990	9.690E-03
6.250	0.200	3.200E-02	0.0276	9.469	1.183	1.513E-05	6.906E+04	2.75	10.569	6.506E-03	6.114E-02	0.292	9.617	0.337	0.997	-1.304E-02
7.500	0.200	2.667E-02	0.0235	10.330	1.183	1.513E-05	6.426E+04	3.02	15.248	6.492E-03	7.483E-02	0.359	7.982	0.406	0.999	-5.762E-03
8.750	0.200	2.286E-02	0.0205	10.342	1.183	1.513E-05	5.609E+04	3.08	18.428	5.761E-03	7.848E-02	0.378	6.838	0.421	1.000	-1.879E-03
10.000	0.200	2.000E-02	0.0182	10.038	1.183	1.513E-05	4.824E+04	3.04	20.590	4.917E-03	7.744E-02	0.374	5.971	0.412	1.000	3.166E-03
9.993	0.616	6.162E-02	0.0560	6.830	1.199	1.514E-05	1.011E+05	1.84	2.478	5.706E-03	8.830E-02	0.139	6.070	0.153	1.000	2.807E-02
15.005	0.616	4.104E-02	0.0385	9.547	1.181	1.544E-05	9.515E+04	2.54	7.302	7.429E-03	1.771E-01	0.281	4.026	0.300	1.000	-1.484E-03
20.000	0.616	3.079E-02	0.0293	11.255	1.184	1.539E-05	8.579E+04	3.05	13.955	7.940E-03	2.566E-01	0.409	3.001	0.430	1.000	-3.933E-03
24.992	0.616	2.464E-02	0.0237	11.591	1.174	1.558E-05	7.051E+04	3.19	19.296	7.012E-03	2.857E-01	0.457	2.395	0.475	1.000	6.850E-03
37.459	0.616	1.644E-02	0.0160	10.701	1.173	1.557E-05	4.401E+04	3.16	28.330	4.563E-03	2.822E-01	0.454	1.591	0.466	1.000	-3.080E-03
50.008	0.616	1.231E-02	0.0121	9.396	1.185	1.541E-05	2.945E+04	2.96	32.854	2.969E-03	2.461E-01	0.397	1.192	0.405	1.000	3.582E-03
Grid independence test:																
1.940	0.160	8.247E-02	0.0544	6.318	1.183	1.519E-05	9.053E+04	0.87	1.934	7.761E-03	1.777E-02	0.105	30.396	0.129	0.807	-6.813E-02
% error from original:									-5.6	-5.2338339	-5.633145	-5.5118	0.263	-5.763	-0.238	

Where: b = semi length of bed wall

h = semi height of side wall

R_h = hydraulic radius

$$\mathrm{Re} = \frac{\overline{U} \times 4R_h}{\nu}$$

F_{sw} = force per unit length on sidewall

F_{bed} = force per unit length on bed wall

τ_ℓ = centreline bed shear stress

Y^+ = value of Y^+ at centre of wall cell on bed centreline

$$\overline{\tau} = \frac{F_{sw} + F_{bed}}{b + h}$$

%E = % error between resistive forces and driving forces

Table 6-1: Inputs and results of simulations in a smooth rectangular duct

Calculation of geometric grid (y direction)								Total Cells	
Aspect ratio	h	$\therefore\Delta$uni	N'_{total}	k	GNY	Δ last cell	GDIS	NY	NZ
1	0.0825	4.125×10^{-3}	22	1.248736	15	3.497×10^{-3}	1.692×10^{-2}	30	30
1.49	0.0825	4.125×10^{-3}	22	1.248736	15	3.497×10^{-3}	1.692×10^{-2}	30	32
1.94	0.0825	4.125×10^{-3}	22	1.248736	15	3.497×10^{-3}	1.692×10^{-2}	30	33
2.50	0.0800	4.000×10^{-3}	22	1.246495	15	3.4103×10^{-3}	1.661×10^{-2}	30	34
3	0.0665	3.325×10^{-3}	21	1.249387	14	2.819×10^{-3}	1.350×10^{-2}	29	34
3.75	0.0535	2.675×10^{-3}	20	1.250000	13	2.270×10^{-3}	1.072×10^{-2}	28	34
5	0.04	2.000×10^{-3}	19	1.244332	12	1.727×10^{-3}	8.159×10^{-3}	27	35
6.25	0.032	1.600×10^{-3}	18	1.244273	11	1.387×10^{-3}	6.429×10^{-3}	26	35
7.5	0.0265	1.325×10^{-3}	17	1.247789	10	1.143×10^{-3}	5.130×10^{-3}	26	34
8.75	0.0230	1.150×10^{-3}	17	1.233490	10	1.031×10^{-3}	4.779×10^{-3}	25	35
10	0.02	1.000×10^{-3}	16	1.241883	9	8.826×10^{-4}	3.886×10^{-3}	25	35
9.993	0.06162	3.081×10^{-3}	21	1.243504	14	2.651×10^{-3}	1.290×10^{-2}	29	40
15.005	0.04104	2.052×10^{-3}	19	1.246580	12	1.762×10^{-3}	8.275×10^{-3}	27	39
20	0.03079	1.539×10^{-3}	18	1.240657	11	1.347×10^{-3}	6.300×10^{-3}	26	40
24.992	0.02464	1.232×10^{-3}	17	1.240445	10	1.084×10^{-3}	4.947×10^{-3}	25	40
37.459	0.01644	0.822×10^{-3}	15	1.245256	8	7.243×10^{-4}	3.041×10^{-3}	24	40
50.008	0.01232	0.616×10^{-3}	14	1.237523	7	5.603×10^{-4}	2.262×10^{-3}	23	40

Notes:

thickness of first cell in geometric region
$= 1.56\times10^{-4}$

h = semi height of duct

Δuni = size of uniform cells

k = expansion factor

GNY = number of cells in geometric region

Δ last cell = size of last (largest) cell in geometric region

GDIS = length of geometric region

NY = total number of cells in y direction

NZ = total number of cells in z direction

Table 6-2: Calculation of grid for smooth rectangular duct tests

Tables relating to simulation of a plate-fin heat exchanger

Inputs		
Pressure gradient in oil flow	1×10^{6}	$Pa\ m^{-1}$
Pressure gradient in water flow	500	$Pa\ m^{-1}$
Temperature of oil at start of period	100	°C
Temperature of water at start of period	70	°C
Conductivity of oil	0.286	$W\ m^{-1}\ °K^{-1}$
Conductivity of water	0.597	$W\ m^{-1}\ °K^{-1}$
Conductivity of steel	381	$W\ m^{-1}\ °K^{-1}$
Conductivity of copper	48	$W\ m^{-1}\ °K^{-1}$
Results		
Flow rate of oil	1.527×10^{-6}	$m^3\ s^{-1}$
Flow rate of water (per channel)	1.264×10^{-6}	$m^3\ s^{-1}$
Heat flux across plate (per period)	0.317	W
Average (local) temperature of oil	98.55	°C
Average (local) temperature of water	77.29	°C

Table 6-3: Inputs and results for simulation of a plate-fin heat exchanger

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

7. SUMMARY AND CONCLUSIONS

7.1 Comparison With Rough Duct Physical Experiments

The two-layer model with periodic boundary conditions was applied to the long angle roughness configurations of Schlichting (1936) to validate the computational model against physical measurements for flow in rough ducts[1] and also to implement the techniques described in Chapter 3. These included the convergence criteria of a linear transverse shear stress distribution and a force balance, and the determination of a dynamically significant rough wall datum position by consideration of streamwise force moments. The opportunity was also taken to test the two-layer model by simulating the roughness configuration of Grass *et al* (1991).

7.1.1 Schlichting's Experiments

The duct employed in Schlichting's experiments was modelled by a two-dimensional domain consisting of one smooth wall and one wall with rib roughness elements of relative heights in the range $0.075 \leq k_h/h \leq 0.081$ and relative period lengths l/k_h = 18.75, 12.90 and 6.67. Each roughness configuration was modelled for Reynolds numbers in the range $1.0 \times 10^5 \leq \mathbf{Re} \leq 2.7 \times 10^5$. The results of these simulations are discussed in Sections 4.1.2 and 4.1.3.

Methodology

The methodology used for the majority of experiments in this thesis are described in Chapter 3. When applied to the simulation of Schlichting's experiments the main conclusions were:

---

[1] The computer code was also validated by simulating the flow in three-dimensional smooth rectangular ducts prior to the rough duct experiments reported here. Simulations of smooth rectangular ducts are described in Section 6.2 and conclusions drawn in Section 7.3.1.

1. A pressure driven, periodically fully developed duct flow with uniformly distributed roughness elements was simulated using the PHOENICS CFD code by utilising the built-in XCYCLE facility. A uniformly distributed source of momentum flux per unit volume was required to account for the pressure gradient.

2. Numerical convergence was satisfactorily determined by two criteria, namely:

   a) By confirming that the transverse distribution of period averaged shear stress was linear and had a slope equal to the reciprocal of the pressure gradient.

   b) By comparing the driving force acting on the fluid due to the source of momentum flux per unit volume with the sum of the boundary resistive forces, to ensure an adequate balance.

3. A dynamically significant method of obtaining the rough wall datum position was achieved by finding the line of action of the period averaged shear stress that gave the same moment of streamwise forces as the original forces distributed about the roughness elements and bed. For the modelled Schlichting roughness configurations this method predicted the datum plane to be located slightly above $k_h / 2$.

Comparison of model with physical experiments

For each flow configuration a pressure gradient equal to that measured by Schlichting was applied in the numerical model and therefore the sum of the period averaged boundary shear stress in model and experiment had to be equal. Given that Schlichting otherwise reported only his measured peak velocity, it was therefore this variable which was used to test the ability of the computational model to reproduce physical reality.

It was concluded that given the excellent agreement between peak velocities at the highest Reynolds numbers simulated (maximum discrepancy of 2.1%), the two-layer model had performed sufficiently well to proceed with confidence in its application to similar computations where the depth of flow would be varied (Depth Scale Roughness Calculations).

A more detailed investigation into the boundary layer shear stress and velocity distributions was also carried out:

Boundary shear stresses

The results presented in Table 4-1 and discussed in Section 4.1.2 show an average discrepancy of 130% between the calculated period averaged shear stress on the smooth wall and that measured by Schlichting. The main conclusions were:

4. Schlichting's assumption that the rough wall did not affect the distribution of flow variables in the smooth wall region was probably incorrect. The modelled turbulence intensity in the region of the smooth wall was found to be higher than that predicted by an isolated smooth wall simulation. This is consistent with a rough wall effect, characterised firstly by increased turbulence kinetic energy and secondly by its convective transport into the smooth wall region, enhanced by cross-stream fluxes created by the roughness elements. Comparison of the numerical model of the smooth wall region in Schlichting's duct with an isolated smooth wall model gave a 15.5% higher shear stress in the former for the same Reynolds number (Plate XX, case 5). This effect, though not accounting for the much greater discrepancy between model and physical experiment, was consistent with a rough wall effect upon the smooth wall flow.

5. The very large discrepancy between the smooth wall boundary shear stress of model and experiment was most readily explained in terms of Schlichting's application of the law of the wall to velocity measurements of dubious quality. The maximum velocity predicted by Schlichting's smooth wall shear stress and the law of the wall was so different from his measured maximum velocity that either the law of the wall did not apply (and the previous conclusion indicates that it did apply approximately) or his measured velocity distribution was in error. The latter is a reasonable conclusion given that he had to determine the logarithmic region in a profile at maximum only 8 mm in extent from the wall to the point of maximum velocity.

6. The streamwise variation in smooth wall shear stress resulting from the periodic acceleration and deceleration of the streamwise velocity (due to the periodic variation in the cross-stream geometry) was investigated and found to be minimal. Diffusion of momentum deficit from the wall occurred at a sufficient rate to ensure that the law of the wall remained roughly in step with the streamwise variation in smooth wall shear stress. Consequently this effect on the smooth wall shear stress was minimal.

Velocity distribution

The logarithmic velocity distribution on the rough wall was used to calculate k_s as described in Section 3.4 and employed in Section 4.1.4. The main conclusions were:

7. The numerical model indicates that the coefficients in the rough wall logarithmic velocity distribution law (Equation 2-30) and the region over which this equation applied were strongly dependent on the datum position (y_0) employed for the rough wall.

8. The use of the geometric mean level to calculate the datum position in the numerical model's results gave a value for B in the logarithmic law which was significantly higher than that quoted in the literature (e.g. Nikuradse 1933 or Jayatilleke 1969). If this represented the physics of Schlichting's experiments then the use of this datum by Schlichting would have given highly inaccurate k_s values. This conclusion agrees with Coleman *et al* (1984).

9. Using streamwise force moments to calculate the datum position resulted in a value for B which was comparable with that suggested by Jayatilleke and this method of determining y_0 is therefore recommended.

7.1.2 Roughness Configuration of Grass *et al.*

The roughness configuration of Grass *et al* (1991, experiment A) was simulated by a two-dimensional domain of depth h = 55.9 mm, with a rough bed periodically

distributed with roughness elements of 5 mm square cross-section, spaced 20 mm apart. The Reynolds number based on duct height was 8091 and the free surface of the water channel was accounted for by a plane of symmetry. For a specified friction velocity, the flow rate was found to be under-predicted by 21.8% in comparison with that measured by Grass *et al.* The main conclusion was:

10. For flow regimes in which the separation streamline from the trailing edge of a roughness element impinges on the vertical face of the next roughness element, the form drag predicted by the numerical model was probably in error. In the depth scale roughness configurations analysed (Chapter 5) the point of re-attachment always occurred on the bed and therefore the condition giving rise to the unsatisfactory solution for Grass *et al*'s experiments did not pertain.

7.2 Depth Scale Roughness Calculations

Calculations were performed on a two-dimensional rough walled duct flow in which the flow depth was varied from very shallow to deep, the range of relative depths being $1.15 \leq h/k_h \leq 40$. Turbulence was accounted for by the two-layer model, periodic boundary conditions were employed to achieve periodically fully developed flow and the duct plane of symmetry was used to represent the free surface in an open channel flow. Two types of roughness were simulated: a square cross-section block roughness (k_h = 50 mm) with a period length of 1 m and a thin strip roughness of height k_h = 3.2 mm with period lengths of 40 and 60 mm.

The outcome of this work is a new proposed method for calculating the channel friction factor f, which is applicable not only to small scale roughness but also to rough wall flows when the roughness length scale is large compared with the depth of flow. If confirmed by physical measurements this method would be a significant advance in the application of boundary layer theory to rough wall flows, where the roughness length scale is large compared with the depth of flow.

The primary conclusions of this analysis are treated in Section 7.2.1 and ancillary conclusions are drawn in Sections 7.2.2 and 7.2.3.

7.2.1 Evaluation of k_s

For relative depths $h/k_h \geq 7$, the equivalent sand roughness length scale (k_s) was obtained from the computational results by analysis of the logarithmic velocity profiles, as described in Section 3.4.1 and performed in Section 5.8.1. Further analysis in Section 5.8.2 led to a resistance equation relating f, k_s and R_h. The main conclusions from this analysis were:

11. For all of the roughness configurations simulated, the resistance equation predicted k_s values which varied in a similar way. This was expected from physical reasoning, and from Figure 5-20 and 5-22 it was tentatively concluded that k_s could be calculated for any depth of flow and any roughness configuration using a relationship of the form

$$\frac{k_s}{k_{s\infty}} = \mathrm{F}\left(\frac{k_s}{h - y_0}\right) \qquad \textit{Equation 7-1}$$

from a knowledge of the flow depth h, the bed datum position y_0 and $k_{s\infty}$, the k_s value for the roughness type at infinitely large flow depths (small scale roughness value). Using a resistance equation of standard form

$$\frac{1}{\sqrt{f}} = a - b\log\frac{k_s}{2R_h} \qquad \textit{Equation 7-2}$$

but modified coefficients, and the previously calculated value of k_s, Figure 5-22 shows that the friction factor could then be deduced for any relative flow depth. Implicit in this calculation procedure was the determination of a dynamically significant value of y_0 which was important at all depths but especially so at the shallow end of the range where it critically affected the hydraulic radius ($R_h = h - y_0$) of the equivalent plane walled duct. The method therefore amounted

to the determination of the dynamically significant length scales, k_s, y_0 and h. This conclusion was tentative depending as it did upon a Reynolds averaged turbulence model implemented by a numerical method and limited in the present application to two-dimensional periodic roughness distributions.

12. A corollary of this determination of y_0 is that the rough wall was thereby transformed into an equivalent plane walled duct. The friction factor was then calculated as

$$f = \frac{8\tau_{y_0}}{\rho \overline{U}_{y_0}^{\,2}} \qquad \textit{Equation 7-3}$$

where τ_{y0} is the shear stress at the bed datum or equivalent plane wall, and $\overline{U}_{y_0}$ is the velocity averaged over the depth $(h - y_0)$, rather than that currently in common use

$$f = \frac{8\tau_w}{\rho \overline{U}^2} \qquad \textit{Equation 7-4}$$

where τ_w is the shear stress with line of action through the base of the roughness elements and $\overline{U}$ is the velocity averaged over the maximum flow depth.

13. The position of the rough wall datum level moved further away from the wall for lower relative depths of flow.

14. The alternative methods for calculating k_s by means of standard velocity profiles (e.g. Nikuradse 1933 or Jayatilleke 1969) described in Section 3.4.1, gave a larger variation of k_s when y_0 was obtained from the geometric mean level than when it was derived by streamwise force moments. The calculation of y_0 using streamwise force moments is therefore recommended.

15. For the thin strip roughness only, the k_s values were found to be approximately independent of flow depth for high relative depths ($h/k_h > 20$) despite the fact

that the k_s value was still of the same order of magnitude as the depth of flow (h/k_s < 10) and therefore did not represent small scale roughness. A constant k_s value was not obtained for the block roughness configuration.

16. The coefficients in the resistance equation (Equation 7-2) obtained from analysis of the velocity profiles agreed with those quoted by French (1986) for flow in open channels.

7.2.2 Velocity Field

The variation of the distribution of streamlines with depth and transverse profiles of streamwise velocity for a typical roughness configuration are presented in Figure 5-10 and are discussed in Section 5.6.5. The same transverse velocity profiles plotted logarithmically are presented in Figure 5-16 and discussed in Section 5.7. The main conclusions were:

17. For the majority of relative depths, the length of the recirculating flow region immediately downstream of each roughness element (the primary recirculating flow region) was independent of flow depth. At very low relative depths however, the presence of high streamwise velocities over the tops of the roughness elements restricted the development of the vena contracta which form due to the reduction in flow cross-sectional area. Consequently the re-attachment length was reduced.

18. For the roughness configurations examined, the re-attachment length associated with the primary recirculating region underwent a slight decrease as the period length decreased. This was probably due to the higher turbulence intensity resulting from the wake of the upstream roughness element and is a characteristic of wake interference flows. The values of re-attachment length were comparable with those reported in the literature (e.g. Senior and Aroussi 1992) for flow over backward facing steps.

19. The length of the recirculating flow region upstream of each roughness element (the secondary recirculating region) increased systematically with increasing

relative depth and lower values of local streamwise velocity. The reason for this is not clear at present.

20. Distributions of the period averaged streamwise velocity support Conclusions 7,8 and 9.

7.2.3 Normal and Shear Stresses at the Boundary

The resistive forces on the boundary were due to the shear stresses on the horizontal surfaces and the normal stresses on the vertical faces of the roughness elements, both of which are discussed in Section 5.6.6. The main conclusions were:

21. For all simulations reported, the largest absolute value of boundary shear stress was located in the primary recirculating flow region. Although the value of local shear stress decreased with an increase in flow depth (due to lower streamwise velocities) as a percentage of the period averaged bed shear stress[2] it increased significantly as the flow depth increased. This was because the form drag on the roughness elements scaled with the square of the velocity, but the skin friction scaled with the velocity.

22. The resistance force due to the bed shear stress was negligible in comparison with the form drag on the roughness elements. In fact the integrated bed shear stress resulted in a force on the fluid in the positive streamwise direction for many of the roughness configurations, due to the regions of reverse flow.

23. The direct stresses on the vertical faces of the roughness elements were nearly uniform, varying little with vertical position. Assuming the resistance force due to the bed shear stress to be negligible (Conclusion 22 above) then for two-dimensional rib roughness the bed datum position could be roughly estimated as a function of the duct geometry only by the consideration of streamwise force moments.

---

[2] Including form drag and as well as viscous shear.

7.3 Other Applications

7.3.1 Simulations of Smooth Rectangular Ducts

The two-layer model was applied to rectangular duct flows of aspect ratios $1.0 \leq b/h \leq 50$ for Reynolds numbers in the range $5.9 \times 10^4 \leq \mathbf{Re} \leq 2.1 \times 10^5$. The primary purpose of these simulations was to perform initial validation tests for the computer code and the implementation of periodic boundary conditions. However the opportunity was taken to investigate the distribution of boundary shear stress and the effect of the side wall upon the centreline bed shear stress. The simulations are described in Section 6.2.1 and the results presented in 6.2.3.

24. Results indicated the correct implementation the of periodic boundary conditions which led to their successful application to the rough duct flows (Conclusion 1).

25. Comparison with the experimental results of Knight and Patel (1985) and Rhodes and Knight (1994) indicated that the inability of the two-layer model to generate turbulence driven secondary flow was significant in the following calculations:

    a) The relationship between the percentage boundary shear force acting on the side walls ($\%SF_w$) and the aspect ratio b/h, computed using the two-layer model, differed significantly from experiment. The difference reflected the enhanced mixing created by turbulence driven secondary flow in the physical model.

    b) For aspect ratios $b/h < 6.5$ (approximately) the side wall effect, represented by the ratio of the centreline bed shear stress to the two-dimensional value, differed between computation and experiment for the same reason as given in Conclusion 25a. For greater aspect ratios the flow was so nearly two-dimensional that the difference became imperceptible.

7.3.2 Simulation of a Plate-Fin Heat Exchanger

A laminar flow solution of a two fluid counterflow plate-fin heat exchanger was carried out as a test of the PHOENICS XCYCLE periodic boundary conditions when applied to

a temperature field. As described in Section 6.1 the change in temperature over one period length was accounted for by applying a source of heat flux per unit volume in one fluid and a corresponding sink in the other fluid. The main conclusions of the results discussed in Section 6.3 were:

26. The change in temperature over one period could adequately be accounted for by a joint source/sink of heat flux per unit volume, enabling periodic boundary conditions to be utilised. This is thought to be a novel method of solution for this problem.

27. The contours of "local" temperature and the velocity vectors proved a useful aid to the design of heat exchangers. For the prototype examined, the heat transfer between the two fluids would have been enhanced by generating more vertical convection in both streams.

7.4 Computational Fluid Dynamics

Several conclusions have been reached with regard to the use of CFD as a tool for simulating flow in rough ducts.

7.4.1 Convergence

One of the limiting factors associated with this project was the inability to rapidly obtain solutions for a wide variety of roughness configurations and high relative depths. Three possible reasons for this are:

28. Grid structure. Despite the simple geometry, large numbers of cells were required to model the viscosity affected region near the boundaries. This also resulted in cells with a very large aspect ratio.

29. Periodic boundary conditions. Periodically fully developed flow was achieved through the use of periodic boundary conditions in order to reduce the

computational resources required. However, the results of this study[3] indicated that although extra cells were not required to model the development length, and therefore there were considerable economies in computer storage, the saving in computational time was considerably less because the total number of calculations was not significantly reduced.

30. High flow depths require longer convergence times. A converged solution is only obtained when the boundary layer is fully developed and therefore for flows of high relative depth, the cross-stream distance over which convective and diffusive transport is required is longer. Also the influence of the roughness elements in the region of the free surface or plane of symmetry diminishes at higher depths and consequently convective and diffusive cross stream transport in this region is low. It was the author's experience that beyond a certain relative depth, converged solutions were unobtainable.

7.4.2 Use of Commercially Available CFD Packages

A major advantage of the general purpose CFD package is that a detailed knowledge of the differential equations, numerical procedures, or the source code is not required to perform simple simulations. However when more detailed calculations are required, an understanding of all these topics becomes essential. For codes written by the user, this information is readily available or already known. In the case of commercial packages however, comprehensive documentation is required to supply the same information. Also access to the source code is needed in order to make knowledgeable adjustments to solution parameters, add user-written coding to the existing program, and to find and correct errors in the original code.

The author found it essential for present purposes to painstakingly develop procedures for checking the physical realism of the computational results. This required an

---

[3] In particular the wall function tests (Section 5.5), which consisted of an extremely simple domain and grid.

understanding of the numerical procedures and led to the methodology described in Chapter 3.

7.5 Suggestions for Further Work

The analysis of depth scale roughness performed in the current study was limited by the results obtained for three principal reasons:

1. Small scale roughness was not achieved. Due to the long convergence times required to simulate ducts of high depth, and the large k_s values obtained, it was not possible to achieve a flow in which the roughness length scale (k_s) was really small in comparison with the flow depth ($k_s / h < 0.1$).

2. Insufficient variation of roughness configurations. Difficulties in obtaining converged solutions for short period lengths resulted in only three roughness configurations being successfully simulated. All of these were probably in the wake interference flow regime and were limited to two-dimensional rib roughness elements.

3. Limitations of a Reynolds averaged turbulence model. When a turbulence model, used within a numerical solution process, is employed to predict any type of flow, there are uncertainties of how accurately the flow physics are represented. In particular, Reynolds averaged turbulence models contain empirically based coefficients which are not applicable to all flow regimes. Also, certain mechanisms such as turbulence driven secondary flows may not be represented. It was concluded in Section 4.2.3 that in modelling flows over two-dimensional rib roughness elements in which the separation streamline impinged on the downstream roughness element, the point of impingement might not be accurately predicted by the two-layer model. When modelling flow over other roughness elements, such as hemispheres, there would also be uncertainties in the prediction of the point of separation.

To confirm that Equation 7-1 is indeed a universal function, further work is required which in particular addresses the three shortcomings outlined above. It is therefore recommended that experimental investigations should be performed, initially on an enclosed duct flow to avoid free surface effects, in which flows over a variety of two and three-dimensional roughness configurations would be examined. One way of undertaking such a project would be to measure the form drag using surface pressure tappings located on the vertical face of a roughness element. The bed shear stress could with care be obtained by measuring the velocity gradient in the viscous sub-layer using high resolution laser Doppler velocimetry (Wei and Willmarth 1989). Before undertaking these experiments, the researcher would need to consider the following points:

1. In order to determine the distribution of the normal stress acting on the vertical faces of the roughness elements, a number of pressure tappings would be required ($\approx$ 10) which would establish the minimum permissible roughness height. For two-dimensional flow, the density of pressure tappings in the vertical direction could be increased by staggering the tappings in the cross-stream direction.

2. In order to achieve small scale roughness, a ratio of $h/k_s > 10$ would be required. The current results for flow over two-dimensional thin strip roughness elements gave $5\,k_h \leq k_s \leq 6.5\,k_h$ corresponding to a ratio of $h/k_h > 65$, which implies an inordinately large duct depth requiring a long development length. It is likely however that other roughness configurations (e.g. three-dimensional roughness) would result in lower k_s values for the same value of k_h and permit a more manageable size of duct.

3. To achieve two-dimensional flow the results of Section 6.2 suggest that an aspect ratio of $b/h > 6.5$ is required for a smooth channel though this would be less for a roughened duct with smooth side walls. Using the argument presented in Appendix 9, Schlichting obtained two-dimensional flow using an aspect ratio b/h

≈ 4.25 for a duct containing one rough wall. A further consideration is that if the channel width were increased, the length of channel required to achieve fully developed flow would also increase.

4. Accurate measurement of bed shear stress over a period length could prove to be difficult to obtain, particularly for three-dimensional roughness configurations for which the flow structure would be complex. For three-dimensional roughness the integrated bed shear stress could also significantly affect the total resistance to flow.

These points indicate that a very large duct might be required for an investigation using this technique. An alternative experimental procedure could be to carry out measurements of the forces and moments on the bed and roughness elements themselves, employing peizo-electric load cells and moment balances, however this is unlikely to permit the use of smaller roughness elements due to the very small forces being measured.

Given that the ultimate application of this work is the prediction of resistance for depth scale roughness in natural open channel flows, the subjects of randomly distributed irregular roughness elements and of the effect of the free surface both require serious investigation. Implicit in the present project is the hypothesis that if through the determination of y_0, any roughness configuration can be reduced to an equivalent plane walled duct, then equations 7-1 and 7-2 derived from regular roughness patterns should be equally applicable to randomly distributed irregular roughness. This hypothesis needs to be tested.

The free surface imposes an extra level of complexity on the determination of channel resistance due to depth scale roughness. This is not a reference to the damping of turbulence fluctuations near the free surface and the consequent effect on secondary flow, important though this may be in the corner regions of channel flow. Rather it is a reference to the significant disturbance of the free surface that occurs when the roughness scale is large compared with the depth. This firstly changes the flow

geometry and secondly adds another route for energy dissipation through wave drag and hydraulic jumps (e.g. Hubbard and Thorne 1994). The problem of depth scale roughness will therefore only be partially treated until these Froude number effects are fully investigated.

Appendix 1: Guide to PHOENICS

Parabolic, Hyperbolic Or Elliptic Numerical Integration Code Series (PHOENICS) is a commercial CFD code written by CHAM Ltd (Wimbledon, London). During this study versions 1.6 to 2.1.1 of the code were used, mounted on a UNIX operating system at the Royal Military College of Science, Shrivenham.

For instructions on the PHOENICS CFD code, the reader is referred to the PHOENICS manuals, now available as an on-line help system called POLIS. However a brief guide to the structure of PHOENICS is given here as a reference aid to some of the features and terms used in this thesis.

PHOENICS consists of two essential computer codes; a pre-processor called SATELLITE and the processor called EARTH. SATELLITE interprets the user-written[1] data file (the Q1 file) and writes a data file (EARDAT) for the processor EARTH. On completion of the solution procedure, EARTH writes two files, the PHI file containing a dump of all flow variables and the RESULT file which contains user specified results in an easily viewed format.

The Q1 file contains instructions written in the PHOENICS input language (PIL) which defines the domain and parameters for each computational study. The Q1 file is organised into the following groups:

GROUP 1	Run title and other preliminaries.
GROUP 2	Transience: time-step specification.
GROUP 3	X-direction grid specification.
GROUP 4	Y-direction grid specification.
GROUP 5	Z-direction grid specification.
GROUP 6	Body-fitted coordinates or grid distortion.
GROUP 7	Variables stored, solved and named.
GROUP 8	Terms (in differential equations) and devices.
GROUP 9	Properties of the medium (or media).
GROUP 10	Interphase-transfer processes and properties.
GROUP 11	Initialization of variable or porosity fields.
GROUP 12	Convection and diffusion adjustments.
GROUP 13	Boundary conditions and special sources.

[1] The Q1 file may alternatively be written automatically using a menu system within SATELLITE.

GROUP 14	Downstream pressure (for free parabolic flow).
GROUP 15	Termination criteria for sweeps and other iterations.
GROUP 16	Termination criteria for inner iterations.
GROUP 17	Under-relaxation and related devices.
GROUP 18	Limits on variables values or increments to them.
GROUP 19	Data communicated by SATELLITE to GROUND.
GROUP 20	Control of preliminary printout.
GROUP 21	Frequency and extent of filed printout.
GROUP 22	Location of spot-values & frequency of residual printout.
GROUP 23	Variable-by-variable field printout and plot and/or tabulation of spot-values/residuals printouts.
GROUP 24	Preparation for continuation runs.

For the majority of cases the domain and parameters can be adequately defined within the Q1 file. However additional functions may be written by the user in a FORTRAN77 subroutine (GROUND.F) which is part of the EARTH module. In the current project the GROUND.F subroutine was employed both to control the fluid calculation and to provide preliminary analysis of the results.

In order to perform these tasks within GROUND.F the values of the variables stored within EARTH needed to be accessed. The following section gives a simplified description of how this is performed.

1.1 GROUND.F Subroutine

The subroutine GROUND.F contains a structure similar to that of the Q1 file, however not all of the groups are active and some groups are sub-divided in sections. One of the most useful groups is group 19, in which each section provides access at each stage of the solution procedure (see Section 2.3.6):

```
C**********************************************************************
C
C--- GROUP 19. Special calls to GROUND from EARTH
C
  19 GO TO (191,192,193,194,195,196,197,198,199,1910,1911),ISC
 191 CONTINUE
C  * ------------------ SECTION 1 ---- Start of time step.
     RETURN
 192 CONTINUE
C  * ------------------ SECTION 2 ---- Start of sweep.
     RETURN
 193 CONTINUE
C  * ------------------ SECTION 3 ---- Start of iz slab.
     RETURN
 194 CONTINUE
C  * ------------------ SECTION 4 ---- Start of iterations over slab.
     RETURN
```

```
 1911 CONTINUE
C   * ------------------ SECTION 11---- After calculation of convection
C                          fluxes for scalars, and of volume
C                          fractions, but before calculation of
C                          scalars or velocities
      RETURN
  199 CONTINUE
C   * ------------------ SECTION 9 ---- Start of solution sequence for
C                                       a variable
      RETURN
 1910 CONTINUE
C   * ------------------ SECTION 10---- Finish of solution sequence for
C                                       a variable
      RETURN
  195 CONTINUE
C   * ------------------ SECTION 5 ---- Finish of iterations over slab.
      RETURN
  196 CONTINUE
C   * ------------------ SECTION 6 ---- Finish of iz slab.
      RETURN
  197 CONTINUE
C   * ------------------ SECTION 7 ---- Finish of sweep.
      RETURN
  198 CONTINUE
C   * ------------------ SECTION 8 ---- Finish of time step.
C
      RETURN
C*************************************************************************
```

Any variable which is STORED or SOLVED in the Q1 file (such as P1, U1, V1, W1, EP, KE etc.) is also stored in EARTH and may be accessed in group 19. Most EARTH variables are known as *full field variables* and have a value for each grid cell (i.e. NX × NY × NZ values). PHOENICS stores all of these variables in a single, one-dimensional array, the size of which may be set by the user. This is termed the F-array. Each EARTH variable is allocated a block of elements NX × NY × NZ long within the F-array which has a first element labelled the "zero F-Location index" or LOF. The variable is then stored for each slab of fixed IZ cells. For a given slab the elements of the F-array are arranged as shown in Figure A1.1.

Thus for example the two-dimensional array of U1 velocities is put into the user array MYU1(IX,IY) for the current slab by the following coding in group 19, section 6:

```
REAL MYU1(NX,NY)
INTEGER L0U1
L0U1=L0F(U1)
DO IX=1,NX
        DO IY=1,NY
                I=IY+NY*(IX-1)
                MYU1(IX,IY)=F(L0U1+I)
        END DO
END DO
```

To aid manipulating segments of the F-array a library of functions (FN0 - FN117) is provided which perform simple arithmetic operations on EARTH variables.

IY increasing

NY-1	2 × NY	3 × NY	⋮	(NX-1×NY)+1	NX × NY
NY-2	2 × NY-1	⋮	⋮	⋮	(NX × NY)-1
⋮	⋮	⋮	⋮	⋮	⋮
3	NY+3	⋮	⋮	⋮	⋮
2	NY+2	⋮	⋮	⋮	⋮
1	NY+1	(2 × NY)+1	⋮	⋮	(NX-1) × NY+1

IX increasing

Figure A1.1: F-array locations of an x-y slab

Appendix 2: Introduction to Tensor Notation

For the reader not familiar with cartesian tensor notation, a short introduction to this subject is provided here. Tensor notation is used in this thesis because it allows three-dimensional equations to be written in a considerably more compact form than is possible with conventional notation.

In cartesian tensor notation, vector quantities are written by attaching an index to the symbol denoting the quantity. The components of a vector in three-dimensional space are obtained by setting the index equal to 1, 2 and 3. For example, the general velocity vector will be:

$$U_\alpha \equiv (U_1, U_2, U_3) \equiv (U, V, W) \qquad \textit{Equation A2-1}$$

A quantity with two indices (eg α and β) is called a *tensor* and has 9 components which can be obtained by permutation of the 2 indices from 1 to 3, so for example, the product of two velocity vectors yields a tensor:

$$U_\alpha U_\beta = \begin{pmatrix} U_1U_1 & U_1U_2 & U_1U_3 \\ U_2U_1 & U_2U_2 & U_2U_3 \\ U_3U_1 & U_3U_2 & U_3U_3 \end{pmatrix} \qquad \textit{Equation A2-2}$$

The Reynolds stress tensor is an example of this type.

In the summation convention used with tensor notation:

$$U_\alpha U_\alpha = \sum_{\alpha=1}^{3} U_\alpha U_\alpha = U_1U_1 + U_2U_2 + U_3U_3 \qquad \textit{Equation A2-3}$$

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

Appendix 3: Navier-Stokes and Reynolds Equations

A3.1 General form of transport equations

All transport equations have the basic form:

$$\underbrace{\frac{\partial \phi}{\partial t}}_{\substack{\text{unsteady}\\ \text{term}}} + \underbrace{\text{div}\,(\mathbf{u}\phi)}_{\substack{\text{convection}\\ \text{term}}} = \underbrace{{}^{1}\!/_{\rho}\,\text{div}\,(\Gamma\,\text{grad}\,\phi)}_{\substack{\text{diffusion}\\ \text{term}}} + S \qquad \textit{Equation A 3-1}$$

where ϕ is a general variable, S is the source term and

$$\text{div}\,(\mathbf{u}\phi) = \frac{\partial(u\phi)}{\partial x} + \frac{\partial(v\phi)}{\partial y} + \frac{\partial(w\phi)}{\partial z}$$

$$\text{grad}\,\phi = \nabla\phi = \frac{\partial \phi}{\partial x}\mathbf{i} + \frac{\partial \phi}{\partial y}\mathbf{j} + \frac{\partial \phi}{\partial z}\mathbf{k}$$

A3.2 Navier-Stokes Equations

If specific momentum (i.e. momentum per unit mass, or velocity) is used as the general variable ϕ, then the Navier-Stokes equations result.

Schlichting (1960) quotes these in full as (using current notation):

$$\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right)=-\frac{\partial p}{\partial x}+\frac{\partial}{\partial x}\left[\mu\left(2\frac{\partial u}{\partial x}-\tfrac{2}{3}\operatorname{div}\mathbf{u}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right]+S_x$$

$$\rho\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}\right)=-\frac{\partial p}{\partial y}+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(2\frac{\partial v}{\partial y}-\tfrac{2}{3}\operatorname{div}\mathbf{u}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+S_y \qquad \textit{Equation A 3-2}$$

$$\rho\left(\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}\right)=-\frac{\partial p}{\partial z}+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(2\frac{\partial u}{\partial z}-\tfrac{2}{3}\operatorname{div}\mathbf{u}\right)\right]+S_z$$

These may be written in an alternative format. Taking the x direction as an example, the RHS of Equation A 3-2 may be written as:

$$=-\frac{\partial p}{\partial x}+\mu\left[2\frac{\partial^2 u}{\partial x^2}-\tfrac{2}{3}\left(\frac{\partial^2 u}{\partial x\partial x}+\frac{\partial^2 v}{\partial x\partial y}+\frac{\partial^2 w}{\partial x\partial z}\right)+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 v}{\partial x\partial y}+\frac{\partial^2 w}{\partial x\partial z}+\frac{\partial^2 u}{\partial z^2}\right]+S_x$$

$$=-\frac{\partial p}{\partial x}+\mu\left[\tfrac{4}{3}\frac{\partial^2 u}{\partial x^2}+\tfrac{1}{3}\frac{\partial^2 v}{\partial x\partial y}+\tfrac{1}{3}\frac{\partial^2 w}{\partial x\partial z}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right]+S_x$$

$$=-\frac{\partial p}{\partial x}+\tfrac{1}{3}\mu\frac{\partial}{\partial x}\left[\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right]+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+S_x$$

Thus the complete Navier-Stokes equations may also be written as:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) + S_x + \left\{\frac{1}{3}\nu\frac{\partial}{\partial x}\left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right]\right\}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) + S_y + \left\{\frac{1}{3}\nu\frac{\partial}{\partial y}\left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right]\right\} \qquad \textit{Equation A 3-3}$$

$$\underbrace{\frac{\partial w}{\partial t}}_{\text{unsteady}} + \underbrace{u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z}}_{\text{convection}} = \underbrace{-\frac{1}{\rho}\frac{\partial p}{\partial z}}_{\text{pressure force}} + \underbrace{\nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right)}_{\text{diffusion}} + \underbrace{S_z}_{\text{body force}} + \underbrace{\left\{\frac{1}{3}\nu\frac{\partial}{\partial z}\left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right]\right\}}_{\text{zero for const } \rho}$$

The source term consists of the pressure forces and the body forces. Note that each term in the Navier-Stokes equations has the units of acceleration, i.e. rate of change of specific momentum (momentum per unit mass, i.e. velocity).

For flows of constant density, the continuity equation:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad \text{or} \quad \text{div}\,\boldsymbol{u} = 0 \qquad \textit{Equation A 3-4}$$

makes the last term on the RHS of Equation A 3-3 equal to zero.

Equation A 3-3 may then be written in vector notation as:

$$\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t} = -\rho\nabla p + \nu\nabla^2\mathbf{u} + S \quad \text{where} \quad \frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t} \text{ is equal to the LHS of Equation A 3-3.}$$

or in tensor notation as:

$$\frac{\partial u_\alpha}{\partial t}+u_\beta\frac{\partial u_\alpha}{\partial x_\beta}=-\frac{1}{\rho}\frac{\partial p}{\partial x_\alpha}+\nu\frac{\partial^2 u_\alpha}{\partial x_\beta^{\,2}}+S_\alpha$$

A3.3 Reynolds Equations

Turbulent flow variables may be represented by a mean quantity plus a fluctuating component. Assuming no body forces apply, and the flow is incompressible, i.e. div $\boldsymbol{u} = 0$, then substituting:

$$u=U+u'$$
$$v=V+v'$$
$$w=W+w'$$
$$p=P+p'$$

into Equation A 3-2 (taking the x component as an example) gives:

$$\rho\left[\frac{\partial(U+u')}{\partial t}+\frac{\partial(U^2+2Uu'+u'^2)}{\partial x}+\frac{\partial(UV+Vu'+v'U+u'v')}{\partial y}+\frac{\partial(UW+Wu'+w'U+u'w')}{\partial z}\right]$$
$$=\frac{\partial(P+p')}{\partial t}+2\mu\frac{\partial}{\partial x}\left(\frac{\partial(U+u')}{\partial x}\right)+\mu\frac{\partial}{\partial y}\left(\frac{\partial(U+u')}{\partial y}+\frac{\partial(V+v')}{\partial x}\right)+\mu\frac{\partial}{\partial z}\left(\frac{\partial(W+w')}{\partial x}+\frac{\partial(U+u')}{\partial z}\right)$$

Taking a time average of this equation will cancel many of the fluctuating components resulting in the following system of equations:

$$\rho\left[\frac{\partial}{\partial x}\left(U^2+\overline{u'^2}\right)+\frac{\partial}{\partial y}\left(UV+\overline{u'v'}\right)+\frac{\partial}{\partial z}\left(UW+\overline{u'w'}\right)\right]=-\frac{\partial P}{\partial x}+\frac{\partial}{\partial x}\left(2\mu\frac{\partial U}{\partial x}\right)+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial W}{\partial x}+\frac{\partial U}{\partial z}\right)\right]$$

$$\rho\left[\frac{\partial}{\partial x}\left(UV+\overline{u'v'}\right)+\frac{\partial}{\partial y}\left(V^2+\overline{v'^2}\right)+\frac{\partial}{\partial z}\left(VW+\overline{v'w'}\right)\right]=-\frac{\partial P}{\partial y}+\frac{\partial}{\partial x}\left(\mu\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)\right)+\frac{\partial}{\partial y}\left[2\mu\frac{\partial V}{\partial y}\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial V}{\partial z}+\frac{\partial W}{\partial y}\right)\right] \qquad \textit{Equation A 3-5}$$

$$\rho\left[\frac{\partial}{\partial x}\left(UW+\overline{u'w'}\right)+\frac{\partial}{\partial y}\left(WV+\overline{v'w'}\right)+\frac{\partial}{\partial z}\left(W^2+\overline{w'^2}\right)\right]=-\frac{\partial P}{\partial z}+\frac{\partial}{\partial x}\left(\mu\left(\frac{\partial W}{\partial x}+\frac{\partial U}{\partial z}\right)\right)+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial W}{\partial y}+\frac{\partial V}{\partial z}\right)\right]+\frac{\partial}{\partial z}\left[2\mu\frac{\partial W}{\partial z}\right]$$

These may be rearranged to give:

$$\rho\left[\frac{\partial}{\partial x}\left(U^2\right)+\frac{\partial}{\partial y}\left(UV\right)+\frac{\partial}{\partial z}\left(UW\right)\right]=-\frac{\partial P}{\partial x}+\frac{\partial}{\partial x}\left(2\mu\frac{\partial U}{\partial x}\right)+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial W}{\partial x}+\frac{\partial U}{\partial z}\right)\right]-\rho\left[\frac{\partial\left(\overline{u'^2}\right)}{\partial x}+\frac{\partial\left(\overline{u'v'}\right)}{\partial y}+\frac{\partial\left(\overline{u'w'}\right)}{\partial z}\right]$$

$$\rho\left[\frac{\partial}{\partial x}\left(UV\right)+\frac{\partial}{\partial y}\left(V^2\right)+\frac{\partial}{\partial z}\left(VW\right)\right]=-\frac{\partial P}{\partial y}+\frac{\partial}{\partial x}\left(\mu\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)\right)+\frac{\partial}{\partial y}\left[2\mu\frac{\partial V}{\partial y}\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial V}{\partial z}+\frac{\partial W}{\partial y}\right)\right]-\rho\left[\frac{\partial\left(\overline{v'u'}\right)}{\partial x}+\frac{\partial\left(\overline{v'^2}\right)}{\partial y}+\frac{\partial\left(\overline{v'w'}\right)}{\partial z}\right]$$

$$\rho\left[\frac{\partial}{\partial x}\left(UW\right)+\frac{\partial}{\partial y}\left(WV\right)+\frac{\partial}{\partial z}\left(W^2\right)\right]=-\frac{\partial P}{\partial z}+\frac{\partial}{\partial x}\left(\mu\left(\frac{\partial W}{\partial x}+\frac{\partial U}{\partial z}\right)\right)+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial W}{\partial y}+\frac{\partial V}{\partial z}\right)\right]+\frac{\partial}{\partial z}\left[2\mu\frac{\partial W}{\partial z}\right]-\rho\underbrace{\left[\frac{\partial\left(\overline{w'u'}\right)}{\partial x}+\frac{\partial\left(\overline{w'v'}\right)}{\partial y}+\frac{\partial\left(\overline{w'^2}\right)}{\partial z}\right]}_{\text{Reynold's Stress terms}}$$

Equation A 3-6

or

$$\frac{\partial}{\partial x}\underbrace{\left[P-2\mu\frac{\partial U}{\partial x}+\rho U^2+\rho\overline{u'^2}\right]}_{-\sigma_x}+\frac{\partial}{\partial y}\underbrace{\left[\rho UV+\rho\overline{v'u'}-\mu\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)\right]}_{-\tau_{yx}}+\frac{\partial}{\partial z}\underbrace{\left[\rho UW+\rho\overline{u'w'}-\mu\left(\frac{\partial W}{\partial x}+\frac{\partial U}{\partial z}\right)\right]}_{-\tau_{zx}}=0$$

$$\frac{\partial}{\partial x}\underbrace{\left[\rho UV+\rho\overline{v'u'}-\mu\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)\right]}_{-\tau_{xy}}+\frac{\partial}{\partial y}\underbrace{\left[P-2\mu\frac{\partial V}{\partial y}+\rho V^2+\rho\overline{v'^2}\right]}_{-\sigma_y}+\frac{\partial}{\partial z}\underbrace{\left[\rho VW+\rho\overline{v'w'}-\mu\left(\frac{\partial V}{\partial z}+\frac{\partial W}{\partial y}\right)\right]}_{-\tau_{zy}}=0 \qquad \textit{Equation A 3-7}$$

$$\frac{\partial}{\partial x}\underbrace{\left[\rho UW+\rho\overline{w'u'}-\mu\left(\frac{\partial U}{\partial z}+\frac{\partial W}{\partial x}\right)\right]}_{-\tau_{xz}}+\frac{\partial}{\partial y}\underbrace{\left[\rho VW+\rho\overline{v'w'}-\mu\left(\frac{\partial V}{\partial z}+\frac{\partial W}{\partial y}\right)\right]}_{-\tau_{yz}}+\frac{\partial}{\partial z}\underbrace{\left[P-2\mu\frac{\partial W}{\partial z}+\rho W^2+\rho\overline{w'^2}\right]}_{-\sigma_z}=0$$

following the shear stress convention whereby the first subscript denotes the plane on which the shear stress acts and the second subscript denotes the direction of the shear stress. Note that $\sigma_x = \tau_{xx}$ etc.

A3.4 Turbulence Modelling

Turbulence modelling describes the way in which the unknown Reynolds stress terms (see Equation A 3-6) are evaluated. The most common methods involve the eddy viscosity concept, as described by Rodi (1984). In tensor notation this is:

$$-\overline{u'_\alpha u'}_\beta = \nu_t\left(\frac{\partial U_\alpha}{\partial x_\beta}+\frac{\partial U_\beta}{\partial x_\alpha}\right)-\tfrac{2}{3}k\delta_{\alpha\beta} \qquad \textit{Equation A 3-8}$$

where k is the kinetic energy of turbulence, and the Kronecker delta $\delta_{\alpha\beta}$ is an operator which implies,

$\delta_{\alpha\beta} = 1$ for $\alpha = \beta$ and $\delta_{\alpha\beta} = 0$ for $\alpha \neq \beta$.

So, for example, the x component Reynolds stress are:

$$-\overline{u'^2} = 2\nu_t \frac{\partial U}{\partial x} - \tfrac{2}{3}k$$

$$-\overline{u'v'} = \nu_t \left(\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x} \right)$$

$$-\overline{u'w'} = \nu_t \left(\frac{\partial U}{\partial z} + \frac{\partial W}{\partial x} \right)$$

and for the y component:

$$-\overline{v'^2} = 2\nu_t \frac{\partial V}{\partial y} - \tfrac{2}{3}k$$

$$-\overline{v'u'} = \nu_t \left(\frac{\partial V}{\partial x} + \frac{\partial U}{\partial y} \right)$$

$$-\overline{v'w'} = \nu_t \left(\frac{\partial V}{\partial z} + \frac{\partial W}{\partial y} \right)$$

and the z component:

$$-\overline{w'^2} = 2\nu_t \frac{\partial W}{\partial z} - \tfrac{2}{3}k$$

$$-\overline{w'u'} = \nu_t\left(\frac{\partial W}{\partial x} + \frac{\partial U}{\partial z}\right)$$

$$-\overline{w'v'} = \nu_t\left(\frac{\partial W}{\partial y} + \frac{\partial V}{\partial z}\right)$$

Noting that the kinetic energy of turbulence is $k = \tfrac{1}{2}\left(\overline{u'^2} + \overline{v'^2} + \overline{w'^2}\right)$, it can be seen that the term $^2/_3 k$ in the normal stress $-\overline{u'^2}$ arises to ensure that the sum of all three components of normal stress is equal to $2k$.

Substituting these Reynolds terms into Equation A 3-7 gives:

$$\frac{\partial}{\partial x}\underbrace{\left[P + \rho U^2 - 2(\mu+\mu_t)\frac{\partial U}{\partial x} + \tfrac{2}{3}\rho k\right]}_{-\sigma_x} + \frac{\partial}{\partial y}\underbrace{\left[\rho UV - (\mu+\mu_t)\left(\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}\right)\right]}_{-\tau_{yx}} + \frac{\partial}{\partial z}\underbrace{\left[\rho UW - (\mu+\mu_t)\left(\frac{\partial W}{\partial x} + \frac{\partial U}{\partial z}\right)\right]}_{-\tau_{zx}} = 0$$

$$\frac{\partial}{\partial x}\underbrace{\left[\rho UV - (\mu+\mu_t)\left(\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}\right)\right]}_{-\tau_{xy}} + \frac{\partial}{\partial y}\underbrace{\left[P + \rho V^2 - 2(\mu+\mu_t)\frac{\partial V}{\partial y} + \tfrac{2}{3}\rho k\right]}_{-\sigma_y} + \frac{\partial}{\partial z}\underbrace{\left[\rho VW - (\mu+\mu_t)\left(\frac{\partial W}{\partial y} + \frac{\partial V}{\partial z}\right)\right]}_{-\tau_{zy}} = 0 \qquad \textit{Equation A 3-9}$$

$$\frac{\partial}{\partial x}\underbrace{\left[\rho UW - (\mu+\mu_t)\left(\frac{\partial U}{\partial z} + \frac{\partial W}{\partial x}\right)\right]}_{-\tau_{xy}} + \frac{\partial}{\partial y}\underbrace{\left[\rho VW - (\mu+\mu_t)\left(\frac{\partial W}{\partial y} + \frac{\partial V}{\partial z}\right)\right]}_{-\tau_{yz}} + \frac{\partial}{\partial z}\underbrace{\left[P + \rho W^2 - 2(\mu+\mu_t)\frac{\partial W}{\partial z} + \tfrac{2}{3}\rho k\right]}_{-\sigma_y} = 0$$

This may be simplified further by using $\Gamma = \mu + \mu_t$ and applying the continuity equation (again using the x component as an example):

$$\frac{\partial(\rho UV)}{\partial y} - \Gamma\frac{\partial}{\partial y}\left(\partial U/\partial y\right) - \Gamma\frac{\partial}{\partial y}\left(\partial V/\partial x\right) + \frac{\partial(\rho UW)}{\partial z} - \Gamma\frac{\partial}{zy}\left(\partial W/\partial x\right) - \Gamma\frac{\partial}{\partial z}\left(\partial U/\partial z\right) + \frac{\partial\left(P + \rho U^2 + 2/3\,\rho k\right)}{\partial x} - 2\Gamma\frac{\partial}{\partial x}\left(\partial U/\partial x\right) = 0$$

$$\rightarrow \quad -\Gamma\frac{\partial^2 V}{\partial y\partial x} - \Gamma\frac{\partial^2 W}{\partial z\partial x} - \Gamma\frac{\partial^2 U}{\partial x\partial x} + \frac{\partial(\rho UV)}{\partial y} - \Gamma\frac{\partial}{\partial y}\left(\partial U/\partial y\right) + \frac{\partial(\rho UW)}{\partial z} - \Gamma\frac{\partial}{\partial z}\left(\partial U/\partial z\right) + \frac{\partial\left(P + \rho U^2 + 2/3\,\rho k\right)}{\partial x} - \Gamma\frac{\partial}{\partial x}\left(\partial U/\partial x\right) = 0$$

$$\rightarrow \quad -\Gamma\frac{\partial}{\partial x}\left(\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z}\right) + \frac{\partial}{\partial x}\left(P + \rho U^2 + 2/3\,\rho k + \Gamma\frac{\partial U}{\partial x}\right) + \frac{\partial}{\partial y}\left(\rho UV - \Gamma\frac{\partial U}{\partial y}\right) + \frac{\partial}{\partial y}\left(\rho UW - \Gamma\frac{\partial U}{\partial z}\right) = 0$$

which, for incompressible flow and all three components gives:

$$\frac{\partial}{\partial x}\left[P + \rho U^2 - \Gamma\frac{\partial U}{\partial x} + 2/3\,\rho k\right] + \frac{\partial}{\partial y}\left[\rho UV - \Gamma\frac{\partial U}{\partial y}\right] + \frac{\partial}{\partial z}\left[\rho UW - \Gamma\frac{\partial U}{\partial z}\right] = 0$$

$$\frac{\partial}{\partial x}\left[\rho UV - \Gamma\frac{\partial V}{\partial x}\right] + \frac{\partial}{\partial y}\left[P + \rho V^2 - \Gamma\frac{\partial V}{\partial y} + 2/3\,\rho k\right] + \frac{\partial}{\partial z}\left[\rho VW - \Gamma\frac{\partial V}{\partial z}\right] = 0 \qquad \textit{Equation A 3-10}$$

$$\frac{\partial}{\partial x}\left[\rho UW - \Gamma\frac{\partial W}{\partial x}\right] + \frac{\partial}{\partial y}\left[\rho VW - \Gamma\frac{\partial W}{\partial y}\right] + \frac{\partial}{\partial z}\left[P + \rho W^2 - \Gamma\frac{\partial W}{\partial z} + 2/3\,\rho k\right] = 0$$

It is the Reynolds equations in this form which are solved by PHOENICS. However note that these individual terms no longer form the complete stresses over a finite control volume.

An alternative form of this equation is:

$$\frac{\partial U^2}{\partial x}+\frac{\partial UV}{\partial y}+\frac{\partial UW}{\partial z}=\Gamma/\rho\left[\frac{\partial^2 U}{\partial y^2}+\frac{\partial^2 V}{\partial y\partial x}+\frac{\partial^2 W}{\partial z\partial x}+\frac{\partial^2 U}{\partial z^2}+2\frac{\partial^2 U}{\partial x^2}\right]-2/3\frac{\partial k}{\partial x}-1/\rho\frac{\partial P}{\partial x}$$

$$\frac{\partial UV}{\partial x}+\frac{\partial V^2}{\partial y}+\frac{\partial VW}{\partial z}=\Gamma/\rho\left[\frac{\partial^2 V}{\partial x^2}+\frac{\partial^2 U}{\partial x\partial y}+\frac{\partial^2 V}{\partial z^2}+\frac{\partial^2 W}{\partial z\partial y}+2\frac{\partial^2 V}{\partial y^2}\right]-2/3\frac{\partial k}{\partial y}-1/\rho\frac{\partial P}{\partial y}$$

Equation A 3-11

$$\underbrace{\frac{\partial UW}{\partial x}+\frac{\partial WV}{\partial y}+\frac{\partial W^2}{\partial z}}_{\text{source}}=\underbrace{\Gamma/\rho\left[\frac{\partial^2 W}{\partial x^2}+\frac{\partial^2 U}{\partial x\partial y}+\frac{\partial^2 W}{\partial y^2}+\frac{\partial^2 V}{\partial y\partial z}+2\frac{\partial^2 W}{\partial z^2}\right]}_{\text{diffusion}}\underbrace{-2/3\frac{\partial k}{\partial z}-1/\rho\frac{\partial P}{\partial z}}_{\text{source}}$$

where $\Gamma/\rho=\nu+\nu_t$ and the turbulent viscosity ν_t may be evaluated from a turbulence model such as the k-ε model:

$$\nu_t=c_\mu\frac{k^2}{\varepsilon}$$

Equation A 3-12

where c_μ = coefficient = 0.09, k = kinetic energy of turbulence and ε = dissipation rate of turbulent kinetic energy.

Appendix 4: Errors and Modifications to the PHOENICS Code

A4.1 List of Errors and Modifications

During the course of this study several errors were found within the PHOENICS code (from versions 1.6 to 2.11). For the purpose of describing the code used in this study, some of the significant errors and modifications employed are shown below. The dates given at the beginning of each entry indicate when the problem was reported to CHAM. In some cases CHAM has corrected the errors in later releases of the code.

1. [24/5/94] Activation of the two-layer model. The PIL command to activate the two-layer model is

```
TURMOD (KEMODL-2L)
```

however this command is not recognised by SATELLITE. To activate this model the lower level commands

```
TURMOD (KEMODL)
IENUTA=8
DISWALL
```

should be used.

2. [2/6/94] Calculation of the wall shear stress. Use of the Q1 command YPLS=T elicits printout of wall shear stress, non-dimensional distance from the wall (Y^+) and the skin friction coefficient. However two errors were found in relation to the wall shear stress:

a) The printout of shear stress (SHSTRESS) is actually τ_w/ρ

b) When using the two-layer model the skin friction inside the laminar sub-layer is incorrectly calculated in the function FNSKIN (subroutine GXWALL.F), part of the CHAM supplied GREX library of GROUND subroutines. The line

```
F(L0STRS+J)=SKINF*F(L0RLVL+J)**2
```

should read

```
F(L0STRS+J)=F(L0SK+J)*F(L0RLVL+J)**2
```
The correct implementation is shown in the printout of FNSKIN in Section A4.2.

3. [8/6/94] Error in definition of blockages. The method of defining blockages in PHOENICS is to use the CONPOR command over a patch of cells; for example from BNXF (block NX first) to BNXL (block NX last) and similarly in the y and z directions:

```
CONPOR (BLOCK,0.0,CELL,BNXF,BNXL,BNYF,BNYL,BNZF,BNZL)
```

However this results in erroneous velocities being calculated inside the upstream cell of the block. To prevent this, the following lines must also be added (for flow in the +ve x direction):

```
PATCH(INIU,INIVAL,BNXF,BNXF,BNYF,BNYL,BNZF,BNZL,1,1)
INIT(INIU,U1,0.0,0.0)
```

This extra code manually sets the velocity on the upstream face of the block to be zero.

4. [3/10/94] Typing error in documentation. In the description of the block location indices of the F-array for patchwise variables (p6.14 of TR200b) the line

```
LF=L0F(LB)+(IY-IYF+1)+(IYL-IYF+1)*(IX-IXF+1)+(IYL-IYF+1)*(IXL-IXF+1)*(IZ-IZF+1)
```

should read

```
LF=L0F(LB)+(IY-IYF+1)+(IYL-IYF+1)*(IX-IXF)+(IYL-IYF+1)*(IXL-IXF+1)*(IZ-IZF+1)
```

5. [3/10/94] Specifying patchwise variables. Contrary to the documentation, the patch name which forms the argument of the GROUND function IPNAME must contain exactly 8 characters. Hence the additional character '8' in the line:

```
L0SB=L0PVAR(PVSTRS,IPNAME('WALLBED8'),IZ)
```

6. [20/3/95] Multi-block grids. Numerous errors were encountered in the first release of the multi-block grid feature. These are listed in Appendix 7: Multi-block Grids.

7. [19/5/95] Error in calculation of non-dimensional wall distance. Further errors were found in the function FNSKIN (subroutine GXWALL.F) relating to the calculation of Y^+. The corrected function is given in Section A4.2 with current modifications commented with a line prefixed by:

```
C#### aks 19/05/95
```

8. [15/9/95] Access to group 19 in GROUND.

a) The group 19 of the ground subroutine is called twice for each sweep from the parent program. This may have consequences in any user code placed here. To ensure that any code in sections 3 and 6 is activated once only, it should be located inside the following IF statement:

```
IF (LOOPZ.EQ.1) THEN
C       user coding
END IF
```

Note that the following COMMON block must also be included;

```
COMMON /GENI/IGFL1(45),LOOPZ,IGFL2(14)
```

b) In a two-dimensional simulation, group 19 is called from ISWEEP=1 to LSWEEP. For a three-dimensional simulation it is called from ISWEEP=0 to LSWEEP.

9. [30/1/96] Typing error in documentation. The POLIS entry for wall functions give two different values for the roughness parameter E for smooth walls (8.6 and 9.0) in different parts of the entry. The actual value used within the PHOENICS wall function is 8.6.

10. [29/5/96] Inaccuracies within wall function. Inaccuracies were found in the calculation of velocity at a near wall cell when using the standard k-ε model with wall functions. The cause of this inaccuracy was traced to the calculation of the skin friction factor, which is calculated within the function FNSKIN by:

$$\frac{\tau}{\rho U^2} = \left[\frac{\kappa}{\ln\left(1.01 + E\,\mathrm{Re}\sqrt{\frac{\tau}{\rho U^2}}\right)} \right]$$

Equation A4-1

The addition of the term 1.01 is to prevent a denominator of zero. Re-arrangement gives the velocity profile as:

$$\frac{U}{U_\tau} = \frac{1}{\kappa}\ln\left(1.01 + E^{U_\tau y}/_{\nu}\right)$$

which is identical to the standard wall function, apart from the inclusion of the term 1.01. It is this term which under some circumstances may produce a significant error. Examination of the possible conditions gives:

1) If $\left|E\,\mathrm{Re}\sqrt{\frac{\tau}{\rho U^2}}\right|$ is large then the addition of 1.01 is inconsequential.

2) If $\left|E\,\mathrm{Re}\sqrt{\frac{\tau}{\rho U^2}}\right|$ is slightly greater than one the addition of 1.01 will still be unnecessary, but will produce significant error.

3) If $E\,\mathrm{Re}\sqrt{\frac{\tau}{\rho U^2}}$ is =1.00000.... then the addition of 1.01 will prevent a division by zero, but will also give a significant error.

For this study an improved scheme was used whereby a conditional statement was included to the effect that if

$$E\,\mathrm{Re}\sqrt{\frac{\tau}{\rho U^2}} = 1.0000$$

then the value would be changed to 1.01 and a warning message would be printed. The 1.01 term could then safely be removed. This revised scheme is also shown in the corrected function FNSKIN given in Section A4.2 (comments prefixed with C### aks 29/5/96).

11. [1/10/96] Typing error in documentation. The POLIS entry for the false time step method of relaxation contains:

".... This has the effect of making ϕ_{next} very close to ϕ_{latest} when DTFALS is **large**, so that changes can occur only after many repetitions of the adjustment process."

This sentence should read:

".... This has the effect of making ϕ_{next} very close to ϕ_{latest} when DTFALS is **small**, so that changes can occur only after many repetitions of the adjustment process."

12. [1/4/97] Error in calculation of IX. At the end of the solution procedure (in GROUND, section 8 of Group 19), the value of the variable IX (denoting the current x cell) becomes equal to NX+1, i.e. greater than the total number of x cells in the domain. Consequently, the standard method of extracting values from the F-array using

```
I=IY+NY*(IX-1)
VAR=F(L0VAR+I)
```

will give incorrect values unless IX is redefined.

A4.2 Corrected version of FNSKIN (part of GXWALL.F)

```
C*********************************************************************
      SUBROUTINE FNSKIN
C**** This subroutine calculates the dimensionless skin-friction
C     factor SKINF = stress/(rho * relvel**2) from one of:-
C                    the Blasius law
C                    the logarithmic law
C                    the generalised log law
C
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      COMMON/SODAL/DBG,LOGIC(4)
      COMMON/RDATA/TINY,RDFIL1(17),ENUL,ENUT,RDFIL2(5),GRND,RDFIL3(59)
      COMMON/IDATA/NX,NY,IDFIL1(23),FSWEEP,LSWEEP,IDFIL2(51),VISL,
     1          IDFIL3(39),VIST,NPHI
      COMMON/TURB3/RTTDKE,AKC,EWC,ACON,TAUDKE
      COMMON/LWFUN2/DHCHKD(100)
      COMMON/LWFUN1/DOSKIN(100)
      COMMON /LWFUN/WALL,STRGNR
      COMMON/LDATA/LDAT1(56),WALPRN,LDAT2(27)
      LOGICAL WALL,STRGNR,DHCHKD,GTZ,ROUGH,DOSKIN,STORKE,STORGN
C#### GTR 21.02.94 ADD NUTNE0
      LOGICAL DBG,LOGIC,NEZ,LDAT1,LDAT2,NUTNE0
      COMMON/NAMFN/NAMFUN,NAMSUB
      CHARACTER*6 NAMFUN,NAMSUB
      INTEGER FSWEEP,VISL,VIST,COGRN
      DATA KOUNT/0/
C
      NAMFUN='FNSKIN'
      COGRN=ISC-1
      if(dbg) then
        call writ40('entry to fnskin called from gxwfun      ')
        call writ2i('indvar  ',indvar,'cogrn   ',cogrn)
      endif
C.... Blasius formula  s=0.0395/(Re)**0.25
C     with low-Reynolds-number modification
C.... ld11 contains laminar viscosity divided by wall distance,
C     ie the laminar wall coefficient.
      L0MUDD=L0F(LD11)
      L0SK=  L0PVAR(PVSKIN,IREG,0)
      L0RLVL=L0PVAR(PVRLVL,IREG,0)
```

```
      L0STRS=L0PVAR(PVSTRS,IREG,0)
      STORKE=.FALSE.
      STORGN=.FALSE.
C
      IF(COGRN.EQ.1) THEN
C.... GRND1
        CTURB=0.0395
        CRIT=0.1
      ELSE
C.... GRND2 or 3
        STORKE=STORE(KE)
C#### GTR 21.02.94 ADD NUTNE0
        NUTNE0=NEZ(ENUT)
        IF(NUTNE0) L0YPLS=L0PVAR(PVYPLS,IREG,0)
C#### mrm 01.03.94 USE NUTNE0
        STORGN=STORE(LGEN1).AND.NUTNE0
        IF(STORGN) THEN
          L0GEN1=L0F(LGEN1)
          L0GENR = L0PVAR(PVGENR,IREG,0)
          L0GENW = L0PVAR(PVDISS,IREG,0)
          IF(ENUT.LT.0.0) L0VIST=L0F(VIST)
          L0SU=L0F(LSU)
          L0M1=L0F(LM1)
        ENDIF
        L0RCDS=L0PVAR(PVRCDS,IREG,0)
        ROUGH=GTZ(WALLA)
        IF(COGRN.EQ.2) THEN
C.... GRND2
          if(dbg) call writ1l('rough   ',rough)
C.... It is possible to test the effect of the number of iterations
C     on the solution by setting ISKINA in SATELLITE
          NUMITS=5+ISKINA
          SHALFM=1.0/11.5
        ELSEIF(COGRN.EQ.3) THEN
C.... GRND3
          L0KE=L0F(KE)
        ENDIF
      ENDIF
C.... Start of DO loop
      J=0
      IPLUS=(IXF-2)*NY
      KOUNT=12
      IF(.NOT.LDAT1(32)) THEN
        IF(KOUNT.EQ.12.AND.ISWEEP.EQ.FSWEEP + 1) KOUNT=0
        IF(ISWEEP.EQ.LSWEEP-1.AND.KOUNT.EQ.0) KOUNT=1
      ENDIF
      DO 20 IX=IXF,IXL
        IPLUS=IPLUS+NY
CDIR$ IVDEP
        DO 20 IY=IYF,IYL
          I=IY+IPLUS
          J=J+1
C.... Reynolds number = relvel/(kinematic viscosity / wall distance)
          REYN=F(L0RLVL+J)/(F(L0MUDD+I) + TINY)
          SKINFL=1.0/(REYN + 1.E-20)
C#### aks 19/595 FNSKIN: correct error in YPLS for laminar flow
          SLHALF=SQRT(SKINFL)
          IF(COGRN.EQ.1) THEN
```

```
C.... Blasius law (GRND1)
          IF(REYN.GT.CRIT) THEN
            SKINF=CTURB * REYN ** (-0.25)
          ELSE
            SKINF=SKINFL
          ENDIF
        ELSEIF(COGRN.EQ.2) THEN
C.... Log law (GRND2, smooth or GRND3, rough)
          IF(ROUGH) THEN
            REYROU=REYN*WALLA*F(L0RCDS+J)
          ELSE
            EREYNO=EWAL*REYN
          ENDIF
          FL0SK=AMAX1(F(L0SK+J),1.E-4)
          SHALF=AMIN1(SQRT(fl0sk),SHALFM)
C.... Iterative solution of: SHALF=AK/LN( E*RE*SHALF)
C### aks 29/5/96          1.01+ removed ^
          DO 10 ITS=1,NUMITS
            IF(ROUGH) THEN
              L0EA2=L0F(EASP2)
              E=EW(SHALF*REYROU)
              EREYNO=E*REYN
            ENDIF
            ARG= EREYNO*SHALF
C### aks 29/5/96  ^ 1.01+ removed
C### aks 29/5/96  check on ARG added.
            IF(ARG.EQ.1.0) THEN
              write(14,*) 'ERROR in FNSKIN !!! '
              write(14,*) 'Argument of skin friction log_e = 1.0 '
              write(14,*) 'Value altered to 1.01 '
              ARG=1.01
            ENDIF

            IF(ISKINB.GT.0) THEN
              AKPLUS=AK/SHALF
              ARG1=(((0.04166667*AKPLUS+0.1666667)*AKPLUS
     1              +0.5)*AKPLUS+1.)*AKPLUS+1.
              ARG=ARG+ARG1
            ENDIF
   10       SHALF=AK/ALOG(ARG)
C#### GTR 21.02.94 ADD CHECK ON ENUT
C#### aks 19/5/95 FNSKIN: correct error in YPLS for laminar flow
          IF(NUTNE0) F(L0YPLS+J)=AMAX1(REYN*SHALF,REYN*SLHALF)
          SKINF=SHALF*SHALF
          IF(ROUGH) F(L0EA2+I)=E
        ELSEIF(COGRN.EQ.3) THEN
C.... Generalised log law, GRND3
          SQRK=SQRT(F(L0KE+I))
          SQKDVS=SQRK*F(L0RCDS+J)/F(L0MUDD+I)
          IF(ROUGH) THEN
            RERDH=RTTDKE*SQKDVS
            TERM=EW(RERDH*WALLA)*RERDH
          ELSE
            TERM=EWC * SQKDVS
          ENDIF
          SKINF=AKC*SQRK/(ALOG( TERM/F(L0RCDS+J)) * F(L0RLVL+J))
C### aks 29/5/96              ^ 1.01+ removed
C#### GTR 21.02.94 ADD CHECK ON ENUT
```

```
C#### aks 19/5/95 FNSKIN: correct error in YPLS for laminar flow
          IF(NUTNE0) F(L0YPLS+J)=AMAX1(RTTDKE*SQRK/F(L0MUDD+I),
     1                               REYN*SLHALF)
        ENDIF
        F(L0SK+J)=AMAX1(SKINF,SKINFL)
C.... Put into patchwise store PVSTRS the turbulent shear stress
C     divided by density
C#### mrm 01/03/94 change STORKE to LGEN1.NE.0 so that near-wall
C          generation rate computed for algebraic turbulence models
        IF(LGEN1.NE.0) THEN
C.... Note that it would be more appropriate to augment the relative
C     velocity by some factor which allowed for the fact that the
C     outer boundary of the cell probably has the same shear stress
C     but a higher relative velocity.
c#### aks 2/6/94 change SKINF to F(L0SK+J)
          F(L0STRS+J)=F(L0SK+J)*F(L0RLVL+J)**2
          IF(STORGN) THEN
C.... Here the generation rate computed in GREX is modified. It has
C     already been added to the source of KE by reason of the KESORC
C     patch created by the turmod(kemodl) command. The generation rate
C     to be used in the k-balance equation is placed in the patch-wise
C     store PVGENR after subtraction of the dissipation rate which is
C     calculated but not stored.
            GENRTN = F(L0STRS+J)*F(L0RLVL+J)
C.... The velocity-squared term is adjusted here to an appropriate
C     value for use in the enthalpy equation
            VLGRD2=GENRTN*0.5*F(L0RCDS+J)
            IF(ENUT.GT.0.0) THEN
              F(L0GEN1+I)=VLGRD2/ENUT
            ELSE
              F(L0GEN1+I)=VLGRD2/(F(L0VIST+I)+TINY)
            ENDIF
            F(L0GENW+J)=F(L0GEN1+I)
            IF(COGRN.EQ.3) THEN
C.... Set to zero the (incorrect) sources resulting from the
C     velocity-gradient-squared and epsilon terms resulting,
C     for near-wall cells, from the KESOURCE patch.
              F(L0SU+I)=0.0
C.... Calculate the dissipation rate from a formula which
C     would make DISS equal GENRTN if tau/rho were equal to
C     the constant TAUDKE. Note that:
C                  GENRTN = (tau/rho) * relvel
C                  DISS   = (tau/rho)**1.5/sqrt(skinf)
C             DISS = (F(L0KE+I)*TAUDKE)**1.5/SQRT(SKINF)
              DISS = (F(L0KE+I)*TAUDKE)**1.5*(ALOG( TERM/
     1              F(L0RCDS+J)))/AK
C### aks 29/5/96                    1.01+ removed ^
              F(L0GENR+J) = GENRTN - DISS
            ENDIF
          ENDIF
        ENDIF
  20 CONTINUE
C.... End of DO loop
      DOSKIN(IREG)=.FALSE.
      if(dbg) then
        CALL WRIT40('exit from FNSKIN called from GXWFUN     ')
        call prnpat('skinfrco',pvskin,ireg)
        if(storke) call prnpat('tau/rho ',pvstrs,ireg)
```

```
      if(storgn.and.cogrn.eq.3) then
        call prnpat('gen-diss',pvgenr,ireg)
        call prnpat('lgenwall',pvdiss,ireg)
      endif
    endif
    NAMFUN='fnskin'
    END

c*********************************************************************
```

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

Appendix 5: Coefficients in the Finite Volume Equations

The coefficients in the FVE (Equation 2-62) have the following meanings:

A5.1 Time Dependence

a_T represents time dependence, i.e. the influence of the past on the present. The coefficient consists of:

$$a_T = \frac{\rho_T (\Delta x \Delta y \Delta z)_T}{\Delta_t} \qquad \textit{Equation A5-1}$$

where $(\Delta x \Delta y \Delta z)_T$ is the cell volume at the earlier time (note that cell volumes may change with time), ρ_T is the density at the earlier time and Δ_t is the magnitude of the time interval. For steady flows i.e. $\partial\phi/\partial t = 0$, such as are examined *in this study, this* coefficient is equal to zero.

A5.2 Convection and Diffusion Coefficients

a_E, a_W etc. express the convection and diffusion contributions from neighbouring cells. As an illustrative example, take the case of a steady one-dimensional situation which contains only convection and diffusion. It will be shown that unless the convective term receives special treatment the solution for ϕ becomes physically unrealistic.

The governing differential equation for one-dimensional incompressible flow is:

$$\frac{\partial}{\partial x}(\rho U \phi) = \frac{\partial}{\partial x}\left(\Gamma \partial\phi/\partial x\right) \qquad \textit{Equation A5-2}$$

Integrating this over the control volume shown in Figure A5-1 gives:

$$(\rho U \phi)_e - (\rho U \phi)_w = \left(\Gamma \partial\phi/\partial x\right)_e - \left(\Gamma \partial\phi/\partial x\right)_w \qquad \textit{Equation A5-3}$$

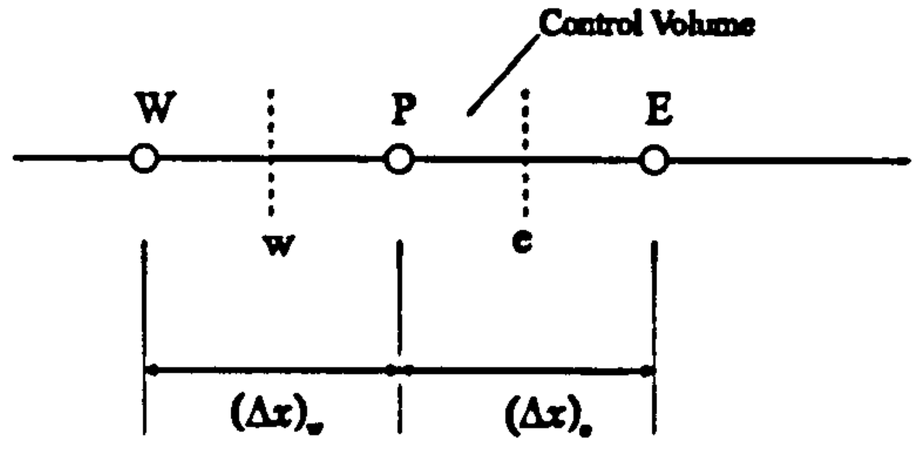


Figure A5-1: One-dimensional control volume

Initially we assume a piece-wise linear profile for ϕ (known as the *central difference* scheme)

$$\phi_e = \frac{1}{2}(\phi_E + \phi_P) \quad \text{and} \quad \phi_w = \frac{1}{2}(\phi_P + \phi_W) \qquad \textit{Equation A5-4}$$

but as will be shown this assumption will lead to a physically unrealistic solution for ϕ. The factor $^1/_2$ arises if the faces 'e' and 'w' are located mid-way between points 'P' and 'E' respectively. Equation A5-3 may now be written as

$$\frac{1}{2}(\rho U)_e(\phi_E + \phi_P) - \frac{1}{2}(\rho U)_w(\phi_P + \phi_W) = \frac{\Gamma_e(\phi_E - \phi_P)}{(\Delta x)_e} - \frac{\Gamma_w(\phi_P - \phi_W)}{(\Delta x)_w} \qquad \textit{Equation A5-5}$$

The values of Γ at faces 'e' and 'w' may be found using either the arithmetic mean

$$\Gamma_e = {(\Gamma_P + \Gamma_E)}/{2}$$

$$\Gamma_w = {(\Gamma_P + \Gamma_W)}/{2} \qquad \textit{Equation A5-6}$$

or the harmonic mean (see Notation for meaning of e(eE) and w(wW))

$$\Gamma_e = \frac{2|PE|}{{|we|}/{\Gamma_P} + {|e(eE)|}/{\Gamma_E}}$$

$$\Gamma_w = \frac{2|PW|}{{|we|}/{\Gamma_P} + {|w(wW)|}/{\Gamma_W}} \qquad \textit{Equation A5-7}$$

both of which are available in PHOENICS through the use of the last argument of the PIL command SOLUTN. In general, harmonic averaging is employed for the momentum equations and the arithmetic mean is used for the continuity equation and turbulence conservation equations.

Equation A5-5 is then cast into the standard form of a FVE:

$$a_P\phi_P = a_E\phi_E + a_W\phi_W \qquad \textit{Equation A5-8}$$

where

$$a_E = \left.\frac{\Gamma}{\Delta x}\right|_e - \frac{\rho U_e}{2}$$

$$a_W = \left.\frac{\Gamma}{\Delta x}\right|_w + \frac{\rho U_w}{2} \qquad \textit{Equation A5-9}$$

$$a_P = a_E + a_W + \underbrace{\rho(U_e - U_w)}_{=0}$$

Note that the last term in a_P goes to zero due to continuity. This now gives the complete FVE. However, problems with this formulation emerge when we attempt to solve the following simple example:

$$\left.\frac{\Gamma}{\Delta x}\right|_e = \left.\frac{\Gamma}{\Delta x}\right|_w = 1 \quad \text{and} \quad \rho U_e = \rho U_w = 4$$

If ϕ_E and ϕ_W are known, as for example: $\phi_E = 200$ and $\phi_W = 100$, then Equation A5-8 gives $\phi_P = 50$, which is clearly unrealistic since the value of ϕ_P cannot lie outside the range of its neighbours. The error is in the assumption of a linear distribution of ϕ across the cell (Equation A5-4).

Improved profile assumption. If Γ is taken to be constant (ρU already being constant from the continuity equation) then the governing equation, Equation A5-2, may in fact be solved analytically. Using the boundary conditions:

$\phi = \phi_0$ at $x = 0$

$\phi = \phi_l$ at $x = l$

gives (Patankar 1980)

$$\frac{\phi - \phi_0}{\phi_l - \phi_0} = \frac{\exp\left(\mathrm{P}\,x/l\right) - 1}{\exp(\mathrm{P}) - 1} \qquad \textit{Equation A5-10}$$

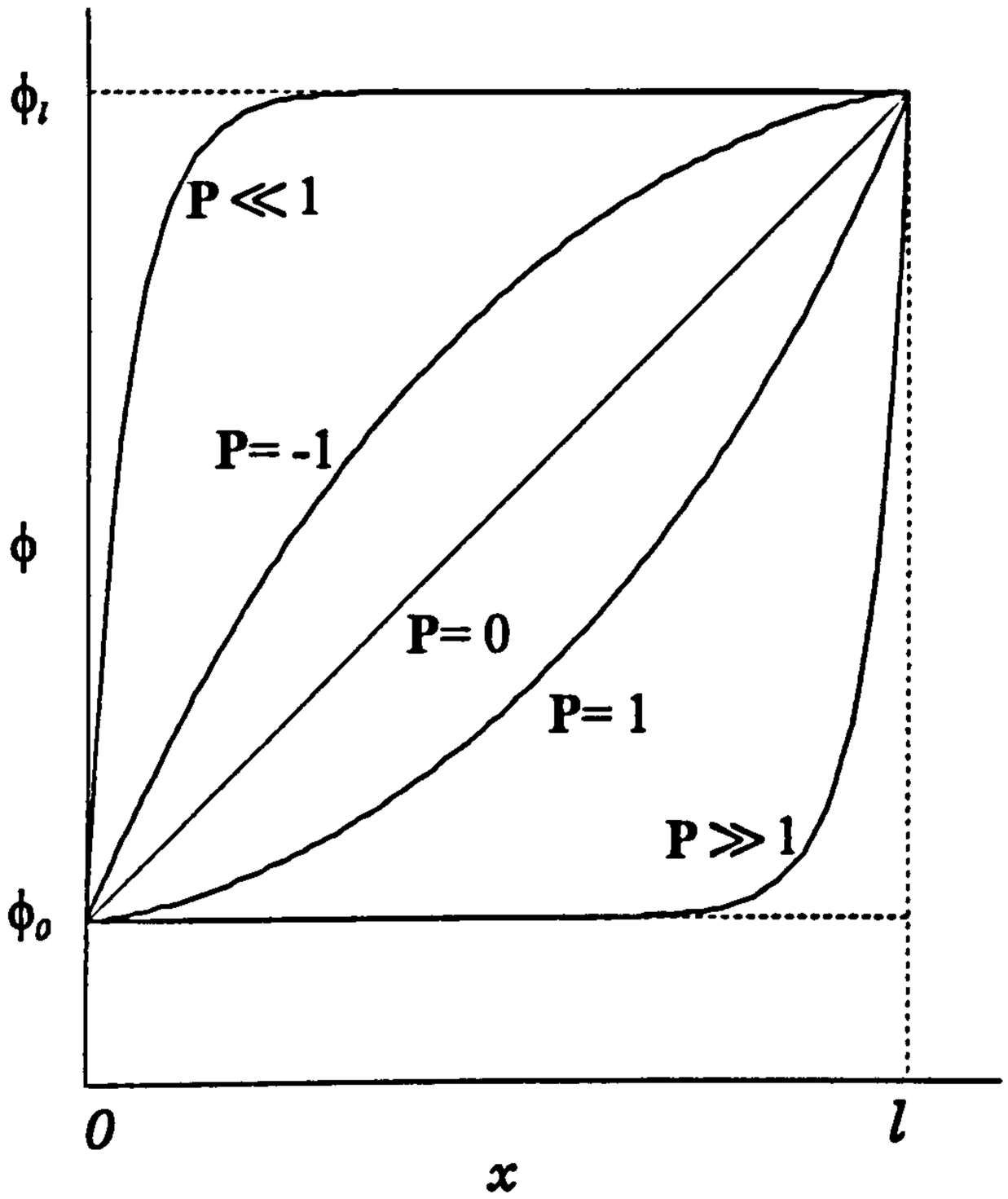


Figure A5-2: Exact solution of the 1D convection-diffusion problem.

where **P** is the Peclet number $= {}^{\rho U l}/_{\Gamma}$ representing the relative strengths of convection and diffusion. The exact solution is plotted in Figure A5-2 for various values of Peclet number. In the limit of zero Peclet number, the pure diffusion (or conduction) problem is obtained and the ϕ - x variation is linear. When the flow is in the positive x direction (i.e. for positive values of **P**), the values of ϕ in the domain are more influenced by the upstream value of ϕ. For a large positive value of **P**, the value of ϕ remains very close to the upstream value ϕ_0 over much of the domain. When the flow is in the negative x direction, ϕ_l becomes the upstream value of ϕ. For a large negative **P** the value of ϕ over most of the domain is very nearly equal to ϕ_l.

It can now be seen that the ϕ - x profile is far from linear except for small values of |P|. When |P| is large the value of ϕ at $x = l/2$ (i.e. the cell interface) is nearly equal to the value of ϕ at the *upwind* boundary, i.e. the value from the upstream cell. Also note that at large |P|, ${}^{d\phi}/_{dx}$ is nearly equal to zero at $x = l/2$, thus the diffusion term in Equation A5-3 is negligible. Several improvements on the simple linear profile have been found, such as the *upwind* scheme, the *exponential* scheme, the *power law* scheme and the *hybrid* scheme (Patankar 1980 p86). PHOENICS by default uses the hybrid scheme which calculates ϕ at the east face of each cell by the following method:

For $-2 \le \mathbf{P_e} \le 2$ $\phi_e = \frac{1}{2}(\phi_E + \phi_P)$ (central difference scheme)

otherwise for $\rho U_e > 0$ $\phi_e = \phi_P$

$\rho U_e < 0$ $\phi_e = \phi_E$ (upwind scheme)

where $\mathbf{P_e}$ is the Peclet number on the east face of the cell. A similar procedure is used for the calculation of ϕ_w.

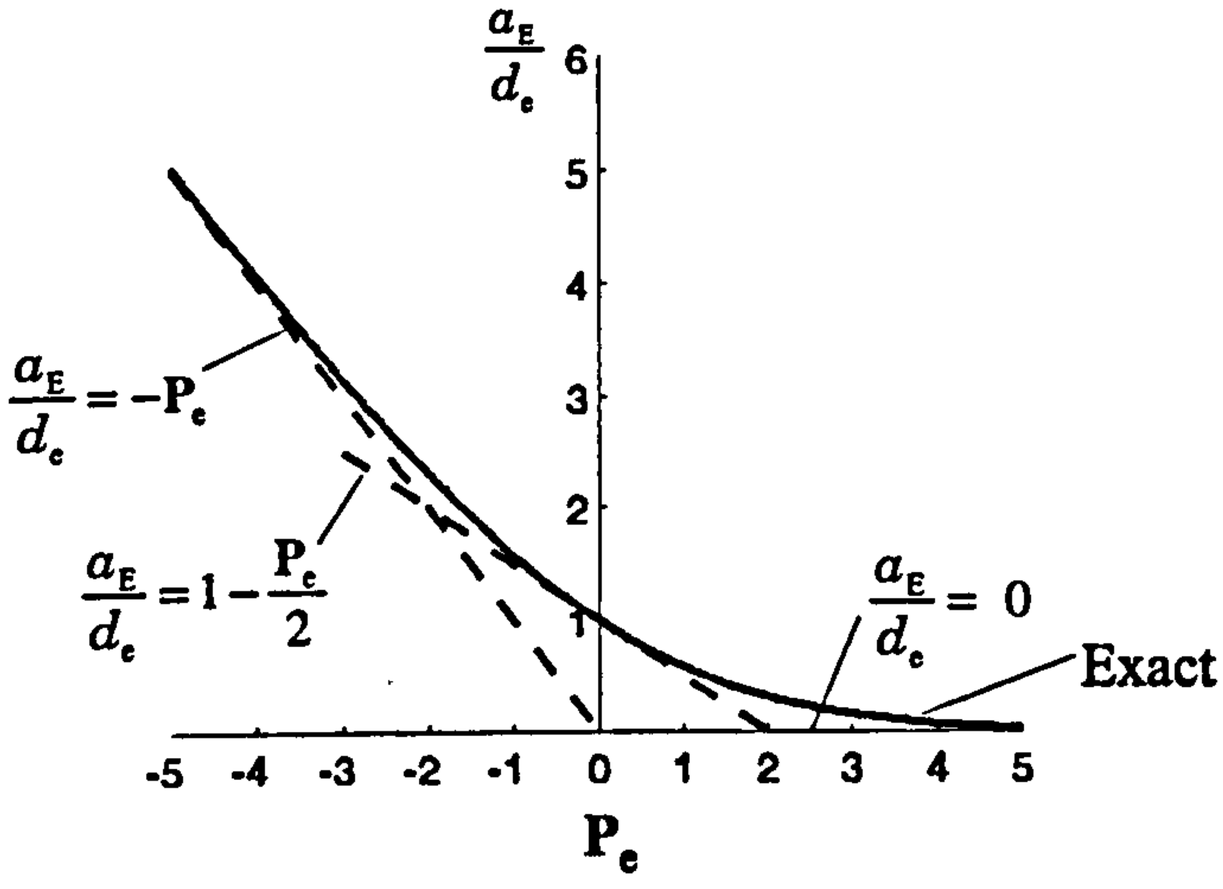


Figure A5-3: Variation of the coefficient a_E with Peclet Number

Further insight may be gained by plotting the dimensionless coefficient a_E/d_e (where $d_e=\Gamma_e/\Delta_x$ for this one-dimensional case) as a function of Peclet number, as shown in Figure A5-3. The solid line shows the exact solution which has the properties:

1. for $\mathbf{P}_e \to \infty$ $a_E/d_e \to 0$
2. for $\mathbf{P}_e \to -\infty$ $a_E/d_e \to -\mathbf{P_e}$
3. at $\mathbf{P}_e = 0$ the tangent is $a_E/d_e = 1 - \mathbf{P_e}/2$

These three straight lines give a reasonable approximation to the exact curve, and constitute the hybrid scheme. Thus:

for $\mathbf{P_e} < -2$ $a_E/d_e = -\mathbf{P_e}$

for $-2 \le \mathbf{P_e} \le 2$ $a_E/d_e = 1 - \mathbf{P_e}/2$

for $\mathbf{P_e} > 2$ $\qquad {}^{a_E}\!/_{d_e} = 0$

These expressions can be combined into a compact form by the use of the operator 'max' which returns the maximum value of its arguments. Thus:

$$a_E = d_e \times \max\left(-\mathbf{P_e}, 1 - {}^{\mathbf{P_e}}\!/_{2}, 0\right) \qquad \textit{Equation A5-11}$$

The coefficient may also be regarded as having a convection term and a diffusion term, the latter being influenced by the convective stream. For strong convection, the diffusion may be reduced to zero. Thus a_E may also be written as:

$$a_E = d_e \times \max\left(0, 1 - 0.5|\mathbf{P_e}|\right) + d_e \times \max\left(0, -\mathbf{P_e}\right) \qquad \textit{Equation A5-12}$$

or

$$a_E = \underbrace{\max\left(0, d_e - 0.5|m_e|\right)}_{\text{diffusion with convective influence}} + \underbrace{\max\left(0, -m_e\right)}_{\text{convection term}} \qquad \textit{Equation A5-13}$$

where m_e is the mass flux through face e. Note that the factor 0.5 may be replaced by the variable α which may be regarded as the value of $1/\mathbf{P_e}$ at which the diffusion term is switched off. The value of α may be changed through the use of the PHOENICS variable DIFCUT. If $\alpha = 0$ then the hybrid scheme returns to the upwind scheme and the value of ϕ at the cell interface is taken to be the value of its upstream neighbour.

Coefficients representing the influence of the W, N, S, H, L cells may be determined in a similar manner.

A5.3 Coefficient of Current Cell

The coefficient of ϕ_P is made up as follows:

$$\begin{aligned} a_P = {} & a_E + a_W + a_N + a_S + a_H + a_L + a_T \\ & + (-S_2) && \text{a source term contribution} \\ & + \left({}^{\rho \Delta x \Delta y \Delta z}\!/_{\Delta t}\right)_P && \text{a term for false time step relaxation} \\ & + \text{ boundary conditions} \end{aligned} \qquad \textit{Equation A5-14}$$

Note that, as suggested by the one-dimensional case in Equation A5-9c, the terms $a_E + a_W + a_N + a_S + a_H + a_L + a_T$ in the expression for a_P are a consequence of mass conservation.

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

Appendix 6: Calculation of the Flow Field

A6.1 Introduction

The flow field may be calculated using the FVE (Equation 2-62) with $\phi = U$, V or W and by noting that the continuity equation and the momentum equations are linked via the pressure field. When the correct pressure field is substituted into the momentum equations, the resulting velocity field will satisfy the continuity equation.

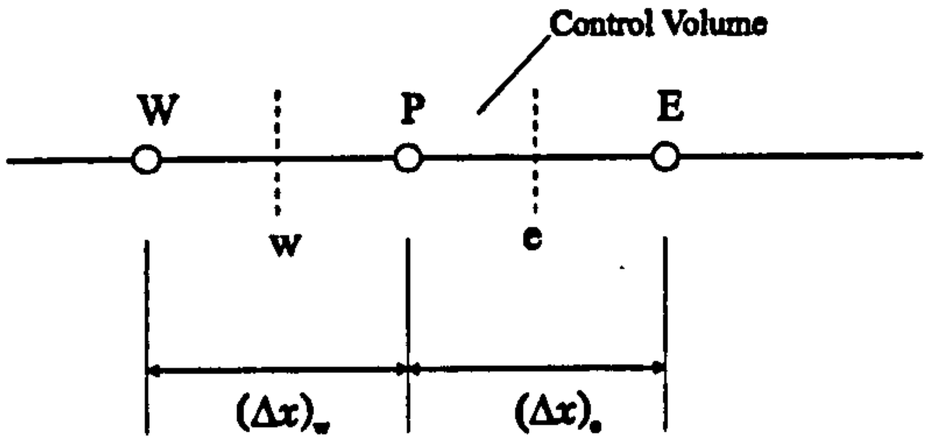


Figure A6-1: One dimensional control volume

However a problem arises when the pressure gradient term of the momentum equation is discretized. Integrating the term $-dP/dx$ over a three point grid cluster such as the one shown in Figure A6-1 and using a piecewise-linear profile (for algebraic convenience only) gives

$$P_w - P_e = \frac{P_W + P_P}{2} - \frac{P_P + P_E}{2} = \frac{P_W - P_E}{2} \qquad \textit{Equation A6-1}$$

Thus the discretised equations contain pressure differences between alternate grid points and not between adjacent ones. The intermediate grid point is skipped, which has the consequence that an unrealistic wavy pressure distribution, for example as shown in

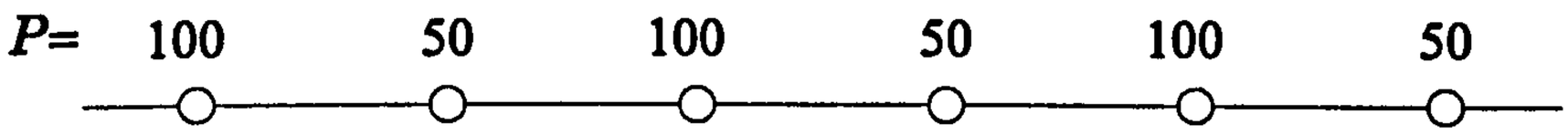


Figure A6-2: Wavy pressure distribution

Figure A6-2, will be seen by Equation A6-1 as a uniform distribution. A similar difficulty arises for the discretization of the continuity equation. For the steady one-dimensional constant density case, the continuity equation is simply

$$\frac{dU}{dx} = 0 \qquad \textit{Equation A6-2}$$

which integrated over the control volume given in Figure A6-1 gives:

$$U_e - U_w = 0 \qquad \textit{Equation A6-3}$$

Again using a piecewise-linear profile for U gives

$$\frac{U_P + U_E}{2} - \frac{U_W + U_P}{2} = 0 \qquad \textit{Equation A6-4}$$

or

$$U_E - U_W = 0 \qquad \textit{Equation A6-5}$$

where again the intermediate point 'P' is skipped, leading to the same possible consequences as described above for the pressure gradient term.

Before a scheme for finding the velocity field from the pressure field could be presented, a solution to the problem posed above was required. This was fulfilled by the *staggered grid*, first used by Harlow and Welch (1965) and which now forms the basis of the SIMPLE procedure of Patankar and Spalding (1972).

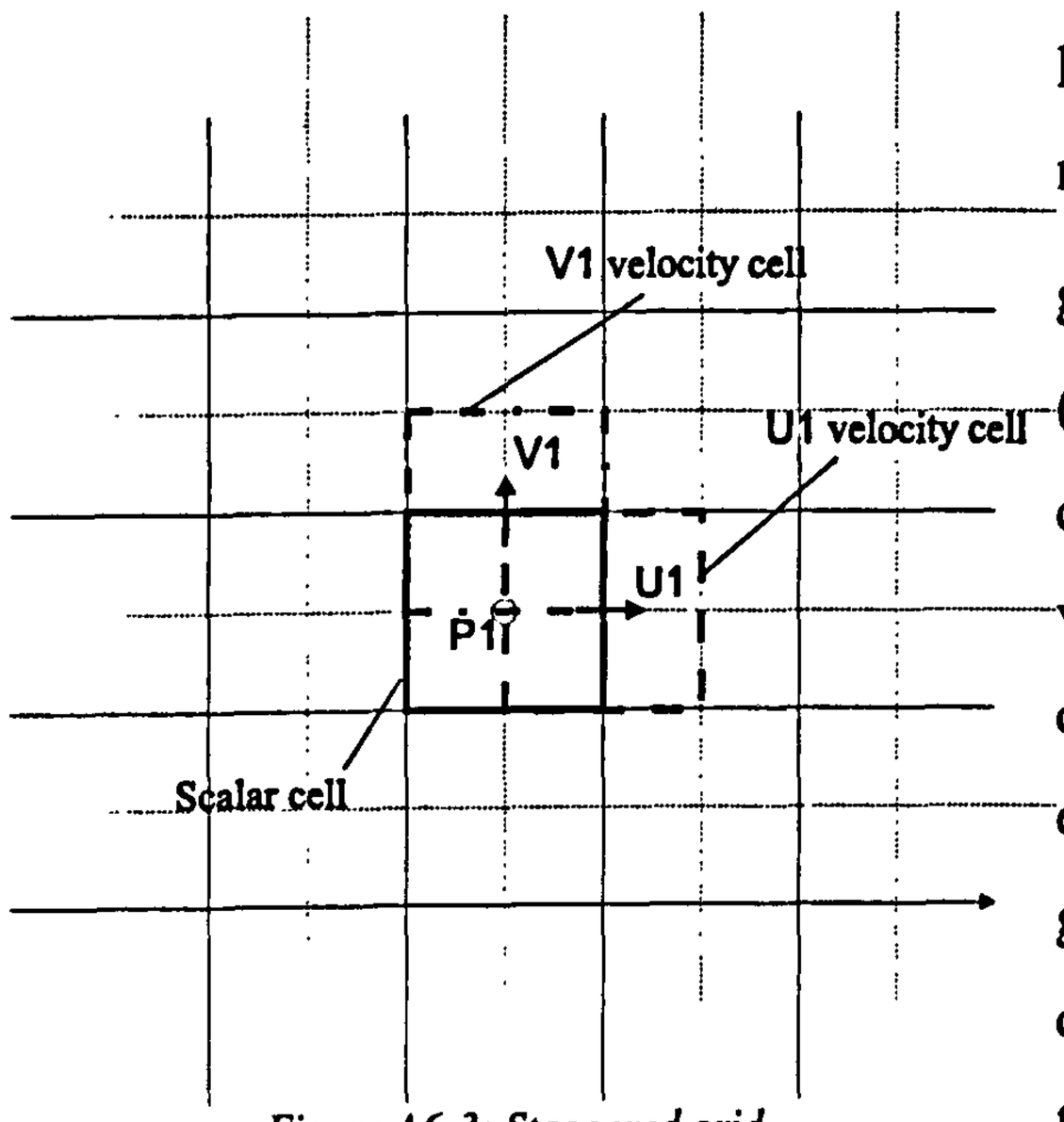


Figure A6-3: Staggered grid

The solution relies on the fact that the location of the control volumes does not need to be the same for all of the general variables ϕ. The staggered grid, (Figure A6-3), uses one grid for the calculation of pressure and other scalar variables, and for each velocity component an additional grid is employed, displaced from the original grid by $^1/_2$ a cell length in the direction of the velocity component. Therefore the velocities may also be viewed as being calculated at the cell faces of the original grid, thus

U_P velocity is calculated on the face 'e'

V_P velocity is calculated on the face 'n'

W_P velocity is calculated on the face 'h'.

The two problems found in discretizing the pressure gradient term and the continuity term are then solved. Integrating the pressure gradient term in the momentum equation over *the staggered velocity* cell corresponding to the 'P' cell results in a pressure difference over the cell which may be represented *by two adjacent pressures*

$$\int_{\text{velocity cell}} \left(-\frac{dP}{dx}\right) = P_{\text{P}} - P_{\text{E}} \qquad \textit{Equation A6-6}$$

Likewise integrating the continuity equation over a pressure cell leads to a velocity difference using stored velocities only and no profile assumption is required.

One consequence of using the staggered grid however is that velocities cannot be calculated at the boundary of the domain due to Equation A6-6 above. For example, the U1 velocity corresponding to cell 'P' is calculated at face 'e'. If this face happens to be a boundary to the domain, P_{E} does not exist and therefore the momentum equation cannot be solved for this particular cell.

A6.2 Momentum Equations

Taking the x direction as an example, the discretization equation for $\phi = U$ over the one-dimensional control volume shown in Figure A6-4 is

$$a_{\text{e}} U_{\text{e}} = \sum a_{\text{nb}} U_{\text{nb}} + b + (P_{\text{P}} - P_{\text{E}}) A_{\text{e}} \qquad \textit{Equation A6-7}$$

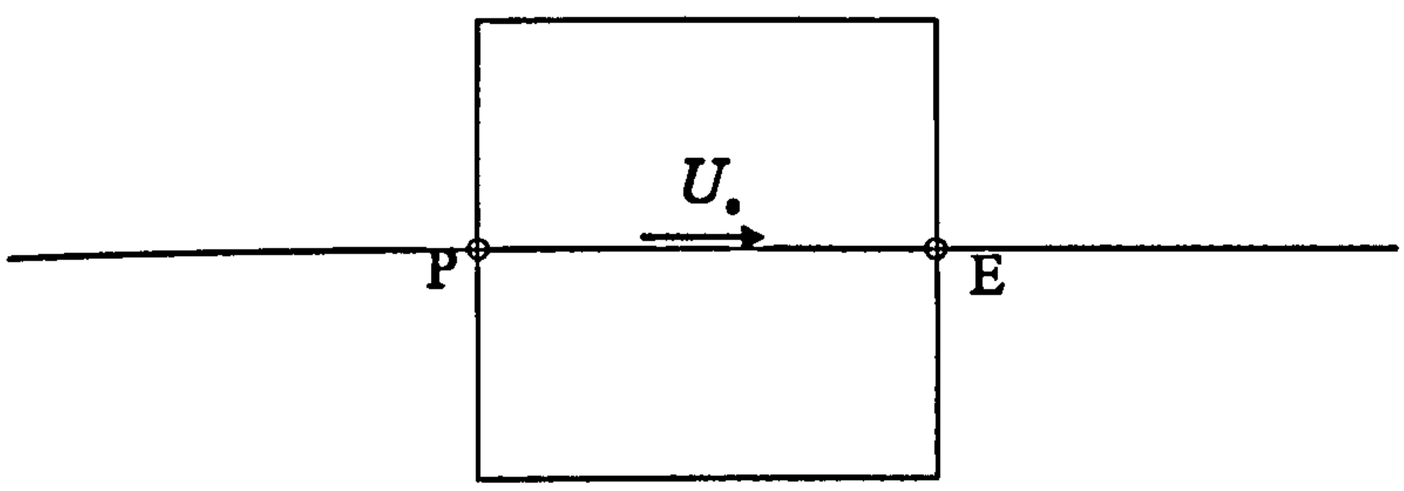


Figure A6-4: Control volume for U velocity

In discretizing the Navier-Stokes equations the pressure gradient term becomes the pressure force term $(P_{\text{P}} - P_{\text{E}})A_{\text{e}}$. Since the pressure and velocity fields are interlinked it would be inconvenient to include this term as part of the source term, as is often done in the Navier-Stokes equations. During the solution procedure only approximate values of

P_P and P_E are known, obtained from the last sweep and denoted here by P_P^* and P_E^*. The resulting velocity field calculated from this imperfect pressure distribution will not satisfy the continuity equation and will be denoted by U^*. The approximate velocity field may then be calculated from the following discretization equation:

$$a_e U_e^* = \sum a_{nb} U_{nb}^* + b + (P_P^* - P_E^*) A_e \qquad \textit{Equation A6-8}$$

The pressure and velocity corrections

A procedure is required which will iteratively improve the approximate pressure P^* such that the resulting velocity field U^* will progressively get closer to satisfying the continuity equation. Writing

$P = P^* + P'$ where P' is termed the pressure correction and

$U = U^* + U'$

and subtracting Equation A6-8 from Equation A6-7 gives

$$a_e U_e' = \sum a_{nb} U_{nb}' + (P_P' - P_E') A_e \qquad \textit{Equation A6-9}$$

The term $\sum a_{nb} U_{nb}'$ may be evaluated in terms of the pressure and velocity corrections of the neighbouring cells, however these neighbours would in turn bring in the influence of their neighbours and so on. Ultimately the velocity correction equation would involve the pressure correction at all grid points in the calculation domain and would be unmanageable. Fortunately this term only affects the rate of convergence and not the final solution (Patankar 1980) and so may be omitted to give

$$U_e' = \frac{A_e}{a_e}(P_P' - P_E') \qquad \textit{Equation A6-10}$$

or

$$U_e = U_e^* + \frac{A_e}{a_e}(P_P' - P_E') \qquad \textit{Equation A6-11a}$$

This velocity correction equation shows how the approximate velocity U_e^* needs to be corrected in response to the pressure corrections to produce a better estimate of velocity, U_e. In a similar manner the correction equation for the other velocity components can be written as

$$V_n = V_n^* + \frac{A_n}{a_n}(P'_P - P'_N) \qquad \textit{Equation A6-11b}$$

$$W_h = W_h^* + \frac{A_h}{a_h}(P'_P - P'_H) \qquad \textit{Equation A6-11c}$$

Pressure correction equation

The pressure correction equation can be obtained by integrating the three-dimensional steady incompressible continuity equation

$$\frac{\partial(\rho U)}{\partial x} + \frac{\partial(\rho V)}{\partial y} + \frac{\partial(\rho W)}{\partial z} = 0 \qquad \textit{Equation A6-12}$$

over a standard grid cell, such as that shown in Figure (ii). This results in

$$\left[(\rho U)_e - (\rho U)_w\right]\Delta y\Delta z + \left[(\rho V)_n - (\rho V)_s\right]\Delta z\Delta x + \left[(\rho W)_h - (\rho W)_l\right]\Delta x\Delta y = 0 \quad \textit{Equation A6-13}$$

Substituting in the velocity correction equations (Equation A6-11) gives (after re-arranging) the pressure correction equation

$$a_P P'_P = a_E P'_E + a_W P'_W + a_N P'_N + a_S P'_S + a_H P'_H + a_L P'_L + b \qquad \textit{Equation A6-14}$$

where

$$a_E = \rho_e \frac{A_e}{a_e}\Delta y\Delta z$$

$$a_W = \rho_w \frac{A_w}{a_w}\Delta y\Delta z$$

$$a_N = \rho_n \frac{A_n}{a_n}\Delta z\Delta x$$

$$a_S = \rho_s \frac{A_s}{a_s}\Delta z\Delta x$$

$$a_H = \rho_h \frac{A_h}{a_h}\Delta x\Delta y$$

$$a_L = \rho_l \frac{A_l}{a_l}\Delta x\Delta y$$

$$a_P = a_E + a_W + a_N + a_S + a_H + a_L$$

$$b = \left[(\rho U^*)_w - (\rho U^*)_e\right]\Delta y\Delta z + \left[(\rho V^*)_s - (\rho V^*)_n\right]\Delta z\Delta x + \left[(\rho W^*)_l - (\rho W^*)_h\right]\Delta x\Delta y$$

Note that the source term b in the pressure correction equation (Equation A6-14) is the discretized continuity equation evaluated for the estimated velocities U^*, V^*, W^*. If these estimated velocities do indeed satisfy the continuity equation then $b = 0$ and no

pressure correction is required. During the solution procedure the term b then represents a mass source which needs to be reduced by means of pressure corrections and their associated velocity corrections. One such solution procedure is the SIMPLE algorithm.

The SIMPLE algorithm

The SIMPLE algorithm (Patankar 1975) stands for Semi-Implicit Method for Pressure Linked Equations. The procedure is semi-implicit due to the omission of the term $\sum a_{nb} U'_{nb}$ in the velocity correction equation. This term represents an indirect or implicit influence of the pressure correction on the velocity field. Since this influence is not included the procedure is only partially implicit.

The procedure may be written as:

1. Guess the pressure field P^*.

2. Solve the momentum equations (such as Equation A6-8) to obtain U^*, V^*, W^*.

3. Solve the pressure correction equation for P'(Equation A6-14).

4. Calculate P from $P = P^* + P'$.

5. Calculate U, V and W from the velocity correction equations (Equation A6-11).

6. Solve the discretization equations for the other variables ϕ (such as turbulence quantities).

7. Make $P^* = P$ and repeat from step 2 until convergence.

Improved algorithms

The SIMPLE algorithm has been used by numerous authors. However in an attempt to improve the rate of convergence, two adaptations have been formulated, the latter of which is used in PHOENICS.

1. SIMPLER (Patankar 1981) stands for SIMPLE Revised. The omission of the term $\sum a_{nb} U'_{nb}$ in the velocity correction equation of SIMPLE over-estimates the pressure correction and therefore slows down convergence. To overcome this an alternative equation for the calculation of the pressure field is employed which uses a pseudo-velocity field. No approximations are used in the derivation of these equations which are fully implicit.

2. SIMPLEST (Spalding 1980) stands for SIMPLE ShorTened. The rate of convergence was found to be particularly slow if fine grids were used. This was traced back to the convection terms in the momentum FVE being non-reciprocal, i.e. the west convection coefficient of point P is not identical to the east convection coefficient of point W. As a consequence an incorrect velocity is convected downstream, increasing the total momentum imbalance in the field. Large pressure corrections are then required to eliminate this imbalance. SIMPLEST therefore only contains diffusion contributions to the momentum equation coefficients a_E, a_W, a_N, a_S, a_H, a_L. The convection terms are added to the momentum source b. The velocity and pressure corrections are consequently smaller and the rate of convergence is increased. In this method the error resulting in omitting the term $\sum a_{nb} U'_{nb}$ is reduced and the procedure used in the original SIMPLE algorithm can be used.

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

Appendix 7: Multi-block Grids

A7.1 Description of Multi-block Grids

Multi-block gridding was first released in version 2.1 (March 1995) of PHOENICS. Its implementation in PHOENICS can be of two types:

1. Linked blocks. Two separate grids are joined along a common line. In this way the domain may be split into several blocks of cells, each containing an independent grid. It is not required for the cells in both grids to match, or that adjoining blocks have identical numbers of cells, however the boundaries of adjoining blocks must match along common grid lines. The total domain may now be of an irregular shape.

2. Embedded fine grids. Where a higher resolution is required a section of the domain may be split into finer cells by "overlaying" an additional, finer grid. The original grid lines are required to match up with a integer multiple of the new grid lines.

Multi-blocking has two advantages over conventional grid techniques:

1. In domains where large areas are blocked off, for example in compound duct flows, the domain may defined to cover only the volume containing the fluid, thus removing the need for unnecessary cells.

2. In domains which require a local fine grid, the fine cells can be placed only where they are required. For example Figure A7.1 shows how a conventional streamwise grid models the vertical boundaries of a roughness element. The layer of fine cells extends to the full height of the duct,

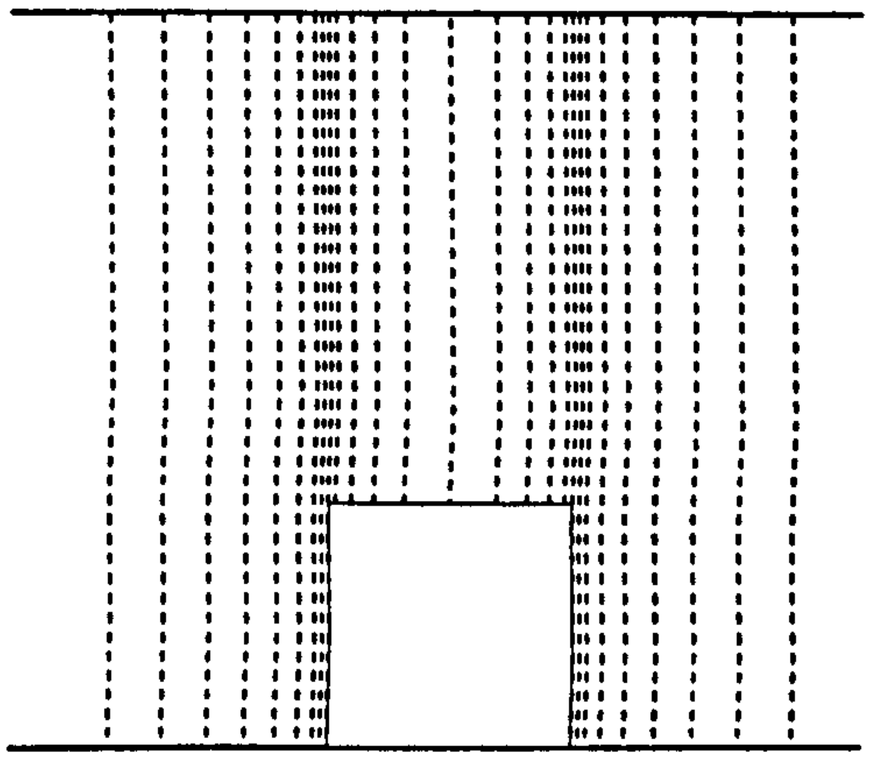


Figure A7.1: Conventional grid (streamwise distribution only)

although they are not required in the main flow region. By using multi-block grids, the fine cells are limited to the region surrounding the block.

It was found however that the use of multi-block gridding in PHOENICS was accompanied by a number of complicating factors:

1. A new numerical scheme. Multi-block grids no longer used the staggered grid system. Instead extra velocities were calculated at the centre of each cell. These were termed "co-located velocities".

2. Storage of values in the F-array. The extraction of values from the F-array was made more complex due to the fact that there were now numerous independent grids, each stored separately. Thus a neighbouring cell in physical space might not be a neighbour in computational space. The layout of the grids in the F-array is illustrated in Figure A7.2.

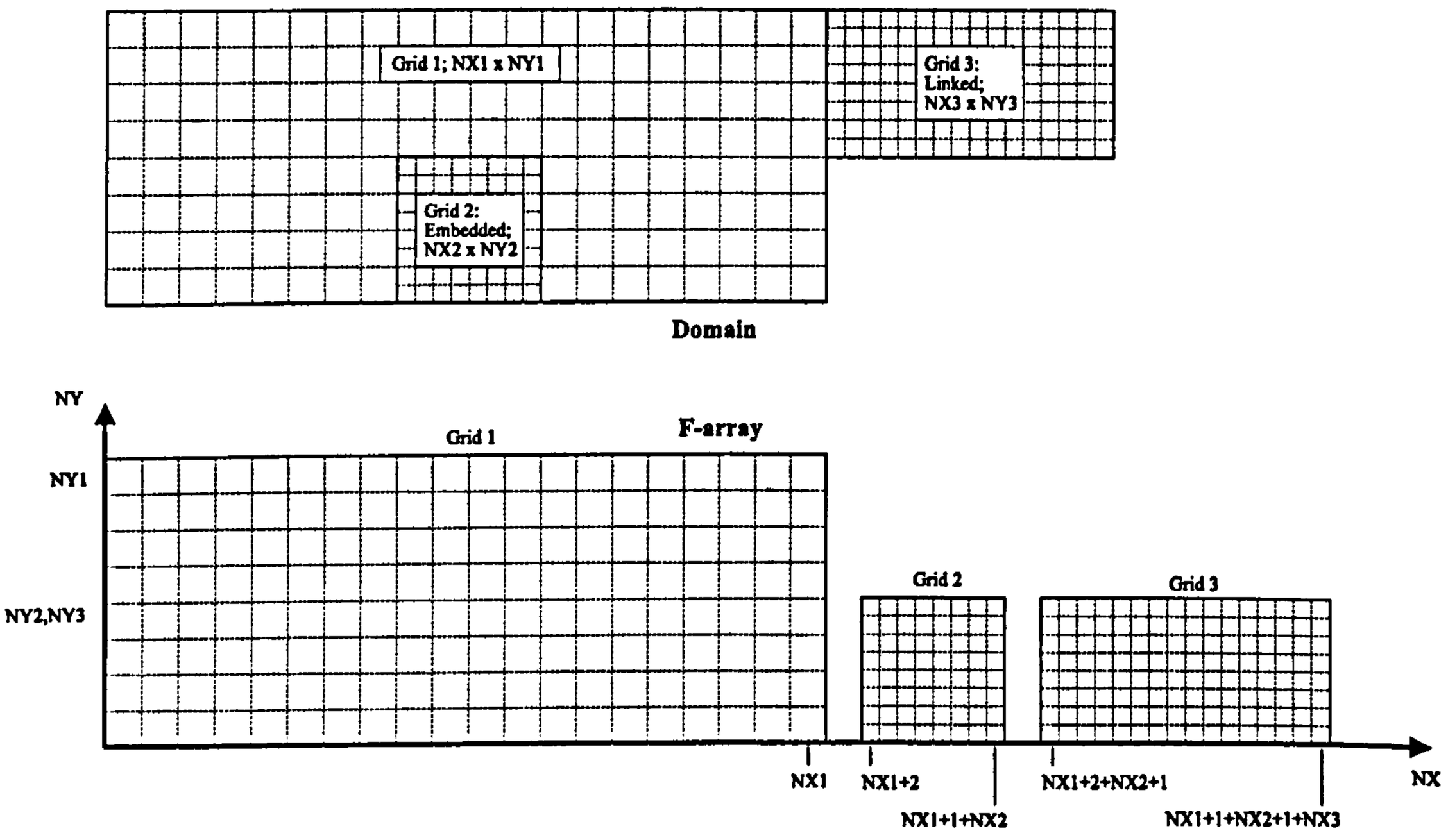


Figure A7.2: F-Array structure of multi-block grids

3. Use of body fitted co-ordinates (BFC). To use multi-block grids, the BFC option in PHOENICS had to be activated, even if the grid required was orthogonal. This resulted in a longer and more tedious grid generation procedure, and less

flexibility in defining each grid, due to the fact that XFRAC, YFRAC and ZFRAC might not be defined explicitly. Only the supplied power law could be used and not user defined geometric expansions.

4. Blockages. Due to a error in PHOENICS, the method of defining blocked off areas, using CONPOR, was found not work with multi-block grids. This problem was later solved with the release of version 2.1.1.

5. A further problem encountered was that the post processor PHOTON was unable to process multi-block grid output files.

Due to the advantages in using multi-block gridding, a thorough examination was undertaken to determine ways of increasing the resolution of grid cells in the region surrounding isolated roughness elements. This is discussed in the following section.

A7.2 Defining a Multi-block Grid over a Roughness Element

Method 1

Conceptually the easiest method of increasing the grid resolution surrounding a roughness element is to use uniform grids which are then sub-divided in the region encompassing the roughness element. A number of sub-divisions may be performed to achieve the required wall cell size. For example Figure A7.3 shows two grid subdivisions of the main grid. Implementation of this method presented several difficulties:

1. The roughness element had to be blocked off using the CONPOR technique in PHOENICS. Due to the error mentioned previously, this was not possible.

2. If the blockage could have been correctly specified, it would have contained lots of very small cells which would have been wasteful of computational resources.

3. Many grid sub-divisions would have been required to gradually reduce the size of the cell from that used in the core flow region to that required adjacent to the solid boundary of the roughness element.

An attempt was also made to find a geometric or power law expansion within these sub-grids, but no adequate method of joining the sub-grid boundaries was found.

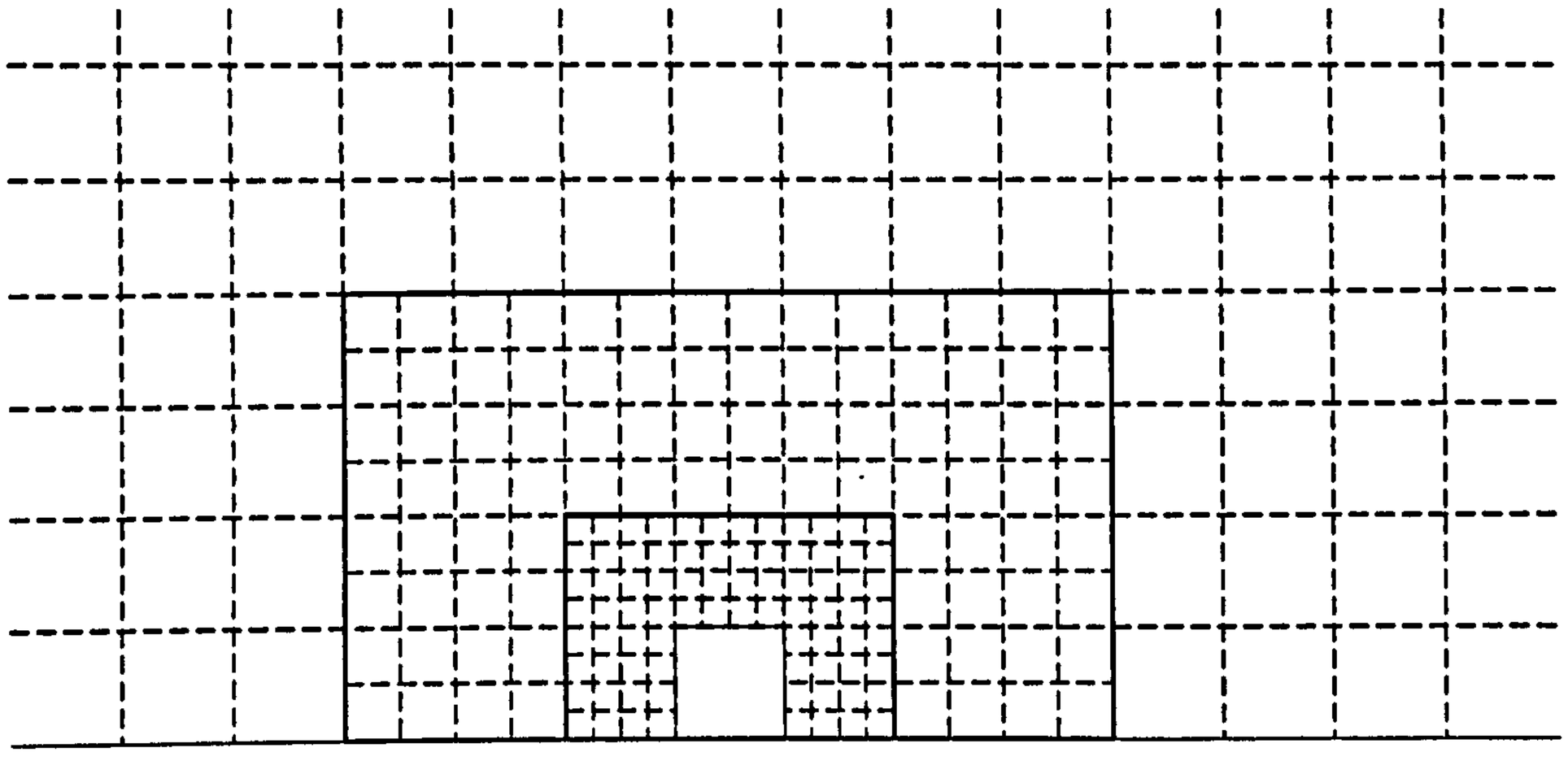


Figure A7.3: Embedded multi-block grids

Method 2

To avoid defining the roughness element as a blockage, the linked blocks method was employed to exclude the roughness element from the domain. Again no solution could be found for joining adjacent expanding grids of the type shown in Figure A7.3 (the y direction only is shown for clarity).

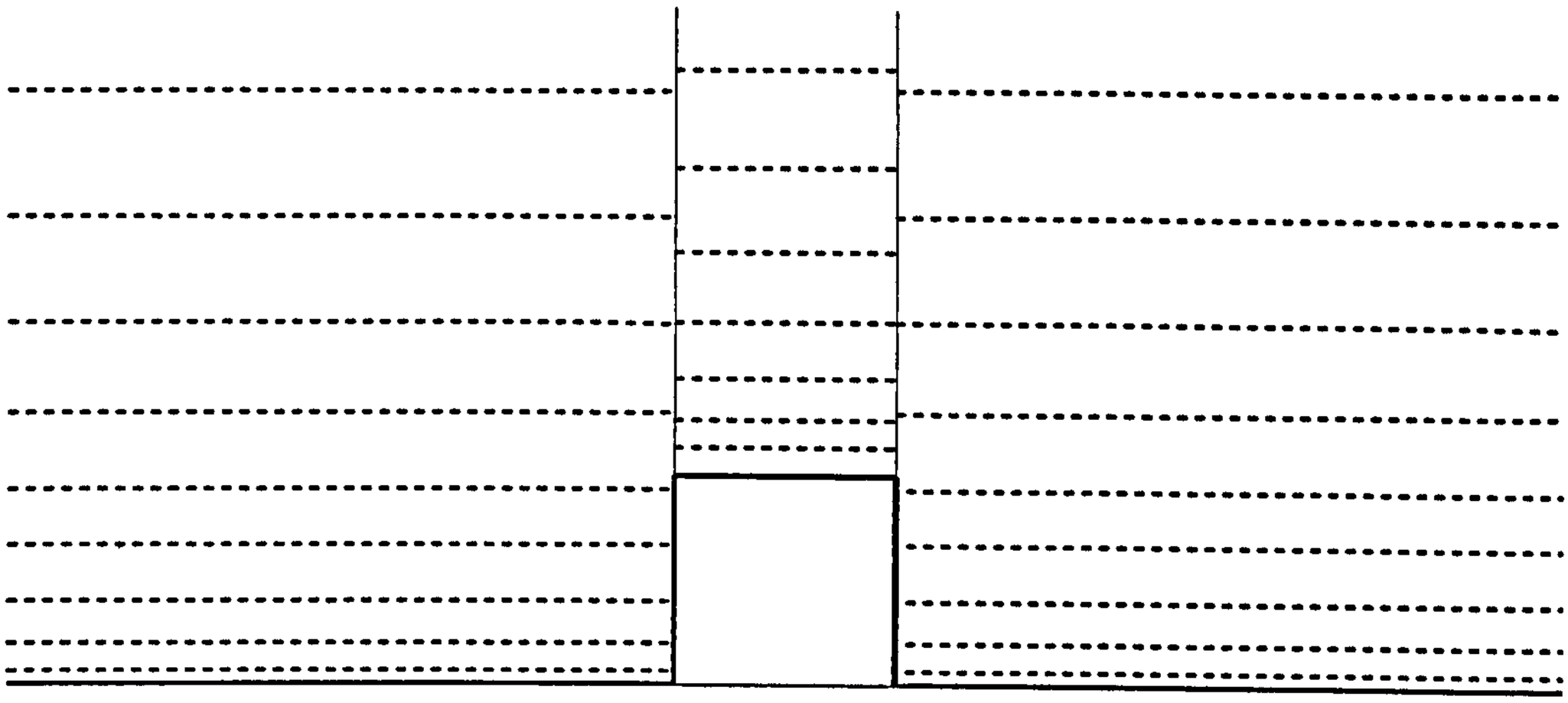


Figure A7.3: Linked multi-block grids (y direction only)

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

Appendix 8: PHOENICS Code for User Controlled Periodic Boundary Conditions

A8.1 Domain and Grid

The user controlled periodic boundary conditions were tested using Schlichting's (1936) roughness plate XX, case 5 ($Re = 2.2 \times 10^5$) for which the domain and grid employed are shown in Figure A8-1. Two periods were used to aid the visualisation of fully developed flow. The Q1 input file (Section A8.2) specified all of the flow parameters. The inlet and outlet boundary conditions, specified in Group 13 of the Q1 file, used the flag GRND which enabled the value of the boundary condition to be set in the GROUND subroutine.

The recycling of the arrays of the variables W1, P1, U1, V1, EP and KE was performed in Group 19, Section 7 of GROUND, in which further subroutines were called. Each variable was extracted from the F-array at the relevant slab and inserted into user defined one-dimensional arrays. Thus the W1 velocities were recycled from cell IZ=NZ−1 to IZ=1 (Figure 3-1), via the GROUND array W1IN. A scale factor, CMASS was calculated as

$$CMASS = \frac{WO}{\sum_{n=1}^{NY \times NX} [\text{cell high area}(n) \times W1IN(n)] \Big/ \text{total high area}}$$

Multiplying the array W1IN by CMASS ensured that the average velocity remained equal to the required averaged velocity, WO. The boundary condition for the pressure variable was the source of mass flow, specified by W1 × CMASS × RHO1. The variables U1, V1, EP and KE were recycled in a similar manner to the W1 variable (though CMASS was not used for these variables).

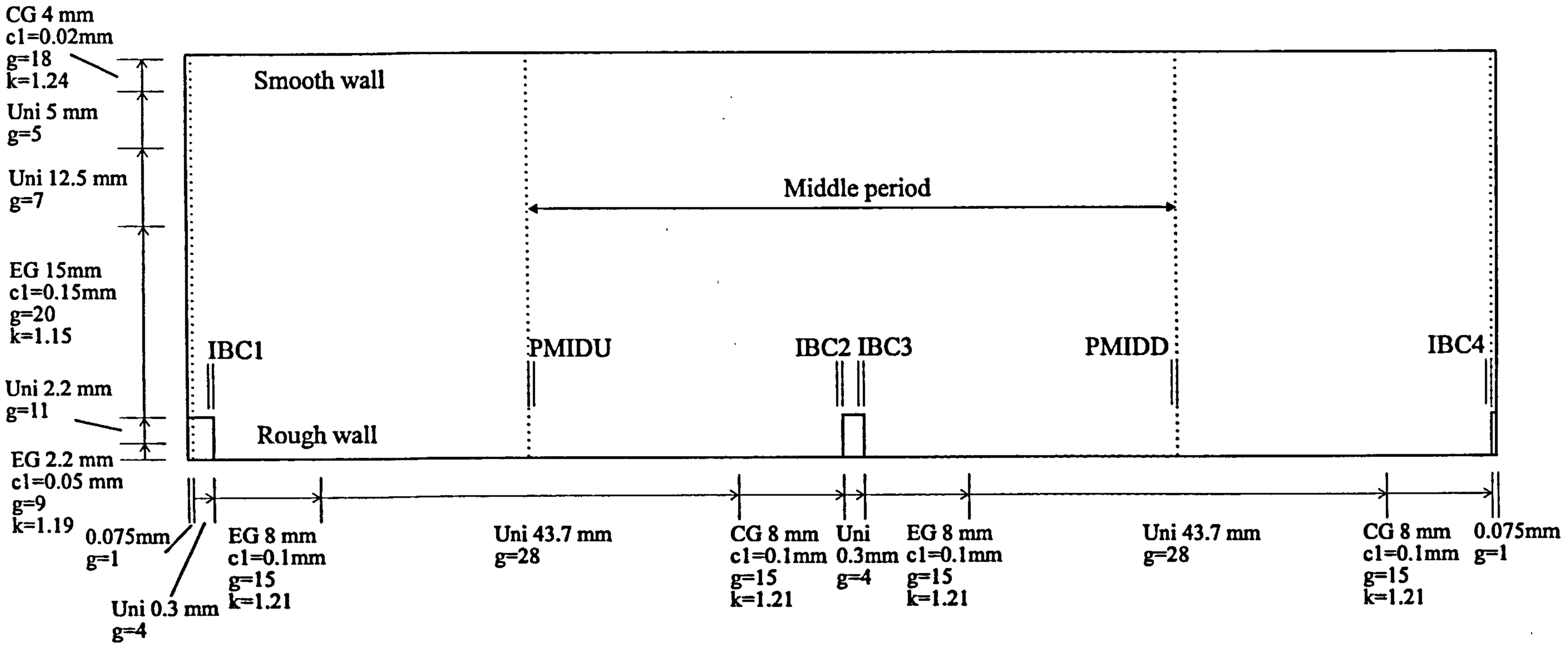


Figure A8-1: Domain and grid for user controlled periodic boundary conditions

Notes:

EG = expanding geometric grid
CG = contracting geometric grid
Uni = uniform grid spacing
c1 = size of first cell
g = number of cells in region
k = expansion factor

IBC1, 2, 3, 4 and PMIDU and PMIDD are cell labels for the roughness elements and the middle period.

The outlet pressure distribution was set by extracting the pressure distribution from IZ=2 and placing it into the GROUND array P1DOWN. The average of this array was calculated by summing the product of the cell pressures with the high face cell areas, from the top of the roughness elements to the top wall. This average value was then taken away from the original P1DOWN to create a new P1DOWN, which was used as the exit boundary condition. This ensured that the average pressure at outlet was zero and gave the correct streamwise pressure gradient.

A8.2 Q1 file

```
TALK=F;RUN( 1, 1);VDU=X11-TERM
NOCOPY=T;NOCOMM=T
SAVE=T
  ************************************************************
  *              CYCLIC DUCT FLOW
  *           ALASTAIR SENIOR, RMCS
  ************************************************************
  * Uses mass flow at inlet, fixed pressure at outlet
  * with recycling of velocites and pressure distribution
  *
  * File name:      frxx
  * Test number(#): 6
  * Date:           14/12/96
  * Running on:     v2.1.1, SUN, SOLARIS
  *
  * Comments:  2d, cyclic b.c.
  *            Case: Sch XX, profile 5 (high Re.)
  *            high P1 exit coeff
  *
  * ************************************************************
  *  GROUP 1.  Run identifiers and other preliminaries.
  *
TEXT(CYCLIC DUCT
PARAB=F
REAL(DEPTH,PLTH,SLTH,WIDTH,WO)
REAL(BLKHGT,BLKLTH,MFDTH)
INTEGER(BNY,BNZ,PNZ,SNZ,MFNY,IBC1,IBC2,IBC3,IBC4)

BLKLTH=0.3e-3
PLTH=60e-3
SLTH=PLTH-BLKLTH
BLKHGT=3.2e-3
DEPTH=39.7e-3
MFDTH=DEPTH-BLKHGT
  * WO = specified bulk velocity(m/s)
WO=5.025
  ************************************************************
  *  GROUP 2.  Time-dependence and related parameters.
STEADY=T
  ************************************************************
```

```
  * GROUP 3.  x-direction grid specification.
CARTES=T
  **********************************************************************
  *
  * GROUP 4.  y-direction grid specification.
  *
BNY=20
MFNY=50
NY=70
YVLAST=DEPTH
YFRAC( 1)=-1;YFRAC( 2)=   1.25945E-03
YFRAC( 3)=1;YFRAC( 4)=   1.49986E-03
YFRAC( 5)=1;YFRAC( 6)=   1.78618E-03
YFRAC( 7)=1;YFRAC( 8)=   2.12715E-03
YFRAC( 9)=1;YFRAC( 10)=   2.53320E-03
YFRAC( 11)=1;YFRAC( 12)=   3.01677E-03
YFRAC( 13)=1;YFRAC( 14)=   3.59265E-03
YFRAC( 15)=1;YFRAC( 16)=   4.27846E-03
YFRAC( 17)=1;YFRAC( 18)=   5.09519E-03
YFRAC( 19)= 11;YFRAC( 20)=   5.03778E-03
YFRAC( 21)=1;YFRAC( 22)=   3.77834E-03
YFRAC( 23)=1;YFRAC( 24)=   4.33739E-03
YFRAC( 25)=1;YFRAC( 26)=   4.97916E-03
YFRAC( 27)=1;YFRAC( 28)=   5.71589E-03
YFRAC( 29)=1;YFRAC( 30)=   6.56163E-03
YFRAC( 31)=1;YFRAC( 32)=   7.53250E-03
YFRAC( 33)=1;YFRAC( 34)=   8.64703E-03
YFRAC( 35)=1;YFRAC( 36)=   9.92647E-03
YFRAC( 37)=1;YFRAC( 38)=   1.13952E-02
YFRAC( 39)=1;YFRAC( 40)=   1.30813E-02
YFRAC( 41)=1;YFRAC( 42)=   1.50168E-02
YFRAC( 43)=1;YFRAC( 44)=   1.72387E-02
YFRAC( 45)=1;YFRAC( 46)=   1.97894E-02
YFRAC( 47)=1;YFRAC( 48)=   2.27175E-02
YFRAC( 49)=1;YFRAC( 50)=   2.60789E-02
YFRAC( 51)=1;YFRAC( 52)=   2.99376E-02
YFRAC( 53)=1;YFRAC( 54)=   3.43672E-02
YFRAC( 55)=1;YFRAC( 56)=   3.94522E-02
YFRAC( 57)=1;YFRAC( 58)=   4.52897E-02
YFRAC( 59)=1;YFRAC( 60)=   5.19909E-02
YFRAC( 61)= 7;YFRAC( 62)=   4.49802E-02
YFRAC( 63)= 5;YFRAC( 64)=   2.51889E-02
YFRAC( 65)=1;YFRAC( 66)=   2.00309E-02
YFRAC( 67)=1;YFRAC( 68)=   1.61293E-02
YFRAC( 69)=1;YFRAC( 70)=   1.29876E-02
YFRAC( 71)=1;YFRAC( 72)=   1.04579E-02
YFRAC( 73)=1;YFRAC( 74)=   8.42088E-03
YFRAC( 75)=1;YFRAC( 76)=   6.78065E-03
YFRAC( 77)=1;YFRAC( 78)=   5.45991E-03
YFRAC( 79)=1;YFRAC( 80)=   4.39642E-03
YFRAC( 81)=1;YFRAC( 82)=   3.54008E-03
YFRAC( 83)=1;YFRAC( 84)=   2.85054E-03
YFRAC( 85)=1;YFRAC( 86)=   2.29531E-03
YFRAC( 87)=1;YFRAC( 88)=   1.84823E-03
YFRAC( 89)=1;YFRAC( 90)=   1.48823E-03
YFRAC( 91)=1;YFRAC( 92)=   1.19835E-03
YFRAC( 93)=1;YFRAC( 94)=   9.64934E-04
YFRAC( 95)=1;YFRAC( 96)=   7.76983E-04
```

YFRAC(97)=1;YFRAC(98)= 6.25641E-04
YFRAC(99)=1;YFRAC(100)= 5.03778E-04

```
*
**********************************************************************
*
*  GROUP 5.  z-direction grid specification.
*
```

NZ=126
BNZ=4
SNZ=58
ZWLAST=0.12015

ZFRAC(1)= -1;ZFRAC(2)= 6.24220E-04
ZFRAC(3)= 4;ZFRAC(4)= 6.24220E-04
ZFRAC(5)=1;ZFRAC(6)= 8.32293E-04
ZFRAC(7)=1;ZFRAC(8)= 1.00917E-03
ZFRAC(9)=1;ZFRAC(10)= 1.22363E-03
ZFRAC(11)=1;ZFRAC(12)= 1.48367E-03
ZFRAC(13)=1;ZFRAC(14)= 1.79896E-03
ZFRAC(15)=1;ZFRAC(16)= 2.18127E-03
ZFRAC(17)=1;ZFRAC(18)= 2.64482E-03
ZFRAC(19)=1;ZFRAC(20)= 3.20688E-03
ZFRAC(21)=1;ZFRAC(22)= 3.88838E-03
ZFRAC(23)=1;ZFRAC(24)= 4.71471E-03
ZFRAC(25)=1;ZFRAC(26)= 5.71665E-03
ZFRAC(27)=1;ZFRAC(28)= 6.93152E-03
ZFRAC(29)=1;ZFRAC(30)= 8.40456E-03
ZFRAC(31)=1;ZFRAC(32)= 1.01906E-02
ZFRAC(33)=1;ZFRAC(34)= 1.23563E-02
ZFRAC(35)= 28;ZFRAC(36)= 1.29897E-02
ZFRAC(37)=1;ZFRAC(38)= 1.23563E-02
ZFRAC(39)=1;ZFRAC(40)= 1.01906E-02
ZFRAC(41)=1;ZFRAC(42)= 8.40456E-03
ZFRAC(43)=1;ZFRAC(44)= 6.93152E-03
ZFRAC(45)=1;ZFRAC(46)= 5.71665E-03
ZFRAC(47)=1;ZFRAC(48)= 4.71471E-03
ZFRAC(49)=1;ZFRAC(50)= 3.88838E-03
ZFRAC(51)=1;ZFRAC(52)= 3.20688E-03
ZFRAC(53)=1;ZFRAC(54)= 2.64482E-03
ZFRAC(55)=1;ZFRAC(56)= 2.18127E-03
ZFRAC(57)=1;ZFRAC(58)= 1.79896E-03
ZFRAC(59)=1;ZFRAC(60)= 1.48367E-03
ZFRAC(61)=1;ZFRAC(62)= 1.22363E-03
ZFRAC(63)=1;ZFRAC(64)= 1.00917E-03
ZFRAC(65)=1;ZFRAC(66)= 8.32293E-04
ZFRAC(67)= 4;ZFRAC(68)= 6.24220E-04
ZFRAC(69)=1;ZFRAC(70)= 8.32293E-04
ZFRAC(71)=1;ZFRAC(72)= 1.00917E-03
ZFRAC(73)=1;ZFRAC(74)= 1.22363E-03
ZFRAC(75)=1;ZFRAC(76)= 1.48367E-03
ZFRAC(77)=1;ZFRAC(78)= 1.79896E-03
ZFRAC(79)=1;ZFRAC(80)= 2.18127E-03
ZFRAC(81)=1;ZFRAC(82)= 2.64482E-03
ZFRAC(83)=1;ZFRAC(84)= 3.20688E-03
ZFRAC(85)=1;ZFRAC(86)= 3.88838E-03
ZFRAC(87)=1;ZFRAC(88)= 4.71471E-03
ZFRAC(89)=1;ZFRAC(90)= 5.71665E-03

```
ZFRAC( 91)=1;ZFRAC( 92)=  6.93152E-03
ZFRAC( 93)=1;ZFRAC( 94)=  8.40456E-03
ZFRAC( 95)=1;ZFRAC( 96)=  1.01906E-02
ZFRAC( 97)=1;ZFRAC( 98)=  1.23563E-02
ZFRAC( 99)= 28;ZFRAC( 100)=  1.29897E-02
ZFRAC( 101)=1;ZFRAC( 102)=  1.23563E-02
ZFRAC( 103)=1;ZFRAC( 104)=  1.01906E-02
ZFRAC( 105)=1;ZFRAC( 106)=  8.40456E-03
ZFRAC( 107)=1;ZFRAC( 108)=  6.93152E-03
ZFRAC( 109)=1;ZFRAC( 110)=  5.71665E-03
ZFRAC( 111)=1;ZFRAC( 112)=  4.71471E-03
ZFRAC( 113)=1;ZFRAC( 114)=  3.88838E-03
ZFRAC( 115)=1;ZFRAC( 116)=  3.20688E-03
ZFRAC( 117)=1;ZFRAC( 118)=  2.64482E-03
ZFRAC( 119)=1;ZFRAC( 120)=  2.18127E-03
ZFRAC( 121)=1;ZFRAC( 122)=  1.79896E-03
ZFRAC( 123)=1;ZFRAC( 124)=  1.48367E-03
ZFRAC( 125)=1;ZFRAC( 126)=  1.22363E-03
ZFRAC( 127)=1;ZFRAC( 128)=  1.00917E-03
ZFRAC( 129)=1;ZFRAC( 130)=  8.32293E-04
ZFRAC( 131)= 1;ZFRAC( 132)=  6.24220E-04

  *
  ***************************************************************
  * GROUP 6.  Body-fitting and other grid distortions.
  ***************************************************************
  * GROUP 7.  Variables (including porosities) named,
  *
       stored & solved.
SOLVE(V1)
SOLVE(W1)
SOLVE(U1)
  * Solve for P1 by whole-field method
SOLVE(P1)
SOLUTN(P1,Y,Y,Y,N,N,N)
STORE(ENUT)
STORE(CWN,CWS,CON,DWN,DVH,SDN,SMW)
STORE(SCN,SDH,WDIF,VDIF,SHR)
  *
  ***************************************************************
  * GROUP 8.  Terms (in differential equations) and devices.
   UCONV=T
   UDIFF=T
  ***************************************************************
  * GROUP 9.  Properties of the medium (or media).
  *
REAL(RREF1)
  * Water in M (from Sch)
RHO1=998.2
ENUL=1.1700e-6
   *
RREF1=RHO1
TURMOD(KEMODL);IENUTA=8;DISWAL
REAL(ENLREF);ENLREF=ENUL
  ***************************************************************
  * GROUP 10. Interphase-transfer processes and properties.
  ***************************************************************
  * GROUP 11. Initialization of fields of variables,
```

```
*           porosities, etc.
*
 RESTRT(ALL)

REAL(KEIN,EPIN)
KEIN=(WO*0.05)**2
EPIN=(0.009*(KEIN**2))/(50*ENUL)

FIINIT(W1)=WO
FIINIT(U1)=1.0E-9
FIINIT(V1)=1.0E-9
FIINIT(P1)=0.0
FIINIT(KE)=KEIN
FIINIT(EP)=EPIN
  *
  ***************************************************************
  * GROUP 12. Convection and diffusion adjustments
  ***************************************************************
  *
  * GROUP 13. Boundary conditions and special sources
  *
  ** inlet bc
INLET(IN,LOW,1,NX,1+BNY,NY,1,1,1,1)
VALUE(IN,P1,GRND)
VALUE(IN,W1,GRND)
VALUE(IN,V1,GRND)
VALUE(IN,KE,GRND)
VALUE(IN,EP,GRND)
  ** Outlet bc
PATCH(OUT,HIGH,1,NX,1+BNY,NY,NZ,NZ,1,1)
COVAL(OUT,P1,1000,GRND)
COVAL(OUT,V1,ONLYMS,0.0)
COVAL(OUT,W1,ONLYMS,0.0)
COVAL(OUT,KE,ONLYMS,0.0)
COVAL(OUT,EP,ONLYMS,0.0)

  ** bed wall
PATCH (WALLSBED,SWALL,1,NX,1,1,1,NZ,1,1)
COVAL(WALLSBED,W1,GRND2,0.0)
COVAL(WALLSBED,KE,1.0,0.0)
COVAL(WALLSBED,LTLS,1.0,0.0)


PATCH (WALLSTOP,NWALL,1,NX,NY,NY,1,NZ,1,1)
COVAL(WALLSTOP,W1,GRND2,0.0)
COVAL(WALLSTOP,KE,1.0,0.0)
COVAL(WALLSTOP,LTLS,1.0,0.0)
```

```
Definition of blocks (z direction)
Given as last cell in region, ie of block or seperation

IBC1=1+BNZ
IBC2=IBC1+SNZ
IBC3=IBC2+BNZ
IBC4=IBC3+SNZ

  ** blockages
```

```
CONPOR (BLOC1,0.0,CELL,-#1,-#1,-1,-BNY,-1,-IBC1)

   * set the w1 vel=0 in cell
CONPOR (BLOC2,0.0,CELL,-#1,-#1,-1,-BNY,-(IBC2+1),-IBC3)
PATCH(INIW2,INIVAL,#1,#1,1,BNY,IBC2,IBC2,1,1)
INIT(INIW2,W1,0.0,0.0)

        NIW1=NIW1+BNZ+SNZ
CONPOR (BLOC3,0.0,CELL,-#1,-#1,-1,-BNY,-(IBC4+1),-(IBC4+1))
PATCH(INIW3,INIVAL,#1,#1,1,BNY,IBC4,IBC4,1,1)
INIT(INIW3,W1,0.0,0.0)
  *
  ***************************************************************
  * GROUP 14. Downstream pressure (for free parabolic flow).
  ***************************************************************
  *
  * GROUP 15. Termination criteria for sweeps and
  *          outer iterations.
  *
  * Number of sweeps
LSWEEP=99999999
  LSWEEP=2
  *
  ***************************************************************
  * GROUP 16. Termination criteria for inner iterations.
  ***************************************************************
  *
  * GROUP 17. Under-relaxation and related devices.
  *
    * Variable declarations
REAL(MAXV,MINL,RELX)
    * Estimate of the maximum velocity within domain
MAXV=WO*1.5
    * Estimate of the minimum cell dimension
MINL=5.0000E-05
    * Level of relaxation (100 - weak, 0.1 - strong)
RELX=30
    * AUTO Linear relaxation applied to P1
    * (0.9 weak, 0.1 strong)
RELAX(P1,LINRLX,0.7)
    * AUTO False time-step relaxation applied to V1
RELAX(V1,FALSDT,MINL/MAXV*RELX)
    * AUTO False time-step relaxation applied to W1
RELAX(W1,FALSDT,MINL/MAXV*RELX)
    * AUTO False time-step relaxation applied to KE
RELAX(KE,FALSDT,MINL/MAXV*RELX)
    * AUTO False time-step relaxation applied to EP
RELAX(EP,FALSDT,MINL/MAXV*RELX)
  *
  ***************************************************************
  * GROUP 18. Limits on variables values or increments
  *          to them.
  ***************************************************************
  * GROUP 19. Data communicated by SATELLITE to GROUND
  * ROUGHGRD.F uses locations 1 to 9
  * CYLICGRD.F uses locations 10 to 19
  *
```

```
   * Bulk vel+ block cells
RG(10)=WO
IG(4)=BNY

   * Recycling stages:
   *
   *    NF       Number of sweeps per stage
IG(10)=3000
   *    RS1      no recycle
IG(11)=0
   *    RS2      vel and turb only
IG(12)=0
   *    RS3      vel, turb. and press
IG(13)=20
   *    RS4      vel and turb only
IG(14)=0
   *    RS5      vel, turb and press
IG(15)=0
   *    RS6      vel and turb only
IG(16)=0
   *    RS7      no recycle
IG(17)=0
   *
   * monitoring. Set IG(20) to: 0 for no monitoring
   *                            1 for complete monitoring
   *                            ? for specific mon. See CYCLEGRD
IG(19)=1
   *

   ***************************************************************
   * GROUP 20. Control of preliminary printout
   ***************************************************************
   * GROUP 21. Frequency and extent of field printout.
YPLS=F
ECHO=T
OUTPUT(P1,Y,N,N,Y,N,Y)
OUTPUT(W1,Y,N,N,Y,N,Y)
OUTPUT(U1,N,N,N,Y,N,Y)
OUTPUT(V1,N,N,N,N,N,N)
OUTPUT(KE,N,N,N,N,N,Y)
OUTPUT(EP,N,N,N,N,N,Y)
OUTPUT(VPOR,N,N,N,N,N,N)
OUTPUT(LTLS,N,N,N,N,N,N)
OUTPUT(ENUT,N,N,N,N,N,N)
   ***************************************************************
   * GROUP 22. Location of spot-value & frequency of
   *           residual printout.
   *
IYMON=3*NY/4
IZMON=NZ/2
IXMON=NX/2
TSTSWP=10
   *
   ***************************************************************
   * GROUP 23. Variable-by-variable field printout and plot
   *           and/or tabulation of spot-values and residuals.
   *
   * Control tabulation & plotting of spot-values/residuals
```

```
  * Tables and plots
    YZPR=T
NZPRIN=1
NYPRIN=1
  *
  **********************************************************************
  * GROUP 24. Preparation for continuation runs.
  **********************************************************************
STOP
```

A8.3 Ground Coding

```
      SUBROUTINE GROUND
C
C  ***************************************************************
C  *          CYCLIC DUCT FLOW - Flow Rate Method
C  *             ALASTAIR SENIOR, RMCS
C  ***************************************************************
C  ***************************************************************
C
C
C    FILE NAME: ground.f.f
C    DATE:     2/12/96
C    VERSION:  2.1.1 (Also suitable for 2.0 - change INCLUDE)
C    COMMENTS: Full 3d cyclic B.C., using 1D arrays.
C              Suitable for rough ducts only.
C              Recyc. every NF+2 sweeps.
C              Recycling in stages: (Set in Q1)
C
C              In Q1 set:
C              GRND as VALUE of P1, W1, U1, V1, EP, KE at 'IN'
C                               P1 at 'OUT'
C              LSWEEP=999999
C
C
C              Also includes monitering of spot values,
C              set in G19 of Q1.
C
C
C
C  Updates and corrections:
C    20/9/95 - clean up
C    20/9/95 - Change calc. of main flow area, MFATOT, (G19, S3)
C              to inc. blockages via VPOR, for V2.1.1, as
C              CONPOR no longer holds blocked area.
C    25/9/95 - Put recyclicing into subroutines, gives
C              more flexibilty in the type of recyclicing.
C              Change q1 to grd array location assignments.
C    27/9/95 - Correction to RS2, G19, S1 [was RS2=RS3+IG(12)]
C
C    27/9/95 - Correction to calc of upstream pressure average,
C              now starts from BNY+1
C              Also changed monitoring to grid independant
C
C
C
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
```

```
C  Version 2.1.1
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C  Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'
C
C 1   Set dimensions of data-for-GROUND arrays here. WARNING: the
C     corresponding arrays in the MAIN program of the satellite
C     and EARTH must have the same dimensions.
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
C
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
C  This common block gives the loopz variable (used in G19)
      COMMON/GENI/IGFL1(45),LOOPZ,IGFL2(14)
      LOGICAL LG
      CHARACTER*4 CG
      SAVE

C     Dimension statments - cyclic coding
C     Real arrays - single dimension, must be over ny*nx
C     NB: These must also be changed in subs RECVEL and RECPRES
C     AAH=high cell areas
C     P1DOWN=press. from iz=2 to 'OUT'
C     W1IN=w1 vel. from nz-1 to 'IN'
C     V1IN=v1  "  "  "
C     U1IN=u1  "  "  "
C     EPIN=EP values from nz-1 to 'IN'
C     KEIN=KE values from nz-1 to 'IN'

      REAL W1IN(100),AAH(100),P1DOWN(100)
      REAL V1IN(100),EPIN(100),KEIN(100),U1IN(100)

C     Real variables
C     WO=specified bulk vel.
C     WB=calc. bluk vel.
C     CMASS=scale factor for mass flow
C     MFATOT=total of main flow high areas
C     PA=outlet average press.

      REAL WO,WB,CMASS,ATOT,MFATOT,PA

C     Integers
C     NC=counter for recycle
C     NF=sweep freq. for recycle
C     N=do loop counter
C     L0***=F-array indicator
C     I=F-array indicial expresion
C     RS1= 1st stage in recyle
C     RS2= 2st stage in recyle
C     RS3= 3st stage in recyle
C     RS4= 4st stage in recyle
C     MONTYPE= monitor type (set in Q1 via IG(2) )
C     BNY= number of cell in roughness element
```

```
      INTEGER NC,NF,N,I,RS1,RS2,RS3,RS4,MONTYPE,BNY

C
c*************************************************************************
c
      IXL=IABS(IXL)
C   NB: some sections have been removed for simplisty.
      IF(IGR.EQ.1) GO TO 1
      IF(IGR.EQ.13) GO TO 13
      IF(IGR.EQ.19) GO TO 19
      RETURN
C*************************************************************************
C
C--- GROUP 1. Run title and other preliminaries
C
    1 GO TO (1001,1002),ISC
 1001 CONTINUE
C
C     User may here change message transmitted to the VDU screen
      IF(IGR.EQ.1.AND.ISC.EQ.1.AND..NOT.NULLPR)
     1  CALL WRYT40('A. Senior. Cyclic b.c: flowrate method. ')
C
      RETURN
 1002 CONTINUE
      RETURN
C*************************************************************************
C
C--- GROUP 13. Boundary conditions and special sources
C                                 Index for Coefficient - CO
C                                 Index for Value       - VAL
   13 CONTINUE
      GO TO (130,131,132,133,134,135,136,137,138,139,1310,
     11311,1312,1313,1314,1315,1316,1317,1318,1319,1320,1321),ISC
  130 CONTINUE
C------------------ SECTION  1 ------------ coefficient = GRND
      RETURN
  131 CONTINUE
C------------------ SECTION  2 ------------ coefficient = GRND1
      RETURN
  132 CONTINUE
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
           !! Material omitted !!
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
C------------------ SECTION 11 ------------ coefficient = GRND10
      RETURN
 1311 CONTINUE
C------------------ SECTION 12 ------------------ value = GRND
C  Set cyclic b.c.
      IF (NPATCH.EQ.'IN') THEN
          IF (INDVAR.EQ.P1) THEN
            CALL SETYX(VAL,W1IN,NY,NX)
            CALL FN25(VAL,RHO1)
            CALL FN25(VAL,CMASS)
          ELSE IF (INDVAR.EQ.W1) THEN
            CALL SETYX(VAL,W1IN,NY,NX)
            CALL FN25(VAL,CMASS)
          ELSE IF (INDVAR.EQ.V1) THEN
            CALL SETYX(VAL,V1IN,NY,NX)
```

```
      ELSE IF (INDVAR.EQ.U1) THEN
        CALL SETYX(VAL,U1IN,NY,NX)
      ELSE IF (INDVAR.EQ.KE) THEN
        CALL SETYX(VAL,KEIN,NY,NX)
      ELSE IF (INDVAR.EQ.EP) THEN
        CALL SETYX(VAL,EPIN,NY,NX)
      END IF

C  Set outlet press distribution to average zero.
      ELSE IF (NPATCH.EQ.'OUT') THEN
         CALL SETYX(VAL,P1DOWN,NY,NX)
      END IF
      RETURN
 1312 CONTINUE
C------------------ SECTION 13 ------------------ value = GRND1
      RETURN
 1313 CONTINUE
C------------------ SECTION 14 ------------------ value = GRND2
      RETURN
 1314 CONTINUE
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
           !! Material omitted !!
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
C------------------ SECTION 21 ------------------ value = GRND9
      RETURN
 1321 CONTINUE
C------------------ SECTION 22 ------------------ value = GRND10
      RETURN
C*********************************************************************
C
C--- GROUP 19. Special calls to GROUND from EARTH
C
   19 GO TO (191,192,193,194,195,196,197,198,199,1910,1911),ISC
  191 CONTINUE
C   * ------------------- SECTION 1 ---- Start of time step.
C     Initialise values
C     NB: cyclicgrd uses array location 10 to 20 for q1 to grd
      CMASS=1.0
      NC=1
      WO=RG(10)
      BNY=IG(4)

      NF=IG(10)
C     set recycle stages (in NF sweeps)
C        no recycle:
      RS1=IG(11)
C        vel (and turb) only:
      RS2=RS1+IG(12)
C        vel and press:
      RS3=RS2+IG(13)
C        vel (and turb) only:
      RS4=RS3+IG(14)
C        vel and press:
      RS5=RS4+IG(15)
C        vel (and turb) only:
      RS6=RS5+IG(16)
C        no recycle:
```

```
      RS7=RS6+IG(17)

C     Set monitoring:
      MONTYPE=IG(19)

C  Initialise cyclic arrays suitable for a restart.
C    Correction 7/9/95
      L0P1=L0F(ANYZ(P1,NZ))
      L0W1=L0F(ANYZ(W1,2))
      L0V1=L0F(ANYZ(V1,2))
      L0U1=L0F(ANYZ(U1,2))
      L0EP=L0F(ANYZ(EP,2))
      L0KE=L0F(ANYZ(KE,2))

      DO 1915 IX=1,NX
       DO 1916 IY=1,NY
         I=IY+NY*(IX-1)
         P1DOWN(I)=F(L0P1+I)
         W1IN(I)=F(L0W1+I)
         V1IN(I)=F(L0V1+I)
         U1IN(I)=F(L0U1+I)
         EPIN(I)=F(L0EP+I)
         KEIN(I)=F(L0KE+I)
1916    CONTINUE
1915  CONTINUE


      RETURN
  192 CONTINUE
C   * ------------------- SECTION 2 ---- Start of sweep.
      RETURN
  193 CONTINUE
C   * ------------------- SECTION 3 ---- Start of iz slab.

C     Used to ensure only one visit in S3 AND S6
      IF(LOOPZ.EQ.1) THEN

C       Get cell area arrays for recycling;
C       (flow area only, ie total area - blockages = main flow area)

        IF (ISWEEP.EQ.1.AND.IZ.EQ.1) THEN
          L0VPOR=L0F(VPOR)
            CALL GETYX(AHIGH,AAH,NY,NX)
            MFATOT=0.0
            DO 1931 IX=1,NX
           DO 1932 IY=1,NY
C             (NB: corrected for V2.1.1 by inc. VPOR on 20/9/95)
              I=IY+NY*(IX-1)
                MFATOT=MFATOT+(AAH(I)*F(L0VPOR+I))
1932         CONTINUE
1931        CONTINUE
C         WRITE(14,*)'main flow area is',MFATOT
        END IF

      END IF

      RETURN
  194 CONTINUE
```

```
C   * ------------------ SECTION 4 ---- Start of iterations over slab.
      RETURN
 1911 CONTINUE
C   * ------------------ SECTION 11---- After calculation of convection
C                            fluxes for scalars, and of volume
C                            fractions, but before calculation of
C                            scalars or velocities
      RETURN
  199 CONTINUE
C   * ------------------ SECTION 9 ---- Start of solution sequence for
C                                        a variable
      RETURN
 1910 CONTINUE
C   * ------------------ SECTION 10---- Finish of solution sequence for
C                                        a variable
      RETURN
  195 CONTINUE
C   * ------------------ SECTION 5 ---- Finish of iterations over slab.
      RETURN
  196 CONTINUE
C   * ------------------ SECTION 6 ---- Finish of iz slab.
C     Used to ensure only one visit in S3 AND S6
      IF(LOOPZ.EQ.1) THEN
C       Convergence monitoring.
        IF (MONTYPE.GT.0) CALL MONITOR(MONTYPE)
      END IF

      RETURN
  197 CONTINUE
C   * ------------------ SECTION 7 ---- Finish of sweep.
C  Working section for recycling.
C  (ie recycle at NF + 2)
      IF (ISWEEP.EQ.((NC*NF)+2)) THEN

C       Recycle vel and turb only
        IF(NC.GT.RS1.AND.NC.LE.RS2) CALL RECVEL(W1IN, V1IN,
     1  U1IN, EPIN, KEIN, AAH, WO, MFATOT, CMASS)

C       Recycle vel and turb AND pressure
        IF(NC.GT.RS2.AND.NC.LE.RS3) THEN
          CALL RECVEL(W1IN, V1IN, U1IN, EPIN, KEIN, AAH,
     1  WO, MFATOT, CMASS)
          CALL RECPRES(P1DOWN, AAH, MFATOT,BNY)

        END IF

C       Recycle vel and turb only
        IF(NC.GT.RS3.AND.NC.LE.RS4) CALL RECVEL(W1IN, V1IN,
     1  U1IN, EPIN, KEIN, AAH, WO, MFATOT, CMASS)

C       Recycle vel and turb AND pressure
        IF(NC.GT.RS4.AND.NC.LE.RS5) THEN
          CALL RECVEL(W1IN, V1IN, U1IN, EPIN, KEIN, AAH,
     1  WO, MFATOT, CMASS)
          CALL RECPRES(P1DOWN, AAH, MFATOT,BNY)
        END IF

C       Recycle vel and turb only
```

```
      IF(NC.GT.RS5.AND.NC.LE.RS6) CALL RECVEL(W1IN, V1IN,
     1  U1IN, EPIN, KEIN, AAH, WO, MFATOT, CMASS)

C       counter for recycling
        NC=NC+1
      END IF
C   End cyclic codeing

C   Finish computation (set LSWEEP)
      IF (ISWEEP.EQ.((NF*RS7)-1)) THEN
          WRITE(14,*)'Finish at isweep = ',ISWEEP
          LSWEEP=ISWEEP+1
          ENUFSW=T
      END IF

C     Temp section 27/9/95
      IF(ISWEEP.EQ.LSWEEP) THEN
      L0P1=L0F(ANYZ(P1,NZ))
      PA=0
      DO 1971 IX=1,NX
         DO 1972 IY=BNY+1,NY
            I=IY+NY*(IX-1)
            PA=PA+(F(L0P1+I)*AAH(I))
1972       CONTINUE
1971  CONTINUE
      PA=PA/MFATOT
      WRITE(14,*)'P1 outlet av = ',PA
      END IF

      RETURN
  198 CONTINUE
C   * ------------------- SECTION 8 ---- Finish of time step.
C
      RETURN

      END
C*********************************************************************
      SUBROUTINE RECVEL(W1IN, V1IN, U1IN, EPIN, KEIN, AAH,
     1  WO, MFATOT, CMASS)

C
C   Rcycles velocity and turbulence, ie fills the arrays:
C   W1IN, V1IN, U1IN, EPIN, KEIN from NZ-1.
C   Also calcs scale factor for mass flow.
C
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C   Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
```

```
      CHARACTER*4 CG

C     Subroutine variables
      REAL W1IN(100),V1IN(100),U1IN(100)
      REAL EPIN(100),KEIN(100),AAH(100)

      REAL WB, MFATOT, WO, CMASS
      INTEGER I

       WRITE(14,*)'Recyc vel at isweep = ',ISWEEP
       L0WI=L0F(ANYZ(W1,NZ-1))
       L0VI=L0F(ANYZ(V1,NZ-1))
       L0UI=L0F(ANYZ(U1,NZ-1))
       L0EPI=L0F(ANYZ(EP,NZ-1))
       L0KEI=L0F(ANYZ(KE,NZ-1))
       DO 10 IX=1,NX
          DO 20 IY=1,NY
             I=IY+NY*(IX-1)
             W1IN(I)=F(L0WI+I)
             V1IN(I)=F(L0VI+I)
             U1IN(I)=F(L0UI+I)
             EPIN(I)=F(L0EPI+I)
             KEIN(I)=F(L0KEI+I)
20         CONTINUE
10     CONTINUE

C      Calculate scale factor
       WB=0.0
       DO 30 N=1,NY*NX
             WB=WB+AAH(N)*W1IN(N)
30     CONTINUE
       WB=WB/MFATOT
       CMASS=WO/WB


       RETURN
       END


C*****************************************************************
      SUBROUTINE RECPRES (P1DOWN, AAH, MFATOT,BNY)
C     Recycles pressure only, ie finds P1DOWN from IZ=2, finds
C     average to give pressure drop and fills array P1DOWN.
C
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C     Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG
```

```
C  Subroutine variables
      REAL P1DOWN(100), AAH(100)
      REAL PA, MFATOT
      INTEGER BNY


      WRITE(14,*)'Recyc pres at isweep = ',ISWEEP
      L0P1=L0F(ANYZ(P1,2))

      DO 10 IX=1,NX
         DO 20 IY=1,NY
            I=IY+NY*(IX-1)
            P1DOWN(I)=F(L0P1+I)
20       CONTINUE
10    CONTINUE

C     Calc dwnstrm pressure profile
C     Find upstrm press. average.
      PA=0.0
      DO 30 IX=1,NX
         DO 40 IY=BNY+1,NY
            I=IY+NY*(IX-1)
            PA=PA+(P1DOWN(I)*AAH(I))
40       CONTINUE
30    CONTINUE
      PA=PA/MFATOT
      DO 50 IX=1,NX
         DO 60 IY=1,NY
            I=IY+NY*(IX-1)
            P1DOWN(I)=P1DOWN(I)-PA
60       CONTINUE
50    CONTINUE
      WRITE(14,*)'Pressure drop is ',PA

      RETURN
      END

C*********************************************************************
```

Appendix 9: Side Wall Effect in Smooth and Rough Ducts

In Section 4.1.1 it was argued that Schlichting's assumption that his flow was two-dimensional, i.e. that the side wall effect on the centreline bed shear stress was negligible, was valid at least for the smooth wall of his duct. The following discussion presents the case for the side wall effect on the rough wall to be also negligible.

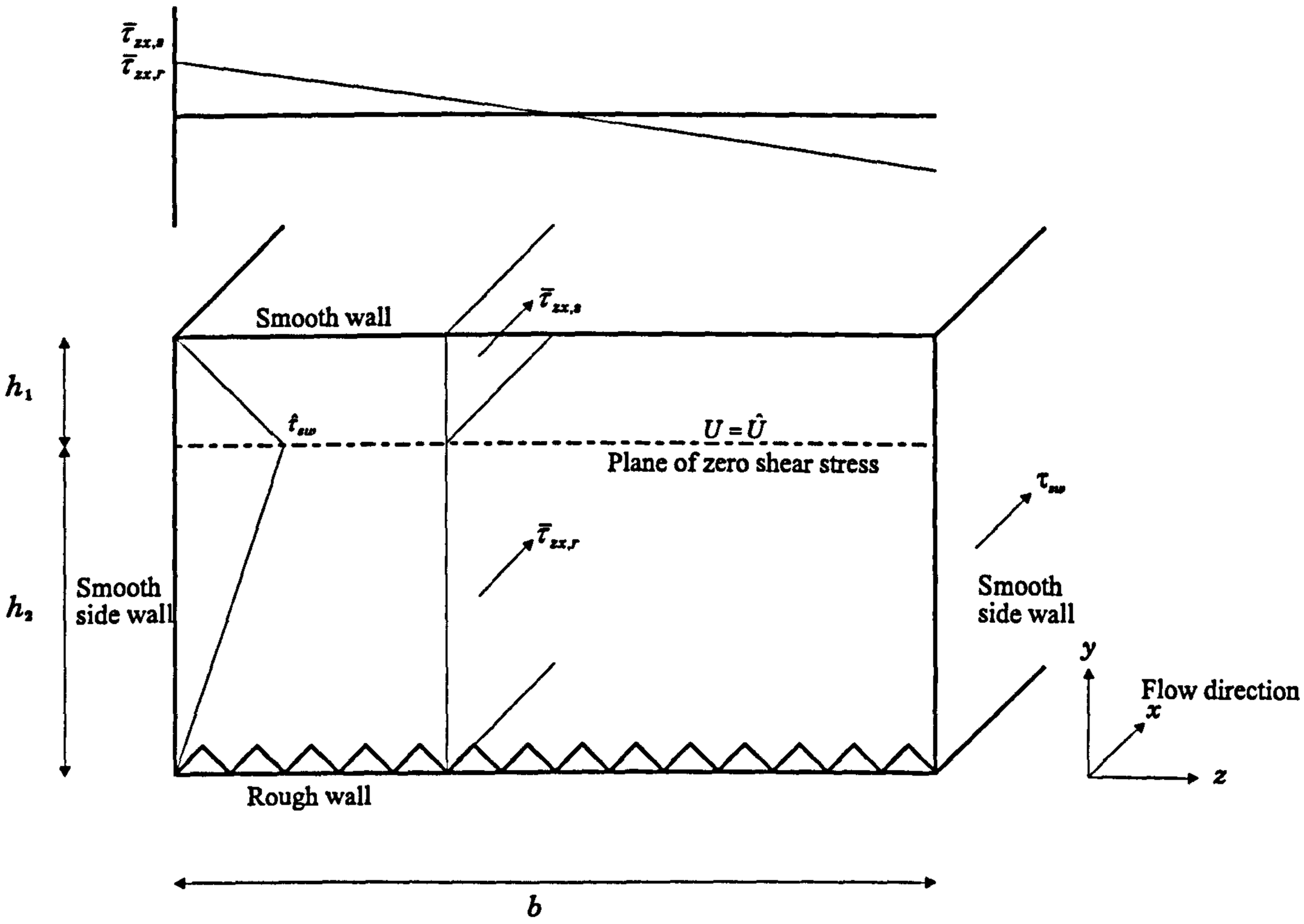


Figure A9-1: Non-symmetrically roughened duct with smooth side walls

Considering the duct shown in Figure A9-1 and assuming that a plane of zero shear stress acts at the y location of maximum streamwise velocity, then the smooth and rough wall regions may be considered as isolated symmetrical ducts with semi-depths h_1 and h_2 respectively. For the duct with smooth walls and neglecting the side wall effect, the two-dimensional bed shear stress would be

$$\tau_{s,2d} = -h_1 \frac{\mathrm{d}P}{\mathrm{d}x}$$ *Equation A9-1*

Including the effect of the side walls, this becomes

$$\tau_s = h_1 \left(-\frac{\mathrm{d}P}{\mathrm{d}x} + \frac{\mathrm{d}\bar{\tau}_{zx,s}}{\mathrm{d}z} \right)$$ *Equation A9-2*

where $\bar{\tau}_{zx,s}$ is the depth averaged value of the shear stress $\tau_{zx,s}$ in the smooth region. The percentage deviation from the two-dimensional value is

$$100 \times \frac{\tau_s - \tau_{s,2d}}{\tau_{s,2d}} = 100 \times \frac{\mathrm{d}\tau_{zx,s} / \mathrm{d}z}{\mathrm{d}P / \mathrm{d}x}$$ *Equation A9-3*

Similarly for a duct with two symmetrical rough walls

$$\tau_{r,2d} = -h_2 \frac{\mathrm{d}P}{\mathrm{d}x}$$ *Equation A9-4*

$$\tau_r = h_2 \left(-\frac{\mathrm{d}P}{\mathrm{d}x} + \frac{\mathrm{d}\bar{\tau}_{zx,r}}{\mathrm{d}z} \right)$$ *Equation A9-5*

and the deviation from the two-dimensional value is

$$100 \times \frac{\mathrm{d}\tau_{zx,r} / \mathrm{d}z}{\mathrm{d}P / \mathrm{d}x}$$ *Equation A9-6*

At the side wall, $\bar{\tau}_{zx} = \bar{\tau}_{sw}$ where $\bar{\tau}_{sw}$ is the average shear stress on the side wall. As a rough approximation, assuming a linear distribution with y of side wall shear stress, then $\bar{\tau}_{sw,r} \approx \bar{\tau}_{sw,s} \approx \frac{1}{2}\hat{\tau}_{sw}$. If the distribution of $\bar{\tau}_{zx}$ with x is also approximately linear, then

$$\frac{\mathrm{d}\bar{\tau}_{zx,s}}{\mathrm{d}z} \approx \frac{\mathrm{d}\bar{\tau}_{zx,r}}{\mathrm{d}z} \approx \frac{\hat{\tau}_{sw}}{b}$$ *Equation A9-7*

where b is the width of the duct. The deviations of centre-line bed shear stress from the two-dimensional bed shear stress values (Equation A9-3 and Equation A9-6) are therefore approximately the same for both the rough and smooth wall regions.

Appendix 10: PHOENICS Code to Model Schlichting's Duct

The following Q1 input file is typical of that used to simulate the flow in Schlichting's rough duct described in Section 4.1. The GROUND coding associated with this Q1 file is similar to that given in Appendix 11 and is therefore not repeated here.

```
TALK=F;RUN( 1, 1);VDU=X11-TERM
NOCOPY=T;NOCOMM=T
SAVE=T
PARAB=F
REAL(DEPTH,PLTH,SLTH,WIDTH,UI,DPDX)
REAL(BLKHGT,BLKLTH,MFDTH)
REAL(KEIN,EPIN)
REAL(MAXV,MINL,RELX,RELXT,RELXDP)
INTEGER(BNY,BNX,PNX,SNX,MFNY,NF,IBU,IBD)
CHAR(ID,NUMBER)
BOOLEAN(SETUI, TWALL,RSTART)
  ************************************************************************
  *
  * Group 1: CASE PARAMETERS SET UP
  * *********************************
    * Set case name and number (max 4 charaters).
ID='xxii'
NUMBER='p6_2'
TEXT(Schlichting Duct
  *
MESG( ************************************************************
MESG( *                CYCLIC DUCT FLOW
MESG( *             ALASTAIR SENIOR, RMCS
MESG( ************************************************************
MESG( * Uses xcycle and momentum source method.
MESG( *
MESG( * Case identification:
MESG(
ID
NUMBER
MESG( * Date:       2/11/96
MESG( * PHOENICS ver:   v2.1.1, SUN
MESG( * GROUND ver:     s3
MESG( *
MESG( * Previous swps: 80 k
MESG( *
MESG( * Comments: Single period. SOLAIRS
MESG( *          Dense grid
MESG( *
MESG( ************************************************************
  * specify a restart run
RSTART=T
  *
```

```
  *  Number of sweeps
LSWEEP=80000
  *
  * Domain dimensions:
BLKLTH=0.0003
PLTH=0.02
SLTH=PLTH-BLKLTH
BLKHGT=0.003
DEPTH=0.0397
MFDTH=DEPTH-BLKHGT
  *
  * y-grid set up
BNY=32
MFNY=82
NY=BNY+MFNY
YVLAST=DEPTH
  *
  * x-xgrid set up
NX=71
BNX=8
SNX=63
XULAST=PLTH
  *
  *  Estimate of the minimum cell dimension
MINL=5.0000E-05
  *
  * Fluid properties, Water in M (from Sch)
RHO1=998.2
ENUL=1.1700e-6
  *
  * Specify flow. Set SETUI =T   to specfiy a average velocity
  *                    =F   to specify a pressure gradient
SETUI=F
  *
IF (SETUI) THEN
+   MESG( Setting required velocity...
    ************************************
    * Specify required average velocity above blocks
+   UI=0
    * specify initial estimate for pressure gradient
+   DPDX=0
    * specify update frequency for pressure gradient
    * (NB: use =0 for manual update of pressure gradient)
+   NF=0
    *
+   IF (NF.GT.0) THEN
+     MESG( with automatic update of pressure gradient.
+   ELSE
+     MESG( with manual update of pressure gradient.
+   ENDIF
ELSE
+   MESG( Setting required pressure gradient.
    *******************************************
    * specify required pressure gradient
+   DPDX=21960.4000
    * Specify initial estimate of velocity field
    * (not required for continuation runs)
+   UI=7
```

```
     * NF is not required; use default of zero
+    NF=0
 ENDIF
   ************************************************************************
   *  End of set up.
   *  ***************
   * Additional preliminaries
   *
   * x grid block labels
 IBD=BNX/2
IBU=IBD+SNX+1
  *
   * Estimate of the maximum velocity within domain
MAXV=UI*1.5
   *
CARTES=T
STEADY=T
   ************************************************************************
   * GROUP 3.  x-direction grid specification.
XFRAC( 1)= -4;XFRAC( 2)=   1.87500E-03
XFRAC( 3)=1;XFRAC( 4)=   2.50000E-03
XFRAC( 5)=1;XFRAC( 6)=   2.76013E-03
XFRAC( 7)=1;XFRAC( 8)=   3.04733E-03
XFRAC( 9)=1;XFRAC( 10)=   3.36442E-03
XFRAC( 11)=1;XFRAC( 12)=   3.71449E-03
XFRAC( 13)=1;XFRAC( 14)=   4.10100E-03
XFRAC( 15)=1;XFRAC( 16)=   4.52772E-03
XFRAC( 17)=1;XFRAC( 18)=   4.99884E-03
XFRAC( 19)=1;XFRAC( 20)=   5.51898E-03
XFRAC( 21)=1;XFRAC( 22)=   6.09325E-03
XFRAC( 23)=1;XFRAC( 24)=   6.72727E-03
XFRAC( 25)=1;XFRAC( 26)=   7.42726E-03
XFRAC( 27)=1;XFRAC( 28)=   8.20009E-03
XFRAC( 29)=1;XFRAC( 30)=   9.05334E-03
XFRAC( 31)=1;XFRAC( 32)=   9.99536E-03
XFRAC( 33)=1;XFRAC( 34)=   1.10354E-02
XFRAC( 35)=1;XFRAC( 36)=   1.21837E-02
XFRAC( 37)=1;XFRAC( 38)=   1.34514E-02
XFRAC( 39)=1;XFRAC( 40)=   1.48511E-02
XFRAC( 41)=1;XFRAC( 42)=   1.63964E-02
XFRAC( 43)=1;XFRAC( 44)=   1.81025E-02
XFRAC( 45)=1;XFRAC( 46)=   1.99861E-02
XFRAC( 47)=1;XFRAC( 48)=   2.20657E-02
XFRAC( 49)=1;XFRAC( 50)=   2.43617E-02
XFRAC( 51)=1;XFRAC( 52)=   2.68966E-02
XFRAC( 53)=1;XFRAC( 54)=   2.96953E-02
XFRAC( 55)=1;XFRAC( 56)=   3.27852E-02
XFRAC( 57)=1;XFRAC( 58)=   3.61966E-02
XFRAC( 59)=1;XFRAC( 60)=   3.99629E-02
XFRAC( 61)= 5;XFRAC( 62)=   3.70000E-02
XFRAC( 63)=1;XFRAC( 64)=   3.99629E-02
XFRAC( 65)=1;XFRAC( 66)=   3.61966E-02
XFRAC( 67)=1;XFRAC( 68)=   3.27852E-02
XFRAC( 69)=1;XFRAC( 70)=   2.96953E-02
XFRAC( 71)=1;XFRAC( 72)=   2.68966E-02
XFRAC( 73)=1;XFRAC( 74)=   2.43617E-02
XFRAC( 75)=1;XFRAC( 76)=   2.20657E-02
XFRAC( 77)=1;XFRAC( 78)=   1.99861E-02
```

XFRAC(79)=1;XFRAC(80)= 1.81025E-02
XFRAC(81)=1;XFRAC(82)= 1.63964E-02
XFRAC(83)=1;XFRAC(84)= 1.48511E-02
XFRAC(85)=1;XFRAC(86)= 1.34514E-02
XFRAC(87)=1;XFRAC(88)= 1.21837E-02
XFRAC(89)=1;XFRAC(90)= 1.10354E-02
XFRAC(91)=1;XFRAC(92)= 9.99536E-03
XFRAC(93)=1;XFRAC(94)= 9.05334E-03
XFRAC(95)=1;XFRAC(96)= 8.20009E-03
XFRAC(97)=1;XFRAC(98)= 7.42726E-03
XFRAC(99)=1;XFRAC(100)= 6.72727E-03
XFRAC(101)=1;XFRAC(102)= 6.09325E-03
XFRAC(103)=1;XFRAC(104)= 5.51898E-03
XFRAC(105)=1;XFRAC(106)= 4.99884E-03
XFRAC(107)=1;XFRAC(108)= 4.52772E-03
XFRAC(109)=1;XFRAC(110)= 4.10100E-03
XFRAC(111)=1;XFRAC(112)= 3.71449E-03
XFRAC(113)=1;XFRAC(114)= 3.36442E-03
XFRAC(115)=1;XFRAC(116)= 3.04733E-03
XFRAC(117)=1;XFRAC(118)= 2.76013E-03
XFRAC(119)=1;XFRAC(120)= 2.50000E-03
XFRAC(121)= 4;XFRAC(122)= 1.87500E-03

****

*

* GROUP 4. y-direction grid specification.

*

YFRAC(1)=-1;YFRAC(2)= 6.29723E-04
YFRAC(3)=1;YFRAC(4)= 7.00477E-04
YFRAC(5)=1;YFRAC(6)= 7.79182E-04
YFRAC(7)=1;YFRAC(8)= 8.66729E-04
YFRAC(9)=1;YFRAC(10)= 9.64113E-04
YFRAC(11)=1;YFRAC(12)= 1.07244E-03
YFRAC(13)=1;YFRAC(14)= 1.19294E-03
YFRAC(15)=1;YFRAC(16)= 1.32697E-03
YFRAC(17)=1;YFRAC(18)= 1.47607E-03
YFRAC(19)=1;YFRAC(20)= 1.64192E-03
YFRAC(21)=1;YFRAC(22)= 1.82640E-03
YFRAC(23)=1;YFRAC(24)= 2.03161E-03
YFRAC(25)=1;YFRAC(26)= 2.25988E-03
YFRAC(27)=1;YFRAC(28)= 2.51380E-03
YFRAC(29)=1;YFRAC(30)= 2.79624E-03
YFRAC(31)=1;YFRAC(32)= 3.11042E-03
YFRAC(33)= 16;YFRAC(34)= 3.14861E-03
YFRAC(35)=1;YFRAC(36)= 2.51889E-03
YFRAC(37)=1;YFRAC(38)= 2.78686E-03
YFRAC(39)=1;YFRAC(40)= 3.08334E-03
YFRAC(41)=1;YFRAC(42)= 3.41136E-03
YFRAC(43)=1;YFRAC(44)= 3.77427E-03
YFRAC(45)=1;YFRAC(46)= 4.17579E-03
YFRAC(47)=1;YFRAC(48)= 4.62003E-03
YFRAC(49)=1;YFRAC(50)= 5.11153E-03
YFRAC(51)=1;YFRAC(52)= 5.65531E-03
YFRAC(53)=1;YFRAC(54)= 6.25694E-03
YFRAC(55)=1;YFRAC(56)= 6.92258E-03
YFRAC(57)=1;YFRAC(58)= 7.65903E-03
YFRAC(59)=1;YFRAC(60)= 8.47383E-03
YFRAC(61)=1;YFRAC(62)= 9.37531E-03
YFRAC(63)=1;YFRAC(64)= 1.03727E-02

YFRAC(65)=1;YFRAC(66)= 1.14762E-02
YFRAC(67)=1;YFRAC(68)= 1.26971E-02
YFRAC(69)=1;YFRAC(70)= 1.40478E-02
YFRAC(71)=1;YFRAC(72)= 1.55423E-02
YFRAC(73)=1;YFRAC(74)= 1.71957E-02
YFRAC(75)=1;YFRAC(76)= 1.90251E-02
YFRAC(77)=1;YFRAC(78)= 2.10491E-02
YFRAC(79)=1;YFRAC(80)= 2.32883E-02
YFRAC(81)=1;YFRAC(82)= 2.57658E-02
YFRAC(83)=1;YFRAC(84)= 2.85069E-02
YFRAC(85)=1;YFRAC(86)= 3.15396E-02
YFRAC(87)=1;YFRAC(88)= 3.48949E-02
YFRAC(89)=1;YFRAC(90)= 3.86071E-02
YFRAC(91)= 8;YFRAC(92)= 3.99874E-02
YFRAC(93)= 8;YFRAC(94)= 1.57431E-02
YFRAC(95)=1;YFRAC(96)= 9.69372E-03
YFRAC(97)=1;YFRAC(98)= 8.78304E-03
YFRAC(99)=1;YFRAC(100)= 7.95792E-03
YFRAC(101)=1;YFRAC(102)= 7.21031E-03
YFRAC(103)=1;YFRAC(104)= 6.53294E-03
YFRAC(105)=1;YFRAC(106)= 5.91920E-03
YFRAC(107)=1;YFRAC(108)= 5.36312E-03
YFRAC(109)=1;YFRAC(110)= 4.85928E-03
YFRAC(111)=1;YFRAC(112)= 4.40278E-03
YFRAC(113)=1;YFRAC(114)= 3.98916E-03
YFRAC(115)=1;YFRAC(116)= 3.61440E-03
YFRAC(117)=1;YFRAC(118)= 3.27484E-03
YFRAC(119)=1;YFRAC(120)= 2.96719E-03
YFRAC(121)=1;YFRAC(122)= 2.68844E-03
YFRAC(123)=1;YFRAC(124)= 2.43587E-03
YFRAC(125)=1;YFRAC(126)= 2.20703E-03
YFRAC(127)=1;YFRAC(128)= 1.99969E-03
YFRAC(129)=1;YFRAC(130)= 1.81183E-03
YFRAC(131)=1;YFRAC(132)= 1.64162E-03
YFRAC(133)=1;YFRAC(134)= 1.48740E-03
YFRAC(135)=1;YFRAC(136)= 1.34766E-03
YFRAC(137)=1;YFRAC(138)= 1.22106E-03
YFRAC(139)=1;YFRAC(140)= 1.10635E-03
YFRAC(141)=1;YFRAC(142)= 1.00241E-03
YFRAC(143)=1;YFRAC(144)= 9.08239E-04
YFRAC(145)=1;YFRAC(146)= 8.22914E-04
YFRAC(147)=1;YFRAC(148)= 7.45606E-04
YFRAC(149)=1;YFRAC(150)= 6.75560E-04
YFRAC(151)=1;YFRAC(152)= 6.12094E-04
YFRAC(153)=1;YFRAC(154)= 5.54591E-04
YFRAC(155)=1;YFRAC(156)= 5.02490E-04
YFRAC(157)=1;YFRAC(158)= 4.55284E-04
YFRAC(159)=1;YFRAC(160)= 4.12512E-04
YFRAC(161)=1;YFRAC(162)= 3.73759E-04
YFRAC(163)=1;YFRAC(164)= 3.38646E-04
YFRAC(165)=1;YFRAC(166)= 3.06832E-04
YFRAC(167)=1;YFRAC(168)= 2.78006E-04
YFRAC(169)=1;YFRAC(170)= 2.51889E-04

```
*
**********************************************************************
*
* GROUP 5.  z-direction grid specification.
```

```
*
*
*****************************************************************
* GROUP 6.  Body-fitting and other grid distortions.
*****************************************************************
* GROUP 7.  Variables (including porosities) named,
*           stored & solved.
SOLVE(V1)
SOLVE(U1)
SOLVE(P1)
   * Y in SOLUTN argument list denotes:
   * 1-stored 2-solved 3-whole-field
   * 4-point-by-point 5-explicit 6-harmonic averaging
SOLUTN(P1  ,Y,Y,N,N,N,N)
SOLUTN(U1  ,Y,Y,N,N,N,Y)
SOLUTN(V1  ,Y,Y,N,N,N,Y)
  *
STORE(ENUT)
   * Storage for effective shear stress variable
STORE(SHR, SCN, SFN, SDN, SDE)
STORE(CUS,CUN,CON,DUN,DVE,UDIF,VDIF)
  *
*****************************************************************
* GROUP 8.  Terms (in differential equations) and devices.
*
* Used to extract convection and diffusion fluxes in ground
UCONV=T
UDIFF=T
*****************************************************************
* GROUP 9.  Properties of the medium (or media).
*
REAL(RREF1)
   *
RREF1=RHO1
TURMOD(KEMODL);IENUTA=8;DISWAL
REAL(ENLREF);ENLREF=ENUL
*****************************************************************
* GROUP 10. Interphase-transfer processes and properties.
*****************************************************************
* GROUP 11. Initialization of fields of variables,
*           porosities, etc.
*
*
IF (RSTART) THEN
+    MESG( Running on from previous PHI file.
+    RESTRT(ALL)
ELSE
+    MESG( Running from scratch.
+    KEIN=(UI*0.05)**2
+    EPIN=(0.009*(KEIN**2))/(50*ENUL)
+    FIINIT(U1)=UI
     * The U1 vel may be set to a different initial value
     * by using the last argument in INIT below
+    PATCH(VEL0,INIVAL,1,NX,1,BNY,1,1,1,1)
+    INIT(VEL0,U1,0,UI/2)
+    FIINIT(V1)=0.0
+    FIINIT(P1)=0.0
+    FIINIT(KE)=KEIN
```

```
+    FIINIT(EP)=EPIN
ENDIF
  *
  ***************************************************************
  * GROUP 12. Convection and diffusion adjustments
  ***************************************************************
  *
  * GROUP 13. Boundary conditions and special sources
  *
  * momentum source for U1
PATCH(U1MT,VOLUME,1,NX,1,NY,1,NZ,1,1)
IF (SETUI.AND.NF.GT.0) THEN
+   MESG( Adjusting pressure gradient via GROUND.
+   COVAL(U1MT,U1,FIXFLU,GRND)
ELSE
+   COVAL(U1MT,U1,FIXFLU,DPDX)
ENDIF
PATCH(BAL,CELL,1,1,NY,NY,1,1,1,1)
COVAL(BAL,P1,FIXP,0.0)
XCYCLE=T

  * bed wall
PATCH (WALLSBED,SWALL,1,NX,1,1,1,NZ,1,1)
COVAL(WALLSBED,U1,GRND2,0.0)
COVAL(WALLSBED,KE,1.0,0.0)
COVAL(WALLSBED,LTLS,1.0,0.0)

  * top wall
TWALL=T
PATCH (WALLSTOP,NWALL,1,NX,NY,NY,1,NZ,1,1)
COVAL(WALLSTOP,U1,GRND2,0.0)
COVAL(WALLSTOP,KE,1.0,0.0)
COVAL(WALLSTOP,LTLS,1.0,0.0)

  * Definition of blocks (x direction)
  ** downstream half of block
CONPOR (BLOCD,0.0,CELL,-1,-IBD,-1,-BNY,1,1)
  ** upstream half of block
  ** (+set the u1 vel=0 in cell prior to block)
CONPOR (BLOCU,0.0,CELL,-IBU,NX,-1,-BNY,1,1)
PATCH(INIU,INIVAL,IBU-1,IBU-1,1,BNY,1,1,1,1)
INIT(INIU,U1,0.0,0.0)
  *
  ***************************************************************
  * GROUP 14. Downstream pressure (for free parabolic flow).
  ***************************************************************
  *
* GROUP 15. Termination criteria for sweeps and
*         outer iterations.
* See G1
*
***************************************************************
* GROUP 16. Termination criteria for inner iterations.
***************************************************************
*
* GROUP 17. Under-relaxation and related devices.
*
* Level of relaxation (100 - weak, 0.1 - strong)
```

```
RELX=30
RELXT=10
  * AUTO Linear relaxation applied to P1
  * (0.9 weak, 0.1 strong)
RELAX(P1,LINRLX,0.7)
  * AUTO False time-step relaxation applied to V1
RELAX(V1,FALSDT,MINL/MAXV*RELX)
  * AUTO False time-step relaxation applied to U1
RELAX(U1,FALSDT,MINL/MAXV*RELX)
  * AUTO False time-step relaxation applied to KE
RELAX(KE,FALSDT,MINL/MAXV*RELXT)
  * AUTO False time-step relaxation applied to EP
RELAX(EP,FALSDT,MINL/MAXV*RELXT)
  * Ground relaxation in calc. of dpdx
RELXDP=0.5
  *
  ********************************************************************
  * GROUP 18. Limits on variables values or increments
  *           to them.
  ********************************************************************
  * GROUP 19. Data communicated by SATELLITE to GROUND
RG(1)=PLTH
RG(2)=BLKLTH
RG(3)=DEPTH
RG(4)=BLKHGT
IG(1)=SNX
IG(2)=BNX
IG(3)=MFNY
IG(4)=BNY
RG(5)=DPDX
RG(6)=UI
RG(7)=RELXDP
CG(1)=':ID:'
CG(2)=':NUMBER:'
LG(1)=TWALL
LG(2)=SETUI
  * Set monitoring. 0 = no monitoring
  *                 1 = complete monitoring
  *                 2 = P1 and U1
  *                 3 = EP and KE
  *                 4 = dpdx calc.
IG(5)=1

  ********************************************************************
  * GROUP 20. Control of preliminary printout
ECHO=T
  * Print out of y+
YPLS=T
  ********************************************************************
  * GROUP 21. Frequency and extent of field printout.
ECHO=T
YPLS=T
OUTPUT(P1,Y,N,N,Y,N,Y)
OUTPUT(U1,Y,N,N,Y,N,Y)
OUTPUT(V1,Y,N,N,N,N,N)
OUTPUT(KE,Y,N,N,Y,N,Y)
OUTPUT(EP,Y,N,N,Y,N,Y)
OUTPUT(VPOR,N,N,N,N,N,N)
```

```
OUTPUT(LTLS,N,N,N,N,N,N)
OUTPUT(WDIS,N,N,N,N,N,N)
OUTPUT(VDIF,N,N,N,N,N,N)
OUTPUT(UDIF,N,N,N,N,N,N)
OUTPUT(DVE,N,N,N,N,N,N)
OUTPUT(DUN,N,N,N,N,N,N)
OUTPUT(CON,N,N,N,N,N,N)
OUTPUT(CUN,N,N,N,N,N,N)
OUTPUT(CUS,N,N,N,N,N,N)
OUTPUT(SDE,N,N,N,N,N,N)
OUTPUT(SDN,N,N,N,N,N,N)
OUTPUT(SFN,N,N,N,N,N,N)
OUTPUT(SCN,N,N,N,N,N,N)
OUTPUT(SHR,Y,N,N,N,N,N)
OUTPUT(ENUT,Y,N,N,N,N,N)
  ***************************************************************
  * GROUP 22. Location of spot-value & frequency of
  *         residual printout.
IYMON=3*NY/4
IZMON=NZ/2
IXMON=NX/2
TSTSWP=100
  *
  ***************************************************************
  * GROUP 23. Variable-by-variable field printout and plot
  *         and/or tabulation of spot-values and residuals.
  * Control tabulation & plotting of spot-values/residuals
  * Tables and plots
ITABL=2
NPRINT=100000
NXPRIN=1
NYPRIN=1
  *
  ***************************************************************
  * GROUP 24. Preparation for continuation runs.
  ***************************************************************
MESG( ***************************************************************
STOP
```

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

Appendix 11: PHOENICS Code for Rough Duct Experiments

The following Q1 input file and GROUND coding was used to simulate flow over roughness elements contained in a duct in which a plane of symmetry was used to represent the free surface. The depth of flow was varied from very shallow to deep as described in Chapter 5.

A11.1 Q1 file

```
TALK=F;RUN( 1, 1);VDU=X11-TERM
NOCOPY=T;NOCOMM=T
SAVE=T
PARAB=F
REAL(DEPTH,PLTH,SLTH,WIDTH,UI,DPDX)
REAL(BLKHGT,BLKLTH,MFDTH,KEIN,EPIN)
REAL(MAXV,MINL,RELX,RELXT,RELXDP)
REAL(RE)
INTEGER(BNY,BNX,PNX,SNX,MFNY,NF,IBU,IBD)
CHAR(ID,NUMBER)
BOOLEAN(SETUI, TWALL,RSTART)
  ************************************************************
  *
  * Group 1: CASE PARAMETERS SET UP
  * *********************************

  * Set case name and number (max 4 charaters).
ID='rsa'
NUMBER='4'
TEXT(Rough Strip Duct
  *
MESG( ************************************************************
MESG( *              CYCLIC DUCT FLOW
MESG( *           ALASTAIR SENIOR, RMCS
MESG( ************************************************************
MESG( * Uses xcycle and momentum source method.
MESG( *
MESG( * Case identification:
MESG(
ID
NUMBER
MESG( * Date:
MESG( * PHOENICS ver:  v2.1.1, SUN
MESG( * GROUND ver:    s12
MESG( *
MESG( * Previous swps:
MESG( *
MESG( * Comments:
```

```
MESG( *
MESG( *
MESG( ************************************************************
MESG( *
  *
  *
  * specify a restart run
RSTART=T
  *
  * Number of sweeps
LSWEEP=10000
  *
  * Domain dimensions (m):
BLKLTH=0.3e-3
PLTH=60e-3
SLTH=PLTH-BLKLTH
BLKHGT=3.2e-3
DEPTH=0.0128
MFDTH=DEPTH-BLKHGT
  *
  * y-grid set up
BNY=36
MFNY=28
NY=BNY+MFNY
YVLAST=DEPTH
  *
  * x-xgrid set up
BNX=8
SNX=114
NX=BNX+SNX
XULAST=PLTH
  *
  * Estimate of the minimum cell dimension
MINL=0.01E-03
  *
  * Fluid properties, Water in M (from Sch)
RHO1=998.2
ENUL=1.17e-6
  *
  * Specify flow. Set SETUI =T   to specfiy a average velocity
  *                    =F   to specify a pressure gradient
SETUI=T
RE=2e5
  *
IF (SETUI) THEN
+   MESG( Setting required velocity...
    ************************************
    * Specify required average velocity above blocks
+   UI=RE*ENUL/(DEPTH-BLKHGT)
+   UI
    * specify initial estimate for pressure gradient
+   DPDX=1.48867E+06
    * specify update frequency for pressure gradient
    * (NB: use =0 for manual update of pressure gradient)
+   NF=0
    *
+   IF (NF.GT.0) THEN
+     MESG( with automatic update of pressure gradient.
```

```
+    ELSE
+      MESG( with manual update of pressure gradient.
+    ENDIF
ELSE
+    MESG( Setting required pressure gradient.
     ********************************************
     * specify required pressure gradient
+    DPDX=0
     * Specify initial estimate of velocity field
     * (not required for continuation runs)
+    UI=0
     * NF is not required; use default of zero
+    NF=0
ENDIF
   ************************************************************************
   *  End of set up.
   *  ***************
   *
   * Additional preliminaries
   *
   * x grid block labels
IBD=BNX/2
IBU=IBD+SNX+1
   *
   * Estimate of the maximum velocity within domain
MAXV=UI*1.5
   *
CARTES=T
STEADY=T
   ************************************************************************
   * GROUP 3. x-direction grid specification.
XFRAC( 1)= -4;XFRAC( 2)=   6.25000E-04
XFRAC( 3)=1;XFRAC( 4)=   8.33333E-04
XFRAC( 5)=1;XFRAC( 6)=   9.20044E-04
XFRAC( 7)=1;XFRAC( 8)=   1.01578E-03
XFRAC( 9)=1;XFRAC( 10)=   1.12147E-03
XFRAC( 11)=1;XFRAC( 12)=   1.23816E-03
XFRAC( 13)=1;XFRAC( 14)=   1.36700E-03
XFRAC( 15)=1;XFRAC( 16)=   1.50924E-03
XFRAC( 17)=1;XFRAC( 18)=   1.66628E-03
XFRAC( 19)=1;XFRAC( 20)=   1.83966E-03
XFRAC( 21)=1;XFRAC( 22)=   2.03108E-03
XFRAC( 23)=1;XFRAC( 24)=   2.24242E-03
XFRAC( 25)=1;XFRAC( 26)=   2.47575E-03
XFRAC( 27)=1;XFRAC( 28)=   2.73336E-03
XFRAC( 29)=1;XFRAC( 30)=   3.01778E-03
XFRAC( 31)=1;XFRAC( 32)=   3.33179E-03
XFRAC( 33)=1;XFRAC( 34)=   3.67847E-03
XFRAC( 35)=1;XFRAC( 36)=   4.06123E-03
XFRAC( 37)=1;XFRAC( 38)=   4.48381E-03
XFRAC( 39)=1;XFRAC( 40)=   4.95036E-03
XFRAC( 41)=1;XFRAC( 42)=   5.46546E-03
XFRAC( 43)=1;XFRAC( 44)=   6.03416E-03
XFRAC( 45)=1;XFRAC( 46)=   6.66203E-03
XFRAC( 47)=1;XFRAC( 48)=   7.35523E-03
XFRAC( 49)=1;XFRAC( 50)=   8.12057E-03
XFRAC( 51)=1;XFRAC( 52)=   8.96554E-03
XFRAC( 53)=1;XFRAC( 54)=   9.89843E-03
```

XFRAC(55)=1;XFRAC(56)= 1.09284E-02
XFRAC(57)=1;XFRAC(58)= 1.20655E-02
XFRAC(59)=1;XFRAC(60)= 1.33210E-02
XFRAC(61)= 56;XFRAC(62)= 1.30060E-02
XFRAC(63)=1;XFRAC(64)= 1.33210E-02
XFRAC(65)=1;XFRAC(66)= 1.20655E-02
XFRAC(67)=1;XFRAC(68)= 1.09284E-02
XFRAC(69)=1;XFRAC(70)= 9.89843E-03
XFRAC(71)=1;XFRAC(72)= 8.96554E-03
XFRAC(73)=1;XFRAC(74)= 8.12057E-03
XFRAC(75)=1;XFRAC(76)= 7.35523E-03
XFRAC(77)=1;XFRAC(78)= 6.66203E-03
XFRAC(79)=1;XFRAC(80)= 6.03416E-03
XFRAC(81)=1;XFRAC(82)= 5.46546E-03
XFRAC(83)=1;XFRAC(84)= 4.95036E-03
XFRAC(85)=1;XFRAC(86)= 4.48381E-03
XFRAC(87)=1;XFRAC(88)= 4.06123E-03
XFRAC(89)=1;XFRAC(90)= 3.67847E-03
XFRAC(91)=1;XFRAC(92)= 3.33179E-03
XFRAC(93)=1;XFRAC(94)= 3.01778E-03
XFRAC(95)=1;XFRAC(96)= 2.73336E-03
XFRAC(97)=1;XFRAC(98)= 2.47575E-03
XFRAC(99)=1;XFRAC(100)= 2.24242E-03
XFRAC(101)=1;XFRAC(102)= 2.03108E-03
XFRAC(103)=1;XFRAC(104)= 1.83966E-03
XFRAC(105)=1;XFRAC(106)= 1.66628E-03
XFRAC(107)=1;XFRAC(108)= 1.50924E-03
XFRAC(109)=1;XFRAC(110)= 1.36700E-03
XFRAC(111)=1;XFRAC(112)= 1.23816E-03
XFRAC(113)=1;XFRAC(114)= 1.12147E-03
XFRAC(115)=1;XFRAC(116)= 1.01578E-03
XFRAC(117)=1;XFRAC(118)= 9.20044E-04
XFRAC(119)=1;XFRAC(120)= 8.33333E-04
XFRAC(121)= 4;XFRAC(122)= 6.25000E-04

```
**********************************************************************
*
* GROUP 4.  y-direction grid specification.
*
```

YFRAC(1)=-1;YFRAC(2)= 7.81250E-04
YFRAC(3)=1;YFRAC(4)= 8.96846E-04
YFRAC(5)=1;YFRAC(6)= 1.02955E-03
YFRAC(7)=1;YFRAC(8)= 1.18188E-03
YFRAC(9)=1;YFRAC(10)= 1.35675E-03
YFRAC(11)=1;YFRAC(12)= 1.55750E-03
YFRAC(13)=1;YFRAC(14)= 1.78795E-03
YFRAC(15)=1;YFRAC(16)= 2.05250E-03
YFRAC(17)=1;YFRAC(18)= 2.35620E-03
YFRAC(19)=1;YFRAC(20)= 2.70483E-03
YFRAC(21)=1;YFRAC(22)= 3.10504E-03
YFRAC(23)=1;YFRAC(24)= 3.56447E-03
YFRAC(25)=1;YFRAC(26)= 4.09188E-03
YFRAC(27)=1;YFRAC(28)= 4.69732E-03
YFRAC(29)=1;YFRAC(30)= 5.39235E-03
YFRAC(31)=1;YFRAC(32)= 6.19021E-03
YFRAC(33)=1;YFRAC(34)= 7.10613E-03
YFRAC(35)=1;YFRAC(36)= 8.15757E-03
YFRAC(37)=1;YFRAC(38)= 9.36459E-03

```
YFRAC( 39)=1;YFRAC( 40)=  1.07502E-02
YFRAC( 41)= 16;YFRAC( 42)=  1.07422E-02
YFRAC( 43)=1;YFRAC( 44)=  3.90625E-03
YFRAC( 45)=1;YFRAC( 46)=  4.37650E-03
YFRAC( 47)=1;YFRAC( 48)=  4.90335E-03
YFRAC( 49)=1;YFRAC( 50)=  5.49363E-03
YFRAC( 51)=1;YFRAC( 52)=  6.15497E-03
YFRAC( 53)=1;YFRAC( 54)=  6.89592E-03
YFRAC( 55)=1;YFRAC( 56)=  7.72607E-03
YFRAC( 57)=1;YFRAC( 58)=  8.65616E-03
YFRAC( 59)=1;YFRAC( 60)=  9.69821E-03
YFRAC( 61)=1;YFRAC( 62)=  1.08657E-02
YFRAC( 63)=1;YFRAC( 64)=  1.21738E-02
YFRAC( 65)=1;YFRAC( 66)=  1.36393E-02
YFRAC( 67)=1;YFRAC( 68)=  1.52812E-02
YFRAC( 69)=1;YFRAC( 70)=  1.71208E-02
YFRAC( 71)=1;YFRAC( 72)=  1.91818E-02
YFRAC( 73)=1;YFRAC( 74)=  2.14910E-02
YFRAC( 75)=1;YFRAC( 76)=  2.40781E-02
YFRAC( 77)=1;YFRAC( 78)=  2.69767E-02
YFRAC( 79)=1;YFRAC( 80)=  3.02243E-02
YFRAC( 81)=1;YFRAC( 82)=  3.38628E-02
YFRAC( 83)=1;YFRAC( 84)=  3.79393E-02
YFRAC( 85)=1;YFRAC( 86)=  4.25065E-02
YFRAC( 87)=1;YFRAC( 88)=  4.76235E-02
YFRAC( 89)=1;YFRAC( 90)=  5.33566E-02
YFRAC( 91)=1;YFRAC( 92)=  5.97798E-02
YFRAC( 93)=1;YFRAC( 94)=  6.69763E-02
YFRAC( 95)=1;YFRAC( 96)=  7.50391E-02
YFRAC( 97)=1;YFRAC( 98)=  8.40725E-02
*****************************************************************
* GROUP 5.  z-direction grid specification.
*****************************************************************
* GROUP 6.  Body-fitting and other grid distortions.
*****************************************************************
* GROUP 7.  Variables (including porosities) named,
*           stored & solved.
SOLVE(V1)
SOLVE(U1)
SOLVE(P1)
  * Y in SOLUTN argument list denotes:
  * 1-stored 2-solved 3-whole-field
  * 4-point-by-point 5-explicit 6-harmonic averaging
SOLUTN(P1  ,Y,Y,N,N,N,N)
SOLUTN(U1  ,Y,Y,N,N,N,Y)
SOLUTN(V1  ,Y,Y,N,N,N,Y)
*
STORE(ENUT)
  * Storage for effective shear stress variable
STORE(SHR, SCN, SFN, SDN, SDE)
STORE(CUS,CUN,CON,DUN,DVE,UDIF,VDIF)
*
*****************************************************************
* GROUP 8.  Terms (in differential equations) and devices.
*
* Used to extract convection and diffusion fluxes in ground
UCONV=T
UDIFF=T
```

```
  ***************************************************************
  * GROUP 9.  Properties of the medium (or media).
  *
REAL(RREF1)
RREF1=RHO1
TURMOD(KEMODL);IENUTA=8;DISWAL
REAL(ENLREF);ENLREF=ENUL
  ***************************************************************
  * GROUP 10. Interphase-transfer processes and properties.
  ***************************************************************
  * GROUP 11. Initialization of fields of variables,
  *          porosities, etc.
  *
IF (RSTART) THEN
+    MESG( Running on from previous PHI file.
+    RESTRT(ALL)
ELSE
+    MESG( Running from scratch.
+    KEIN=(UI*0.05)**2
+    EPIN=(0.009*(KEIN**2))/(50*ENUL)
+    PATCH(U1INPROF,INIVAL,1,NX,1,NY,1,1,1,1)
+    INIT(U1INPROF,U1,0,GRND)
+    FIINIT(V1)=0.0
+    FIINIT(P1)=0.0
+    FIINIT(KE)=KEIN
+    FIINIT(EP)=EPIN
ENDIF
  *
  ***************************************************************
  * GROUP 12. Convection and diffusion adjustments
  ***************************************************************
  *
  * GROUP 13. Boundary conditions and special sources
  *
  * momentum source for U1
PATCH(U1MT,VOLUME,1,NX,1,NY,1,NZ,1,1)
IF (SETUI.AND.NF.GT.0) THEN
+  MESG( Adjusting pressure gradient via GROUND.
+  COVAL(U1MT,U1,FIXFLU,GRND)
ELSE
+  COVAL(U1MT,U1,FIXFLU,DPDX)
ENDIF
PATCH(BAL,CELL,1,1,NY,NY,1,1,1,1)
COVAL(BAL,P1,FIXP,0.0)
XCYCLE=T

  * bed wall
PATCH (WALLSBED,SWALL,1,NX,1,1,1,NZ,1,1)
COVAL(WALLSBED,U1,GRND2,0.0)
COVAL(WALLSBED,KE,1.0,0.0)
COVAL(WALLSBED,LTLS,1.0,0.0)

  * top wall
TWALL=F

  * Definition of blocks (x direction)
  ** downstream half of block
CONPOR (BLOCD,0.0,CELL,-1,-IBD,-1,-BNY,1,1)
```

```
  ** upstream half of block
  ** (+set the u1 vel=0 in cell prior to block)
CONPOR (BLOCU,0.0,CELL,-IBU,NX,-1,-BNY,1,1)
PATCH(INIU,INIVAL,IBU-1,IBU-1,1,BNY,1,1,1,1)
INIT(INIU,U1,0.0,0.0)
  *
  ***************************************************************
  * GROUP 14. Downstream pressure (for free parabolic flow).
  ***************************************************************
  *
  * GROUP 15. Termination criteria for sweeps and
  *           outer iterations.
  * See G1
  * Switch off auto resref
SELREF =   F
RESFAC = 1.000E-03
RESREF(P1  ) =  1.000E-08 ;RESREF(U1  ) =  1.000E-08
RESREF(V1  ) =  1.000E-08 ;RESREF(KE  ) =  1.000E-08
RESREF(EP  ) =  1.000E-08 ;RESREF(LTLS) =  1.000E-08
  *
  ***************************************************************
  * GROUP 16. Termination criteria for inner iterations.
  ***************************************************************
  *
  * GROUP 17. Under-relaxation and related devices.
  *
  * Level of relaxation (100 - weak, 0.1 - strong)
RELX=70
RELXT=70
  * AUTO Linear relaxation applied to P1
  * (0.9 weak, 0.1 strong)
RELAX(P1,LINRLX,0.8)
  * AUTO False time-step relaxation applied to V1
RELAX(V1,FALSDT,MINL/MAXV*RELX)
  * AUTO False time-step relaxation applied to U1
RELAX(U1,FALSDT,MINL/MAXV*RELX)
  * AUTO False time-step relaxation applied to KE
RELAX(KE,FALSDT,MINL/MAXV*RELXT)
  * AUTO False time-step relaxation applied to EP
RELAX(EP,FALSDT,MINL/MAXV*RELXT)
  * Ground relaxation in calc. of dpdx
RELXDP=0.5
  *
  ***************************************************************
  * GROUP 18. Limits on variables values or increments
  *           to them.
  ***************************************************************
  * GROUP 19. Data communicated by SATELLITE to GROUND
RG(1)=PLTH
RG(2)=BLKLTH
RG(3)=DEPTH
RG(4)=BLKHGT
IG(1)=SNX
IG(2)=BNX
IG(3)=MFNY
IG(4)=BNY
RG(5)=DPDX
RG(6)=UI
```

```
RG(7)=RELXDP
CG(1)=':ID:'
CG(2)=':NUMBER:'
LG(1)=TWALL
LG(2)=SETUI
  * Set monitoring. 0 = no monitoring
  *                 1 = complete monitoring
  *                 2 = P1 and U1
  *                 3 = EP and KE
  *                 4 = dpdx calc.
IG(5)=1
  ***************************************************************
  * GROUP 20. Control of preliminary printout
  ***************************************************************
  * GROUP 21. Frequency and extent of field printout.
ECHO=T
YPLS=T
OUTPUT(P1,Y,N,N,Y,N,Y)
OUTPUT(U1,Y,N,N,Y,N,Y)
OUTPUT(V1,Y,N,N,N,N,N)
OUTPUT(KE,Y,N,N,Y,N,Y)
OUTPUT(EP,Y,N,N,Y,N,Y)
OUTPUT(VPOR,N,N,N,N,N,N)
OUTPUT(LTLS,N,N,N,N,N,N)
OUTPUT(WDIS,N,N,N,N,N,N)
OUTPUT(VDIF,N,N,N,N,N,N)
OUTPUT(UDIF,N,N,N,N,N,N)
OUTPUT(DVE,N,N,N,N,N,N)
OUTPUT(DUN,N,N,N,N,N,N)
OUTPUT(CON,N,N,N,N,N,N)
OUTPUT(CUN,N,N,N,N,N,N)
OUTPUT(CUS,N,N,N,N,N,N)
OUTPUT(SDE,N,N,N,N,N,N)
OUTPUT(SDN,N,N,N,N,N,N)
OUTPUT(SFN,N,N,N,N,N,N)
OUTPUT(SCN,N,N,N,N,N,N)
OUTPUT(SHR,Y,N,N,N,N,N)
OUTPUT(ENUT,Y,N,N,N,N,N)
  ***************************************************************
  * GROUP 22. Location of spot-value & frequency of
  *           residual printout.
  *
IYMON=3*NY/4
IZMON=NZ/2
IXMON=NX/2
TSTSWP=100
  ***************************************************************
  * GROUP 23. Variable-by-variable field printout and plot
  *           and/or tabulation of spot-values and residuals.
  *
  * Control tabulation & plotting of spot-values/residuals
  * Tables and plots
ITABL=2
NPRINT=100000
NXPRIN=1
NYPRIN=1
```

```
*
*******************************************************************
* GROUP 24. Preparation for continuation runs.
*******************************************************************
MESG( *******************************************************************
STOP
```

A11.2 GROUND code

```
C23456789012345678901234567890123456789012345678901234567890123456789012
      SUBROUTINE GROUND
C
C ****************************************************************
C *              ROUGH DUCT ANALYSIS
C *             ALASTAIR SENIOR, RMCS
C *                 ROUGHGRD.F
C ****************************************************************
C
C ****************************************************************
C
C *      Single Period Rough Walled Cyclic Duct Flow        *
C
C This ground routine is to be used with a Y-X domain using the
C XCYCLE/ momentum source method of cyclic boundary conditions.
C The domain should consist of 1 full period in the X (streamwise)
C direction: 1/2 block - space - 1/2 block.
C The blocks should be named 'BLOCD' and 'BLOCU' in the Q1 file.
C
C Coding includes options of a plane of symmetry or a top wall
C and specification of a required pressure gradient or a required
C average velocity. If the latter is selected, the pressure gradient
C may be updated automatically wrt the required average velocity
C or may be updated manually at the end of each run.
C
C The elapsed run time is printed in the file "result2".
C
C Output files:
C    phi     - standard PHOENICS output.
C    result  - standard PHOENICS output.
C    result2 - Analysis and summary of results.
C    res_pau.m - matlab input file for period averaged velocity
C    res_sig.m - matlab input file for direct stress on block
C    res_tau.m - matlab input file for shear stress on block
C    res_shr.m - matlab input file for effective shear stress
C                                (con/diff method)
C   res_mshr.m - matlab input file for effective shear stress
C                                (manual method)
C    res_ynf.m - matlab input file for north face y coords from
C                BNY+1 to NY for use with res_shr.m and res_mshr.m
C
C Before using this code, CHECK THE ARRAY DIMENSIONS below.
C
C   FILE NAME: ground.f
C   DATE:     31/3/97
C   VERSION:  s12
C   COMMENTS:
C
C     Q1     to     Ground variables:
```

```
C
C   PLTH     RG(1)    PLEN
C   BLKLTH   RG(2)    BLKLEN
C   DEPTH    RG(3)    DEPTH
C   BLKHGT   RG(4)    BLKHGT
C   DPDX     RG(5)    DPDX
C   UI       RG(6)    UI
C   RELXDP   RG(7)    RELXDP
C   KSI      RG(8)    KS
C   SNX      IG(1)    SNX
C   BNX      IG(2)    BNX
C   MFNY     IG(3)    MFNY
C   BNY      IG(4)    BNY
C    /       IG(5)    MONTYPE
C   NF       IG(6)    NF
C   TWALL    LG(1)    TWALL
C   SETUI    LG(2)    SETUI
C
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C  Version 2.1.1
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
C
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
C  This common block gives the loopz variable (used in G19)
      COMMON/GENI/IGFL1(45),LOOPZ,IGFL2(14)
      LOGICAL LG
      CHARACTER*4 CG
      SAVE

C  User common block, ROUGH1
C  Integers:
C     IBU=label for IX cell of upstream face of block
C     IBD=label for IX cell of downstream face of block
C     BNY=Number of Y cells in block
C  Reals:
C     PLEN=Period length
C     DPDX=Pressure gradient

      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C  Subroutine variables
C  **********************

C  Real Arrays
C  ***************************************************************
C
C  Check that the dimensions in the corrisponding subs are the same!
C  ***************************************************************
C
C       in IX;
C    ASFTW=(subs ground, SHFOR,APRN) array shear force top wall
C    ASFBW=(subs ground, SHFOR,APRN) array shear force bottom wall

      REAL ASFTW(600),ASFBW(600)
```

```
C      in NY;
C     MBU=(subs ground,PBLK) mass flow through cell upstream of block
C     MBD=(subs ground,PBLK) mass flow through cell dwnstream of block
C     DBU=(subs ground,PBLK) diff. coeff through cell upstream of block
C     DBD=(subs ground,PBLK) diff. coeff through cell dwnstream of block

      REAL MBU(600), MBD(600), DBU(600), DBD(600)

C  Integers:
C     MONTYPE=label for type of convergance monitoring required
C     BNX=number of X cells in block
C     SNX=number of cells in the spaces between blocks
C     MFNY=number of cells in the Y direction above the blocks
C          (Main Flow area)
C     I=location index for F-array
C     IW=additional location index, used for extracting west cells.
C     IXX=loop counter if ix
C     IYY=loop counter if iy
C     L0AE=location zero point of area east segment of F-array
C     L0U1=location zero point of U1 vel segment of F-array
C     NF=DPDX effective step in sweeps
C     NC=counter of NF
C     MUIY=IY cell of period ave max vel
C     BNY=NY of block cells
C  Integer arrays:
C     STIME(20)=array giving run start time
C  Logicals:
C     SETUI=equal to .TRUE. if the velocity is specifed in the Q1
C  Reals:
C     TPFB=total pressure force on block
C     TPMU=total press. moment on upstrm face of block
C     TPMD=total press. moment on dwnstrm face of block
C     TSFB=total shear force on block
C     UI=inputed U1 velocity
C     UC=calculated U1 velocity
C     UMAX=estimate of max vel used for initial conditions
C     MU=real, max vel
C     AMFET=area, main flow, east face, total
C     BLKLEN=length (in streamwise dir) of block
C     DEPTH=total depth of domain
C     BLKHGT=height of roughness element (block)
C     DPDX=pressure grad
C     KS=estimate of Ks used in initial conditions
C     Y0=estimate of Y0 (gml) used in initial conditions
C     UTAU=estimate of Utau based on Y0 above
C  note, following used for SETUI=TRUE only
C     FLOWF=flow factor used to calc DPDX
C     RELXDP=relaxation factor for DPDX
C     ERR=error used in calc of RELXDP

      REAL TPFB,TPMU,TPMD,TSFB,UI,UC,AMFET,FLOWF, UMAX
      REAL RELXDP,ERR, DEPTH, BLKLEN, BLKHGT, MU, DPDX, KS, Y0, UTAU
C     Integers
      INTEGER MONTYPE, BNX, SNX, MFNY, I, IW, IXX, IYY, MUIY
      INTEGER L0AE, L0U1, BNY
      INTEGER STIME(20)
C     Logicals
```

```
      LOGICAL SETUI
C
c*********************************************************************
c
      IXL=IABS(IXL)
C   NB: Some sections have been removed for simplisty.
      IF(IGR.EQ.1) GO TO 1
      IF(IGR.EQ.8) GO TO 8
      IF(IGR.EQ.11) GO TO 11
      IF(IGR.EQ.13) GO TO 13
      IF(IGR.EQ.19) GO TO 19
      RETURN
C*********************************************************************
C
C--- GROUP 1. Run title and other preliminaries
C
    1 GO TO (1001,1002),ISC
 1001 CONTINUE
C
C     User may here change message transmitted to the VDU screen
      IF(IGR.EQ.1.AND.ISC.EQ.1.AND..NOT.NULLPR)
     1  CALL WRYT40('Single Period Rough Duct Ground called. ')
C
      CALL MAKE(YG2D)
      CALL MAKE(YV2D)
      CALL MAKE(XG2D)
      CALL MAKE(XU2D)


      RETURN
 1002 CONTINUE
      RETURN
C*********************************************************************
C
C--- GROUP 8. Terms (in differential equations) & devices
C
    8 CONTINUE
      IF(ISC.EQ.8) GO TO 88
      IF(ISC.EQ.9) GO TO 89
      RETURN

   88 CONTINUE
C   * ------------------ SECTION 8 ---- Convection fluxes
C--- Entered when UCONV =.TRUE.; block-location indices are:
C    LD11 for east and north (accessible at the same time),
C    LD12 for west and south (accessible at the same time),
C    LD2  for high (which becomes low for the next slab).
C    User should provide INDVAR and NDIREC IF's as appropriate.
      IF (ISWEEP.EQ.LSWEEP) THEN

C   Extract convection flux through north and south faces of U1 cell
C   for use in calc effective shear stress
        IF (INDVAR.EQ.U1.AND.NDIREC.EQ.1) THEN
          CALL FN0(LBNAME('CUN'),LD11)
          CALL FN0(LBNAME('CUS'),LD12)
        END IF

C   Convection terms for use in calc force over block
```

```
      IF (INDVAR.EQ.U1.AND.NDIREC.EQ.3) THEN

        L0LD11=L0F(LD11)
        L0LD12=L0F(LD12)
C   Setup block upstrm total mass flow arrays
        IXX=IBU-1
C         write(14,*)'ix is',IXX
        DO IYY=1, BNY
          I=IYY+NY*(IXX-1)
          IW=IYY+NY*(IXX-2)
C           write(14,*)'iy is',IYY
C           write(14,*)'mass bu east is',F(L0LD12+I)
C           write(14,*)'mass bu west is (-)',F(L0LD11+IW)
C           write(14,*)' '
          MBU(IYY)=-F(L0LD11+IW)+F(L0LD12+I)
        END DO
C   Setup block dwnstrm total mass flow arrays
        IXX=IBD+1
C         write(14,*)'ix is',IXX
        DO IYY=1, BNY
          I=IYY+NY*(IXX-1)
          IW=IYY+NY*(IXX-2)
C           write(14,*)'iy is',IYY
C           write(14,*)'mass bd east is',F(L0LD12+I)
C           write(14,*)'mass bd west is (-)',F(L0LD11+IW)
C           write(14,*)' '
          MBD(IYY)=-F(L0LD11+IW)+F(L0LD12+I)
        END DO
      END IF

      END IF

      RETURN
   89 CONTINUE
C   * ------------------- SECTION 9 ---- Diffusion coefficients
C--- Entered when UDIFF =.TRUE.; block-location indices are LAE
C    for east, LAW for west, LAN for north, LAS for
C    south, LD11 for high, and LD11 for low.
C    User should provide INDVAR and NDIREC IF's as above.
C    EARTH will apply the DIFCUT and GP12 modifications after the user
C    has made his settings.
C
      IF(ISWEEP.EQ.LSWEEP) THEN

C  Find diffusion coeff throu' north face of U1 cell for eff. stress
        IF(INDVAR.EQ.U1.AND.NDIREC.EQ.1) CALL FN0(LBNAME('DUN'),LAN)

C  Find diffusion coeff up/ dwn of block for force over block
        IF (INDVAR.EQ.U1.AND.NDIREC.EQ.3) THEN
          L0LAE=L0F(LAE)
C   Setup block upstrm diffusion arrays
          IXX=IBU-2
          DO IYY=1, BNY
            I=IYY+NY*(IXX-1)
            DBU(IYY)=F(L0LAE+I)
          END DO
C   Setup block dwnstrm diffusion arrays
          IXX=IBD
```

```
        DO IYY=1, BNY
          I=IYY+NY*(IXX-1)
          DBD(IYY)=F(L0LAE+I)
        END DO

      END IF

    END IF

    RETURN
C*****************************************************************************
C
C--- GROUP 11. Initialization of variable or porosity fields
C                                        Index VAL
   11 CONTINUE

     IF (NPATCH.EQ.'U1INPROF') THEN
C  define U1 inital field from 1/7th pwr law vel profile
C  (under blocks is set to const value of U1 at BNY+1)
C  NB: This can not be used if top wall is present!!
       DEPTH=RG(3)
       UI=RG(6)
       BNY=IG(4)
       UMAX=8*UI/7
       L0VAL=L0F(VAL)
       L0Y=L0F(YG2D)
       DO 111 IXX=1,NX
       DO 111 IY=1,NY
         I=IY+(IXX-1)*NY
         A=(F(L0Y+I)/DEPTH)**0.1428
         F(L0VAL+I)=A*UMAX
111    CONTINUE
      DO 112 IXX=1,NX
      DO 112 IY=1,BNY
        I=IY+(IXX-1)*NY
        F(L0VAL+I)=F(L0VAL+BNY+1)
112    CONTINUE

     ELSEIF (NPATCH.EQ.'U1KSPROF') THEN
C  define U1 inital field from Nik rough wall vel profile
C  (under blocks is set to const value of U1 at BNY)
C  NB: This can not be used if top wall is present!!
     WRITE(14,*)' Setting initial U1 field using CHAM WF law. '
       DEPTH=RG(3)
       BLKHGT=RG(4)
       BLKLEN=RG(2)
       KS=RG(8)
       DPDX=RG(5)
       PLEN=RG(1)
       BNY=IG(4)

       Y0=(BLKHGT*BLKLEN)/PLEN
       UTAU=SQRT((DPDX*(DEPTH-Y0))/RHO1)
       A=8.2710904
       B=5.75

       L0VAL=L0F(VAL)
       L0Y=L0F(YG2D)
```

```
      DO 113 IXX=1,NX
      DO 113 IY=BNY,NY
        I=IY+(IXX-1)*NY
        F(L0VAL+I)=UTAU*(A+B*LOG((F(L0Y+I)-Y0)/KS))
113   CONTINUE
      DO 114 IXX=1,NX
      DO 114 IY=1,BNY-1
        I=IY+(IXX-1)*NY
        F(L0VAL+I)=F(L0VAL+BNY)
114   CONTINUE

    END IF

    RETURN
C*************************************************************************
C
C--- GROUP 13. Boundary conditions and special sources
C                              Index for Coefficient - CO
C                              Index for Value      - VAL
  13 CONTINUE
    IF(ISC.EQ.12) GO TO 1311
    RETURN

 1311 CONTINUE
C------------------ SECTION 12 ------------------ value = GRND

C  Set dpdx - press grad used for momentum source.
    IF (NPATCH.EQ.'U1MT') THEN
       CALL FN1(VAL,DPDX)
    END IF

    RETURN
C*************************************************************************
C
C--- GROUP 19. Special calls to GROUND from EARTH
C
  19 GO TO (191,192,193,194,195,196,197,198,199,1910,1911),ISC
 191 CONTINUE
C   * ------------------- SECTION 1 ---- Start of time step.
C     Set start time
    CALL MCLOCK(STIME)
C     Initialise values
    PLEN=RG(1)
    BLKLEN=RG(2)
    DEPTH=RG(3)
    BLKHGT=RG(4)
    SNX=IG(1)
    BNX=IG(2)
    MFNF=IG(3)
    BNY=IG(4)
    DPDX=RG(5)
    UI=RG(6)
    RELXDP=RG(7)
    MONTYPE=IG(5)
    SETUI=LG(2)
    NF=IG(6)
    NC=1
    FLOWF=1.0
```

```
      UC=0.0

C     Definition of blocks (x direction)
      IBD=BNX/2
      IBU=IBD+SNX+1

C     Find true depth from coord of y north face
      WRITE(14,*)' '
      WRITE(14,*)'Input depth is ',DEPTH
      DEPTH=F(L0F(YV2D)+NY)
      WRITE(14,*)'True depth is ',DEPTH

C     Print y coords of north cell faces (used for p2m.f)
      CALL PRN('ynf',YV2D)

      RETURN
  192 CONTINUE
C   * ------------------ SECTION 2 ---- Start of sweep.

      RETURN
  193 CONTINUE
C   * ------------------ SECTION 3 ---- Start of iz slab.
      RETURN
  194 CONTINUE
C   * ------------------ SECTION 4 ---- Start of iterations over slab.

      RETURN
 1911 CONTINUE
C   * ------------------ SECTION 11---- After calculation of convection
C                            fluxes for scalars, and of volume
C                            fractions, but before calculation of
C                            scalars or velocities
      RETURN
  199 CONTINUE
C   * ------------------ SECTION 9 ---- Start of solution sequence for
C                                        a variable
      RETURN
 1910 CONTINUE
C   * ------------------ SECTION 10---- Finish of solution sequence for
C                                        a variable
      RETURN
  195 CONTINUE
C   * ------------------ SECTION 5 ---- Finish of iterations over slab.
      RETURN
  196 CONTINUE
C   * ------------------ SECTION 6 ---- Finish of iz slab.
C   Calculate improved pressure grad.
      IF(SETUI.AND.NF.GT.0.AND.ISWEEP.EQ.(NF*NC)+2) THEN
      WRITE(14,*)'Updating pressure grad at sweep = ',ISWEEP
      NC=NC+1
      IXX=NX
      L0AE=L0F(AEAST)
      L0U1=L0F(U1)
      UC=0.0
      AMFET=0.0
      DO 1961 IYY=BNY+1,NY
         I=IYY+NY*(IXX-1)
         UC=UC+(F(L0U1+I)*F(L0AE+I))
```

```
          AMFET=AMFET+F(L0AE+I)
1961    CONTINUE
        UC=UC/AMFET
        FLOWF=UI/UC
C         WRITE(14,*)'UI is ',UI
C         WRITE(14,*)'UC is ',UC
C         WRITE(14,*)'flow factor is ',FLOWF
C       Set relaxation for calc of dpdx
        ERR=FLOWF-1
        ERR=ERR*RELXDP
        FLOWF=ERR+1
C         WRITE(14,*)'flow factor after relax is ',FLOWF
C         WRITE(14,*)'dpdx old is ',DPDX
        DPDX=DPDX*FLOWF
        IF(DPDX.LE.0.001) DPDX=0.001
C         WRITE(14,*)'dpdx new is ',DPDX
C         WRITE(14,*)' '
C     recalc flowf for printing
            FLOWF=UI/UC
      END IF

      RETURN
  197 CONTINUE
C   * ------------------- SECTION 7 ---- Finish of sweep.

C     Convergance monitoring.
      IF (MONTYPE.GT.0) CALL MONITOR(MONTYPE,SETUI,UC,FLOWF)

      RETURN
  198 CONTINUE
C   * ------------------- SECTION 8 ---- Finish of time step.
C
C   Calculate period average effective shear stress.
C   Method 2: Convection/diffusion method.
      CALL EFFSS
C  Calculate period average effective shear stress.
C   Method 3: manual method. (25/11/96)
      CALL MANEFF
C   Finds shear force on walls/blocks
      CALL SHFOR(ASFTW,ASFBW,TSFB)
C   Finds total pressure force/moment over block
      CALL PBLK(TPFB,TPMB,BLKHGT,MBU,MBD,DBU,DBD,DEPTH)
C   prints period average U1 velocity
      CALL PAVEL(DEPTH,TPAU,MU,MUIY)
C   stream function
      CALL STREAM
C   Analsyis and printout
      CALL APRN(TPFB,TPMB,ASFTW,ASFBW,TSFB,
     1   DEPTH,BLKHGT,BLKLEN,UI,STIME,SETUI,TPAU,FLOWF,MU,MUIY)
C   writes contour files (except stream)
      CALL CONTOUR

      RETURN
      END
C*********************************************************************
C*********************************************************************
      SUBROUTINE EFFSS
C Finds period average effective shear stress by convection/diffusion
```

```
C method.

      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C   Version 2.0
C     INCLUDE 'lp2/d_includ/satear'
C     INCLUDE 'lp2/d_includ/grdloc'
C     INCLUDE 'lp2/d_includ/grdear'
C     INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C     My common block for rough duct analysis subroutines
      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C   Subroutine variables:
C     Real Arrays:
C   PASHR(IY)=Final array of period averaged effective shear stress
C     Reals:
C   A,B,B1,B2,B3=Temp coeff used in func's
C   ALPA=Coef used in diff. term. Hybrid scheme=0.5
C     Integers:
C   ISHR=x cell location of period ave. SHR
C   I=location index for F-array
C   II=location index for F-array of p. ave SHR at ISHR
C   LSHR=F-array name for SHR
C   LSCN=F-array name for SCN
C   LSFN=F-array name for SFN
C   LSDN=F-array name for SDN
C   LCUS=F-array name for CUS
C   LCUN=F-array name for CUN
C   LCON=F-array name for CON
C   LDUN=F-array name for DUN
C   LUDIF=F-array name for UDIF
C   L0SHR=location zero point of SHR segment of F-array
C   L0SCN=location zero point of SCN segment of F-array
C   L0SFN=location zero point of SFN segment of F-array
C   L0SDN=location zero point of SDN segment of F-array
C   L0CUS=location zero point of CUS segment of F-array
C   L0CUN=location zero point of CUN segment of F-array
C   L0CON=location zero point of CON segment of F-array
C   L0U1=location zero point of U1 segment of F-array
C   L0AE=location zero point of AEAST segment of F-array
C   L0YNF=location zero point of YV2D (y north face) segment of F-array

      REAL PASHR(600)
      REAL A,B,B1,B2,B3,ALPA
      INTEGER ISHR, I, II
      INTEGER LSHR,LSCN,LSFN,LSDN,LCUS,LCUN,LCON,LDUN,LUDIF
      INTEGER L0SHR,L0SCN,L0SFN,L0SDN,L0CUS,L0CUN,L0CON,L0U1,L0AE

C   Earth arrays (nx*ny), part of F-array:
```

```
C  (see TR99 under U-vel equ. for terms).
C  SHR=Period averaged shear stress, for each cell
C    =(U_N-U_P)[max(0,d_en-alpalm_enl)+max(0,-m_en)]-m_en.U_P
C  consisting of the following sub terms:
C     SDN= "   "   diffusion over north face
C       =(U_N-U_P)[max(0,d_en-alpalm_enl)]
C     SCN=stress due to convection over north face
C       =(U_N-U_P)[max(0,-m_en)]
C     SFN= "   "   additional flow term
C       =U_P . m_en
C  other earth arrays used in calc:
C   CUS=mass flow (conv) of U1 through south face of every cell
C   CUN=mass flow (conv) of U1 through north face of every cell
C     =[max(0,-m_en)]
C   CON=total mass flow (conv flux) throu' n. face of every cell
C     =m_en
C   DUN=diffusion coeff. of U1 through north face of every cell
C     =d_en
C   UDIF=U_north - U_current =(U_N-U_P)= U1(ix,iy+1)-U1(ix,iy)

C    Set up EARTH variables
C       Prefix 'L' means LBNAME
      LSHR=LBNAME('SHR')
      LSCN=LBNAME('SCN')
      LSFN=LBNAME('SFN')
      LSDN=LBNAME('SDN')
      LCUS=LBNAME('CUS')
      LCUN=LBNAME('CUN')
      LCON=LBNAME('CON')
      LDUN=LBNAME('DUN')
      LUDIF=LBNAME('UDIF')
C       For accessing F array directly
      L0SHR=L0F(LSHR)
      L0SCN=L0F(LSCN)
      L0SFN=L0F(LSFN)
      L0SDN=L0F(LSDN)
      L0CUS=L0F(LCUS)
      L0CUN=L0F(LCUN)
      L0CON=L0F(LCON)
      L0U1=L0F(U1)
      L0AE=L0F(AEAST)
      L0YNF=L0F(YV2D)


      WRITE(14,*)'sub effss '

C  For debugging:
C     CALL PRN('cun',LBNAME('CUN'))

C    calc of total convective flux over north face of U1 cell
C    by: CON(ix,iy)= -CUN(ix,iy) + CUS(ix,iy+1)
      DO 10 IXX=1,NX
      DO 10 IY=BNY+1, NY-1
         I=IY+NY*(IXX-1)
         F(L0CON+I)=F(L0CUS+I+1)
10    CONTINUE
      DO 20 IXX=1,NX
      DO 20 IY=BNY+1, NY
```

```
      I=IY+NY*(IXX-1)
      F(L0CON+I)=F(L0CON+I)-F(L0CUN+I)
20    CONTINUE

C     Calculate sub term SDN; diffusion over north face
C       Use SDN=abs(CON) as temp store)
      CALL FN0(LSDN,LCON)
      CALL FN40(LSDN)
C       Fn10 gives SDN = DUN - 0.5(SDN)
      A=0.0
      B=1.0
      ALPA=-0.5
      CALL FN10(LSDN,LDUN,LSDN,A,B,ALPA)
      CALL FN22(LSDN,0)
      CALL FN103(LUDIF,U1,1)
      CALL FN26(LSDN,LUDIF)

C     Calculate sub term SCN; convection over north
C       face of U1 cell; SCN= UDIF*CUN
      A=0.0
      B=1.0
      CALL FN21(LSCN,LCUN,LUDIF,A,B)

C     Calc sub term SFN;additional north convection term =-CON*U1
      A=0
      B=-1
      CALL FN21(LSFN,LCON,U1,A,B)

C     Now have the 3 sub terms SCN, SFN, SDN.
C       The total effective shear stress is the sum of these:
      A=0.0
      B1=1.0
      B2=1.0
      B3=1.0
      CALL FN12(LSHR,LSDN,LSCN,LSFN,A,B1,B2,B3)

C     Open Matlab files
      OPEN (41,FILE='res_shr.m')
      WRITE(41,*)'% MATLAB array in y of effective shear stress'
      WRITE(41,*)'% from NY=1 to NY-1'
      WRITE(41,*)'% (using convection/diffusion method) '
      WRITE(41,*)' '
      WRITE(41,*)'shr=[ '
      OPEN (42,FILE='res_ynf.m')
      WRITE(42,*)'% MATLAB array in y of y north face co-ords'
      WRITE(42,*)'% from NY=1 to NY-1'
      WRITE(42,*)' '
      WRITE(42,*)'ynf=[ '

C     Find period average, write to array PASHR and write
C     matlab files res_shr.m and res_ynf.m
      DO 40 IY=1,NY-1
        DO 50 IXX=1,NX
          I=IY+NY*(IXX-1)
          PASHR(IY)=PASHR(IY)+F(L0SHR+I)
50        CONTINUE
        PASHR(IY)=PASHR(IY)/PLEN
        WRITE(41,*) PASHR(IY)
```

```
      WRITE(42,*) F(L0YNF+I)
40   CONTINUE
     WRITE(41,*)'];'
     WRITE(41,*)' '
     WRITE(42,*)'];'
     WRITE(42,*)' '

     RETURN
     END
C*********************************************************************
     SUBROUTINE MANEFF
C Finds period average effective shear stress by manual method
C
C    tau_yx = -rho * UV + Gamma(dU/dy + dV/dx)          (1)
C
C  and then
C
C    tau_eff = 1/Plen SUM ( tau_yx * north face cell area )
C
C  Shear stresses are taken over the north face of a scaler cell
C  from BNY+1 to NY-1
C
C  Note the term dV/dx in eq (1) should be omitted
C
C  Period average shear stress is given in matlab file 'maneff.m' .
C
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C   Version 2.0
C     INCLUDE 'lp2/d_includ/satear'
C     INCLUDE 'lp2/d_includ/grdloc'
C     INCLUDE 'lp2/d_includ/grdear'
C     INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C     My common block for rough duct analysis subroutines
      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C   Subroutine variables:
C     Real Arrays
C     EFFS = array in IY of effective shear stress

      REAL EFFS(600)

C     Reals:
C     SUMTAU = sum of tau*area
C     UCP    = U1 vel at centre of P cell
C     UCN    = U1 vel at centre of N cell
C     UNF    = U1 vel at centre of north face of P cell
C     ENUTNF = ENUT at centre of north face of P cell
C     T1     = 1st term in equation
```

```
C    T2    = 2nd term in equation
C    GAM   = rho*(enul+enut)
C    ADUDY  = dU/dy at north face of P cell
C    ADVDX  = dV/dx at north face of P cell
C    DELX  = dx in ADVDX
C
      REAL SUMTAU,UCP,UCN,UNF,ENUTNF,T1,T2,GAM,ADUDY,ADVDX,DELX
C
C    Integers:
C    I   = location index for F-array of current (P) cell
C    IN  = " " of North cell
C    IW  = " " of West cell
C    INW  = " " of North West cell
C    IE  = " " of East cell
C    IW  = " " of West cell
C    IW  = " " of West cell
C
C    Set up EARTH variables
      L0U1=L0F(U1)
      L0V1=L0F(V1)
      L0AN=L0F(ANORTH)
      L0YC=L0F(YG2D)
      L0XC=L0F(XG2D)
      L0YNF=L0F(YV2D)
      LENUT=LBNAME('ENUT')
      L0ENUT=L0F(LENUT)

      WRITE(14,*)'sub maneff '
      OPEN (49,FILE='res_mshr.m')
      WRITE(49,*)'% MATLAB script of manual eff shear stress'
      WRITE(49,*)'% from BNY+1 to NY-1 '
      WRITE(49,*)' '
      WRITE(49,*)'mshr=[ '

      DO 10 IY=BNY+1, NY-1

       SUMTAU=0
       DO 20 IXX=1,NX

        I=IY+NY*(IXX-1)
        IN=(IY+1)+NY*(IXX-1)
        IF (IXX.EQ.1) THEN
C         set IW as slab NX
         IW=IY+NY*(NX-1)
         INW=(IY+1)+NY*(NX-1)
        ELSE
         IW=IY+NY*(IXX-2)
         INW=(IY+1)+NY*(IXX-2)
        END IF
        IF (IXX.EQ.NX) THEN
C         set IE as slab IX=1
         IE=IY
        ELSE
         IE=IY+NY*(IXX)
        END IF

C       Interpolate to find values of variables on north face of
C       scaler cell.
```

```
      UCP=0.5*(F(L0U1+I)+F(L0U1+IW))
      UCN=0.5*(F(L0U1+IN)+F(L0U1+INW))

      UNF=(F(L0YC+IN)-F(L0YNF+I))*(UCN-UCP)
      UNF=UCN-(UNF/(F(L0YC+IN)-F(L0YC+I)))
      ENUTNF=(F(L0YC+IN)-F(L0YNF+I))*(F(L0ENUT+IN)-F(L0ENUT+I))
      ENUTNF=F(L0ENUT+IN)-(ENUTNF/(F(L0YC+IN)-F(L0YC+I)))

      T1=-1*RHO1*UNF*F(L0V1+I)
      GAM=RHO1*(ENUL+ENUTNF)

      ADUDY=(UCN-UCP)/(F(L0YC+IN)-F(L0YC+I))
      IF (IXX.EQ.1) THEN
        DELX=PLEN-F(L0XC+IW)+F(L0XC+IE)
C         WRITE(14,*)'delx1= ',delx
      ELSE IF (IXX.EQ.NX) THEN
        DELX=PLEN-F(L0XC+IW)+F(L0XC+IE)
C         WRITE(14,*)'delxnx= ',delx
      ELSE
        DELX=(F(L0XC+IE)-F(L0XC+IW))
      END IF
      ADVDX=(F(L0V1+IE)-F(L0V1+IW))/DELX

C       comment this term to use ADVDX term
      ADVDX=0

      T2=GAM*(ADUDY+ADVDX)
      TAU=(T1+T2)*F(L0AN+I)
      SUMTAU=SUMTAU+TAU
C     The following two lines print out the local Peclet number
C     to the result file.
C         PE=RHO1*F(L0V1+I)*(F(L0YC+IN)-F(L0YC+I))/GAM
C         WRITE(14,*)'IY is ',IY,' IX is ',IXX,' Pe is ',PE

20     CONTINUE
      EFFS(IY)=SUMTAU/PLEN
      WRITE(49,*) EFFS(IY)

10   CONTINUE
     WRITE(49,*)']; '
     CLOSE(49)

     RETURN
     END

C*********************************************************************

     SUBROUTINE SHFOR(ASFTW,ASFBW,TSFB)
C  Finds shear force on walls/blocks
     INCLUDE 'lp21/d_includ/satear'
     INCLUDE 'lp21/d_includ/grdloc'
     INCLUDE 'lp21/d_includ/grdear'
     INCLUDE 'lp21/d_includ/grdbfc'
C  Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'
```

```
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C     My common block for rough duct analysis subroutines
      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C     Subroutine variables
C     Real Arrays in IX (check dimensions!!!)
C     ****************************************
C ASFTW=(sub ground, APRN) array shear force top wall
C ASFBW=(sub ground, APRN)array shear force bottom wall
      REAL ASFTW(600),ASFBW(600)

C     Reals:
C TSFB=total shear force block
C U1AVE=U1 vel average between U_p and U_e
C     Logicals
C TWALL=.TRUE. if top wall exits, else a plane of sym.
C     Integers
C SIGN=sign of U1AVE and therefore of shear stress
C I=integer to give F-array location of current cell
C IW=integer to give F-array location of previous (west) cell
C ID=integer to give F-array location of patch wise varables
C IYF=first y cell of patch
C IYL=last y cell of patch
C IXF=first x cell of patch
C IXL=last x cell of patch
C L0AN=location zero point of area north segment of F-array
C L0U1=location zero point of U1 segment of F-array
C L0SB=location zero point of `shear on bed` segment of F-array
C L0SBD=loc. zero pt. of `shear on dwnstm 1/2 of block` segment of F-array
C L0SBU=loc. zero pt. of `shear on upstm 1/2 of block` segment of F-array

      REAL TSFB, U1AVE
      INTEGER SIGN,I,IW,ID,IYF,IYL,IXF,IXL,L0AN,L0U1,L0SB,L0SBD,L0SBU
      LOGICAL TWALL
      SIGN=1
      TWALL=LG(1)
C     Location of north cell areas
      L0AN=L0F(ANORTH)
      L0U1=L0F(U1)
      L0XC=L0F(XG2D)
      IF (TWALL) THEN
       L0ST=L0PVAR(PVSTRS,IPNAME('WALLSTOP'),0)
      END IF
      L0SB=L0PVAR(PVSTRS,IPNAME('WALLSBED'),0)
      L0SBD=L0PVAR(PVSTRS,IPNAME('BLOCD-SW'),0)
      L0SBU=L0PVAR(PVSTRS,IPNAME('BLOCU-SW'),0)

      WRITE(14,*)' '
      WRITE(14,*)'sub SHFOR '

C     Write out to MATLAB file (open file)
      OPEN (51,FILE='res_tau.m')
```

```
      WRITE(51,*)'% MATLAB script of wall shear stress results.'
      WRITE(51,*)' '


C     Array shear force top wall
      IY=NY
      IYF=NY
      IYL=NY
      IXF=1
      IXL=NX
      IXX=1
C     * If no top wall then plane of sym. ie free surface.
      IF (.NOT.TWALL) THEN
       DO 5 IXX=1,NX
            ASFTW(IXX)=0.0
5      CONTINUE
      ELSE
C       Write out to MATLAB file (header)
       WRITE(51,*)'twtau=[ '

       DO 10 IXX=1,NX
C       Find sign of shear force, acting on the fluid
        I=IY+NY*(IXX-1)
        IW=IY+NY*(IXX-2)
        U1AVE=(F(L0U1+I)+F(L0U1+IW))/2
        IF (U1AVE.GT.0.0) THEN
          SIGN=-1
        ELSE
          SIGN=1
        END IF
C       Find shear force
         ID=(IY-IYF+1)+(IYL-IYF+1)*(IXX-IXF)
         ASFTW(IXX)=F(L0ST+ID)*SIGN*F(L0AN+I)*RHO1
C          WRITE(14,*)' '
C          WRITE(14,*)'t/r topw is',(F(L0ST+ID)*SIGN)
C          WRITE(14,*)'cell area is ',F(L0AN+I)
C          WRITE(14,*)' '
C          Write out to MATLAB file (shear stress*sign)
         WRITE(51,*) (F(L0ST+ID)*SIGN*RHO1)
10     CONTINUE
C      Write out to MATLAB file (footer)
      WRITE(51,*)'];'
      WRITE(51,*)' '

      END IF

C     Array shear force bottom wall
C     NB IXF,IXL must apply to PATCH WALLSBED, but loop is
C     over space between blocks only.
      IY=1
      IYF=1
      IYL=1
      IXF=1
      IXL=NX
C     Write out to MATLAB file (header)
      WRITE(51,*)'      xcc       bed shear stress '
      WRITE(51,*)'bwtau=[ '
```

```
      DO 20 IXX=IBD+1,IBU-1
C       Find sign of shear force, acting on the fluid
        I=IY+NY*(IXX-1)
        IW=IY+NY*(IXX-2)
        IF (IXX.EQ.IBD+1) THEN
          U1AVE=F(L0U1+I)/2
        ELSEIF (IXX.EQ.IBU-1) THEN
          U1AVE=F(L0U1+IW)/2
        ELSE
          U1AVE=(F(L0U1+I)+F(L0U1+IW))/2
        END IF
        IF (U1AVE.GT.0.0) THEN
          SIGN=-1
        ELSE
          SIGN=1
        END IF
C       Find shear force
        ID=(IY-IYF+1)+(IYL-IYF+1)*(IXX-IXF)
        ASFBW(IXX)=F(L0SB+ID)*SIGN*F(L0AN+I)*RHO1
        WRITE(14,*)' '
        WRITE(14,*)'ix is ',IXX
        WRITE(14,*)'t/r bed1 is',(F(L0SB+ID)*SIGN)
        WRITE(14,*)'cell area is ',F(L0AN+I)
        WRITE(14,*)'sign is ',SIGN
        WRITE(14,*)'U1 ave is ',U1AVE
        WRITE(14,*)' '
C         Write out to MATLAB file (shear stress*sign)
        WRITE(51,*) F(L0XC+I), (F(L0SB+ID)*SIGN*RHO1)
20    CONTINUE
C     Write out to MATLAB file (footer)
      WRITE(51,*)'];'
      WRITE(51,*)' '

C     Total shear force block
C     Write out to MATLAB file (header)
      WRITE(51,*)'     xcc       block shear stress '
      WRITE(51,*)'blktau=[ '
      IY=BNY+1
      IYF=BNY+1
      IYL=BNY+1
      TSFB=0.0
C     Upstream 1/2 of block (NB: dwnstream end of domain)
      IXF=IBU
      IXL=NX
      DO 40 IXX=IXF,IXL
C       Find sign of shear force, acting on the fluid
        I=IY+NY*(IXX-1)
        IW=IY+NY*(IXX-2)
        U1AVE=(F(L0U1+I)+F(L0U1+IW))/2
        IF (U1AVE.GT.0.0) THEN
          SIGN=-1
        ELSE
          SIGN=1
        END IF
C       Find shear force
        ID=(IY-IYF+1)+(IYL-IYF+1)*(IXX-IXF)
        TSFB=TSFB+(F(L0SBU+ID)*SIGN*F(L0AN+I))
C       Write out to MATLAB file (shear stress*sign)
```

```
        WRITE(51,*) F(L0XC+I),(F(L0SBU+ID)*SIGN*RHO1)
40    CONTINUE
C     Dwnstream 1/2 of block (NB: upstream end of domain)
      IXF=1
      IXL=IBD
      DO 45 IXX=IXF,IXL
C       Find sign of shear force, acting on the fluid
        I=IY+NY*(IXX-1)
        IF (IXX.EQ.1) THEN
          IW=IY+NY*(NX-1)
        ELSE
          IW=IY+NY*(IXX-2)
        END IF
        U1AVE=(F(L0U1+I)+F(L0U1+IW))/2
        IF (U1AVE.GT.0.0) THEN
          SIGN=-1
        ELSE
          SIGN=1
        END IF
C       Find shear force
        ID=(IY-IYF+1)+(IYL-IYF+1)*(IXX-IXF)
        TSFB=TSFB+(F(L0SBD+ID)*SIGN*F(L0AN+I))
C       Write out to MATLAB file (shear stress*sign)
        WRITE(51,*) F(L0XC+I),(F(L0SBD+ID)*SIGN*RHO1)
45    CONTINUE

      TSFB=TSFB*RHO1

C       Write out to MATLAB file (footer and close)
      WRITE(51,*)'];'
      WRITE(51,*)' '
      CLOSE (51)

       RETURN
      END

C*********************************************************************
      SUBROUTINE PBLK(TPFB,TPMB,BLKHGT,MBU,MBD,DBU,DBD,DEPTH)
C     Finds total pressure force/moment over block (ie sigma_x)
C   Force is taken to act on the fluid in the +ve X direction.
C     For method see notes (in printout) dated 2/11/95, A. Senior.
C     (Corrected 22/11/95)
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C     Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C     My common block for rough duct analysis subroutines
      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
```

```
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C   Subroutine variables
C   Real Arrays in BNY (check dimensions!!!)
C   ****************************************
C MBU=(sub ground) array of mass flow across face P upstream of block
C MBD=(sub ground) array of mass flow across face P dwnstream of block
C DBU=(sub ground) array of diff. coeff across face P upstream of block
C DBD=(sub ground) array of diff. coeff across face P dwnstream of block
      REAL MBU(600), MBD(600), DBU(600), DBD(600)

C TPFU=total pressure force upstream (of block)
C TPFD=total pressure force downstream (of block)
C TPFB=total pressure force over block
C TPMU=total pressure moment upstream (of block) from IY=NY
C TPMD=total pressure moment downstream (of block) from IY=NY
C TPMB=total pressure moment over block from IY=NY
C ALPA=coeff of hybrid scheme = 0.5
C AE=convection/diff coeff for east face of U1 cell
C DEPTH=duct depth (=flow area for 2D)
C UDIF= U1_w - U1_e
C JP=total flux across face P / area of cell east face
C TJPU=total force due to JP on upstrm side of the block
C TJPD=total force due to JP on dwnstrm side of the block
C PFU=pressure force upstream (of block)
C PFD=pressure force downstream (of block)
C I=integer to give F-array location of current cell
C IE=integer to give F-array location of next (east) cell
C L0YC=location zero point of y cell center segment of F-array
C L0AE=location zero point of area east segment of F-array
C L0P1=location zero point of P1 segment of F-array
C L0U1=location zero point of U1 segment of F-array

      REAL TPFU,TPMU,TPFD,TPMD,TPFB, ALPA, AE, UDIF,JP, PFU, PFD
      REAL TJPU, TJPD
      INTEGER I, IE ,L0P1, L0U1, L0AE, L0YC

C    Set up EARTH variables
      L0P1=L0F(P1)
      L0U1=L0F(U1)
      L0AE=L0F(AEAST)
      L0YC=L0F(YG2D)

      WRITE(14,*)' '
      WRITE(14,*)'sub PBLK '
      OPEN (60,FILE='res_sig.m')
      WRITE(60,*)'% MATLAB file of sigma_x on upstream face of block '
      WRITE(60,*)'%                    -sigbup '
      WRITE(60,*)'% and sigma_x on downstream stream face of block '
      WRITE(60,*)'%                    -sigbdwn '
      WRITE(60,*)'% '
      WRITE(60,*)'%   ycc      sigma_x       P1       JP
     1 area_east   '
      WRITE(60,*)' '
61    FORMAT (5(E11.5,2X))

C    Find upstream pressure forces
```

```
C    First find Jp, based on flow throu' east face
C    of cell Uw, ie IBU-2
     WRITE(60,*)'sigbup=[ '
     TPFU=0.0
     TPMU=0.0
     TJPU=0.0
     ALPA=0.5
     IXX=IBU-2
C      WRITE(14,*)'ix is ',IXX
     DO 10 IY=1,BNY
C      WRITE(14,*)'iy is ',IY
       I=IY+NY*(IXX-1)
       IE=IY+NY*(IXX)
       AE=AMAX1(0.0,(DBU(IY)-( ALPA*ABS(MBU(IY)))))
C      WRITE(14,*)'ae_1 is ',AE
       AE=AE+AMAX1(0.0,-MBU(IY))
C      WRITE(14,*)'ae_2 is ',AE
       UDIF=F(L0U1+I)
C      WRITE(14,*)'u1 is ',UDIF
       JP=((AE*UDIF)+(MBU(IY)*F(L0U1+I)))
       PFU=(F(L0P1+IE))*F(L0AE+I)
       TJPU=TJPU+JP
       TPFU=TPFU+PFU
       TPMU=TPMU+((PFU+JP)*(DEPTH-F(L0YC+IE)))
C      WRITE(14,*)'p1 is ',F(L0P1+IE)
C      WRITE(14,*)'pfu is ',PFU
C    write to MATLAB file...
     WRITE(60,61) F(L0YC+I),(JP+PFU)/F(L0AE+I),F(L0P1+IE),JP,F(L0AE+I)
10   CONTINUE
C      WRITE(14,*)'tpfu is ',TPFU
C      WRITE(14,*)'tjpu is ',TJPU
C      WRITE(14,*)' '
     TPFU=-1*(TPFU+TJPU)
     WRITE(60,*)'] '
     WRITE(60,*)' '

C    Find dwnstream pressure forces
C    First find Jp, based on flow throu' east face
C    of i/2 wall cell, ie IBD
     WRITE(60,*)'sigbdwn=[ '
     TPFD=0.0
     TPMD=0.0
     TJPD=0.0
     ALPA=0.5
     IXX=IBD
C      WRITE(14,*)'ix is ',IXX
     DO 20 IY=1,BNY
C      WRITE(14,*)'iy is ',IY
       I=IY+NY*(IXX-1)
       IE=IY+NY*(IXX)
       AE=AMAX1(0.0,(DBD(IY)-( ALPA*ABS(MBD(IY)))))
C      WRITE(14,*)'ae_1 is ',AE
       AE=AE+AMAX1(0.0,-MBD(IY))
C      WRITE(14,*)'ae_2 is ',AE
       UDIF=-F(L0U1+IE)
C      WRITE(14,*)'u1 is ',UDIF
       JP=AE*UDIF
       PFD=(F(L0P1+IE))*F(L0AE+IE)
```

```
      TPFD=TPFD+PFD
      TJPD=TJPD+JP
      TPMD=TPMD+((PFD+JP)*(DEPTH-F(L0YC+IE)))
C     WRITE(14,*)'p1 is ',F(L0P1+IE)
C     WRITE(14,*)'pfd is ',PFD
C    write to MATLAB file...
      WRITE(60,61) F(L0YC+I),(JP+PFD)/F(L0AE+I),F(L0P1+IE),JP,F(L0AE+I)
20    CONTINUE
C     WRITE(14,*)'tpfd is ',TPFD
C     WRITE(14,*)'tjpd is ',TJPD
      TPFD=TPFD+TJPD
      TPFB=TPFD+TPFU
      TPMB=-1*(TPMU-TPMD)
C     WRITE(14,*)'tpfd2 is ',TPFD
C     WRITE(14,*)'tpfb  is ',TPFB
C     WRITE(14,*)'tpmu is ',TPMU
C     WRITE(14,*)'tpmd is ',TPMD
C     WRITE(14,*)'tpmb is ',TPMB
C     WRITE(14,*)' '
      WRITE(60,*)'] '
      WRITE(60,*)' '
      CLOSE(60)

      RETURN
      END

C*************************************************************************
      SUBROUTINE PAVEL(DEPTH,TPAU,MU,MUIY)
C Finds period average U1 vel. and writes results to matlab
C file pavel.m. Also intergrates to find vol. flow rate.

      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C  Version 2.0
C     INCLUDE 'lp2/d_includ/satear'
C     INCLUDE 'lp2/d_includ/grdloc'
C     INCLUDE 'lp2/d_includ/grdear'
C     INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C     My common block for rough duct analysis subroutines
      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C     Subroutine variables:
C PAU(IY)=Real array in y of period averaged U1 vel
C TPAU=Real, average of PAU(IY)
C MU=real, max vel
C MUIY=int, IY cell of MU
C UCUR=real, u current in calc UM

      REAL PAU(600)
```

```
      REAL TPAU,TAN, MU, UCUR
      INTEGER MUIY

      L0AN=L0F(ANORTH)
      L0AE=L0F(AEAST)
      L0U1=L0F(U1)
      L0YCC=L0F(YG2D)

C  Calc period average vel
      DO IY=1,NY
        TAN=0.0
       DO IXX=1,NX
         I=IY+NY*(IXX-1)
         IE=IY+NY*(IXX)
         PAU(IY)=PAU(IY)+( F(L0U1+I)* (F(L0AN+I)+F(L0AN+IE))/2 )
         TAN=TAN+((F(L0AN+I)+F(L0AN+IE))/2)
       END DO

       PAU(IY)=PAU(IY)/TAN
C        write(14,*)'tan is ',TAN
       TPAU=TPAU+(PAU(IY)*F(L0AE+I))
      END DO
      TPAU=TPAU/DEPTH

C  Write out to MATLAB file
      OPEN (50,FILE='res_pau.m')
      WRITE(50,*)'% MATLAB script to plot U1 period average.'
      WRITE(50,*)' '
      WRITE(50,*)'pau=[ '
      DO IY=1,NY
        WRITE(50,*) PAU(IY)
      END DO
      WRITE(50,*)'];'
      WRITE(50,*)' '
      CLOSE (50)

C  Write out y cell centre coords to MATLAB file ycc.m
C  NB: the program p2m.f also creates an identical file.
      OPEN (52,FILE='ycc.m')
      WRITE(52,*)'% MATLAB script of y cell centre coords.'
      WRITE(52,*)' '
      WRITE(52,*)'ycc=[ '
      IXX=1
      DO IY=1,NY
        I=IY+NY*(IXX-1)
        WRITE(52,*) F(L0YCC+I)
      END DO
      WRITE(52,*)'];'
      WRITE(52,*)' '
      CLOSE (52)

C  calc max vel
      MU=-10E10
      DO I=1,NY
       IF (PAU(I).GT.MU) THEN
         MU=PAU(I)
         MUIY=I
       END IF
```

```
      END DO

      RETURN
      END

C*********************************************************************

      SUBROUTINE APRN(TPFB,TPMB,ASFTW,ASFBW,TSFB,
     1   DEPTH,BLKHGT,BLKLEN,UI,STIME,SETUI,TPAU,FLOWF,MU,MUIY)

C  Analsyis and printout
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C  Version 2.0
C     INCLUDE 'lp2/d_includ/satear'
C     INCLUDE 'lp2/d_includ/grdloc'
C     INCLUDE 'lp2/d_includ/grdear'
C     INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C  My common block for rough duct analysis subroutines
      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C  Subroutine variables
C    brought in:
C  Real Arrays in IX (check dimensions!!!)
C  ****************************************
C ASFTW=(sub ground, SHFOR) array shear force top wall
C ASFBW=(sub ground, SHFOR)array shear force bottom wall
      REAL ASFTW(600),ASFBW(600)

C  Reals
C TPFB=total pressure force over block
C TPMB=total pressure moment over block from IY= north NY
C TSFB=total shear force block
C DEPTH=duct height (=flow area for 2D)
C BLKHGT=block height
C BLKLEN=block length
C UI=U1 velocity (over top of blk) inputed from Q1 file (desired vel)
C MU=max period ave vel.
C TPAU=total, ie averaged, period averaged U1 vel
C FLOWF=flow factor used to calc dpdx from UI
C  Integers (array)
C STIME= stime time array
C  Integers
C MUIY=IY cell of MU
C  Logicals
C SETUI=.TRUE. if average U1 is set in Q1

      REAL TPFB,TPMU,TPMD,TSFB,DEPTH,BLKHGT,BLKLEN, MU
```

```
      REAL UI,TPAU,FLOWF
      INTEGER STIME(20), MUIY
      LOGICAL SETUI

C     internal only:
C     Reals
C STWP=(total) shear (force) top wall (over the) period
C SBW=(total) shear (force) bottom wall
C TRFB=total resistive force bed
C TRMB=total resistive moment bed (taken from IY=NY)
C DP=pressure drop over period due to body force
C DF=driving force over period (-block body force)
C EBSSP1=equivalent bed shear stress position
C EBSS1=equivalent bed shear stress
C A=quadratic coeff (e^2) used in EBSS
C B=quadratic coeff (e^1) used in EBSS
C C=quadratic coeff (e^0) used in EBSS
C UC=calculated U1 average velocity
C MFAET=main flow area east total
C US=calc of bed vel. stationary point (= -ve if adj. vel change sign)
C PEDF=% error in boundary forces wrt driving force
C PEU= % error in average velocity wrt UI

C   integers
C I=integer to give F-array location of current cell
C IW=integer to give F-array location of west cell
C NS=index for array CUS (vel bed stationary points)
C CUS(20)=array, Cell numbers where bed U vel is Stationary.
C STIME(3)=start date
C STIME(4)=start hours
C STIME(5)=start mins
C FTIME(3)=finish date
C FTIME(4)=finish hours
C FTIME(5)=finish mins
C TH=time, hours
C TM=time, minutes
C TIME=total time in minues
C L0U1=location zero point of U1 segment of F-array
C L0AE=location zero point of area east segment of F-array
C L0YC=location zero point of y cell centers segment of F-array

C   charaters
C S/F1-5 are all start/finish time charaters
C ID= case identification
C NUMBER=case number

C   Logicals
C TWALL=.TRUE. if top wall exits, else a plane of sym.


      REAL STWP,SBW
      REAL TRFB, TRMB, DP, DPB, DPC, DF, EBSSP1, EBSS1
      REAL A, B, C, EBSSP2, EBSS2
      REAL UCUR, ARWU, RWA, UC, MFAET, US, PEDF, PEU
      CHARACTER*4 ID, NUMBER, ST1, FT1
      CHARACTER*2 ST4, ST5, ST3, ST2, FT4, FT5, FT3, FT2
      INTEGER FTIME(20), CUS (20)
      INTEGER TH, TM, TIME, L0U1,L0AE,L0YC,I, NS
```

```
      LOGICAL TWALL


C     Earth location variables
      L0U1=L0F(U1)
      L0AE=L0F(AEAST)
      L0YC=L0F(YG2D)
      L0XC=L0F(XG2D)

      ID=CG(1)
      NUMBER=CG(2)
      TWALL=LG(1)

C     Calc times
C **NB: If running on a CRAY, comment out this section, and the
C ** relevent section in the Print header.
      CALL MCLOCK(FTIME)
      WRITE(ST4,'(I2.2)') STIME(4)
      WRITE(ST5,'(I2.2)') STIME(5)
      WRITE(ST3,'(I2.2)') STIME(3)
      WRITE(ST2,'(I2.2)') STIME(2)
      WRITE(FT4,'(I2.2)') FTIME(4)
      WRITE(FT5,'(I2.2)') FTIME(5)
      WRITE(FT3,'(I2.2)') FTIME(3)
      WRITE(FT2,'(I2.2)') FTIME(2)
      WRITE(ST1,'(I4.4)') STIME(1)
      WRITE(FT1,'(I4.4)') FTIME(1)
      TIME=(FTIME(3)-STIME(3))*24*60
      TIME=TIME+((FTIME(4)-STIME(4))*60)+ (FTIME(5)-STIME(5))
      TH=TIME/60
      TM=MOD(TIME,60)

      WRITE(14,*)' '
      WRITE(14,*)'sub APRN '
      WRITE(14,*)' '
      WRITE(14,*)'See file "result2"'
      OPEN(4,FILE='result2')

C Print header
      WRITE(4,*)' '
      WRITE(4,*)' '
      WRITE(4,*)'*********************************************************** '
      WRITE(4,*)'*          A. Senior. RMCS.                                *'
      WRITE(4,*)'* File = "result2"                                        *'
      WRITE(4,*)'* Gives GROUND results of xcyclic rough ducts             *'
      WRITE(4,*)'* for single period domains.                              *'
      WRITE(4,*)'*                                                         *'
      WRITE(4,*)'* GROUND update:  31/3/97                                 *'
      WRITE(4,*)'* GROUND version: s12                                     *'
      WRITE(4,*)'* All units are standard SI unless stated in Q1.          *'
      WRITE(4,*)'***********************************************************'
      WRITE(4,*)'*                                                         *'
      WRITE(4,*)'* Case ID name: ',ID,'                                    *'
      WRITE(4,*)'* Case number:  ',NUMBER,
     1                '                                    *'
      WRITE(4,*)'*                                                         *'
      WRITE(4,*)'* Started at : ',ST4,':',ST5,' on ',
     1           ST3,'/',ST2,'/',ST1,'                    *'
```

```
      WRITE(4,*)'* Finished at: ',FT4,':',FT5,' on ',
     1                 FT3,'/',FT2,'/',FT1,'            *'
      WRITE(4,*)'* Total run time is: ',TH,'hrs',TM,
     1                          'mins          *'
      IF (FT2.NE.ST2.OR.FT1.NE.ST1) THEN
        WRITE(4,*)'***** CHECK MONTH AND DAYS******'
      END IF

C Print inputs
      WRITE(4,*)'*                                                *'
      WRITE(4,*)'**************************************************'
      WRITE(4,*)' '
      WRITE(4,*)'INPUTS: '
      WRITE(4,*)'======= '
      WRITE(4,*)' '
      WRITE(4,*)'Domain geometry: '
      WRITE(4,*)'    Duct depth (from north face coord) is',DEPTH
      WRITE(4,*)'    Period length is                   ',PLEN
      WRITE(4,*)'    Block height is                    ',BLKHGT
      WRITE(4,*)'    Block length is                    ',BLKLEN
      WRITE(4,*)' ..... X direction cells:'
      WRITE(4,*)'    Downstream cell of block is      ',IBD
      WRITE(4,*)'    Upstream cell of block is        ',IBU
      WRITE(4,*)'    Total number of X cells is     ',NX
      WRITE(4,*)' ..... y direction cells:'
      WRITE(4,*)'    Block Y cells is           ',BNY
      WRITE(4,*)'    Total number of Y cells is  ',NY
      WRITE(4,*)' '
      WRITE(4,*)'Fluid properties: '
      WRITE(4,*)'    Density (rho) is             ',RHO1
      WRITE(4,*)'    Kinematic viscosity (ENUL) is ',ENUL
      WRITE(4,*)' '
      IF(SETUI) THEN
        WRITE(4,*)'Input average velocity,     ',UI
          WRITE(4,*)'over top of blocks:'
      ELSE
        WRITE(4,*)'Input pressure gradient: ',DPDX
      END IF
      WRITE(4,*)' '
      WRITE(4,*)'Current run of ',ISWEEP,' sweeps.'

C     Calculate results

C 1a] find average vel from above block:
      IXX=1
C     find average vel
      UC=0.0
      MFAET=0.0
      DO IY=BNY+1, NY
        I=IY+NY*(IXX-1)
        UC=UC+(F(L0U1+I)*F(L0AE+I))
        MFAET=MFAET+F(L0AE+I)
      END DO
      UC=UC/MFAET

C 1b] Find points of zero U1 vel on the rough bed
      IY=1
      NS=1
```

```
      DO IXX=IBD+1,IBU-1
        I=IY+NY*(IXX-1)
        IW=IY+NY*(IXX)
        US=F(L0U1+I)*F(L0U1+IW)
        IF(US.LE.0.0) THEN
          CUS(NS)=IXX
          NS=NS+1
        END IF
      END DO

C 3] Force balance over period
C Body force pressure drop over whole period domain:
      DP=DPDX*PLEN
C Driving force
      DF=(DP*DEPTH)-(DPDX*BLKLEN*BLKHGT)

C find shear (force) on walls (over) period etc.
      STWP=0
      SBW=0
      DO 30 IXX=1,NX
           STWP=STWP+ASFTW(IXX)
30    CONTINUE
      DO 40 IXX=IBD+1,IBU-1
           SBW=SBW+ASFBW(IXX)
40    CONTINUE
C  find total resistive force of bed
      TRFB=TPFB+SBW+TSFB

C 2] Equivalent bed shear stress, body force method, 20/12/95
C    Finds 'e' and equivalent bed shear stress using con. ang. momt.
C find total resistive moment of the bed
      TRMB=TPMB+(TSFB*(DEPTH-BLKHGT))+(DEPTH*SBW)
C multipty TRMB, STWP by -1 to agree with convection of notes
C (ie. modulas only. Sign is incorperated in following theory)
      TRMB=-1*TRMB
      STWP=-1*STWP
C find quadratic coeffs
      A=-0.5*PLEN*DPDX
      B=DEPTH*PLEN*DPDX-STWP
      C=DPDX*(BLKHGT*BLKLEN*(DEPTH-0.5*BLKHGT)-PLEN*DEPTH**2)+TRMB+
     1      DEPTH*STWP

C find position (displacement 'e' from bed)
      EBSST=B**2-4*A*C
      IF (EBSST.LE.0) THEN
        WRITE(4,*) 'ERROR in effective bed shear stress;'
           WRITE(4,*) 'negative square root. See APRN item 2].'
      ELSE
        EBSSP1=(-B+SQRT(B**2-4*A*C))/(2*A)
C    .... force
        EBSS1=DPDX*PLEN*(DEPTH-EBSSP1)-STWP
C    .... and stress
        EBSS1=EBSS1/PLEN
C 2nd root
        EBSSP2=(-B-SQRT(B**2-4*A*C))/(2*A)
        EBSS2=DPDX*PLEN*(DEPTH-EBSSP2)-STWP
        EBSS2=EBSS2/PLEN
      END IF
```

```
C Return -ve sign to STWP
      STWP=-1*STWP



C Print out results
      WRITE(4,*)' '
      WRITE(4,*)'********************************************** '
      WRITE(4,*)' '
      WRITE(4,*)'RESULTS'
      WRITE(4,*)'======='
      WRITE(4,*)' '
     WRITE(4,*)'1) Velocities. '
     WRITE(4,*)'-------------- '
     WRITE(4,*)' '
     IF (SETUI.AND.NF.GT.0) THEN
       WRITE(4,*)'( Flow factor to obtain desired velocity is ',
    1     FLOWF,' )'
       WRITE(4,*)' '
     END IF
     WRITE(4,*)'Average U1 over middle of block is ',UC
     WRITE(4,*)'     corresponding flow rate is ',UC*(DEPTH-BLKHGT)
     WRITE(4,*)'Average U1 from period averaged profile is ',TPAU
     WRITE(4,*)'                corresponding flow rate is',TPAU*DEPTH
     WRITE(4,*)' '
     WRITE(4,*)'From period averaged profile (see file res_pau.m): '
     WRITE(4,*)'Maxium U1 velocity is      ',MU
     WRITE(4,*)'          IY cell is      ',MUIY
     WRITE(4,*)'          Y position is   ',F(L0YC+MUIY)
     WRITE(4,*)' '
     DO NS=1,20
      IF (CUS(NS).GT.0) THEN
       WRITE(4,*) 'Bed velocity stationary point',NS,' is at '
       WRITE(4,*)'     x cell ',CUS(NS)
       WRITE(4,*)'      coord ',F(L0XC+(NY*(CUS(NS)-1)))
      END IF
     END DO
     WRITE(4,*)' '
     WRITE(4,*)'2) Shear stresses over period. '
     WRITE(4,*)'------------------------------------------ '
     WRITE(4,*)' '
     WRITE(4,*)'See file res_tau.m for wall shear stresses. '
     WRITE(4,*)' '
     IF (TWALL) THEN
      WRITE(4,*)'Smooth wall shear stress  is   ', STWP/PLEN
      WRITE(4,*)'          shear velocity is   ',
    1       SQRT(ABS(STWP/PLEN)/RHO1)
      WRITE(4,*)' '
     END IF
     WRITE(4,*)'Equivalent bed shear stress / datum levels. '
     WRITE(4,*)'  y0 geometric roughness level is  ',BLKHGT*BLKLEN/PLEN
     IF (MUIY.EQ.NY) THEN
C     max vel located at upper boundary; use DEPTH as pos of 0 shear stress
      WRITE(4,*)'                          tau  is ',
    1       ((BLKHGT*BLKLEN/PLEN)-DEPTH)*DPDX*(-1)
     ELSE
      WRITE(4,*)'                          tau  is ',
    1     ((BLKHGT*BLKLEN/PLEN)-F(L0YC+MUIY))*DPDX*(-1)
```

```
      END IF
      WRITE(4,*)' (calc from theortical linear shear stress profile)'
      WRITE(4,*)' '
      WRITE(4,*)'     y0 from con. ang. momt.  is ',EBSSP1
      WRITE(4,*)'                          tau  is ',EBSS1
      WRITE(4,*)' (calc from conservation of angular momentum) '
      IF (MUIY.EQ.NY) THEN
C     max vel located at upper boundary; use DEPTH as pos of 0 shear stress
       WRITE(4,*)'                          tau  is ',
     1      (EBSSP1-DEPTH)*DPDX*(-1)
      ELSE
       WRITE(4,*)'                          tau  is ',
     1      (EBSSP1-F(L0YC+MUIY))*DPDX*(-1)
      END IF
      WRITE(4,*)' (calc from theortical linear shear stress profile) '
      WRITE(4,*)'  '
      WRITE(4,*)' Note that con. ang. momt. gives 2nd roots as: '
      WRITE(4,*)' tau is      ',EBSS2
      WRITE(4,*)' position is ',EBSSP2
      WRITE(4,*)' '
      WRITE(4,*)' Also see file: res_shr '
      WRITE(4,*)' for period averaged effective shear stress. '
      WRITE(4,*)' '
      WRITE(4,*)' '
      WRITE(4,*)'4) Force balance over period. '
      WRITE(4,*)'------------------------------------ '
      WRITE(4,*)' (NB; forces are acting on the fluid)'
      WRITE(4,*)' '
      WRITE(4,*)'Driving forces: '
      WRITE(4,*)'The driving force is due to a body force acting '
      WRITE(4,*)'on each fluid cell, dependant on the press. grad.'
      WRITE(4,*)'  Pressure gradient is                 ',DPDX
      WRITE(4,*)'  and the pres. drop is                ',DP
      WRITE(4,*)'  area over which this acts (flow depth) is ',DEPTH
      WRITE(4,*)'  gives a force of                     ',DP*DEPTH
      WRITE(4,*)'However, the body force will not apply on the block;'
      WRITE(4,*)'  block body force pressure drop is ',DPDX*BLKLEN
      WRITE(4,*)'  block body force is       ',DPDX*BLKLEN*BLKHGT
      WRITE(4,*)' '
      WRITE(4,*)'Thus total driving force is ',DF
      WRITE(4,*)' '
      WRITE(4,*)'Resistive forces: '
      WRITE(4,*)'  total shear force on smooth top is ',STWP
      WRITE(4,*)'  total shear force on bed is       ',SBW
      WRITE(4,*)'  total shear force on block is     ',TSFB
      WRITE(4,*)'  total press. force on block is    ',TPFB
      WRITE(4,*)'Total resistive force is           ',TRFB+STWP
      WRITE(4,*)' '
      PEDF=100*(DF-(-1*(TRFB+STWP)))/DF
      WRITE(4,*)' % error in driving/ resistive forces is',PEDF
      WRITE(4,*)'(with respect to driving force). '
      WRITE(4,*)' '
      WRITE(4,*)' '
      WRITE(4,*)'******************************************** '
      WRITE(4,*)' '
      WRITE(4,*)'Results summary: '
      WRITE(4,*)'  Total run time is: ',TH,'hrs',TM,
     1                    'mins '
```

```
      WRITE(4,*)' '
      IF (SETUI.AND.NF.GT.0) THEN
        WRITE(4,*)' Flow convergance for a required average  '
        WRITE(4,*)' velocity with automatic update of DPDX: '
        WRITE(4,*)' '
        WRITE(4,*)'  Flow factor (UI/UC) is ',FLOWF
        WRITE(4,*)'  Latest pressure gradient is ',DPDX
        WRITE(4,*)'  % error in driving/ resistive forces is',PEDF
        WRITE(4,*)' '

      ELSEIF (SETUI.AND.NF.LE.0) THEN
        WRITE(4,*)' Flow convergance for a required average  '
        WRITE(4,*)' velocity with manual update of DPDX: '
        WRITE(4,*)' '
        WRITE(4,*)'  Pressure gradient used is ',DPDX
        WRITE(4,*)'  Average velocity required is ',UI
        WRITE(4,*)'  Average velocity calculated is ',UC
C        find % error wrt UI
        PEU=100*(UI-UC)/UI
        WRITE(4,*)'       % error is ',PEU
        WRITE(4,*)' '
        WRITE(4,*)' % error in driving/ resistive forces is ',PEDF
        WRITE(4,*)' '
        WRITE(4,*)'To continue solution: '
        IF(ABS(PEDF).GT.5) THEN
          WRITE(4,*)' Hold value of DPDX and run on. '
        ELSE
          IF (ABS(PEU).LT.5) THEN
            IF (ABS(PEDF).LT.0.1) THEN
              WRITE(4,*)'Case converged!!! '
            ELSE
              WRITE(4,*)'Reynolds number (and U1 average) correct '
              WRITE(4,*)'Hold DPDX and run on to convergance. '
            END IF
          ELSE
            WRITE(4,*)'Forces adiquatly converged. '
            WRITE(4,*)'Run on with new value of DPDX= ',
     1                                    DPDX*(UI/UC)**2
          END IF
        END IF
        WRITE(4,*)' '

      ELSE
        WRITE(4,*)' Flow convergance for a required pressure gradient:'
        WRITE(4,*)' '
        WRITE(4,*)' % error in driving/ resistive forces is',PEDF
        WRITE(4,*)' '

      END IF
      WRITE(4,*)' '
      WRITE(4,*)'End of file "result2". '
      WRITE(4,*)'****************************************** '
      CLOSE(4)

      RETURN
      END
C*********************************************************************
      SUBROUTINE MONITOR(MONTYPE,SETUI,UC,FLOWF)
```

```
C  Creates matlab files for convergance monitoring
C  Set IG(5) in Q1 to:    0 for no monitoring
C                          1 for complete monitoring
C                          2 for P1 and U1
C                          3 for EP and KE
C                          4 for dpdx calc.

   INCLUDE 'lp21/d_includ/satear'
   INCLUDE 'lp21/d_includ/grdloc'
   INCLUDE 'lp21/d_includ/grdear'
   INCLUDE 'lp21/d_includ/grdbfc'
C  Version 2.0
C    INCLUDE 'lp2/d_includ/satear'
C    INCLUDE 'lp2/d_includ/grdloc'
C    INCLUDE 'lp2/d_includ/grdear'
C    INCLUDE 'lp2/d_includ/grdbfc'
   PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
   COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
   LOGICAL LG
   CHARACTER*4 CG

C  My common block for rough duct analysis subroutines
   COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
   REAL PLEN,DPDX
   INTEGER IBU,IBD,BNY

C  Subroutine variables
   REAL MON,UC,FLOWF
   INTEGER MONTYPE
   LOGICAL SETUI

   IF (MONTYPE.EQ.1.OR.MONTYPE.EQ.2) THEN
C    p1 at inlet
    IXX=1
    IY=3*NY/4
    I=IY+NY*(IXX-1)
    IF (ISWEEP.EQ.1) THEN
     OPEN (20,FILE='mon_p1_in.m')
     WRITE(20,*)'% MATLAB monitoring plotting:'
     WRITE(20,*)'% of P1 at IX=',IXX,' IY=',IY
     WRITE(20,*)' '
     WRITE(20,*)'mon=[ '
    END IF
    MON=F(L0F(P1)+I)
    WRITE(20,*) MON
    IF (ISWEEP.EQ.LSWEEP) THEN
     WRITE(20,*)'];'
     WRITE(20,*)' '
     WRITE(20,*)'plot (mon)'
     WRITE(20,*)'title ("Spot values of P1 at IX=',IXX,' IY=',
  1          IY,' ")'
     WRITE(20,*)'xlabel("sweep number")'
     WRITE(20,*)'ylabel("spot value")'
     CLOSE (20)
    END IF
   END IF

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

Material removed!
>>

```
      RETURN
      END
C***************************************************************
C***************************************************************
      SUBROUTINE STREAM
C Calculates field values of stream function at north east
C corner of each cell.
C Outputs array PSI to file res_stm in text (excel) format:
C
C        ynf1  ynf2  ynf3  ynf4 ...
C   xef1   *     *     *     *
C   xef2   *     *     *     *
C   xef3   *     *     *     *
C    :
C

      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C  Version 2.0
C     INCLUDE 'lp2/d_includ/satear'
C     INCLUDE 'lp2/d_includ/grdloc'
C     INCLUDE 'lp2/d_includ/grdear'
C     INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C   My common block for rough duct analysis subroutines
      COMMON/ROUGH1/IBU,IBD,BNY,PLEN,DPDX
      REAL PLEN,DPDX
      INTEGER IBU,IBD,BNY

C  Subroutine variables:
C PSI(NY,NX)=Real array in y of period averaged U1 vel

C         NY+1*NX+1
      REAL PSI(120,125)

      WRITE(14,*)'sub stream '

      L0AN=L0F(ANORTH)
      L0AE=L0F(AEAST)
      L0U1=L0F(U1)
      L0XE=L0F(XU2D)
      L0YN=L0F(YV2D)


C   write x-east on 1st col of PSI
      IYY=1
      DO 30 IXX=1,NX
        I=IYY+NY*(IXX-1)
        PSI(1,IXX+1)=F(L0XE+I)
```

```
30    CONTINUE

C     write y-north on 1st row of PSI
      IXX=1
      DO 40 IYY=1,NY
         I=IYY+NY*(IXX-1)
         PSI(IYY+1,1)=F(L0YN+I)
40    CONTINUE

C     write stream function, starting at array (2,2)
      DO 10 IXX=1,NX
        IYY=1
        I=IYY+NY*(IXX-1)
        PSI(IYY+1,IXX+1)=(F(L0U1+I)*F(L0AE+I))
        DO 20 IYY=2,NY
          I=IYY+NY*(IXX-1)
          PSI(IYY+1,IXX+1)=PSI(IYY,IXX+1)+ (F(L0U1+I)*F(L0AE+I))
20      CONTINUE
10    CONTINUE



C     write to file
      OPEN (71,FILE='con_stm')
C           NY (dimension as above)
72    FORMAT(120(E10.4,2X))
      WRITE(71,72) PSI
      CLOSE(71)

      RETURN
      END
```

Appendix 12: PHOENICS Code For Simulating Flow in Smooth Rectangular Ducts

This Appendix contains an example of a Q1 input file and the GROUND subroutine which was used in the numerical simulation of flow in smooth rectangular three-dimensional ducts, described in Section 6.2.

A12.1 Q1 Input file

```
TALK=F;RUN( 1, 1);VDU=X11-TERM
NOCOPY=T;NOCOMM=T
SAVE=T
PARAB=F

REAL(HDEPTH,HWID,LENGTH,ASPECT)
REAL(UI,DPDX)
REAL(KEIN,EPIN)
REAL(MAXV,MINL,RELX,RELXT,RELXDP)
REAL(GYDIS,GZDIS)
REAL(KFRAC,DEL,DELT1,DUMMY,AUN,UC,DELTZ)
INTEGER(GNY,GNZ,UNY,UNZ,LNX,JJM,NF)
CHAR(ID,NUMBER)
BOOLEAN(SETUI)
  * GROUP 1.  Run identifiers and other preliminaries.
  *
  * Inputs:
  **********
  * Set case name and number (max 4 charaters).
ID='sd'
NUMBER='10'
DPDX=2.5
ASPECT=10
HWID=0.20
UI=10.038
RHO1=1.18293
ENUL=1.51319E-05
  * y grid
DELT1=1.56e-04
KFRAC=1.2418838966258
GNY=9
GYDIS=3.88661e-03

TEXT(SMOOTH DUCT
MESG( ************************************************************
MESG( *                CYCLIC DUCT FLOW
MESG( *             ALASTAIR SENIOR, RMCS
MESG( ************************************************************
MESG( *
MESG( * Case identification:
```

```
MESG(
ID
NUMBER
MESG( * Date:          12/1/96
MESG( * Running on:    v2.1.1, SUN
MESG( *
MESG( * Previous sweeps: 0
MESG( *
MESG( * Investigations into shear stress distrubtions
MESG( * in a smooth wide duct. Uses xcycle and momentum
MESG( * source method and the 2 layer KE model.
MESG( * Input requires bulk velocity and a dpdx estimate.
MESG( * Only 1/4 duct is modelled. Dim; metres. Fluid; air.
MESG( *
MESG( *
MESG( *
MESG( *                  /                    #
MESG( *  # = wall        /                     #
MESG( * - - = symmetry   /                      #
MESG( *                  - - - - - - - - - - - - #
MESG( *                 |                      #
MESG( *          hdepth|      .                 #     #
MESG( *           (h)  |    .                   #    #
MESG( *     y  x       |  .                     #  #
MESG( *     | /        |.                      # #
MESG( *     |/___ z    ################################
MESG( *                      hwid
MESG( *                       (b)
MESG( *
MESG( *
MESG( ***************************************************************

  * Run inputs
LSWEEP=100000

  *  Use this section for continuation runs.
KEIN=(UI*0.05)**2
EPIN=(0.009*(KEIN**2))/(50*ENUL)

RESTRT(ALL)
   FIINIT(U1)=UI
   FIINIT(V1)=1.0E-9
   FIINIT(W1)=1.0E-9
   FIINIT(P1)=0.0
   FIINIT(KE)=KEIN
   FIINIT(EP)=EPIN

  * general preliminaries:
HDEPTH=HWID/ASPECT
SETUI=T

  * grid
  **Uniform cell size: 1/20 hdepth
AUN=(HDEPTH-GYDIS)/(HDEPTH/20)
UNY=AUN

UNY
GNY
```

```
  * grid - z dir. kfrac and delt1 as y
  * last cell geo cell =1/20 of hwid

UC=HWID/20
DELTZ=DELT1
GZDIS=DELTZ
GNZ=1
LABEL LOOP
DELTZ=DELTZ*KFRAC
GNZ=GNZ+1
GZDIS=GZDIS+DELTZ
IF (DELTZ.LT.UC) THEN
GOTO LOOP
ENDIF
   * go back one
GZDIS=GZDIS-DELTZ
GNZ=GNZ-1
  **Uniform cell size: 1/20 HWID
AUN=(HWID-GZDIS)/UC
UNZ=AUN

UNZ
GNZ

  * grid x dir (streamwise)
LNX=2
LENGTH=LNX*UC

  * finish of grid:
NY=UNY+GNY
NZ=UNZ+GNZ
NX=LNX

NZ
NY
NX

  *  Estimate of the maximum velocity within domain
MAXV=UI*1.5
  *  Estimate of the minimum cell dimension
MINL=DELT1
  ***********************************************************************

PARAB=F

  ***********************************************************************
  *  GROUP 2.  Time-dependence and related parameters.
 STEADY=T
  ***********************************************************************
  *  GROUP 3.  x-direction grid specification.
CARTES=T
NREGX=1;REGEXT(X,1)
IREGX=1;GRDPWR(X,LNX,LENGTH,1)
  ***********************************************************************
  *
  *  GROUP 4.  y-direction grid specification.
  *
```

```
DEL=DELT1/HDEPTH
NREGY=2

  ** two region grid, geo...
  ** define uniform grid initially
IREGY=1;GRDPWR(Y,GNY,GYDIS,1.0)
IREGY=2;GRDPWR(Y,UNY,(HDEPTH-GYDIS),1.0)

  ** alter yfrac for geo grid, expanding from north boundary
YFRAC(1)=DEL

DO JJ=2,GNY
+ DEL=KFRAC*DEL
+ JJM=JJ-1
+ YFRAC(JJ)=YFRAC(JJM)+DEL
ENDDO

  *
  **********************************************************************
  *
  *  GROUP 5.  z-direction grid specification.
  *
DEL=DELT1/HWID
  ** two region grid, uniform....
NREGZ=2
IREGZ=1;GRDPWR(Z,UNZ,(HWID-GZDIS),1.0)
  ** and geo. Define uniform grid initially
IREGZ=2;GRDPWR(Z,GNZ,GZDIS,1.0)
  ** compute expanding grid from high Z boundary
ZFRAC(NZ)=1
ZFRAC(NZ-1)=1-DEL
DUMMY=DEL

DO JJ=NZ-2,NZ-(GNZ-1),-1
+ DEL=KFRAC*DEL
+ JJM=JJ+1
+ DUMMY=DUMMY+DEL
+ ZFRAC(JJ)=1.0-DUMMY
ENDDO

  *
  **********************************************************************
  *  GROUP 6.  Body-fitting and other grid distortions.
  **********************************************************************
  *  GROUP 7.  Variables (including porosities) named,
  *            stored & solved.
SOLVE(V1)
SOLVE(W1)
SOLVE(U1)
  *  Solve for P1 by whole-field method
SOLVE(P1)
SOLUTN(P1,Y,Y,Y,N,N,N)
STORE(ENUT)

  *
  **********************************************************************
  *  GROUP 8.  Terms (in differential equations) and devices.
  **********************************************************************
```

```
  * GROUP 9.  Properties of the medium (or media).
  *
REAL(RREF1)
RREF1=RHO1
   * Turbulance model (2 layer ke)

TURMOD(KEMODL);IENUTA=8;DISWAL

REAL(ENLREF);ENLREF=ENUL
  ***************************************************************
  * GROUP 10. Interphase-transfer processes and properties.
  ***************************************************************
  * GROUP 11. Initialization of fields of variables,
  *          porosities, etc.
  *
  * See group 1
  *
  ***************************************************************
  * GROUP 12. Convection and diffusion adjustments
  ***************************************************************
  *
  * GROUP 13. Boundary conditions and special sources
  *
  * momentum source for U1
PATCH(U1MT,VOLUME,1,NX,1,NY,1,NZ,1,1)
COVAL(U1MT,U1,FIXFLU,GRND)
PATCH(BAL,CELL,1,1,NY,NY,1,1,1,1)
COVAL(BAL,P1,FIXP,0.0)
XCYCLE=T
  ** walls
PATCH(WALL_BED,SWALL,1,NX,1,1,1,NZ,1,1)
COVAL(WALL_BED,U1,GRND2,0.0)
COVAL(WALL_BED,W1,GRND2,0.0)
COVAL(WALL_BED,KE,1.0,0.0)
COVAL(WALL_BED,LTLS,1.0,0.0)

PATCH(WALLSIDE,HWALL,1,NX,1,NY,NZ,NZ,1,1)
COVAL(WALLSIDE,U1,GRND2,0.0)
COVAL(WALLSIDE,V1,GRND2,0.0)
COVAL(WALLSIDE,KE,1.0,0.0)
COVAL(WALLSIDE,LTLS,1.0,0.0)

  *
  ***************************************************************
  * GROUP 14. Downstream pressure (for free parabolic flow).
  ***************************************************************
  *
* GROUP 15. Termination criteria for sweeps and
*          outer iterations.
*
*
***************************************************************
* GROUP 16. Termination criteria for inner iterations.
***************************************************************
*
* GROUP 17. Under-relaxation and related devices.
*
```

```
RELX=30
RELXT=10
  * AUTO Linear relaxation applied to P1
  * (0.9 weak, 0.1 strong)
RELAX(P1,LINRLX,0.7)
  * AUTO False time-step relaxation applied to V1
RELAX(V1,FALSDT,MINL/MAXV*RELX)
  * AUTO False time-step relaxation applied to U1
RELAX(U1,FALSDT,MINL/MAXV*RELX)
  * AUTO False time-step relaxation applied to KE
RELAX(KE,FALSDT,MINL/MAXV*RELXT)
  * AUTO False time-step relaxation applied to EP
RELAX(EP,FALSDT,MINL/MAXV*RELXT)
  * Ground relaxation in calc. of dpdx
RELXDP=0.5
  *
  ***************************************************************
  * GROUP 18. Limits on variables values or increments
  *          to them.
  ***************************************************************
  * GROUP 19. Data communicated by SATELLITE to GROUND
RG(1)=UI
RG(2)=HDEPTH
RG(3)=HWID
RG(4)=DPDX
RG(5)=RELXDP
CG(1)=':ID:'
CG(2)=':NUMBER:'
LG(1)=SETUI
  * Set monitoring. 0 = no monitoring
  *                 1 = complete monitoring
IG(2)=1
  ***************************************************************
  * GROUP 20. Control of preliminary printout
ECHO=t
   * Printout values of y+
YPLS=T
 ***************************************************************
  * GROUP 21. Frequency and extent of field printout.
  ***************************************************************
  * GROUP 22. Location of spot-value & frequency of
  *          residual printout.
  *
IYMON=NY/2
IXMON=NX/2
IZMON=NZ/2
TSTSWP=10
  ***************************************************************
  * GROUP 23. Variable-by-variable field printout and plot
  *          and/or tabulation of spot-values and residuals.
  *
  * Control tabulation & plotting of spot-values/residuals
  * Tables and plots
NZPRIN=1
NYPRIN=1
NXPRIN=1
  ***************************************************************
```

```
   * GROUP 24. Preparation for continuation runs.
   ************************************************************
STOP
```

A12.2 Ground Subroutine

```
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
      SUBROUTINE GROUND
C
C   ************************************************************
C   *            SMOOTH DUCT ANALYSIS
C   *           ALASTAIR SENIOR, RMCS
C   *               SMOOTHGRD.F
C   ************************************************************
C   ************************************************************
C
C  This ground routine is to be used with a Y-X domain using the
C  XCYCLE/ momentum source method of cyclic boundary conditions.
C  It is suitable for 3D domains.
C
C  Before using this code, CHECK THE ARRAY DIMENSIONS below.
C
C    FILE NAME: ground.f
C    DATE:     11/12/95
C    VERSION:   2.1.1 (Also suitable for 2.0 - change INCLUDE)
C    COMMENTS:
C
C
C
C    Q1 to Ground variables:
C    DPDX     RG(5)    DPDX
C    UI       RG(6)    UI
C    RELXDP   RG(7)    RELXDP

C
C
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C   Version 2.1.1
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C   Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'
C
C 1   Set dimensions of data-for-GROUND arrays here. WARNING: the
C     corresponding arrays in the MAIN program of the satellite
C     and EARTH must have the same dimensions.
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
C
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG
```

```
      SAVE

C     variables used to calc DPDX
C  Logicals:
C     SETUI=equal to .TRUE. if the vol flow is specifed in the Q1
C  Real:
C     VFLOW=calculated vol flow rate
C     UI=average velocity from Q1
C     VFLOWI=vol flow rate calculated from UI
C     FLOWF=flow factor used to calc DPDX
C     DPDX=pressure gradient
C     RELXDP=relaxation factor for DPDX
C     ERR=error used in calc of RELXDP

      LOGICAL SETUI
      REAL VFLOW, UI, VFLOWI, FLOWF, DPDX, RELXDP, ERR

C     variables used for smooth duct analysis
C  Real:
C     HWID=
C     HDEPTH=

      REAL AAN(100), AAH(100)
      REAL HWID, HDEPTH
      INTEGER MONTYPE
      INTEGER STIME(20)

c**************************************************************************
c
      IXL=IABS(IXL)
C   NB: Some sections have been removed for simplisty.
      IF(IGR.EQ.1) GO TO 1
      IF(IGR.EQ.13) GO TO 13
      IF(IGR.EQ.19) GO TO 19
      RETURN
C**************************************************************************
C
C--- GROUP 1. Run title and other preliminaries
C
    1 GO TO (1001,1002),ISC
 1001 CONTINUE
C
C     User may here change message transmitted to the VDU screen
      IF(IGR.EQ.1.AND.ISC.EQ.1.AND..NOT.NULLPR)
     1  CALL WRYT40('A. Senior smooth duct ground.f called   ')
C
C      CALL MAKE(YG2D)
C      CALL MAKE(YV2D)

      RETURN
 1002 CONTINUE
      RETURN
C**************************************************************************
C
C--- GROUP 13. Boundary conditions and special sources
C                                 Index for Coefficient - CO
C                                 Index for Value       - VAL
   13 CONTINUE
```

```
      IF(ISC.EQ.12) GO TO 1311
      RETURN

 1311 CONTINUE
C------------------- SECTION 12 ------------------- value = GRND

C  Set dpdx - press grad used for momentum source.
      IF (NPATCH.EQ.'U1MT') THEN
         CALL FN1(VAL,DPDX)
      END IF

      RETURN
C*******************************************************************
C
C--- GROUP 19. Special calls to GROUND from EARTH
C
   19 GO TO (191,192,193,194,195,196,197,198,199,1910,1911),ISC
  191 CONTINUE
C   * ------------------- SECTION 1 ---- Start of time step.
C     Set start time
      CALL MCLOCK(STIME)
C     Initialise values
      UI=RG(1)
      HDEPTH=RG(2)
      HWID=RG(3)
      DPDX=RG(4)
      RELXDP=RG(5)
      SETUI=LG(1)
      MONTYPE=IG(2)
      RETURN
  192 CONTINUE
C   * ------------------- SECTION 2 ---- Start of sweep.

C     Set calculated vol flow rate=0 for use in calc DPDX
      VFLOW=0.0

      RETURN
  193 CONTINUE
C   * ------------------- SECTION 3 ---- Start of iz slab.
      RETURN
  194 CONTINUE
C   * ------------------- SECTION 4 ---- Start of iterations over slab.

      RETURN
 1911 CONTINUE
C   * ------------------- SECTION 11---- After calculation of convection
C                              fluxes for scalars, and of volume
C                              fractions, but before calculation of
C                              scalars or velocities
      RETURN
  199 CONTINUE
C   * ------------------- SECTION 9 ---- Start of solution sequence for
C                                        a variable
      RETURN
 1910 CONTINUE
C   * ------------------- SECTION 10---- Finish of solution sequence for
C                                        a variable
      RETURN
```

```
 195 CONTINUE
C   * ------------------ SECTION 5 ---- Finish of iterations over slab.
      RETURN
 196 CONTINUE
C   * ------------------ SECTION 6 ---- Finish of iz slab.

      IF(SETUI) THEN
C   Calculate improved pressure grad.
C   First find total volume flow, VFLOW
        IXX=NX
        L0AE=L0F(AEAST)
        L0U1=L0F(U1)
        DO IYY=1,NY
          I=IYY+NY*(IXX-1)
          VFLOW=VFLOW+(F(L0U1+I)*F(L0AE+I))
        END DO
      END IF

C     find array in IZ of north cell area arrays (used in PRN)
      L0AN=L0F(ANORTH)
      IYY=1
      IXX=1
      I=IYY+NY*(IXX-1)
      AAN(IZ)=F(L0AN+I)
      write(14,*)' area north is ',AAN(IZ)
C     find array in IY of high cell area arrays (used in PRN)
      IF(IZ.EQ.NZ) THEN
        L0AH=L0F(AHIGH)
        IXX=1
        DO IYY=1,NY
          I=IYY+NY*(IXX-1)
          AAH(IYY)=F(L0AH+I)
C       write(14,*)' area high is ',AAH(IYY)

        END DO
      END IF

      RETURN
 197 CONTINUE
C   * ------------------ SECTION 7 ---- Finish of sweep.
C       WRITE(14,*)'sweep ',ISWEEP

      IF(SETUI.AND.ISWEEP.GT.2) THEN
C   Calculate improved pressure grad. (cont)
C   then find flow factor and calc dpdx

        VFLOW=VFLOW/(HWID*HDEPTH)
        FLOWF=UI/VFLOW

C       Set relaxation for calc of dpdx
        ERR=FLOWF-1
        ERR=ERR*RELXDP
        FLOWF=ERR+1
        DPDX=DPDX*FLOWF
        IF(DPDX.LE.0.1) THEN
          DPDX=0.11
```

```
        FLOWF=1.001
      END IF


    END IF


C     Convergance monitoring.
      IF(ISWEEP.GE.1) THEN
        IF (MONTYPE.GT.0) CALL MONITOR(MONTYPE,VFLOW,FLOWF,DPDX)
      END IF
      RETURN
  198 CONTINUE
C   * ------------------- SECTION 8 ---- Finish of time step.
      CALL PRNT (HWID,HDEPTH,DPDX,STIME,UI,FLOWF,VFLOW,AAN,AAH)
C
C
      RETURN
      END
C*********************************************************************
C*********************************************************************
      SUBROUTINE PRNT (HWID,HDEPTH,DPDX,STIME,UI,FLOWF,VFLOW,AAN,AAH)
C   Analsyis and printout

C   Version 2.1.1
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C   Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'

      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C   Subroutine variables
C SB(IZ)=array in NZ of shear stress on bed wall
C AAN(IZ)=array in NZ of cell area north force
C SW(IY)=array in NY of shear stress on side wall
C AAH(IY)=array in Ny of cell area high face
C FB=total bed shear force
C FW=total side wall shear force
C FP=percentage of shear force on sidewall
C DX=dis from cell centre of ix=1 to 2, or x len. of one cell (uni grid)
C DF=driving force due to momentum source
C RF=resistive force due to shear stresses
C RDPDX=resistive press. grad, from RF
C TWODSH=2D soln. for (centre line) bed shear stress
C FLOWF=flow factor for DPDX calc
C VFLOW=calculated average U1 vel
C DPDX=pressure gradient used as input for momentum source
C UI=inputted average U1 vel
```

```
C STIME=start or run time
C FTIME=finish of run time


      REAL SB(100),SW(100),AAN(100),AAH(100)
      REAL FB,FW,DX,DF,RF,RDPDX,TWODSH, ASPECT,HWID,FP
      REAL HDEPTH, WID, DEPTH, FLOWF, VFLOW, DPDX, UI
      CHARACTER*4 ID, NUMBER, ST1, FT1
      CHARACTER*2 ST4, ST5, ST3, ST2, FT4, FT5, FT3, FT2
      INTEGER FTIME(20), STIME(20)

      INTEGER TH, TM, TIME
      INTEGER IXPP,IYPP

      WRITE(14,*)' '
      WRITE(14,*)'sub PRN '
      WRITE(14,*)' '
      WRITE(14,*)'See file "result2"'

C     Initialisation
      ID=CG(1)
      NUMBER=CG(2)
      DEPTH=HDEPTH*2
      WID=HWID*2
      ASPECT=HWID/HDEPTH

C     Analysis:
C     Find F array values for wall_bed
      IY=1
      IYF=1
      IYL=1
      IX=NX
      IXF=NX
      IXL=NX
      IZF=1
      IZL=NZ
      DO 10 IZ=IZF,IZL
      L0SB=L0PVAR(PVSTRS,IPNAME('WALL_BED'),IZ)
        I=(IY-IYF+1)+(IYL-IYF+1)*(IX-IXF)
        SB(IZ)=F(L0SB+I)
        write(14,*)' '
        write(14,*)'ix is ',ix
        write(14,*)'iz is ',iz
        write(14,*)'iy is ',iy
        WRITE(14,*)'tau/rho for wall bed is ',SB(IZ)
        SB(IZ)=SB(IZ)*RHO1
        write(14,*)'aan is ',AAN(IZ)
        FB=FB+SB(IZ)*AAN(IZ)
10    CONTINUE

C     Find F array values for wallside
c     note: wall is now at high Z
      IYF=1
      IYL=NY
      IX=NX
      IXF=NX
      IXL=NX
      IZ=NZ
```

```
      IZF=NZ
      IZL=NZ
      L0SW=L0PVAR(PVSTRS,IPNAME('WALLSIDE'),IZ)
      DO 20 IY=IYF,IYL
         I=(IY-IYF+1)+(IYL-IYF+1)*(IX-IXF)
         SW(IY)=F(L0SW+I)
C          write(14,*)' '
C          write(14,*)'ix is ',ix
C          write(14,*)'iz is ',iz
C          write(14,*)'iy is ',iy
C          WRITE(14,*)'tau/rho for wall side is ',SW(IY)
         SW(IY)=SW(IY)*RHO1
C          write(14,*)'aah is ',AAH(IY)
          FW=FW+SW(IY)*AAH(IY)
20    CONTINUE
      FP=FW/(FW+FB)

C     Find DX
      CALL GETONE(DXG2D,DX,NY,2)
C      WRITE(14,*)'DX is ',DX

C     Driving force from momentum source
      DF=DPDX*DX*HWID*HDEPTH
C     Resitive force
      RF=FW+FB
      RDPDX=RF/(HWID*HDEPTH*DX)
C      2D soln for bed shear
      TWODSH=RDPDX*HDEPTH

C     Calc times
C **NB: If running on a CRAY, comment out this section, and the
C ** relevent section in the Print header.
      CALL MCLOCK(FTIME)
      WRITE(ST4,'(I2.2)') STIME(4)
      WRITE(ST5,'(I2.2)') STIME(5)
      WRITE(ST3,'(I2.2)') STIME(3)
      WRITE(ST2,'(I2.2)') STIME(2)
      WRITE(FT4,'(I2.2)') FTIME(4)
      WRITE(FT5,'(I2.2)') FTIME(5)
      WRITE(FT3,'(I2.2)') FTIME(3)
      WRITE(FT2,'(I2.2)') FTIME(2)
      WRITE(ST1,'(I4.4)') STIME(1)
      WRITE(FT1,'(I4.4)') FTIME(1)
      TIME=(FTIME(3)-STIME(3))*24*60
      TIME=TIME+((FTIME(4)-STIME(4))*60)+ (FTIME(5)-STIME(5))
      TH=TIME/60
      TM=MOD(TIME,60)

C     **** Print out to file result2

      OPEN (4,FILE='result2')
      WRITE(4,*)' '
      WRITE(4,*)' '
      WRITE(4,*)'************************************************** '
      WRITE(4,*)'*          A. Senior. RMCS.                     *'
      WRITE(4,*)'* File = "result2"                              *'
      WRITE(4,*)'* Gives GROUND results of xcyclic rough ducts.  *'
```

```
WRITE(4,*)'* All units are standard SI unless stated in Q1. *'
WRITE(4,*)'*************************************************'
WRITE(4,*)'*                                                 *'
WRITE(4,*)'* Case ID name: ',ID,'                            *'
WRITE(4,*)'* Case number:  ',NUMBER,
1                    '                          *'
WRITE(4,*)'*                                                 *'
WRITE(4,*)'* Started at : ',ST4,':',ST5,' on ',
1                 ST3,'/',ST2,'/',ST1,'              *'
WRITE(4,*)'* Finished at: ',FT4,':',FT5,' on ',
1                 FT3,'/',FT2,'/',FT1,'              *'
WRITE(4,*)'* Total run time is: ',TH,'hrs',TM,
1                      'mins             *'
IF (FT2.NE.ST2.OR.FT1.NE.ST1) THEN
  WRITE(4,*)'***** CHECK MONTH AND DAYS******'
END IF
WRITE(4,*)'*                                                 *'
WRITE(4,*)'* Finished on sweep number: ',ISWEEP,'            *'
WRITE(4,*)'*                                                 *'

WRITE(4,*)'*************************************************'
WRITE(4,*)' '
WRITE(4,*)'INPUTS: '
WRITE(4,*)'======= '
WRITE(4,*)' '
WRITE(4,*)'Domain geometry:'
write(4,*)' '
write(4,*)'   Aspect Ratio is   ',ASPECT
write(4,*)'   Width, b, (z dir) is  ',2*HWID
write(4,*)'   Height, h, (y dir) is ',2*HDEPTH
write(4,*)'   1/2 width is      ',HWID
write(4,*)'   1/2 height is     ',HDEPTH
write(4,*)'   NY is',NY,'   NX is',NX,'   NZ is',NZ
write(4,*)' '
WRITE(4,*)'Average input velocity is ',UI
WRITE(4,*)' '
WRITE(4,*)'************************************************ '
WRITE(4,*)' '
WRITE(4,*)'RESULTS'
WRITE(4,*)'======='
WRITE(4,*)' '
WRITE(4,*)'1) Check on convergance: '
WRITE(4,*)'    Flow factor is       ',FLOWF
WRITE(4,*)'    Average velocity is  ',VFLOW
write(4,*)'    Pressure gradient is ',DPDX
WRITE(4,*)' '
WRITE(4,*)'2) Shear forces:'
write(4,*)'shear force on sidewall is ',FW
write(4,*)'shear force on bedwall is   ',FB
write(4,*)'percentage of shear force on sidewall is ',FP*100
write(4,*)' '
write(4,*)'shear force per unit length on sidewall is ',FW/DX
write(4,*)'shear force per unit length on bedwall is  ',FB/DX
write(4,*)'NB: Force given is for 1/4 duct only. '
write(4,*)' '
write(4,*)' '
write(4,*)'3) Force balance: '
write(4,*)'   Driving force is ', DF
```

```
write(4,*)'  Resitive force is ',RF
write(4,*)'Percentage error is        ',100*(DF-RF)/DF
write(4,*)' '
write(4,*)'4) Centerline shear stresses: '
write(4,*)'  2D theortical solution is ',TWODSH
write(4,*)' '
write(4,*)'  Shear stress on bed center line is ',SB(1)
write(4,*)'  Ratio: tau/2D tau is ', SB(1)/TWODSH
write(4,*)'  the % difference is ',((TWODSH-SB(1))*100)/SB(1)
write(4,*)' '
write(4,*)' [ for inverse aspect ratio = ',HDEPTH/HWID,' ]'
write(4,*)' [ Shear stress on wall center line is ',SW(NY),' ]'
write(4,*)' [ Ratio: tau/2D tau is ', SW(NY)/TWODSH,' ]'
write(4,*)' [ the % difference is ',
1                  ((TWODSH-SW(NY))*100)/SW(NY),' ]'
 write(4,*)' '
 write(4,*)' '
 write(4,*)'Results summary: '
 write(4,*)'---------------- '
 write(4,*)'percentage of shear force on side is',FP*100
 write(4,*)'Ratio: tau/2D tau is              ', SB(1)/TWODSH
 write(4,*)'% error in force balance is ',100*(DF-RF)/DF
 write(4,*)' '
 write(4,*)' '
 write(4,*)'End of file Result2 '

 CLOSE (4)

 RETURN
 END

C*********************************************************************
      SUBROUTINE MONITOR(MONTYPE,VFLOW,FLOWF,DPDX)
C  Creates matlab files for convergance monitoring
C  Set IG(2) in Q1 to:    0 for no monitoring
C                         1 for complete monitoring

      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
C  Version 2.0
C       INCLUDE 'lp2/d_includ/satear'
C       INCLUDE 'lp2/d_includ/grdloc'
C       INCLUDE 'lp2/d_includ/grdear'
C       INCLUDE 'lp2/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C  Subroutine variables
      REAL MON
      INTEGER MONTYPE

      IF (MONTYPE.EQ.1.OR.MONTYPE.EQ.2) THEN
C       u1 at true duct centre
        IX=NX
```

```
IY=NY
IZ=1
L0U1=L0F(ANYZ(U1,IZ))
I=IY+NY*(IX-1)
IF (ISWEEP.EQ.1) THEN
  OPEN (20,FILE='mon_u1.m')
  WRITE(20,*)'% MATLAB monitoring plotting:'
  WRITE(20,*)'% of U1 at IX=',IX,' IY=',IY,' IZ=',IZ
  WRITE(20,*)' '
  WRITE(20,*)'mon=[ '
END IF
MON=F(L0U1+I)
   WRITE(20,*) MON
   IF (ISWEEP.EQ.LSWEEP) THEN
    WRITE(20,*)'];'
    WRITE(20,*)' '
    WRITE(20,*)'plot (mon)'
    WRITE(20,*)'title ("Spot values of U1 at IX=',IX,' IY=',
1             IY,' IZ=',IZ,' ")'
    WRITE(20,*)'xlabel("sweep number")'
    WRITE(20,*)'ylabel("spot value")'
    CLOSE (20)
  END IF
 END IF

 IF (MONTYPE.EQ.1.OR.MONTYPE.EQ.4) THEN
C    VFLOW, calculated bulk flow vel
  IF (ISWEEP.EQ.1) THEN
   OPEN (29,FILE='mon_uc.m')
   WRITE(29,*)'% MATLAB monitoring plotting:'
   WRITE(29,*)'% of VFLOW'
   WRITE(29,*)' '
   WRITE(29,*)'mon=[ '
  END IF
  MON=VFLOW
  WRITE(29,*) MON
  IF (ISWEEP.EQ.LSWEEP) THEN
   WRITE(29,*)'];'
   WRITE(29,*)' '
   WRITE(29,*)'plot (mon)'
   WRITE(29,*)'title ("Calculated values of VFLOW.") '
   WRITE(29,*)'xlabel("sweep number")'
   WRITE(29,*)'ylabel("bulk velocity")'
   CLOSE (29)
  END IF
END IF

IF (MONTYPE.EQ.1.OR.MONTYPE.EQ.4) THEN
C    FLOWF, flow factor
 IF (ISWEEP.EQ.1) THEN
  OPEN (30,FILE='mon_flowf.m')
  WRITE(30,*)'% MATLAB monitoring plotting:'
  WRITE(30,*)'% of flowfactor'
  WRITE(30,*)' '
  WRITE(30,*)'mon=[ '
 END IF
 MON=FLOWF
 WRITE(30,*) MON
```

```
  IF (ISWEEP.EQ.LSWEEP) THEN
    WRITE(30,*)'];'
    WRITE(30,*)' '
    WRITE(30,*)'plot (mon)'
    WRITE(30,*)'title ("Values of flow factor.") '
    WRITE(30,*)'xlabel("sweep number")'
    WRITE(30,*)'ylabel("flow factor")'
    CLOSE (30)
  END IF
END IF

IF (MONTYPE.EQ.1.OR.MONTYPE.EQ.4) THEN
C     DPDX, pressure grad.
      IF (ISWEEP.EQ.1) THEN
        OPEN (31,FILE='mon_dpdx.m')
        WRITE(31,*)'% MATLAB monitoring plotting:'
        WRITE(31,*)'% of pressure grad'
        WRITE(31,*)' '
        WRITE(31,*)'mon=[ '
      END IF
      MON=DPDX
      WRITE(31,*) MON
      IF (ISWEEP.EQ.LSWEEP) THEN
        WRITE(31,*)'];'
        WRITE(31,*)' '
        WRITE(31,*)'plot (mon)'
        WRITE(31,*)'title ("Values of DPDX.") '
        WRITE(31,*)'xlabel("sweep number")'
        WRITE(31,*)'ylabel("DPDX")'
        CLOSE (31)
      END IF
END IF


RETURN
END
```

ALL MISSING PAGES ARE BLANK

IN

ORIGINAL

Appendix 13: PHOENICS Code for Simulating Flow in a Plate-Fin Heat Exchanger

The following Q1 and GROUND code was used to calculate fully developed periodic flow in a two fluid counterflow plate-fin heat exchanger of the type described in *Section 6.3.*

A13.1 Q1 data file

```
TALK=F;RUN( 1, 2);VDU=X11-TERM
NOCOPY=T;NOCOMM=T
SAVE=T
PARAB=F
REAL(TW,TS,TC,TO,IH,TP,S,WDTH)
INTEGER(NTW,NTS,NTC,NIH,NTO,NTP,NS,NWDTH)
INTEGER(STW1,FNW1,STS1,FNS1,STC1,FNC1,STOIL,FNOIL,STC2,FNC2)
INTEGER(STS2,FNS2,STW2,FNW2,FNB,STT,YMID)
INTEGER(FX2A,FX1A,FX2B,FX1B,FZ2)
REAL(DPDXO,DPDXW,TKOIL,TKWAT,TSOUC)
REAL(UIO,UIW,MAXV,MINL,RELX,RELXDP)
CHAR(ID,NUMBER)
BOOLEAN(CALCV)
  *
  * Set case name and number (max 4 charaters).
ID='hx'
NUMBER='aa20'
TEXT(HEAT EXCHANGER
  *
MESG( ************************************************************
MESG( *          CYCLIC HEAT EXCHANGER FLOW
MESG( *            ALASTAIR SENIOR, RMCS
MESG( ************************************************************
MESG( * recyclic bc
MESG( *
MESG( * Case identification:
MESG(
ID
NUMBER
MESG( * Date:          21/4/96
MESG( * Running on:     v2.1.1, SUN
MESG( *
MESG( *
MESG( *
MESG( ************************************************************
  *
  * GROUP 1.  Run identifiers and other preliminaries.
  *
  * Number of sweeps
LSWEEP=1000
  *
  * Domain dimensions:
TW=1.5E-3
```

```
TS=0.2E-3
TC=0.1E-3
TO=3E-3
TP=0.2E-3
S=2E-3
IH=TO-(2*TP)

WDTH=2E-3
  * Domain cells: (Note: NS and NW must be an even number)
NTW=8
NTS=4
NTC=2
  * NB: NIH should be an odd number
NIH=15
NTP=4
NS=6
NTO=NIH+(2*NTP)

NWDTH=12
  *
  * Set flow, dpdx estimate
DPDXO=1e6
DPDXW=500
  * set initial temp source estimate
TSOUC=1.2E-7
  * required U vel
UIO=0.1
UIW=0.1
  *
MESG( * Grid locations:
  * grid locations -Y
STW1=1
FNW1=NTW
STS1=NTW+1
FNS1=STS1+NTS-1
STC1=FNS1+1
FNC1=STC1+NTC-1
STOIL=FNC1+1
FNB=STOIL+NTP-1
STT=STOIL+NTO-NTP
FNOIL=STOIL+NTO-1
STC2=FNOIL+1
FNC2=STC2+NTC-1
STS2=FNC2+1
FNS2=STS2+NTS-1
STW2=FNS2+1
FNW2=STW2+NTW-1
YMID=(FNW2+1)/2
STW1
FNW1
STS1
FNS1
STC1
FNC1
STOIL
FNB
STT
FNOIL
STC2
FNC2
STS2
```

```
FNS2
STW2
FNW2
YMID
  * grid locations -X
FX2A=NS/2+1
FX1A=FX2A+NTP+NS
FX2B=FX1A+NTP+NS
FX1B=FX2B+NTP+NS
FX2A
FX1A
FX2B
FX1B
  * grid locations -Z
FZ2=(NWDTH/2)+1
FZ2
  *
  * calc vel
CALCV=T
  *  Estimate of the minimum cell dimension
MINL=TO/NTO
  *  Estimate of the maximum velocity within domain
MAXV=UIO*1.5
  *
  **********************************************************************
  *  GROUP 2.  Time-dependence and related parameters.
STEADY=T
  **********************************************************************
  *  GROUP 3.  x-direction grid specification.
CARTES=T
NREGX=9
IREGX=1;GRDPWR(X,NS/2,S/2,1)
IREGX=2;GRDPWR(X,NTP,TP,1)
IREGX=3;GRDPWR(X,NS,S,1)
IREGX=4;GRDPWR(X,NTP,TP,1)
IREGX=5;GRDPWR(X,NS,S,1)
IREGX=6;GRDPWR(X,NTP,TP,1)
IREGX=7;GRDPWR(X,NS,S,1)
IREGX=8;GRDPWR(X,NTP,TP,1)
IREGX=9;GRDPWR(X,NS/2,S/2,1)

  **********************************************************************
  *
  *  GROUP 4.  y-direction grid specification.
  *
NREGY=9
IREGY=1;GRDPWR(Y,NTW,TW,1)
IREGY=2;GRDPWR(Y,NTS,TS,1)
IREGY=3;GRDPWR(Y,NTC,TC,1)
IREGY=4;GRDPWR(Y,NTP,TP,1)
IREGY=5;GRDPWR(Y,NIH,IH,1)
IREGY=6;GRDPWR(Y,NTP,TP,1)
IREGY=7;GRDPWR(Y,NTC,TC,1)
IREGY=8;GRDPWR(Y,NTS,TS,1)
IREGY=9;GRDPWR(Y,NTW,TW,1)
  *
  **********************************************************************
  *
  *  GROUP 5.  z-direction grid specification.
NREGZ=2
```

```
IREGZ=1;GRDPWR(Z,NWDTH/2,WDTH/2,1)
IREGZ=2;GRDPWR(Z,NWDTH/2,WDTH/2,1)
  *
  *
  ***************************************************************
  * GROUP 6.  Body-fitting and other grid distortions.
  ***************************************************************
  * GROUP 7.  Variables (including porosities) named,
  *
        stored & solved.
SOLVE(V1)
SOLVE(U1)
SOLVE(W1)
  * Solve for P1 by whole-field method
SOLVE(P1)
SOLUTN(P1,Y,Y,Y,N,N,N)
  * Solve for TEM1
SOLVE(TEM1)
  * store?,solve?,wholefield?,pt-by-pt?,exp(trans)? har. ave?
SOLUTN(TEM1,Y,N,Y,N,N,Y)
  *
STORE(ENUL)
STORE(PRPS)
STORE(BLOK)
  *
  ***************************************************************
  * GROUP 8.  Terms (in differential equations) and devices.
  *
  *
  ***************************************************************
  * GROUP 9.  Properties of the medium (or media).
  *
RHO1=GRND10
PRNDTL(TEM1)=-GRND10
  * Laminar kinematic viscosity (m^2/s)
ENUL=GRND10
CP1=GRND10
  ***************************************************************
  * GROUP 10. Interphase-transfer processes and properties.
  ***************************************************************
  * GROUP 11. Initialization of fields of variables,
  *          porosities, etc.
  *
FIINIT(U1)=UIO
FIINIT(V1)=1.0E-9
FIINIT(W1)=1.0E-9
FIINIT(P1)=0.0
FIINIT(TEM1)=273+70
   * set initial properties
INIADD=F
PATCH(INW1,INIVAL,1,NX,STW1,FNW1,1,NZ,1,1)
PATCH(INO1,INIVAL,1,NX,STOIL,FNOIL,1,NZ,1,1)
PATCH(INW2,INIVAL,1,NX,STW2,FNW2,1,NZ,1,1)

INIT(INW1,U1,0,UIW)
INIT(INW1,BLOK,0,67)
INIT(INW1,PRPS,0,67)

INIT(INO1,U1,0,UIO)
```

```
INIT(INO1,BLOK,0,65)
INIT(INO1,PRPS,0,65)

INIT(INW2,U1,0,UIW)
INIT(INW2,BLOK,0,67)
INIT(INW2,PRPS,0,67)

  * blocks
CONPOR(STE1,-1,CELL,-1,-NX,-STS1,-FNS1,-1,-NZ)
COVAL(STE1,PRPS,0.0,111)
COVAL(STE1,BLOK,0.0,2)

CONPOR(COP1,-1,CELL,-1,-NX,-STC1,-FNC1,-1,-NZ)
COVAL(COP1,PRPS,0.0,103)
COVAL(COP1,BLOK,0.0,3)

CONPOR(COP2,-1,CELL,-1,-NX,-STC2,-FNC2,-1,-NZ)
COVAL(COP2,PRPS,0.0,103)
COVAL(COP2,BLOK,0.0,4)

CONPOR(STE2,-1,CELL,-1,-NX,-STS2,-FNS2,-1,-NZ)
COVAL(STE2,PRPS,0.0,111)
COVAL(STE2,BLOK,0.0,5)

  *Set fins etc. - check vel just before fins
CONPOR (F1A,-1,CELL, -FX1A,-(FX1A+NTP-1), -STOIL,-FNOIL,$
-1,-(FZ2-1))
COVAL(F1A,PRPS,0.0,111)
COVAL(F1A,BLOK,0.0,6)

CONPOR (F1B,-1,CELL, -FX1B,-(FX1B+NTP-1), -STOIL,-FNOIL,$
-1,-(FZ2-1))
COVAL(F1B,PRPS,0.0,111)
COVAL(F1B,BLOK,0.0,7)

CONPOR (F2A,-1,CELL, -FX2A,-(FX2A+NTP-1), -STOIL,-FNOIL,$
-FZ2,-NZ)
COVAL(F2A,PRPS,0.0,111)
COVAL(F2A,BLOK,0.0,8)

CONPOR (F2B,-1,CELL, -FX2B,-(FX2B+NTP-1), -STOIL,-FNOIL,$
-FZ2,-NZ)
COVAL(F2B,PRPS,0.0,111)
COVAL(F2B,BLOK,0.0,9)

  * set base fins
CONPOR (B1A,-1,CELL, -1,-(FX1A-1),   -STOIL,-(STOIL+NTP-1),$
-1,-(FZ2-1))
COVAL(B1A,PRPS,0.0,111)
COVAL(B1A,BLOK,0.0,10)

CONPOR (B1B,-1,CELL, -(FX1B+NTP),-NX, -STOIL,-(STOIL+NTP-1),$
-1,-(FZ2-1))
COVAL(B1B,PRPS,0.0,111)
COVAL(B1B,BLOK,0.0,11)

CONPOR (B2A,-1,CELL, -1,-(FX2A-1),   -STOIL,-(STOIL+NTP-1),$
-FZ2,-NZ)
COVAL(B2A,PRPS,0.0,111)
COVAL(B2A,BLOK,0.0,12)
```

```
CONPOR (B2B,-1,CELL, -(FX2B+NTP),-NX, -STOIL,-(STOIL+NTP-1),$
-FZ2,-NZ)
COVAL(B2B,PRPS,0.0,111)
COVAL(B2B,BLOK,0.0,13)

  * Definition of top fins
CONPOR (T1,-1,CELL, -(FX1A+NTP),-(FX1B-1), -(FNOIL-NTP+1),$
-FNOIL,  -1,-(FZ2-1))
COVAL(T1,PRPS,0.0,111)
COVAL(T1,BLOK,0.0,14)

CONPOR (T2,-1,CELL, -(FX2A+NTP),-(FX2B-1), -(FNOIL-NTP+1),$
-FNOIL,  -FZ2,-NZ)
COVAL(T2,PRPS,0.0,111)
COVAL(T2,BLOK,0.0,15)

  *
  ***************************************************************
  *  GROUP 12. Convection and diffusion adjustments
  ***************************************************************
  *
  *  GROUP 13. Boundary conditions and special sources
  *
PATCH(WAT1,VOLUME,1,NX,STW1,FNW1,1,NZ,1,1)
PATCH(OIL1,VOLUME,1,NX,STOIL,FNOIL,1,NZ,1,1)
PATCH(WAT2,VOLUME,1,NX,STW2,FNW2,1,NZ,1,1)

  * momentum source
COVAL(WAT1,U1,FIXFLU,DPDXW)
PATCH(RPW1,CELL,1,1,STW1,STW1,NZ/2,NZ/2,1,1)
COVAL(RPW1,P1,FIXP,0.0)

COVAL(OIL1,U1,FIXFLU,DPDXO)
PATCH(RPO1,CELL,1,1,YMID,YMID,NZ/2,NZ/2,1,1)
COVAL(RPO1,P1,FIXP,0.0)

COVAL(WAT2,U1,FIXFLU,DPDXW)
PATCH(RPW2,CELL,1,1,FNW2,FNW2,NZ/2,NZ/2,1,1)
COVAL(RPW2,P1,FIXP,0.0)

XCYCLE=T
  *
  *  GROUP 17. Under-relaxation and related devices.
  *
  *  Level of relaxation (100 - weak, 0.1 - strong)
RELX=10
  *  AUTO Linear relaxation applied to P1
  *  (0.9 weak, 0.1 strong)
RELAX(P1,LINRLX,0.3)
  *  AUTO False time-step relaxation applied to V1
RELAX(V1,FALSDT,MINL/MAXV*RELX)
  *  AUTO False time-step relaxation applied to W1
RELAX(W1,FALSDT,MINL/MAXV*RELX)
  *  AUTO False time-step relaxation applied to U1
RELAX(U1,FALSDT,MINL/MAXV*RELX)
RELXDP=0.5
  *
  ***************************************************************
  *  GROUP 18. Limits on variables values or increments
  *          to them.
  ***************************************************************
```

```
  * GROUP 19. Data communicated by SATELLITE to GROUND

LG(1)=CALCV
RG(6)=TO
RG(7)=TW
IG(1)=STW1
IG(2)=FNW1
IG(3)=STS1
IG(4)=FNS1
IG(5)=STC1
IG(6)=FNC1
IG(7)=STOIL
IG(8)=FNB
IG(9)=STT
IG(10)=FNOIL
IG(11)=STC2
IG(12)=FNC2
IG(13)=STS2
IG(14)=FNS2
IG(15)=STW2
IG(16)=FNW2
IG(17)=YMID

  **********************************************************************
  * GROUP 20. Control of preliminary printout
ECHO=T
  **********************************************************************
  * GROUP 21. Frequency and extent of field printout.
  **********************************************************************
  * GROUP 22. Location of spot-value & frequency of
  *         residual printout.
  *
IYMON=3*NY/4
IXMON=NX/2
TSTSWP=10
  *
  **********************************************************************
  * GROUP 23. Variable-by-variable field printout and plot
  *         and/or tabulation of spot-values and residuals.
  *
  * Control tabulation & plotting of spot-values/residuals
  * Tables and plots
ITABL=2
NPRINT=100000
NXPRIN=1
NYPRIN=1
  **********************************************************************
  * GROUP 24. Preparation for continuation runs.
NOWIPE=T
NSAVE='phi1'
  **********************************************************************
STOP

  * Run 2 - calc temp field
TEXT (HEAT EXCHANGER
  * temp inputs:
TKOIL=273+100
TKWAT=273+70
  *
NAMFI='phi1'
RESTRT(P1,U1,V1,W1,TEM1)
```

```
CALCV=F
LG(1)=CALCV

RG(8)=TKWAT
RG(9)=TSOUC

  **Switch off solution of pressure and velocities and cut
    out printout
SOLUTN(P1,Y,N,N,N,N,N);SOLUTN(U1,Y,N,N,N,N,N)
SOLUTN(V1,Y,N,N,N,N,N);SOLUTN(W1,Y,N,N,N,N,N)
OUTPUT(P1,N,N,N,N,N,N);OUTPUT(W1,N,N,N,N,N,N)
OUTPUT(U1,N,N,N,N,N,N);OUTPUT(V1,N,N,N,N,N,N)
OUTPUT(BLOK,N,N,N,N,N,N)
  **Activate whole-field solution of tem1
SOLUTN(TEM1,Y,Y,Y,N,N,Y)
OUTPUT(TEM1,Y,N,N,Y,Y,Y)
  **Activate block solver
ISOLBK=1;IVARBK=50;ISOLX=1;ISOLY=1
   * source of heat in oil flow
COVAL(OIL1,TEM1,FIXFLU,GRND)
PATCH(TREF,CELL,1,1,STOIL+NTO/2,STOIL+NTO/2,NZ/2,NZ/2,1,1)
COVAL(TREF,TEM1,FIXVAL,TKOIL)
   * sink of heat in water flow
COVAL(WAT1,TEM1,FIXFLU,GRND)
COVAL(WAT2,TEM1,FIXFLU,GRND)

   * Set sweeps
FSWEEP=LSWEEP
LSWEEP=LSWEEP+500;LITER(TEM1)=20

NSAVE='phi2'
STOP
```

A13.2 GROUND coding

```
C FILE NAME GROUND.FTN----------------------------081294
c#### dbs/hqq 08.12.94 UCONV comments provided
c#### dbs/mrm 10.08.94 new access point on group 19, section 11
      SUBROUTINE GROUND
C
C  ***************************************************************
C  *          HEAT EXCHANGER GROUND CODING
C  *            ALASTAIR SENIOR, RMCS
C  ***************************************************************
C  ***************************************************************
C
C     FILE NAME: aa20.grd
C     DATE:     21/4/96
C     VERSION:  2.1.1
C     COMMENTS:
C
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX USER SECTION STARTS:
C
```

```
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
C
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG
C
C  Subroutine variables:
      REAL UI1,UI2,UC,FLOWA,DPDX1,DPDX2,FLOWF,RELXDP,ERR,VOLF,IH,WH
      REAL FRW,ATW,TVOLW,ATO,TVOLO,KSTEEL,DELY,Q,TAQ,TQB,VQA,VQB
      INTEGER STW1,FNW1,STS1,FNS1,STC1,FNC1,STOIL,FNOIL,STC2,FNC2
      INTEGER STS2,FNS2,STW2,FNW2,FNB,STT,YMID
      LOGICAL CALCV
C
C*************************************************************************
c
      IXL=IABS(IXL)
      IF(IGR.EQ.1) GO TO 1
      IF(IGR.EQ.13) GO TO 13
      IF(IGR.EQ.19) GO TO 19
   25 CONTINUE
      RETURN
C*******************************************************************
C
C--- GROUP 1. Run title and other preliminaries
C
    1 GO TO (1001,1002),ISC
 1001 CONTINUE
C
C    User may here change message transmitted to the VDU screen
      IF(IGR.EQ.1.AND.ISC.EQ.1.AND..NOT.NULLPR)
     1  CALL WRYT40('GROUND file is GROUND.F  ~hx2~; 22/4/96  ')
C
      CALL MAKE(YG2D)

      RETURN
 1002 CONTINUE
      RETURN
C*******************************************************************
C
C--- GROUP 13. Boundary conditions and special sources
C                              Index for Coefficient - CO
C                              Index for Value       - VAL
   13 CONTINUE
      GO TO (130,131,132,133,134,135,136,137,138,139,1310,
     11311,1312,1313,1314,1315,1316,1317,1318,1319,1320,1321),ISC
  130 CONTINUE
C------------------- SECTION  1 ------------- coefficient = GRND
      RETURN

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
MATERIAL OMITTED !!
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

 1311 CONTINUE
C------------------- SECTION 12 ------------------- value = GRND
C  Set sources, either momentum or temp.
      IF (CALCV) THEN
C  Set momentum source
      write(14,*)'ERROR!! - ground setting mt source. '
      ELSE
C  Set tem1 source - temp grad.
```

```
      IF (NPATCH.EQ.'OIL1') THEN
        CALL FN1(VAL,TSOUC)
C         CALL PRN('v1',VAL)
      END IF
      IF (NPATCH.EQ.'WAT1') THEN
        CALL FN1(VAL,TSOUC)
C         CALL PRN('v2',VAL)
        CALL FN25(VAL,-1.0)
C         CALL PRN('v3',VAL)
        CALL FN25(VAL,VOLF)
C         write(14,*)'volf is ',VOLF
C         CALL PRN('v4',VAL)
      END IF
      IF (NPATCH.EQ.'WAT2') THEN
        CALL FN1(VAL,TSOUC)
        CALL FN25(VAL,-1.0)
        CALL FN25(VAL,VOLF)
      END IF

    END IF


    RETURN
 1312 CONTINUE
C------------------- SECTION 13 ------------------- value = GRND1
    RETURN

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
MATERIAL OMITTED !!
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

 1321 CONTINUE
C------------------- SECTION 22 ------------------- value = GRND10
    RETURN
C*********************************************************************
C
C--- GROUP 19. Special calls to GROUND from EARTH
C
  19 GO TO (191,192,193,194,195,196,197,198,199,1910,1911),ISC
 191 CONTINUE
C   * ------------------- SECTION 1 ---- Start of time step.

C    Initialise
     IH=RG(6)
     WH=RG(7)
     TREF=RG(8)
     TSOUCI=RG(9)

     STW1=IG(1)
     FNW1=IG(2)
     STS1=IG(3)
     FNS1=IG(4)
     STC1=IG(5)
     FNC1=IG(6)
     STOIL=IG(7)
     FNB=IG(8)
     STT=IG(9)
     FNOIL=IG(10)
     STC2=IG(11)
     FNC2=IG(12)
     STS2=IG(13)
```

```
FNS2=IG(14)
STW2=IG(15)
FNW2=IG(16)
YMID=IG(17)

CALCV=LG(1)
IF(CALCV) THEN
  write(14,*)'solving for vel '
ELSE
  write(14,*)'solving for tem1 '
END IF

TSOUCI=0.8E+6
TSOUC=TSOUCI
TMPF=1.0
NSWP=1
TACT=1.0
VOLF=IH/(2*WH)
write(14,*)'water height ,*2 is ',2*WH
write(14,*)'oil height is ',IH
write(14,*)'volf 1 is ',VOLF

C    For itergration of results:
FRW=0.0
FRO=0.0
ATW=0.0
ATO=0.0
TVOLW=0.0
TVOLO=0.0
TQA=0.0
TQB=0.0
VQA=0.0
VQB=0.0
TOTQ=0.0
totan=0.0

    RETURN
 192 CONTINUE
C   * ------------------ SECTION 2 ---- Start of sweep.
    RETURN
 193 CONTINUE
C   * ------------------ SECTION 3 ---- Start of iz slab.
    RETURN
 194 CONTINUE
C   * ------------------ SECTION 4 ---- Start of iterations over slab.

    RETURN
1911 CONTINUE
C   * ------------------ SECTION 11---- After calculation of convection
C                        fluxes for scalars, and of volume
C                        fractions, but before calculation of
C                        scalars or velocities
    RETURN
 199 CONTINUE
C   * ------------------ SECTION 9 ---- Start of solution sequence for
C                                        a variable
    RETURN
1910 CONTINUE
C   * ------------------ SECTION 10---- Finish of solution sequence for
C                                        a variable
    RETURN
```

```
 195 CONTINUE
C   * ------------------- SECTION 5 ---- Finish of iterations over slab.
    RETURN
 196 CONTINUE
C   * ------------------- SECTION 6 ---- Finish of iz slab.

C       Find actual temp at cell WALT, wall ref for update of Tsource
C       point at IY=1, IX=1
     IF(.NOT.CALCV) THEN
      IF(IZ.EQ.NZ/2) THEN
       L0T1=L0F(LBNAME('TEM1'))
       IY=1
       IX=1
       I=IY+NY*(IX-1)
       TACT=F(L0T1+I)
      END IF
     END IF

C  Intergrate results
C  Flow rate of water and oil, taken at ix=NX/2

     IF(ISWEEP.EQ.LSWEEP) THEN

       L0U1=L0F(U1)
       L0AE=L0F(AEAST)
       IXX=NX/2
       DO IYY=1,FNW1
        I=IYY+NY*(IXX-1)
        FRW=FRW+(F(L0U1+I)*F(L0AE+I))
       END DO
       DO IYY=STOIL,STT-1
        I=IYY+NY*(IXX-1)
        FRO=FRO+(F(L0U1+I)*F(L0AE+I))
       END DO

C  Ave. temp of water
       L0T1=L0F(LBNAME('TEM1'))
       L0VL=L0F(VOL)
       DO IYY=1,FNW1
        DO IXX=1,NX
         I=IYY+NY*(IXX-1)
         ATW=ATW+(F(L0T1+I)*F(L0VL+I))
         TVOLW=TVOLW+F(L0VL+I)
     WRITE(14,*)'IY is ',IY
     WRITE(14,*)'IX is ',IX
     WRITE(14,*)'vol wat is ',F(L0VL+I)
        END DO
       END DO
C  Ave. temp of oil [NB: PRPS(oil)=65]
       L0PR=L0F(LBNAME('PRPS'))
       DO IYY=STOIL,FNOIL
        DO IXX=1,NX
         I=IYY+NY*(IXX-1)
         IF(F(L0PR+I).EQ.65) THEN
          ATO=ATO+(F(L0T1+I)*F(L0VL+I))
          TVOLO=TVOLO+F(L0VL+I)
         END IF
        END DO
       END DO

C  Ave temps in steel plate, used to calc Q through plates
```

```
 C  temp ref points
       IYA=9
       IYB=12
       L0YC=L0F(YG2D)
       L0AN=L0F(ANORTH)
       DO IXX=1,NX
         IA=IYA+NY*(IXX-1)
         IB=IYB+NY*(IXX-1)
         DT=F(L0T1+IB)-F(L0T1+IA)
         DY=F(L0YC+IB)-F(L0YC+IA)
         TOTQ=TOTQ+((DT/DY)*F(L0AN+IA))
         totan=totan+F(L0AN+IA)
        END DO

      END IF

      RETURN
  197 CONTINUE
C   * ------------------- SECTION 7 ---- Finish of sweep.

C  For Calculation of temperature only
C       Calculate improved TEM1 source.
      IF(.NOT.CALCV) THEN
        WRITE(14,*)'temp act is ',TACT
        WRITE(14,*)'temp ref is ',TREF
        TMPF=TACT/TREF
        WRITE(14,*)'tempfac is ',TMPF
        TSOUC=TSOUC*TMPF
C       Convergance monitoring.
        CALL MONTEMP(NSWP,TACT,TMPF,TSOUC)
        NSWP=NSWP+1
      END IF

      RETURN
  198 CONTINUE
C   * ------------------- SECTION 8 ---- Finish of time step.
      WRITE(14,*)' '
      WRITE(14,*)'^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ '
      WRITE(14,*)' '
      WRITE(14,*)'Results: '
      WRITE(14,*)' '
      WRITE(14,*)'Flow rate of water is ',FRW
      WRITE(14,*)'Flow rate of oil is ',FRO
      WRITE(14,*)'Ave temp of water is ',ATW/TVOLW
      WRITE(14,*)'Ave temp of oil is ',ATO/TVOLO
      WRITE(14,*)'Temp. source is ',TSOUC
      WRITE(14,*)' '

C     To calc q across the wall
      KSTEEL=48.0
      Q=KSTEEL*TOTQ
      WRITE(14,*)' Q is ',Q

C
      RETURN
C*****************************************************************
      END

C*****************************************************************
      SUBROUTINE MONTEMP(NSWP,TACT,TMPF,TSOUC)
```

```
      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG

C     Subroutine variables
      REAL MON,TACT,TMPF,TSOUC

C       Act. temp
       IF (NSWP.EQ.1) THEN
        OPEN (20,FILE='mon_tact.m')
        WRITE(20,*)'% MATLAB monitoring plotting:'
        WRITE(20,*)'% of TACT'
        WRITE(20,*)' '
        WRITE(20,*)'mon=[ '
       END IF
       MON=TACT
       WRITE(20,*) MON
       IF (ISWEEP.EQ.LSWEEP) THEN
        WRITE(20,*)'];'
        WRITE(20,*)' '
        WRITE(20,*)'plot (mon)'
        WRITE(20,*)'title ("Spot values of TACT ")'
        WRITE(20,*)'xlabel("sweep number")'
        WRITE(20,*)'ylabel("spot value")'
        CLOSE (20)
       END IF

C       Temp factor
       IF (NSWP.EQ.1) THEN
        OPEN (21,FILE='mon_tf.m')
        WRITE(21,*)'% MATLAB monitoring plotting:'
        WRITE(21,*)'% of TMPF'
        WRITE(21,*)' '
        WRITE(21,*)'mon=[ '
       END IF
       MON=TMPF
       WRITE(21,*) MON
       IF (ISWEEP.EQ.LSWEEP) THEN
        WRITE(21,*)'];'
        WRITE(21,*)' '
        WRITE(21,*)'plot (mon)'
        WRITE(21,*)'title ("Spot values of TMPF ")'
        WRITE(21,*)'xlabel("sweep number")'
        WRITE(21,*)'ylabel("spot value")'
        CLOSE (21)
       END IF
C       temp source
       IF (NSWP.EQ.1) THEN
        OPEN (22,FILE='mon_ts.m')
        WRITE(22,*)'% MATLAB monitoring plotting:'
        WRITE(22,*)'% of TSOUC'
        WRITE(22,*)' '
        WRITE(22,*)'mon=[ '
       END IF
       MON=TSOUC
       WRITE(22,*) MON
       IF (ISWEEP.EQ.LSWEEP) THEN
```

```
        WRITE(22,*)'];'
        WRITE(22,*)' '
        WRITE(22,*)'plot (mon)'
        WRITE(22,*)'title ("Spot values of TSOUC ")'
        WRITE(22,*)'xlabel("sweep number")'
        WRITE(22,*)'ylabel("spot value")'
        CLOSE (22)
      END IF
C      DPDX1
      IF (NSWP.EQ.1) THEN
        OPEN (23,FILE='mon_dpdx1.m')
        WRITE(23,*)'% MATLAB monitoring plotting:'
        WRITE(23,*)'% of DPDX1'
        WRITE(23,*)' '
        WRITE(23,*)'mon=[ '
      END IF
      MON=DPDX1
      WRITE(23,*) MON
      IF (ISWEEP.EQ.LSWEEP) THEN
        WRITE(23,*)'];'
        WRITE(23,*)' '
        WRITE(23,*)'plot (mon)'
        WRITE(23,*)'title ("Spot values of DPDX1 ")'
        WRITE(23,*)'xlabel("sweep number")'
        WRITE(23,*)'ylabel("spot value")'
        CLOSE (23)
      END IF
C      DPDX2
      IF (NSWP.EQ.1) THEN
        OPEN (24,FILE='mon_dpdx2.m')
        WRITE(24,*)'% MATLAB monitoring plotting:'
        WRITE(24,*)'% of DPDX2'
        WRITE(24,*)' '
        WRITE(24,*)'mon=[ '
      END IF
      MON=DPDX2
      WRITE(24,*) MON
      IF (ISWEEP.EQ.LSWEEP) THEN
        WRITE(24,*)'];'
        WRITE(24,*)' '
        WRITE(24,*)'plot (mon)'
        WRITE(24,*)'title ("Spot values of DPDX2 ")'
        WRITE(24,*)'xlabel("sweep number")'
        WRITE(24,*)'ylabel("spot value")'
        CLOSE (24)
      END IF

      RETURN
      END
C*****************************************************************
      SUBROUTINE MONVEL(NSWP,DPDX1,DPDX2)


      INCLUDE 'lp21/d_includ/satear'
      INCLUDE 'lp21/d_includ/grdloc'
      INCLUDE 'lp21/d_includ/grdear'
      INCLUDE 'lp21/d_includ/grdbfc'
      PARAMETER (NLG=100, NIG=200, NRG=200, NCG=100)
      COMMON/LGRND/LG(NLG)/IGRND/IG(NIG)/RGRND/RG(NRG)/CGRND/CG(NCG)
      LOGICAL LG
      CHARACTER*4 CG
```

```
C  Subroutine variables
      REAL MON,DPDX1,DPDX2

C       DPDX1
       IF (NSWP.EQ.1) THEN
        OPEN (23,FILE='mon_dpdx1.m')
        WRITE(23,*)'% MATLAB monitoring plotting:'
        WRITE(23,*)'% of DPDX1'
        WRITE(23,*)' '
        WRITE(23,*)'mon=[ '
       END IF
       MON=DPDX1
       WRITE(23,*) MON
       IF (ISWEEP.EQ.LSWEEP) THEN
        WRITE(23,*)'];'
        WRITE(23,*)' '
        WRITE(23,*)'plot (mon)'
        WRITE(23,*)'title ("Spot values of DPDX1 ")'
        WRITE(23,*)'xlabel("sweep number")'
        WRITE(23,*)'ylabel("spot value")'
        CLOSE (23)
       END IF
C       DPDX2
       IF (NSWP.EQ.1) THEN
        OPEN (24,FILE='mon_dpdx2.m')
        WRITE(24,*)'% MATLAB monitoring plotting:'
        WRITE(24,*)'% of DPDX2'
        WRITE(24,*)' '
        WRITE(24,*)'mon=[ '
       END IF
       MON=DPDX2
       WRITE(24,*) MON
       IF (ISWEEP.EQ.LSWEEP) THEN
        WRITE(24,*)'];'
        WRITE(24,*)' '
        WRITE(24,*)'plot (mon)'
        WRITE(24,*)'title ("Spot values of DPDX2 ")'
        WRITE(24,*)'xlabel("sweep number")'
        WRITE(24,*)'ylabel("spot value")'
        CLOSE (24)
       END IF

      RETURN
      END
C*****************************************************************************
```

REFERENCES

1) **Ackers, P.** (1958). *Resistance to fluids flowing in channels and pipes*. Hydraulics Research Paper No 1. H.M.S.O. London.

2) **Adachi, S.** (1964). *On the artificial strip roughness*. Bulletin 69, Disaster Prevention Research Institute. Kyoto University, Tokyo, Japan.

3) **Aguire-Pe, J. and Fyentes, R.** (1990). Resistance to flow in steep rough streams. *J. Hydr Eng*, ASCE, **116** (11), 1374-1387.

4) **Alhamid, A. A.** (1991). Boundary shear stress and velocity distributions in differentially roughened trapezoidal open channels. *Ph.D. Thesis*, Birmingham University.

5) **Ambrose, H. H.** (1956). *The effect of surface roughness on velocity distribution and boundary resistance*. University of Tennessee, Dept. of Civil Eng. Contract 811(03). Office of Naval Research, Dept. of the Navy.

6) **ASCE Task Force.** (1963). Report of the ASCE Task Force on friction factors in open channels. *J. Hydr Div*, ASCE, **89** (HY2), 97-143.

7) **Bathurst, J.** (1988). Velocity profile in high gradient, boulder-bed channels. *Proc. Int. Conf. on Fluvial Hydraulics*. 30 May-3 June Budapest, Hungary 29-33.

8) **Beale, S. B.** (1989). Laminar fully developed flow and heat transfer in an offset rectangular plate-fin surface. *PHOENICS Journal*, **3** (1), 1-38.

9) **Bergeron, N. E.** (1994). An analysis of flow velocity profiles, stream bed roughness and resistance to flow in natural gravel bed streams. *Hydraulic Engineering*, Vol 1, ASCE. Buffalo, New York. August 1-5, 692-696.

10) **Cebeci, T. and Chang, K. C.** (1978). Calculation of incompressible rough-wall boundary-layer flows. *AIAA J.*, **16** (7), 730-735.

11) **Celik, I. and Rodi, W.** (1984). Simulation of free surface effects in turbulent channel flows. *Physico-Chemical Hydrodynamics*, **5**, 217-227.

12) **Cetina, M. and Rajar, R.** (1993). Mathematical simulation of flow in a kayak racing channel. *Proc. 5th Int. Symposium on Refined Flow Modelling and Turbulence Measurements*, 7-10 September, Paris, 637-644.

13) **Chang, B. H. and Mills, A. F.** (1991). Turbulent flow in a tube with repeated rectangular rib roughness. *PHOENICS Journal*, **4** (3), 262-288.

14) **Christoph, G. H.** (1981). *Skin-friction and heat transfer calculations on ablating axisymmetic vehicles*. AIAA paper 81-0419.

15) **Christoph, G. H. and Plecher, R. H.** (1983). Prediction of rough wall skin friction and heat transfer. *AIAA J.*, **21** (4), 509-515.

16) **Clauser, F. H.** (1956). *The turbulent boundary layer*. Adv. Appl. Mech., Vol 4, 1-51.

17) **Colebrook, C. R.** (1939). Turbulent flow in pipes with particular reference to the transition region between the smooth and rough pipe laws. *J. Inst. Civil Eng.*, **11,** 133-156.

18) **Coleman, H. W., Hodge, B. K. and Taylor, R. P.** (1984). A re-evaluation of Schlichting's surface roughness experiment. *J. Fluids Eng.*, ASME, **106**, 60-65.

19) **Coles, D.** (1956). The law of the wake in the turbulent boundary layer. *J. Fluid. Mech.*, **1**, 191-226.

20) **Concentration, Heat and Momentum (CHAM) Ltd.** (1987). *The PHOENICS Reference Manual*, TR200b, Wimbledon, London.

21) **Coon, W. F.** (1994). Design of Mannings roughness coefficient study in New York state. *Hydraulic Engineering*, Vol 1, ASCE. Buffalo, New York. August 1-5, 677-681.

22) **Cornish, R. J.** (1928). Flow in pipe of rectangular cross section. *Proc. Royal Society,* **120** (A), 691-700.

23) **Crow, E. L., Davis, F. A. and Maxfield, M. W.** (1960) *Statistics manual*. Dover Publications Inc., New York.

24) **Cunge, J. A., Holly, F. M. and Verwey, A.** (1980). *Practical Aspects of Computational River Hydraulics*. Pitman, London.

25) **Dean, R. B.** (1978). Reynolds number dependence of skin friction and other bulk variables in two-dimensional rectangular duct flow. *J. Fluids Eng.*, **100**, 215-223.

26) **Economist magazine** (1993). And the waters prevailed. 17 July, 41-42.

27) **Einstein, H. A. and Barbarossa, N. L.** (1952). River channel roughness. *Trans. ASCE*, **117**, 1121-1146.

28) **Ferro, V. and Baiamonte, G.** (1994). Flow velocity profiles in gravel bed rivers. *J. Hydr Eng*, ASCE, **120** (1), 60-80.

29) **Ferro, V. and Giordano, G.** (1991). Experimental study of flow resistance in gravel bed rivers. *J. Hydr. Eng.*, ASCE, **117** (10), 1239-1246.

30) **Finley, P. J., Phoe, K. C. and Poh, C. J.** (1966). Velocity measurements in a thin turbulent water layer. *La Houille Blanche*, **21** (6), 713-721.

31) **Finson, M. L.** (1982). *A model for rough wall turbulent heating and skin friction.* AIAA Paper 82-0199.

32) **Finson, M. L. and Clarke, A. S.** (1980). *The effect of surface roughness character on turbulent re-entry heating.* AIAA Paper 80-1459.

33) **Finson, M. L. and Wu, P. K. S.** (1979). *Analysis of rough wall turbulent heating with application to blunted flight vehicles.* AIAA Paper 79-008.

34) **French, R. H.** (1986). *Open channel hydraulics.* McGraw-Hill, New York.

35) **Grace, A.** (1990). *Optimization Toolbox User Guide.* (For use with MATLAB). The Mathworks, Inc. MA, USA.

36) **Granville, P. S. and Taylor, D. W.** (1984). Mixing length formulations for turbulent boundary layers over arbitrary rough surfaces. *Proc. 9th US Air Force and Federal Republic of Germany Data Exchange Agreement Meeting - Viscous and Interacting Flow Field Effects*, Ohio, USA.

37) **Grass, A. J., Stuart, R. J. and Mansour-Tehrani, M.** (1991). Vortical structures and coherent motion in turbulent flow over smooth and rough boundaries. *Phil. Trans. R. Soc. Lond.* **336** (A), 35-65.

38) **Gessner, F. B.** (1981). Corner flow. *1980-81 AFOSR-HTTM Stanford Conference on Complex Turbulent Flows*, Vol 1, 182-212.

39) **Haaland, S. E.** (1983). Simple and explicit formulas for the friction factor in turbulent pipe flow. *J. Fluids Eng.*, ASME, **105**, 89-90.

40) **Hama, F.** (1954). Boundary layer characteristics for smooth and rough surfaces. *Trans Soc. Naval Architects and Marine Eng.*, **62**, 333-358.

41) **Haque, M. I. and Mahmood, K.** (1983). Analytical determination of form friction factor. *J. Hydr Eng*, ASCE, **109** (4), 590-610.

42) **Harlow, F. H. and Welch, J. E.** (1965). Numerical calculation of time dependent viscous incompressible flow of fluid with free surface. *Phys. Fluids.*, **8**, 2182-2189.

43) **Hirt, C. W. and Nichols, B. D.** (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Computational Physics,* **39**, 201-225.

44) **Hosni, M. H., Coleman, H. W. and Taylor, R. P.** (1993). Measurement and calculation of fluid dynamics characteristics of rough-wall turbulent boundary layer flows. *J. Fluids Eng.*, ASME, **115** (3), 383-388.

45) **Hubbard, L. C. and Thorne, R. C.** (1994). Flow Patterns in a mountain stream. *Hydraulic Engineering*, Vol 1, ASCE. Buffalo, New York. August 1-5, 737-741.

46) **Jayatilleke, C. L. V.** (1969). The influence of Prandtl number and surface roughness on the resistance of the laminar sublayer to momentum and heat transfer. *Prog. Heat and Mass Trans.* Vol 1, Pergamon Press.

47) **Jha, A. K., Akiyama, J. and Ura, M.** (1994). Modelling unsteady open-channel flows: modification to Beam and Warming scheme. *J. Hydr. Eng.*, ASCE, **120** (4), 461-475.

48) **Johns, B. and Soulsby, R. L. and Xing, J.** (1993). A comparison of numerical model experiments of free surface flow over topography with flume and field observations. *J. Hydr Research,* **31** (2), 215-228.

49) **Jones, W. P. and Launder, B. E.** (1972). The prediction of laminarization with a two equation model of turbulence. *Int. J. Heat and Mass Transfer*, **15**, 301-314.

50) **Keulegan, G. H.** (1938). Laws of turbulent flow in open channels. *J. Research of the National Bureau of Standards*, **21**, 707- 741.

51) **Kirksoz, M.** (1989). Turbulent velocity profiles for smooth and rough open channel flow. *J. Hydr Eng*, ASCE, **115** (12), 1543-1561.

52) **Kironoto, B. A. and Graf, W. H.** (1994). Turbulence characteristics in rough uniform open-channel flow. *Proc. Inst. Civ. Engrs. Wat., Marit. & Energy,* **106,** 333-344.

53) **Knight, D. W. and MacDonald, J. A.** (1979). Hydraulic resistance of artificial strip roughness. *J. Hydr. Div.*, ASCE, **105** (HY6), 675-690.

54) **Knight, D. W. and Patel, H. S.** (1985). Boundary shear in smooth rectangular ducts. *J. Hydr. Eng.*, ASCE, **111** (1), 29-47.

55) **Knight, D. W., Alhamid, A. A. I. and Yuen, K. W. H.** (1992). Boundary shear in differentially roughened trapezoidal channels. *2nd Int. Conf. Hydraulic and*

Environmental Modelling of Coastal, Estuarine and River Waters. Vol 1, Bradford, UK.

56) **Knight, D. W., Shiono, K. and Pirt, J.** (1989). Prediction of depth mean velocity and discharge in natural rivers with overbank flow. *Proc. 1st Int. Conf. Hydraulic and Environmental Modelling of Coastal, Estuarine and River Waters.* Bradford, UK, 419-428, Gower Technical.

57) **Kumar, S. and Roberson, J. A.** (1980). General algorithm for rough conduit resistance. *J. Hydr. Div.*, ASCE, **106** (HY11), 1745-1764.

58) **Lam, C. K. G. and Bremhorst, K.** (1981). A modified form of the k-ε model for predicting wall turbulence. *J. Fluids Eng.*, ASME, **103**, 456-460.

59) **Launder, B. E. and Spalding, D. B.** (1974). The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering,* **3**, 269-289.

60) **Lawler, D.** (1994). Tales of the river bank. *Geography Review*, **8** (1), 2-6.

61) **Leutheusser, H. J.** (1963). Turbulent flow in rectangular ducts. *J. Hydr. Div.*, ASCE, **89** (HY3), 1-17.

62) **Lewis, M. J.** (1975). An elementary analysis for predicting the momentum and heat transfer characteristics of a hydraulically rough surface. *Journal of Heat Transfer,* **97**, 249-254.

63) **Liggett, J. A.** (1994). *Fluid Mechanics*. McGraw-Hill. Singapore.

64) **Liu, J., Tominaga, A. and Nagao, M.** (1995). Numerical study of turbulent structure over strip roughness in open channel flow. *Hydra 2000,* Vol 1, *IAHR*, London, UK, Sept 11-15, 165-170.

65) **Marchand, J. D., Jarrett, R. D. and Jones, L. L.** (1984). Velocity profile, water-surface and bed material size for selected streams in Colorado. US Geological Survey Open Files report 84-733. *Proc. 9th US Air Force and Federal Republic of Germany Data Exchange Agreement Meeting - Viscous and Interacting flow Field effects*. Ohio USA.

66) **Massey, B. S.** (1989). *Mechanics of fluids*. 6th Edn. Van Nostrand Reinhold (international) Co. Ltd. London.

67) **Matthew, G. D.** (1990). The Colebrook-White equation - an oft cited result but neglected derivation? *Proc. Inst. Civ. Engrs.*, **89** (2), 39-45.

68) **Maxwell, T.** (1977). Numerical modelling of free-surface flows. *Ph.D. Thesis*, University of London.

69) **Miller, B. A. and Wenzel, L. L.** (1985). Analysis and simulation of low flow hydraulics. *J. Hydr. Eng.*, ASCE, **111** (12), 1429-1446.

70) **Monin A. S. and Yaglom A. M.** (1971). *Statistical fluid mechanics*. Vol 1, MIT Press.

71) **Moore, W. L.** (1951). An experimental investigation of the boundary layer development along a rough surface. *Ph. D. Thesis*, University of Iowa.

72) **Morris, H. M.** (1954). *Flow in rough conduits*. ASCE paper 2745.

73) **Neat, J. D., Jackson, L. D. and Falconer, R. N.** (1989). Mathematical modelling of the Rivers Aire and Calder. *Proc. 1st Int. Conf. Hydraulic and Environmental Modelling of Coastal, Estuarine and River Waters*. Bradford, UK, 463-473, Gower Technical.

74) **Nezu, I. and Rodi, W.** (1985). Experimental study on secondary currents in open channel flow. *Proc. XXI IAHR Congress*, Vol 2, Melbourne, Australia, 115-119.

75) **Nezu, I. and Rodi, W.** (1986). Open channel flow measurements with a laser Doppler anemometer. *J. Hydr. Eng.*, ASCE, **112** (5), 335-355.

76) **Nikuradse, J.** (1933). *Laws of flow in rough pipes*. National Advisory Committee for Aeronautics. Technical Memo 1292.

77) **Norris, L. H. and Reynolds, W. C.** (1975). *Turbulent channel flow with a moving wavy boundary*. Report No. FM-10, Stanford University, Dept. Mech. Eng.

78) **Paeschke, W.** (1937). Experimentelle untersuchungen zum rauhigkeits- und stabilitatsproblem in der bodennahahen huftschicht. *Beitr. Phys. fr. Atmos.*, **24**, 163-189.

79) **Parker, G. and Peterson, A. W.** (1980). Bar resistance of gravel bed streams. *J. Hydr. Div.*, ASCE, **106** (10), 1559-1575.

80) **Patankar, S. V.** (1975). *Numerical prediction of three-dimensional flows*. In Studies in convection: theory, measurement and applications, Vol 1, Ed. B. E. Launder, Academic Press, New York.

81) **Patankar, S. V.** (1980). *Numerical heat transfer and fluid flow*. Hemisphere Publishing Corporation. USA.

82) **Patankar, S. V.** (1981). A calculation procedure for two-dimensional elliptic situations. *Num. Heat Transfer*, **4**, 409-426.

83) **Patankar, S. V. and Spalding, D. B.** (1972). A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat and Mass Transfer*, **15**, 1787-1806.

84) **Patankar, S. V., Liu, C. H. and Sparrow, E. M.** (1977). Fully developed flow and heat transfer in ducts having streamwise-periodic variations of cross-sectional area. *J. Heat Transfer*, **99**, 180-186.

85) **Perry, A. E. and Joubert. P. N.** (1963). Rough wall boundary layers in adverse pressure gradients. *J. Fluid. Mech.*, **17**, 193-211.

86) **Perry, A. E., Scofield, W. H. and Joubert, P. N.** (1969). Rough wall turbulent boundary layers. *J. Fluid. Mech.*, **37** (2), 383-413.

87) **Peyret, R. and Taylor, T. D.** (1986). *Computational methods for fluid flow*. Springer-Verlag, New York.

88) **Prakash, C.** (1985). *Computation of laminar flow and heat transfer in a corrugated duct passage*. Phoenics Demonstration Report. Ref: PDR/CHAM NA/6. Concentration, Heat and Momentum Ltd. Wimbledon, London.

89) **Pritchard, V.** (1996). The new flood warning system. *Proc. Inst. Civ. Engrs.*, *Mun. Engr.*, **115**, 20-27.

90) **Pyle, R. and Novak, P.** (1981). Coefficient of friction in conducts with large roughness. *J. Hydr. Research,* **19** (2), 119-140.

91) **Raju, K. G. R. and Garde, R. J.** (1970). Resistance to flow over two-dimensional strip roughness. *J. Hyd. Div.*, ASCE, **96** (HY3), 815-834.

92) **Rhodes D. G. and Knight, D. W.** (1994). Distribution of shear force on the boundary of a smooth rectangular duct. *J. Hydr. Eng.*, ASCE, **120** (7), 787-807.

93) **Rhodes, D. G.** (1991). An experimental investigation of the mean flow structure in wide ducts of simple rectangular and compound trapezoidal cross-section, examining in particular zones of high lateral shear. *Ph.D. Thesis,* Birmingham University.

94) **Roberson, J. A.** (1970). Flow in conduits with low roughness concentration. *J. Hydr. Div.*, ASCE, **96** (HY4), 941-957.

95) **Rodi, W.** (1984). *Turbulence models and their applications in hydraulics - a state of the art review*. 2nd Edn. IAHR. Delft, The Netherlands.

96) **Rodi, W.** (1991). Experience with two-layer models combining the k-ε model with a one-equation model near the wall. *29th Aerospace Science Meeting, AIAA.* January 7-10, Nevada, USA.

97) **Rosten, H. I. and Spalding, D. B.** (1987). *The PHOENICS Equations*, TR99, Concentration, Heat and Momentum Ltd. Wimbledon, London.

98) **Samuels, P. G.** (1997). Private communication.

99) **Sayre, W. W. and Albertson, M. L.** (1963). Roughness spacing in rigid open channels. *J. Hydr. Div.*, ASCE, **128** (1), 343-373.

100) **Schlichting, H.** (1936). Experimentelle untersuchungen zum rauhigkoitsproblem. *Ingeniour-Archiv.*, **VII** (1) 1-34. English translation by the National Advisory Committee for Aeronautics, Technical memo. No 823, 1937.

101) **Schlichting, H.** (1960). *Boundary layer theory.* 2nd English Edn. McGraw-Hill. New York.

102) **Schmitz, G., Seus, G. J. and Czirwitzky, H. J.** (1983). Simulating two-dimensional flood flow. *Int. Conf. Hydraulic Aspects of Floods and Flood Control.* BHRA, London. 195-206.

103) **Senior, A. K.** (1992). Effects of inlet characteristics on the flow reattachment behind a rearward facing step. *Undergraduate project.* Nottingham University.

104) **Senior, A. K.** (1994). An investigation of resistance due to depth scale roughness with a view to application in computational river models. *M.Phil./Ph.D. Transfer Thesis*, Cranfield University.

105) **Senior, A. K. and Aroussi, A.** (1992). Correlation between reattachment length downstream of a backstep and the structure of the incident flow. *13th Symp. Turbulence.* Missouri, USA.

106) **Shen, H. W., Fehlman, H. M. and Mendoza, C.** (1990). Bed form resistance in open channels. *J. Hydr Eng*, ASCE, **116** (6), 799-813.

107) **Simpson, R. L.** (1972). A generalised correlation of roughness density effects on the turbulent boundary layer. *AIAA J.*, **11** (2), 242-244.

108) **Spalding D. B.** (1980). *Mathematical modelling of fluid mechanics, heat transfer and chemical-reaction processes.* A lecture course. CFDU Report HTS/80/1.

109) **Streeter V. and Chu, H.** (1949). *Fluid flow and heat transfer in artificially roughened pipes.* Armour Research Foundation, Final Report, Project 4918.

110) **Tayor, R. P., Coleman, H. W. and Hodge, B. K.** (1985). Prediction of turbulent rough wall skin friction using a discrete element approach. *J. Fluids Eng.*, ASME, **107**, 251-257.

111) **Tennekes, H. and Lumley, J. L.** (1987). *A first course in turbulence*. MIT Press, USA.

112) **Tritton, D. J.** (1977). *Physical fluid dynamics*. Van Nostrand Reinhold (international) Co. Ltd. London.

113) **Vivash, R., Ottosen, O., Janes, M. D. and Sørensen, H. V.** (1996). River restoration: benefits for integrated catchment management, the river restoration works and related practical aspects. *River Restoration Conference '96: The Physical Dimension*, Silkeborg, Denmark, 9-13 September. (In print).

114) **Wei, T. and Willmarth, W. W.** (1989). Reynolds-number effects on the structure of a turbulent channel flow. *J. Fluid Mech.*, **204**, 57-95.

115) **Willcox, D.C.** (1994). *Turbulence modeling for CFD*. Griffin Printing, California, USA.

116) **Young, A. D. and Maas, J. N.** (1936). *The behaviour of a pitot tube in a transverse total pressure gradient*. Aeronautical Research Committee Reports and Memoranda, No. 1770.