A finite volume method for steady state 2D shallow water flows

J.G. Zhou and I.M. Goodwill
Department of Civil Engineering, University of Leeds, Leeds, UK

Received August 1994
Revised March 1996

Nomenclature

A	=	Area of control volume face
Cb	=	Bed friction coefficient
Cw	=	Friction factor at wall of channel
D_u^1, D_u^2, D_z^u	=	Coefficients in equation (25)
D_v^1, D_v^2, D_z^v	=	Coefficients in equation (26)
f_e, f_n, f_s	=	Geometric interpolation factors
g	=	Gravitational acceleration (= 9.81 m^2/s)
H_u, H_v	=	Terms related to neighbouring velocities in equations (25) and (26)
h	=	Water depth
k	=	Von Kármán constant (= 0.4)
$\vec{n}$	=	Unit vector normal to the cell surface outward
Q_e, Q_w, Q_n, Q_s	=	Discharges through the cell faces, e, w, n, s
S_x, S_y	=	Source terms in equations (23) and (24)
$\vec{U}_{wn}, \vec{U}_{ws}$	=	Velocity vectors normal and parallel to the wall respectively
u, v	=	Depth-averaged x and y direction components of the velocity
u_*	=	Shear velocity
$\vec{V}$	=	Velocity vector
x, y	=	Cartesian co-ordinates
z	=	Elevation of channel bed above arbitrary datum
[[A, B]]	=	Maximum of A and B
α	=	Under-relaxation factor for velocity
α_h	=	Relaxation factor for depth
ξ, η	=	Axes of curvilinear co-ordinates
ε	=	Depth-averaged diffusion coefficient
ρ	=	Fluid density
$\vec{\tau}_w$	=	Shear stress vector at the wall
τ_{bx}, τ_{by}	=	Bed shear stresses in x and y directions
Ω	=	Volume of the control volume
∇	=	Vector differential operator

Superscripts

*	=	Value of the quantity from the latest iteration
**	=	Value of the quantity from the previous iteration
'	=	Quantity correction

Subscripts

E, W, N, S, P	=	Grid points (Figure 2)
e, w, n, s	=	Cell face points (Figure 2)

Introduction

If the wind and Coriolis forces are neglected then the governing equations for steady state two dimensional shallow flow in an open channel can be written in Cartesian co-ordinates as

$$\frac{\partial(uh)}{\partial x} + \frac{\partial(vh)}{\partial y} = 0 \tag{1}$$

International Journal of Numerical Methods for Heat & Fluid Flow
Vol. 7 No. 1, 1997, pp. 4-23.

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial x}-\frac{\tau_{bx}}{\rho h}+\frac{\partial}{\partial x}\left(\varepsilon\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\varepsilon\frac{\partial u}{\partial y}\right) \quad (2)$$

$$u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=-\frac{1}{\rho}\frac{\partial p}{\partial y}-\frac{\tau_{by}}{\rho h}+\frac{\partial}{\partial x}\left(\varepsilon\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial y}\left(\varepsilon\frac{\partial v}{\partial y}\right) \quad (3)$$

where x and y are the co-ordinates in the longitudinal and transverse directions respectively, u and v are the depth averaged velocity components in the x and y directions, h is the water depth, ρ is the fluid density, ε is the depth averaged diffusion coefficient, τ_{bx} and τ_{by} are the bed shear stresses in the x and y directions and p is the hydrostatic pressure which is defined by equation (4)

$$p=\rho g(h+z) \quad (4)$$

where g is gravitational acceleration and z is the elevation of the channel bed above an arbitrary datum. The bed shear stresses, τ_{bx} and τ_{by} can be described using the depth averaged velocities as follows

$$\tau_{bx}=\rho C_b u\sqrt{u^2+v^2} \qquad \tau_{by}=\rho C_b v\sqrt{u^2+v^2} \quad (5)$$

where C_b is the bed friction coefficient. If a linear distribution of the shear stress is assumed then the depth averaged diffusion coefficient, ε, is given by

$$\varepsilon=\frac{k}{6}u_* h \quad (6)$$

where k is the von Kármán constant (= 0.4); and u_* is the shear velocity which can be derived from equation (5) as

$$u_*=\sqrt{C_b(u^2+v^2)} \quad (7)$$

The momentum equations (2) and (3) can be written in weak conservation form as[1]

$$\frac{\partial(uuh)}{\partial x}+\frac{\partial(vuh)}{\partial y}=-\frac{h}{\rho}\frac{\partial p}{\partial x}-\frac{\tau_{bx}}{\rho}+\frac{\partial}{\partial x}\left(h\varepsilon\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(h\varepsilon\frac{\partial u}{\partial y}\right) \quad (8)$$

$$\frac{\partial(uvh)}{\partial x}+\frac{\partial(vvh)}{\partial y}=-\frac{h}{\rho}\frac{\partial p}{\partial y}-\frac{\tau_{by}}{\rho}+\frac{\partial}{\partial x}\left(h\varepsilon\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial y}\left(h\varepsilon\frac{\partial v}{\partial y}\right) \quad (9)$$

(Here the z is not taken into account, because it is unchanged for a fixed bed.)

By substituting equations (4) and (5) into equations (8) and (9), the two momentum equations with three primitive variables u, v and h in strong conservation form can be written as follows:

$$\frac{\partial(uuh)}{\partial x}+\frac{\partial(vuh)}{\partial y} = -g\frac{\partial}{\partial x}(\frac{h^2}{2})-gh\frac{\partial z}{\partial x}-C_b u\sqrt{u^2+v^2} + \frac{\partial}{\partial x}\left(h\varepsilon\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(h\varepsilon\frac{\partial u}{\partial y}\right) \quad (10)$$

$$\frac{\partial(uvh)}{\partial x}+\frac{\partial(vvh)}{\partial y} = -g\frac{\partial}{\partial y}(\frac{h^2}{2})-gh\frac{\partial z}{\partial y}-C_b v\sqrt{u^2+v^2} + \frac{\partial}{\partial x}\left(h\varepsilon\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial y}\left(h\varepsilon\frac{\partial v}{\partial y}\right) \quad (11)$$

The continuity equation remains unchanged, since it is already written in strong conservation form.

The set of governing flow equations, (1), (10) and (11), together with the bed shear velocity equation (7) can be written in vector notation as

$$\nabla\cdot(h\vec{V})=0 \quad (12)$$

$$\nabla\cdot(h\vec{V}\vec{V})=-\frac{g}{2}\nabla h^2-gh\nabla z-C_b|\vec{V}|\vec{V}+\nabla\cdot(h\varepsilon\nabla\vec{V}) \quad (13)$$

$$u_*=\sqrt{C_b}|\vec{V}| \quad (14)$$

where $\vec{V}$ is the depth-averaged velocity vector.

Equations (12) and (13) contain three primitive variables (u, v and h) and are in strong conservation form whereas equations (1), (8) and (9) contain four variables (u, v, h and p) and are in a weak conservation form. In the weak conservation form the depth, h, is related to the pressure, p, through equation (4). In the strong conservation form the velocity is related to the depth by both linear and quadratic terms, whereas in the weak conservation form the velocity is linearly proportional to depth.

The majority of previous solutions to this problem have been based on the solution of equations (1), (2) and (3) using a straightforward finite difference approach[2]. Early attempts to use the control volume approach assumed that the change in water depth was sufficiently small so that the continuity equation could be written as

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0 \quad (15)$$

While this may be an acceptable approximation for some flow situations, it is not so in a channel bend, where the transverse variation in depth affects the whole of the velocity field. In a recent paper, Zhou[3] studied the velocity depth coupling and proposed a SIMPLE-like scheme to preserve the coupling using a solution which incorporated the control volume method. The principal advantage of this approach is that the solution maintains the strong conservation form of the equations, thus considerably improving the accuracy

of the solution. However there is a disadvantage in that the solution is made on a staggered grid which makes the methodology difficult to apply to problems with complex boundary geometries.

Mathematical model

Grid generation

Since the discretized equations contain only first order terms, neither orthogonality nor smoothness of the grid lines are particularly important hence a simple algebraic method can be used to generate the grid lines. This has several advantages, the principal ones being that the grid can quickly and easily be generated by the computer and the points can readily be placed where they are most needed. The method used here is due to Demirdzic[1] and can be written as

$$x_{i,j} = x_{i,1} + \frac{l_j}{L}(x_{i,N_j} - x_{i,1}) \qquad y_{i,j} = y_{i,1} + \frac{l_j}{L}(y_{i,N_j} - y_{i,1}) \quad (16)$$

where N_i and N_j denote the numbers of grid points in the ξ and η directions. This is shown diagrammatically in Figure 1.

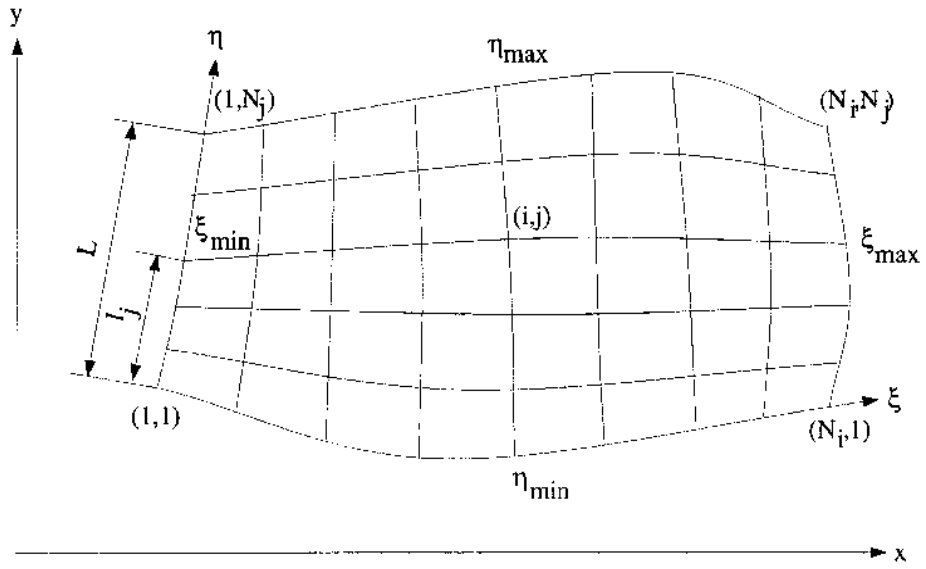

Figure 1.
Grid generation

Discretized flow equations in the computational domain

The strong conservation form of the equations in vector notation, equations (12) and (13), was used for discretization since this guarantees conservation of the components of the vectors. The continuity equation can be integrated over the area of the control volume (shaded area in Figure 2) to give

$$\int\int \nabla \cdot (h\vec{V}) d\Omega = 0 \quad (17)$$

where Ω is the area of the solution domain.

By applying the Gauss divergence theorem to this equation it can be integrated around the boundary of the domain to yield

$$\oint_A \vec{n} \cdot (h\vec{V}) dA = 0 \tag{18}$$

where $\vec{n}$ is the outwardly directed unit vector normal to the surface of the cell and A is the length of the closed line of the cell. By expanding equation (18) the following is established

$$\int_{A_e} \vec{n} \cdot (h\vec{V}) dA + \int_{A_n} \vec{n} \cdot (h\vec{V}) dA + \int_{A_w} \vec{n} \cdot (h\vec{V}) dA + \int_{A_s} \vec{n} \cdot (h\vec{V}) dA = 0 \tag{19}$$

which can be much more simply written as

$$Q_e - Q_w + Q_n - Q_s = 0 \tag{20}$$

where Q_e, Q_w, Q_n and Q_s are the discharges through the cell faces e, w, n and s, respectively and A_e, A_n, A_w and A_s are the areas of corresponding cell faces e, n, w and s. Clearly equation (20) is a discretized form of the law of conservation of mass with constant fluid density.

The momentum equation can be similarly integrated over the control volume hence equation (13) becomes

$$\begin{aligned} \int\int \nabla \cdot (h\vec{V}\vec{V}) d\Omega &= -\frac{g}{2} \int\int \nabla h^2 d\Omega - g \int\int h \nabla z d\Omega \\ &- C_b \int\int |\vec{V}| \vec{V} d\Omega + \int\int \nabla \cdot (h\varepsilon \nabla \vec{V}) d\Omega \end{aligned} \tag{21}$$

By applying the Gauss divergence theorem, the equation can be integrated around the control volume to yield

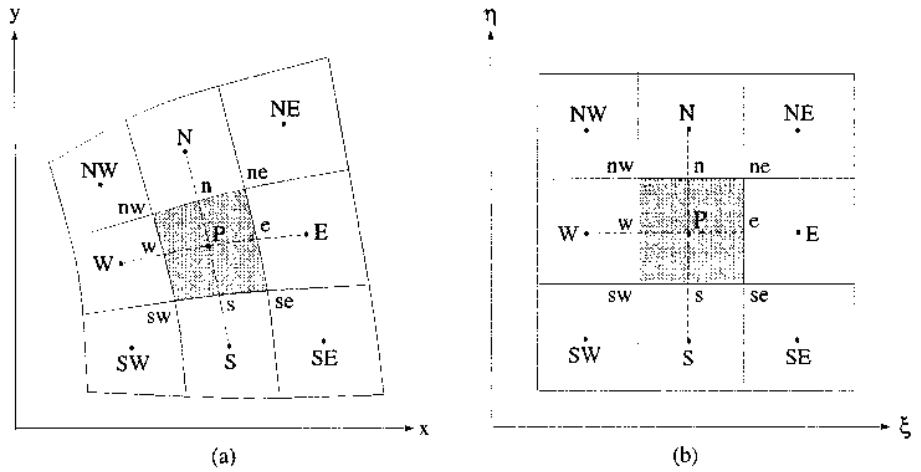

Figure 2.
Control volume in (a) physical domain, and (b) computational domain

$$\oint_A \vec{n}\cdot(h\vec{V}\vec{V})dA = -\frac{g}{2}\Omega_P(\nabla h^2)_P - gh_P\Omega_P(\nabla z)_P - C_b\Omega_P|\vec{V}_P|\vec{V}_P + \oint_A \vec{n}\cdot(h\varepsilon\nabla\vec{V})dA \quad (22)$$

If an upwind scheme is used for the convection terms, equation (22) can readily be discretized in the x and y directions. In the x direction the equation becomes

$$a_P u_P = a_E u_E + a_N u_N + a_W u_W + a_S u_S + S_x - \frac{g}{2}[A^1_{xP}(h^2_e - h^2_w) + A^2_{xP}(h^2_n - h^2_s)] - gh_P[A^1_{xP}(z_e - z_w) + A^2_{xP}(z_n - z_s)] \quad (23)$$

where A^i_{xP} and A^i_{yP} are the Cartesian components of the area vector at grid point P in the physical domain, i.e. $\vec{A^1_P} = \overline{sn}\,\vec{n}_1$ and $\vec{A^2_P} = \overline{we}\,\vec{n}_2$ (see Figure 2); $\vec{n}_1$, $\vec{n}_2$ are the unit vectors normal to the grid lines $\overline{sn}$ and $\overline{we}$ outward respectively and

$$a_E = [[0, -Q_e]] + \frac{(A^1_{xe})^2 + (A^1_{ye})^2}{\Omega_e}(h\varepsilon)_e, \qquad a_N = [[0, -Q_n]] + \frac{(A^2_{xn})^2 + (A^2_{yn})^2}{\Omega_n}(h\varepsilon)_n$$

$$a_W = [[0, Q_w]] + \frac{(A^1_{xw})^2 + (A^1_{yw})^2}{\Omega_w}(h\varepsilon)_w, \qquad a_S = [[0, Q_s]] + \frac{(A^2_{xs})^2 + (A^2_{ys})^2}{\Omega_s}(h\varepsilon)_s$$

$$a_P = a_E + a_N + a_W + a_S + C_b\Omega_P\sqrt{u_P^2 + v_P^2}$$

$$S_x = \frac{A^1_{xe}A^2_{xe} + A^1_{ye}A^2_{ye}}{\Omega_e}(h\varepsilon)_e(u_{ne} - u_{se}) + \frac{A^1_{xn}A^2_{xn} + A^1_{yn}A^2_{yn}}{\Omega_n}(h\varepsilon)_n(u_{ne} - u_{nw}) - \frac{A^1_{xw}A^2_{xw} + A^1_{yw}A^2_{yw}}{\Omega_w}(h\varepsilon)_w(u_{nw} - u_{sw}) - \frac{A^1_{xs}A^2_{xs} + A^1_{ys}A^2_{ys}}{\Omega_s}(h\varepsilon)_s(u_{se} - u_{sw})$$

Here $[[A, B]]$ denotes the maximum of A and B, and the subscripts E, W, N and S signify the grid points at east, west, north and south with reference to the central grid point P (see Figure 2).

It may be noted that all the terms in equation (23) have a clear physical meaning, for example, the term, $A^1_{xP}(h^2_e - h^2_w)$, means the pressure force exerted on the control volume in the x direction.

Similarly in the y direction it becomes

$$a_P v_P = a_E v_E + a_N v_N + a_W v_W + a_S v_S + S_y - \frac{g}{2}[A^1_{yP}(h^2_e - h^2_w) + A^2_{yP}(h^2_n - h^2_s)] - gh_P[A^1_{yP}(z_e - z_w) + A^2_{yP}(z_n - z_s)] \quad (24)$$

where

$$S_y = \frac{A^1_{xe}A^2_{xe} + A^1_{ye}A^2_{ye}}{\Omega_e}(h\varepsilon)_e(v_{ne} - v_{se}) + \frac{A^1_{xn}A^2_{xn} + A^1_{yn}A^2_{yn}}{\Omega_n}(h\varepsilon)_n(v_{ne} - v_{nw}) - \frac{A^1_{xw}A^2_{xw} + A^1_{yw}A^2_{yw}}{\Omega_w}(h\varepsilon)_w(v_{nw} - v_{sw}) - \frac{A^1_{xs}A^2_{xs} + A^1_{ys}A^2_{ys}}{\Omega_s}(h\varepsilon)_s(v_{se} - v_{sw})$$

The terms in S_x and S_y such as $(u_{ne} - U_{se})$ may be calculated as

$$u_{ne} - u_{se} = f_e(u_n - u_s)_E + (1 - f_e)(u_n - u_s)_P$$

Here

$$u_n = f_n u_N + (1 - f_n)u_P \qquad u_s = f_s u_P + (1 - f_s)u_S$$

where f_e, f_n and f_s are the geometric interpolation factors defined by

$$f_e = \frac{\overline{Pe}}{\overline{Pe} + \overline{eE}} \qquad f_n = \frac{\overline{Pn}}{\overline{Pn} + \overline{nN}} \qquad f_s = \frac{\overline{Ss}}{\overline{Ss} + \overline{sP}}$$

The details of the computation for all the geometric quantities such as Ω and $\vec{A}^i$ follow standard methods which can be found in many texts[4].

It will be noticed that all the coefficients in equations (23) and (24) satisfy the Scarborough condition and the four basic rules of Patankar[5]; hence the convergence of the discretized equations should be achieved without difficulty.

Coupling of velocity and depth

If an under-relaxation factor α ($0 < \alpha < 1.0$) is introduced into the discretized momentum equations then

$$u_P = H_u - D^1_u(h^2_e - h^2_w) - D^2_u(h^2_n - h^2_s) - D^u_z h_P + (1 - \alpha)u^{**}_P \quad (25)$$

$$v_P = H_v - D^1_v(h^2_e - h^2_w) - D^2_v(h^2_n - h^2_s) - D^v_z h_P + (1 - \alpha)v^{**}_P \quad (26)$$

where u^{**}_P and v^{**}_P are the values of u_P and v_P from the previous iteration and the other newly introduced variables are defined as

$$H_u = \alpha \frac{(a_E u_E + a_W u_W + a_N u_N + a_S u_S + S_x)}{a_P}$$

$$H_v = \alpha \frac{(a_E v_E + a_W v_W + a_N v_N + a_S v_S + S_y)}{a_P}$$

$$D_u^1 = \frac{\alpha g}{2a_P} A_{xP}^1 \qquad D_u^2 = \frac{\alpha g}{2a_P} A_{xP}^2 \qquad D_v^1 = \frac{\alpha g}{2a_P} A_{yP}^1$$

$$D_z^u = \frac{\alpha g}{a_P}[A_{xP}^1(z_e - z_w) + A_{xP}^2(z_n - z_s)]$$

$$D_z^v = \frac{\alpha g}{a_P}[A_{yP}^1(z_e - z_w) + A_{yP}^2(z_n - z_s)]$$

It is noted that in equations (25) and (26) the velocity components (u and v) are no longer linearly proportional to the depth, but are proportional to the square of the depth (h^2) and the depth (h) itself, whereas the velocity field is linearly proportional to the pressure field in the Navier-Stokes equations for general flows. Moreover all the coefficients in these equations are functions of both velocity and depth, while the coefficients for general flows are constants or functions of velocity only. These characteristics make the use of solution schemes based on a SIMPLE-like approach fail for shallow water flows, thus a modification to the method is required, this is achieved as a two part process, a predictor stage followed by a corrector one, these are described in turn below.

Predictor stage: an initial estimate of the depth is made and input into equations (27) and (28) to yield the following equations

$$u_P^* = H_u^* - (D_u^1)^*(h_e^{*2} - h_w^{*2}) - (D_u^2)^*(h_n^{*2} - h_s^{*2}) - (D_z^u)^* h_P^* + (1-\alpha)u_P^{**} \quad (27)$$

$$v_P^* = H_v^* - (D_v^1)^*(h_e^{*2} - h_w^{*2}) - (D_v^2)^*(h_n^{*2} - h_s^{*2}) - (D_z^v)^* h_P^* + (1-\alpha)v_P^{**} \quad (28)$$

where h^* is the estimate of depth and u_p^* and v_p^* are the velocities evaluated from h^*. When a. conventional linear interpolation method was used to solve equations (27) and (28) it was found that the depths oscillated. This was overcome by using the momentum interpolation approach[6] to determine the velocities and discharges at the cell faces. The assumption underlying this approach is that, with the exception of the terms containing differences in depth normal to the cell face, the variation in the values of all the terms between the grid points is assumed to be linear. The terms containing the differences in depth normal to the cell faces are calculated directly by using the differences between the values at the adjacent grid points.

Thus the velocities at the face of cell e can be written as

$$u_e^* = (H_u^1)_e^* - (D_u^1)_e^*(h_E^{*2} - h_P^{*2}) - (D_z^u)_e^* h_e^* + (1-\alpha)u_e^{**} \tag{29}$$

$$v_e^* = (H_v^1)_e^* - (D_v^1)_e^*(h_E^{*2} - h_P^{*2}) - (D_z^v)_e^* h_e^* + (1-\alpha)v_e^{**} \tag{30}$$

where

$$(H_u^1)^* = H_u^* - (D_u^2)^*(h_n^{*2} - h_s^{*2}) \qquad (H_v^1)^* = H_v^* - (D_v^2)^*(h_n^{*2} - h_s^{*2})$$

In these equations the under relaxation factor, α, is used in the same manner as for equation (25) and equation (26) such that the solution to the equations is independent of the value of the under-relaxation factor chosen[7].

The coefficients for the velocities at cell face e such as u_e^* are interpolated as follows

$$\frac{1}{(a_P)_e} = \frac{f_e}{(a_P)_E} + \frac{(1-f_e)}{(a_P)_P} \qquad (H_u^1)_e = f_e(H_u^1)_E + (1-f_e)(H_u^1)_P$$

and similarly for cell face n.

$$u_n^* = (H_u^2)_n^* - (D_u^2)_n^*(h_N^{*2} - h_P^{*2}) - (D_z^u)_n^* h_n^* + (1-\alpha)u_n^{**} \tag{31}$$

$$v_n^* = (H_v^2)_n^* - (D_v^2)_n^*(h_N^{*2} - h_P^{*2}) - (D_z^v)_n^* h_n^* + (1-\alpha)v_n^{**} \tag{32}$$

where

$$(H_u^2)^* = H_u^* - (D_u^1)^*(h_e^{*2} - h_w^{*2}) \qquad (H_v^2)^* = H_v^* - (D_v^1)^*(h_e^{*2} - h_w^{*2})$$

The velocities at cell faces w and s can be similarly expressed and the discharges can be calculated, for example the estimate of the discharge through cell face e, Q_e, can be expressed as

$$Q_e^* = (h^*\vec{V}^*)_e \cdot \vec{A}_e^1 = h_e^* u_e^* A_{xe}^1 + h_e^* v_e^* A_{ye}^1 \tag{33}$$

The depth h^* and velocities u^* and v^* do not normally satisfy the continuity equation therefore it is necessary to correct them in order that the continuity equation is satisfied. This is achieved in the corrector stage of the solution.

Corrector stage: if the depth correction is h' then the discharges corresponding to this correction can be determined, for example the correction to Q_e can be written as

$$Q_e = Q_e^* - h_e^*(h_E^* + h_P^*)[(D_u^1)_e^* A_{xe}^1 + (D_v^1)_e^* A_{ye}^1](h'_E - h'_P) \tag{34}$$

It will be noted that the term containing the difference in depths is the only one affected by the correction, the effect of the correction on the other terms is ignored. This omission may affect the speed of convergence but will have no effect on the final solution. The depth correction equation can be obtained by substituting all the discharges obtained from expressions similar to Equation (34) into the continuity equation (see Equation (20)) to yield

$$c_P h'_P = c_E h'_E + c_W h'_W + c_N h'_N + c_S h'_S + c_o \tag{35}$$

where

$$c_E = h_e^*(h_E^* + h_P^*)[(D_u^1)_e^* A_{xe}^1 + (D_v^1)_e^* A_{ye}^1]$$

$$c_W = h_w^*(h_P^* + h_W^*)[(D_u^1)_w^* A_{xw}^1 + (D_v^1)_w^* A_{yw}^1]$$

$$c_N = h_n^*(h_N^* + h_P^*)[(D_u^2)_n^* A_{xn}^2 + (D_v^2)_n^* A_{yn}^2]$$

$$c_S = h_s^*(h_P^* + h_S^*)[(D_u^2)_s^* A_{xs}^2 + (D_v^2)_s^* A_{ys}^2]$$

$$c_P = c_E + c_W + c_N + c_S \qquad c_o = Q_w^* - Q_e^* + Q_s^* - Q_n^*$$

Equation (35) is solved to yield the depth correction h' which is then used to update the depth and velocities as follows

$$h = h^* + \alpha_h h' \tag{36}$$

$$u_P = u_P^* - D_u^1(h_e + h_w)(h_e' - h_w') - D_u^2(h_n + h_s)(h_n' - h_s')] - D_z^u h_P' \tag{37}$$

$$v_P = v_P^* - D_v^1(h_e + h_w)(h_e' - h_w') - D_v^2(h_n + h_s)(h_n' - h_s') - D_z^v h_P' \tag{38}$$

where α_h is a relaxation factor for depth ($0 < \alpha_h \leq 1.0$).

Solution procedure

The set of discretized equations can be solved by a variety of methods, depending on the type of computer available. Normally an iteration method is used to save computer time and space. Without losing generality the iterative solution procedure can be split into a series of steps as follows:

1. Make an estimate of the initial depth field h^*.
2. Solve equations (27) and (28) with $\alpha \approx 0.5$ to obtain the velocities u^* and v^*.
3. Solve equation (35) to obtain the depth correction hence the new estimate of depth from equation (36).
4. Correct velocities u^* and v^* by using equations (37) and (38) to obtain u and v.
5. Repeat the procedure from step 2 until convergence is reached.

Boundary conditions

The determination of an open channel surface profile is the solution of a boundary value problem hence the boundary conditions must be determined

accurately, otherwise an incorrect solution will be obtained. If secondary flows are ignored then the flow is essentially one dimensional and the boundary conditions can be defined at one end of the channel as the downstream end for subcritical flow and the upstream end for supercritical flow. Once secondary flows are taken into account these simple conditions are insufficient to fully describe the boundary conditions since the flow situation at the side walls must also be considered. The boundary conditions for secondary flow have to be known for the full length of the channel and can be defined as follows

- At inlet: the discharge Q or velocity u is specified; $v = 0$; and h is defined for supercritical flow.
- At outlet: $\partial u/\partial s = 0$; $\partial v/\partial s = 0$; and h is defined for subcritical flow.
- At the side wall: the velocity vector normal to the wall is set to zero, i.e. $\vec{U}_{wn} = 0$; and a slip velocity vector $\vec{U}_{ws}$ which is parallel to the wall is specified through the following equation:

$$\frac{\vec{\tau}_w}{\rho} = \varepsilon \frac{\partial \vec{U}_{ws}}{\partial n} = -C_w |\vec{U}_{ws}| \vec{U}_{ws} \tag{39}$$

in which $\vec{\tau}_w$ is the shear stress vector at the wall, C_w is friction factor at the wall and n is the co-ordinate perpendicular to the wall.

Verification and application

This model is tested by analysing the flow pattern in a strongly curved channel. This problem has the advantage of having been extensively studied in the past, hence both experimental and analytical results are readily available for comparison; furthermore the secondary flows generated in the bend provide quite a severe test for any numerical model. For comparative purposes, the experimental results provided by Rozovski[8] are used, the data for his Run 8 are given in Table I. Figure 3 shows the principal dimensions of the experimental apparatus.

The grid generation

Since the geometry of the channel is regular then it is a straightforward matter to generate the computational grid. A fine grid is required in those parts of the solution domain where there is a large variation in either the geometry or the dependent variables. For this particular problem a fine mesh of grid points is required in the entry and exit areas and close to the side walls. Two types of grid were used, one semi-uniform where grid lines were spaced closely together near the channel walls and uniformly in the longitudinal direction, this is shown in Figure 3. A second form is non-uniform where the longitudinal density of the grid also varies, this is shown in Figure 4. In order to ensure that the solutions were not grid dependent, both types of grid were refined until the results were unchanging. The grids shown in Figures 3 and 4 produce grid independent

Discharge (l/s)	12.3
Mean entrance velocity (m/s)	0.25
Entrance depth (m)	0.063
Width of channel (m)	0.8
Bend radius (m)	
Inner	0.4
Centre	0.8
Outer	1.2
Bend angle (degrees)	180
Cross-section	Rectangular
Bottom slope	Zero

Table I.
Experimental condition of Rozovski's Run 8 data

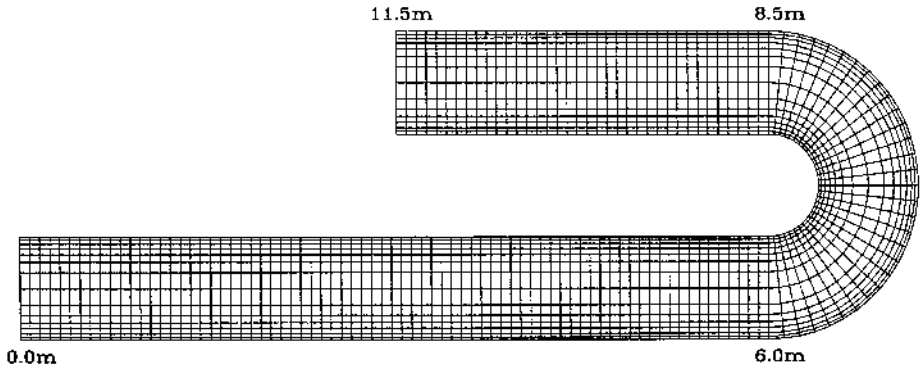

Figure 3.
Plan view of channel and semi-uniform grid lines

results, Figure 3 contains 142 × 14 control volumes and Figure 4 contains 82 × 14 control volumes.

The discretized equations were solved using a line-by-line iteration method and the computations were undertaken on a SUN SPARC Station model *IPC*. Both grids produced similar results however the results produced from the non-uniform grid compare slightly more favourably with the experimental results. It takes 3,369 seconds CPU time on semi-uniform grids to obtain the solution and 2,121 seconds CPU time on non-uniform grids; that is about 37 per cent less CPU time.

The computed results

Except where otherwise stated, all the results in this section were obtained using non-uniform grid lines, the parameters used were $C_b = 0.021$, $C_w = 0.005$, $\alpha = 0.5$ and $\alpha_h = 0.9$.

Velocity vectors computed from the two grid types are plotted in Figure 5 and Figure 6. They clearly show the acceleration on the inner wall of the

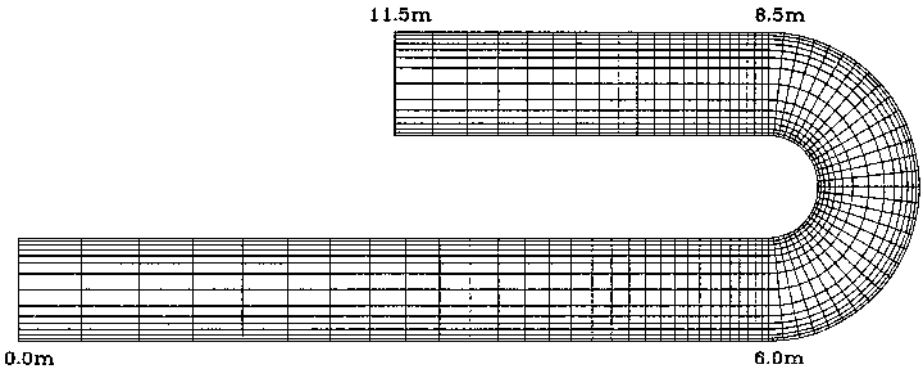

Figure 4.
Non-uniform grid lines

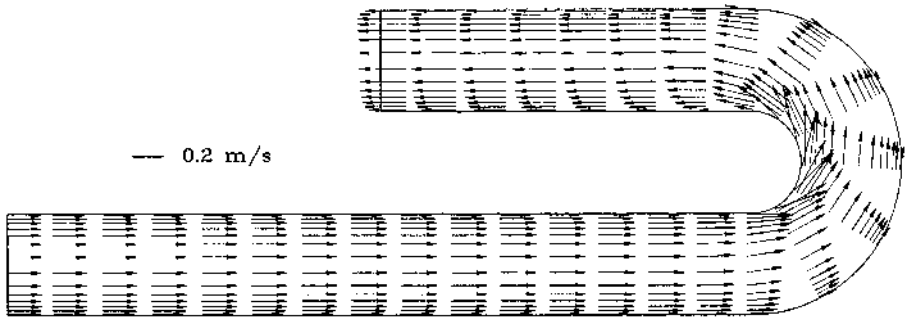

Figure 5.
Velocity vectors based on semi-uniform grids

channel and the deceleration on the outer wall as the fluid enters the bend. Furthermore the velocity distributions at the entrance and exit of the bend follow closely the patterns shown in Rozovski's experiments. The velocity distribution at entry is characterized by an acceleration on the inside of the bend and a similar acceleration is seen at the outside of the bend as the fluid leaves.

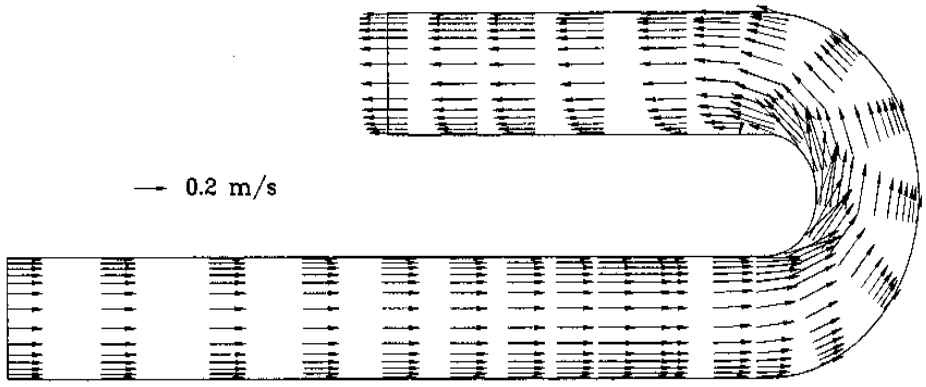

Figure 6.
Velocity vectors based on non-uniform grids

It is difficult to make comparisons between the computed and laboratory water level variations round the bend because of the difficulty of measuring the experimentally determined levels accurately. However the numerical model produced the expected transverse level differences round the bend. A comparison between the computed levels and Rozovski's measured ones is given in Figure 7 and Table II. The latter shows that the comparison is good with errors less than 1mm except at the three points marked with asterisks. These larger errors may be due to inaccuracies in the experimental data, there do not appear to be any irregularities in the computed depths at these points.

Figure 8 and Table III show the measured and computed values of the tangential velocities at various positions around the channel bend. With a few exceptions the errors are less than 10 per cent.

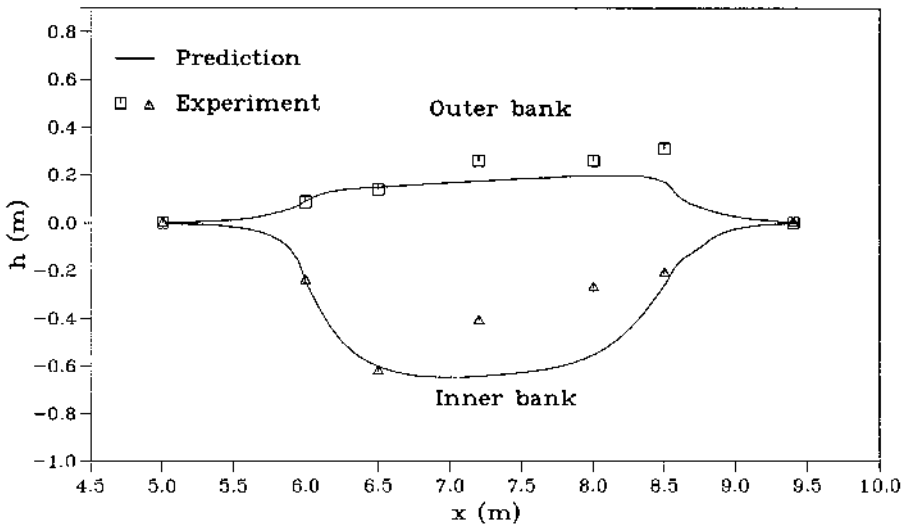

Figure 7.
Water surface super-elevation in bend

S (m)	h_{er} (cm)	h_{cr} (cm)	Error (cm)	h_{el} (cm)	h_{cl} (cm)	Error (cm)
5.0	0.000	0.002	0.002	0.000	–0.002	0.002
6.0	0.090	0.091	0.001	–0.240	–0.245	0.005
6.5	0.140	0.148	0.008	–0.620	–0.592	0.028
7.2	0.260	0.175	0.085	–0.410	–0.628	0.218*
8.0	0.260	0.196	0.064	–0.270	–0.541	0.271*
8.5	0.310	0.165	0.145*	–0.210	–0.245	0.035
9.4	0.000	0.003	0.003	0.000	–0.001	0.001

Note: h_{er} and h_{el} are experimental data of super-elevation on the outer and inner banks, respectively, h_{cr} and h_{cl} are the corresponding values from the computation

Table II.
Comparison of super-elevation relative to central depth in bend

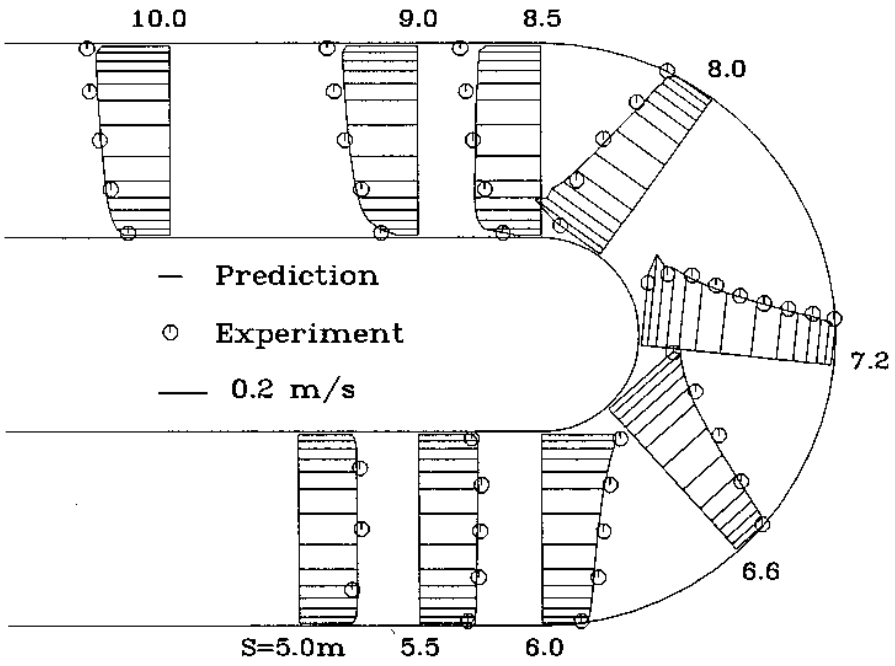

Figure 8.
Profiles of depth-averaged tangential velocities

The accuracy of the numerical procedure

The numerical procedure proposed is subject to some numerical diffusion, this is probably caused by the upwind approximation of the convection terms. Patankar[5] has shown that numerical diffusion is particularly severe when the direction of flow is not parallel to one of the grid lines. Consideration of Figure 5 and Figure 6 shows that such flows occur within the channel bend, especially in the regions close to the inner bank, for example velocity vectors at s = 6.6 m, s = 7.2 m and s = 8.0 m (see Figure 5). Furthermore the use of momentum interpolation may produce physically unrealistic velocities in regions of the flow where the depth is changing rapidly. In an attempt to further investigate the accuracy of the numerical solution, five cross sections were chosen at several points in the channel and the magnitudes of various terms in the momentum equations were investigated. The positions of these cross sections are shown in Figure 9. In order to estimate the amount of diffusion induced by the upwind scheme and the extent of the errors in the computation of the cell velocities caused by the momentum interpolation, a central difference approximation was used to determine the values of the convection terms and cell velocities at the five cross sections. Using this approach the terms in the momentum equation no longer sum to zero hence the magnitude of the closing residual provides a measure of the second order truncation errors introduced into the solution by the momentum interpolation and the upwind differencing. Figure 10 shows the balance of all the terms of the momentum equation in the x direction at sections

S (m)	V_e (m/s)	V_c (m/s)	Er	S (m)	V_e (m/s)	V_c (m/s)	Er
5.0	0.228	0.231	0.011	7.2	0.307	0.287	0.065
5.0	0.268	0.231	0.137	7.2	0.303	0.328	0.082
5.0	0.261	0.232	0.112	7.2	0.265	0.296	0.117
5.5	0.196	0.208	0.064	8.0	0.223	0.203	0.089
5.5	0.241	0.230	0.047	8.0	0.245	0.222	0.092
5.5	0.248	0.233	0.059	8.0	0.267	0.250	0.062
5.5	0.254	0.237	0.068	8.0	0.257	0.285	0.110
5.5	0.215	0.221	0.027	8.0	0.195	0.230	0.180
6.0	0.160	0.174	0.087	8.5	0.326	0.252	0.226
6.0	0.230	0.209	0.089	8.5	0.304	0.258	0.151
6.0	0.252	0.229	0.091	8.5	0.277	0.264	0.049
6.0	0.280	0.259	0.076	8.5	0.228	0.267	0.172
6.0	0.322	0.301	0.064	8.5	0.157	0.114	0.276
6.6	0.147	0.141	0.039	9.0	0.362	0.299	0.173
6.6	0.199	0.193	0.029	9.0	0.336	0.292	0.131
6.6	0.261	0.231	0.116	9.0	0.294	0.276	0.063
6.6	0.313	0.284	0.091	9.0	0.228	0.248	0.088
6.6	0.359	0.352	0.019	9.0	0.150	0.161	0.074
7.2	0.183	0.178	0.024	10.0	0.335	0.293	0.126
7.2	0.202	0.191	0.054	10.0	0.325	0.291	0.105
7.2	0.213	0.202	0.051	10.0	0.285	0.278	0.025
7.2	0.225	0.225	0.001	10.0	0.240	0.260	0.082
7.2	0.245	0.237	0.032	10.0	0.170	0.193	0.135
7.2	0.278	0.262	0.056				

Notes: S is the distance of the central line of the channel from the entrance

$Er = |V_c - V_e|/V_e$, where V_e is experimental value and V_c the computational one

Table III.
Comparison of depth-averaged tangential velocity

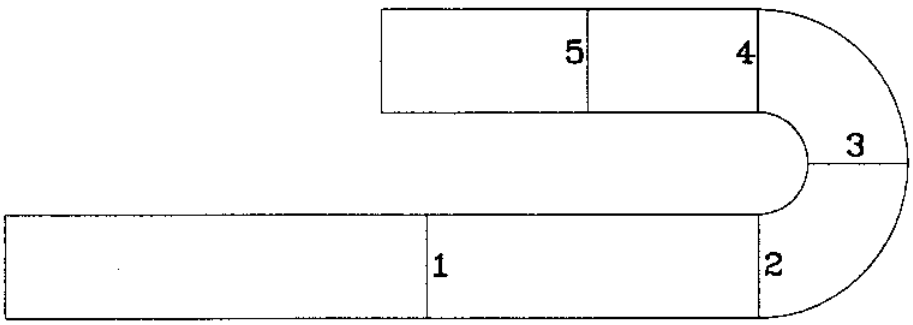

Figure 9.
Five cross-sections in the channel

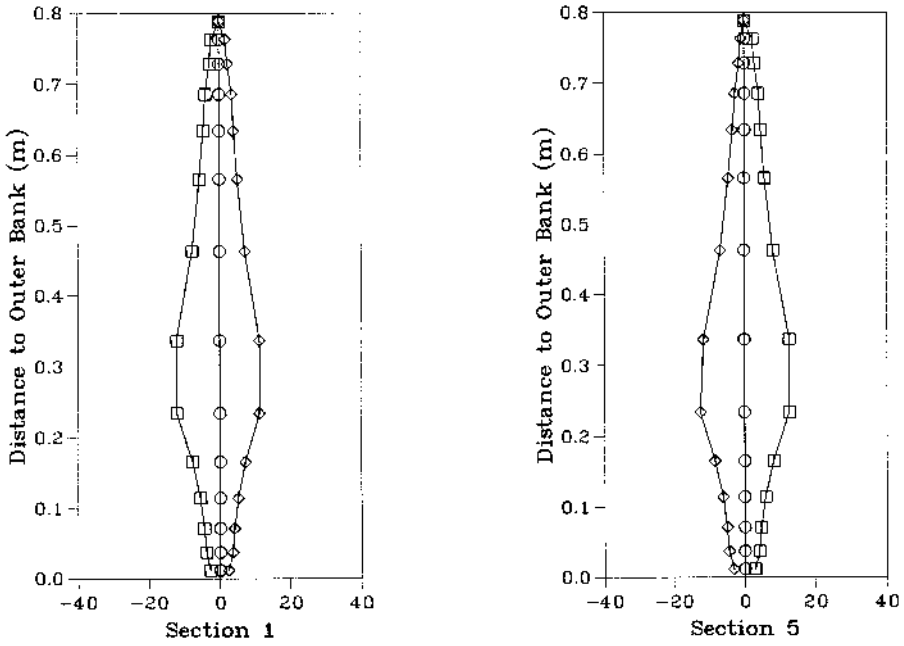

Figure 10.
Balance of x direction momentum equation evaluated from numerical solution. The symbols mean: □ = depth gradient $g/2\nabla h^2$; Δ = convection $\nabla \cdot (h\vec{V}\vec{V})$; ◊ = bottom shear stress $\vec{\tau}_b/\rho$; ○ = closing term (Σ all the terms)

1 and 5. The corresponding balance in the transverse direction is too small to be of significance. It should be noted that in this and subsequent figures only significant terms are plotted for clarity. As seen in the figures, at cross-sections 1 and 5, there are balances between the depth gradients and the bottom shear stresses and no numerical diffusion is apparent. The closing term suggests that no physically unrealistic velocities are present at these two sections.

In Figure 11 the balances of all the terms in the x and y direction momentum equations at section 2 are plotted. It is clearly seen that there are balances among the depth gradients, the convection terms and the bottom shear stresses and the other terms are insignificant. For detailed consideration of the balances the section is split into two regions, the flow in the inner half of the bend and the flow in the outer half. In the outer half the positive super-elevation of the water surface leads to a large negative y depth gradient, negative x convection and positive y convection terms which carry the water towards the centre of the channel. Also, a small positive x depth gradient has a slight influence on the x direction velocity. In the inner half, the negative super-elevation of water and the positive x convection term lead to a large negative y depth gradient and positive y convection term which carry the water towards the centre of the channel. All these details are consistent with the expected flow pattern in the upstream part of the bend. As previously, the closing term suggests that numerical diffusion is not present and the effect of non-physical velocities is insignificant.

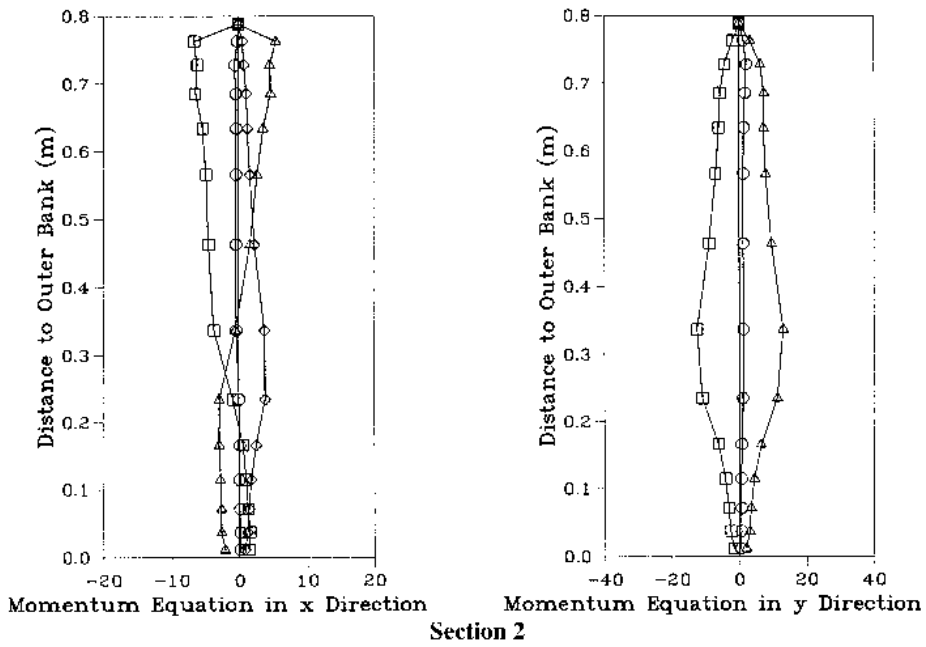

Figure 11. Balance of momentum equation evaluated from numerical solution

Furthermore, the situation at section 4 is nearly the reverse of that at section 2, hence the similar comments apply (see Figure 12).

As shown in Figure 13, there are balances at section 3 between the depth gradient and the convection term for the *x* momentum equation and between the depth gradient and the bottom shear stress for the *y* momentum equation. Clearly, at the middle section of the 180° bend, the large positive *x* depth gradient and large negative *x* convection term are the dominant features of the flow. For the *y* momentum equation there is some error in the balance, probably this is caused by the presence of both numerical diffusion and non-physical velocities. The magnitude of the closing term is small compared with the values of the dominant terms, therefore it has little effect on the solution.

The above analysis indicates that the accuracy of the solution is independent of the grid type chosen. Furthermore the results indicate that there is little numerical diffusion in bends where secondary flows are small[9] – such as the present one – and that the possible non-physical velocities induced by momentum interpolation do not have any noticeable effect on the results[10].

Conclusions

A finite volume method for steady state 2-D shallow water flows is developed which can be used to solve flow problems in channels with arbitrary or

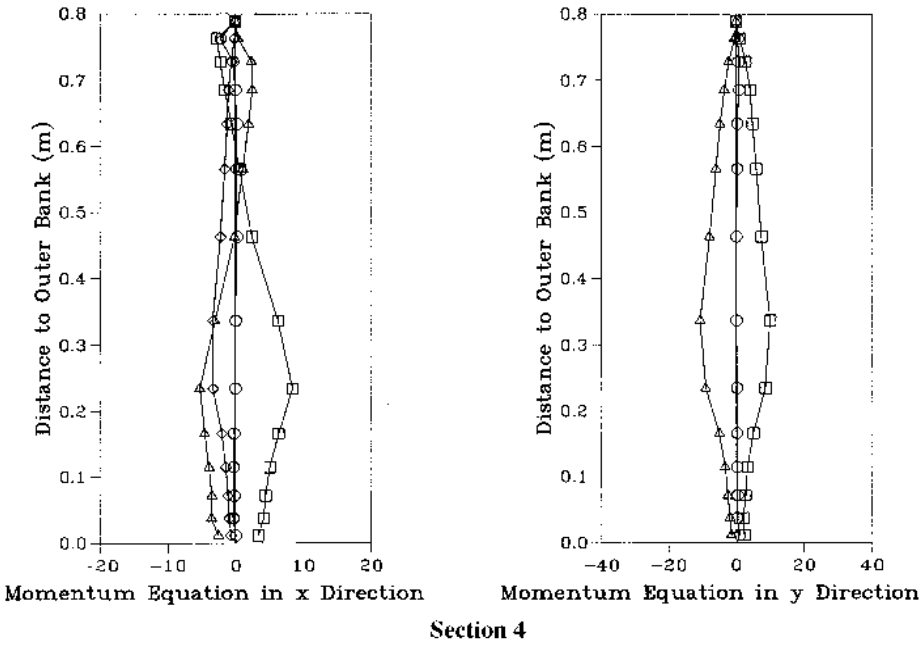

Figure 12.
Balance of momentum equation evaluated from numerical solution

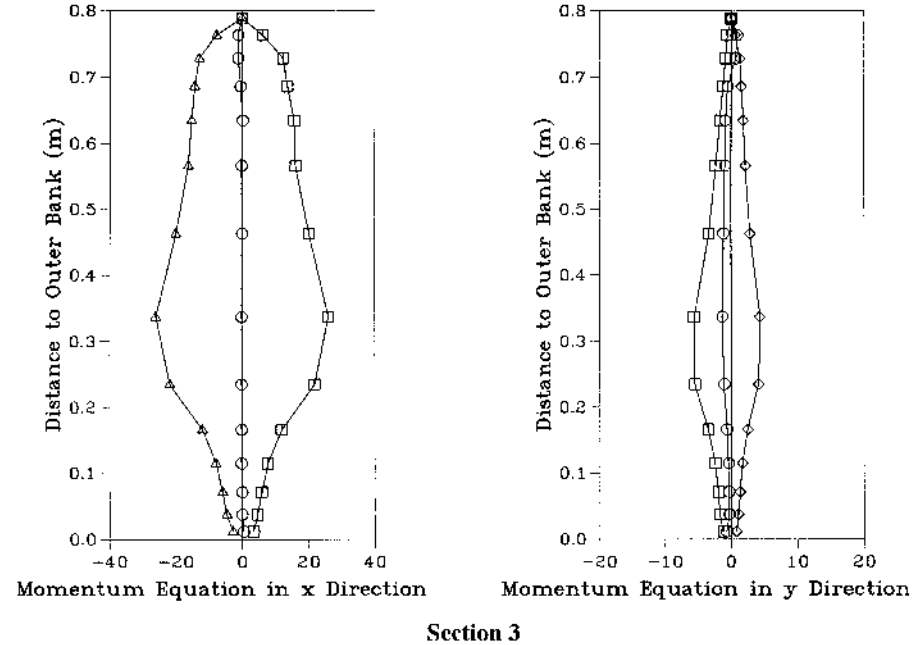

Figure 13.
Balance of momentum equation evaluated from numerical solution

irregular boundary shapes by using a collocated body-fitted co-ordinate system. The momentum interpolation method is used to avoid depth oscillations and a modified SIMPLE-like scheme is described to deal with the velocity-depth coupling in which the velocity field is not linearly proportional to the depth. This feature in the construction of the model guarantees that there is no numerical instability and that a physically realistic solution is obtained. The comparison between the computed and laboratory results has shown that the velocity and the super-elevation in the bend can be predicted correctly. The analysis of numerical accuracy shows that neither numerical diffusion nor non-physical velocities are found to have any noticeable effect on the solution. The study indicates that the use of momentum interpolation is valid and that the model may be applied to practical problems.

References

1. Demirdžić, I.A. A., *Finite Volume Method for Computation of Fluid Flow in Complex Geometries*, PhD thesis, London University, UK, 1982.
2. Grubeft, J.P., Numerical computation of two-dimensional flows, *Journal of Waterway Harbors and Coastal Engineering Division, ASCE*, 1976.
3. Zhou, J. G., Velocity-depth coupling in shallow water flows, *Journal Hydr. Engineering Division, ASCE*, Vol. 121 No. 10, 1995, pp. 717-24.
4. Thompson, J.F., Warsi, Z.U.A. and Mastin, C.W., *Numerical Grid Generation: Foundations and Applications*, North-Holland, New York, NY, 1985.
5. Patankar, S.V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere, New York, NY, 1980.
6. Rhie, C.M., *A Numerical Study of the Flow Past an Isolated Airfoil with Separation*, PhD thesis, University of Illinois at Urbana-Champaign, 1981.
7. Majumdar, S., "Role of underrelaxation in momentum interpolation for calculation of flow with nonstaggered grids", *Numerical Heat Transfer*, Vol. 13, 1988, pp. 125-32.
8. Rozovski, I.L., *Flow of Water in Bends of Open Channels,* Israel Program for Scientific Translation, Jerusalem, Israel, 1965.
9. McGuirk, J. J. and Rodi, W., "A depth-averaged mathematical model for the field of the near side discharges into open-channel flow", *J. Fluid Mech.*, Vol. 86, 1978, pp. 761-81.
10. Miller, T.F. and Schmidt, F.W., "Use of a pressure-weighted interpolation method for the solution of the incompressible Navier-Stokes equations on a nonstaggered grid system", *Numerical Heat Transfer,* Vol. 14, 1988, pp. 213-33.