INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **48**:775–799
Published online 18 March 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.956

Time accurate local time stepping for the unsteady shallow water equations

A. J. Crossley and N. G. Wright*,†

School of Civil Engineering, University of Nottingham, Nottingham NG7 2RD, U.K.

SUMMARY

Two time accurate local time stepping (LTS) strategies originally developed for the Euler equations are presented and applied to the unsteady shallow water equations of open channel flow. Using the techniques presented allows individual cells to be advanced to different points in time, in a time accurate fashion. The methods shown are incorporated into an explicit finite volume version of Roe's scheme which is implemented in conjunction with an upwind treatment for the source terms. A comparison is made between the results obtained using the conventional time stepping approach and the two LTS methods through a series of test cases. The results illustrate a number of benefits of using LTS which included reduced run times and improved solution accuracy. In addition it is shown how using an upwind source term treatment can be beneficial for flows dominated by the geometry. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: local time stepping; Roe's approximate Riemann solver; upwind source terms; open channel; shallow water; unsteady flow

1. INTRODUCTION

The flow in open channels is frequently modelled using the shallow water equations for which both 1D and 2D formulations exist. Advances in the numerical treatment of these equations has been made by following new developments in the field of aeronautics. Whereas much research into computational hydraulics has focused on using methods such as the Preismann or McCormack schemes [1, 2], surveying more recent literature shows the increasing application of upwinded high resolution shock capturing methods [3, 4], which have proved successful for application to the Euler equations. Such techniques are generally based on the Godunov formulation and a solution is obtained by solving a series of Riemann problems. In particular

*Correspondence to: N. G. Wright, School of Civil Engineering, University of Nottingham, University Park, Nottingham NG7 2RD, U.K.
†E-mail: Nigel.Wright@nottingham.ac.uk

Contract/grant sponsor: University of Nottingham

Received 7 March 2004
Revised 24 January 2005
Accepted 31 January 2005

the flux difference splitting approach of Roe has proved especially popular for open channel flows [5–9].

In this paper, another concept originally developed for the Euler equations is applied to the shallow water equations, namely the application of local time stepping, or temporal adaptivity. Following the conventional approach (global time stepping), the solution is advanced in time through a series of updates in which all the cells are integrated to the same point in time. When an explicit scheme is used, this necessitates using the smallest permissible time step for the mesh based on stability criteria and the CFL condition. In the case of an implicit method, though stability constraints may not place a restriction of the size of the time step, the issue of accuracy is still a consideration, which in turn may limit the size of the time step used. Implicit adaptations of explicit schemes can be formulated (see Reference [10] for example), which then may permit the use of higher CFL numbers. However, generally these methods are less accurate than the corresponding explicit counterparts, particularly if discontinuities are present in the flow. In contrast to global time stepping, the idea behind local time stepping is to integrate different cells to different points in time using each cell's individual permissible time step. This concept is readily applied to steady state problems in which a pseudo time stepping strategy is adopted and used to accelerate convergence, and reduce the overall run time. The application to transient problems, however, is not so straightforward. Here it is necessary to ensure that information is correctly propagated between different cells. In this paper two time accurate local time stepping strategies developed for aeronautical flows are applied to the 1D shallow water equations. The algorithms are implemented in conjunction with Roe's scheme together with an upwind treatment for the source terms. The concept of local time stepping may equally be applied to other numerical methods and serves not only to reduce run times, but can also improve solution accuracy.

2. THE 1D SHALLOW WATER EQUATIONS

The 1D shallow water or Saint Venant equations which can be used to represent the flow in open channels, can be written as [1]

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0$$

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A} + gI_1\right) = gA(S_{\mathrm{o}} - S_{\mathrm{f}}) + gI_2$$

where A is the cross sectional area in m^2, Q is the discharge in $\mathrm{m}^3\,\mathrm{s}^{-1}$, S_{o} is the bed slope and S_{f} represents the friction slope which can be defined by either the Manning or Chezy formula. I_1 is an 'effective pressure' and can be evaluated from

$$I_1 = \int_0^{h(x)} [h(x) - \eta]\sigma(x,\eta)\,\mathrm{d}\eta$$

where h is the level of the free surface, η is the depth integration variable and σ corresponds the channel width at a particular depth. For a rectangular channel, $I_1 = A^2/2b$, where b is the width of the channel. I_2 is a wall pressure term represented by

$$I_2 = \int_0^{h(x)} \left(h(x) - \left[\frac{\partial \sigma}{\partial x} \right]_{h=h_0} \right) \mathrm{d}\eta$$

which is equal to zero for a prismatic channel. This particular formulation of the equations was chosen as the equations are written in conservative form and can be classified as conservation laws for which Riemann based methods were designed. Using vector notation it is convenient to express the system of equations in the form

$$\mathbf{U}_t + \mathbf{F}_x = \mathbf{R} \tag{1}$$

where

$$\boldsymbol{U} = \begin{pmatrix} A \\ Q \end{pmatrix}, \quad \boldsymbol{F} = \begin{pmatrix} Q \\ Q^2/A + gI_1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{R} = \begin{pmatrix} 0 \\ gA(S_\mathrm{o} - S_\mathrm{f}) + gI_2 \end{pmatrix}$$

Written in this form, the vector $\boldsymbol{U}$ is the vector of conserved variables, $\boldsymbol{F}$ is the flux vector and $\boldsymbol{R}$ is the source term vector. It is also possible to write Equation (1) in the form

$$\mathbf{U}_t + J\mathbf{U}_x = \mathbf{R}$$

using the Jacobian matrix J of the flux vector $\boldsymbol{F}$ where

$$J = \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \begin{pmatrix} 0 & 1 \\ c^2 - u^2 & 2u \end{pmatrix}$$

noting that $u = Q/A$ and $c = \sqrt{(gA/B)}$ (B is the channel width at the free surface) together with corresponding eigenvalues and eigenvectors

$$\lambda^{1,2} = (u \pm c) \quad \text{and} \quad \boldsymbol{e}^{1,2} = \begin{pmatrix} 1 \\ u \pm c \end{pmatrix}$$

3. ROE'S SCHEME

Roe's scheme was initially developed for the Euler equations [11] and subsequently extended to the Saint Venant equations for channels of infinite width by Glaister [5]. Numerous authors have since applied the method to both 1D and 2D shallow water flows and the formulation adopted here is as reported by Alcrudo *et al.* [6]. The method is based on the approach of Godunov, which involves solving a series of Riemann problems at the cell interfaces using the cell values $\boldsymbol{U}_\mathrm{L}$ and $\boldsymbol{U}_\mathrm{R}$ in order to evaluate the numerical flux used to update the homogeneous equations. By introducing a local linearization, the exact Riemann problems are replaced by approximate ones, from which it becomes necessary to determine an approximate Jacobian matrix $\tilde{J}$ of the form

$$\tilde{J} = \begin{pmatrix} 0 & 1 \\ \tilde{c}^2 - \tilde{u}^2 & 2\tilde{u} \end{pmatrix}$$

with eigenvalues and eigenvectors

$$\tilde{\lambda}^{1,2} = (\tilde{u} \pm \tilde{c}) \quad \text{and} \quad \tilde{\mathbf{e}}^{1,2} = \begin{pmatrix} 1 \\ \tilde{u} \pm \tilde{c} \end{pmatrix}$$

where the values denoted by tildes are known as the Roe averaged quantities. Following Roe's analysis, the expressions for $\tilde{u}$ and $\tilde{c}$ are found to be

$$\tilde{u} = \frac{\sqrt{A_L} u_L + \sqrt{A_R} u_R}{\sqrt{A_L} + \sqrt{A_R}}$$

$$\tilde{c} = \begin{cases} g \dfrac{I_{1_R} - I_{1_L}}{A_R - A_L} & \text{if } A_R - A_L \neq 0 \\ (c_L + c_R)^2 & \text{if } A_L = A_R \end{cases} \tag{2}$$

Note that using Equation (2) for $\tilde{c}$ it is necessary to consider the case where $A_L = A_R$. Alternatively García-Navarro and Vázquez-Cendón [12] have proposed using the form

$$\tilde{c} = \sqrt{\frac{g}{2}\left[\left(\frac{A}{B}\right)_L + \left(\frac{A}{B}\right)_R\right]}$$

which is the approximation used in this work. The conservative formulation of the scheme is utilized whereby the numerical flux is formulated as

$$\boldsymbol{F}^*_{i+1/2} = \frac{1}{2}(\boldsymbol{F}_{i+1} + \boldsymbol{F}_i) - \frac{1}{2}\sum_{k=1}^{2} \tilde{\alpha}^k_{i+1/2} |\tilde{\lambda}^k_{i+1/2}| \tilde{\boldsymbol{e}}^k_{i+1/2} \tag{3}$$

where $\tilde{\alpha}^k_{i+1/2}$ are the wave strengths defined by

$$\tilde{\alpha}_1 = \frac{(\tilde{c} - \tilde{u})\Delta A + \Delta Q}{2\tilde{c}}$$

$$\tilde{\alpha}_2 = \frac{(\tilde{c} + \tilde{u})\Delta A - \Delta Q}{2\tilde{c}}$$

given that the Δ operator refers to the difference between the right and left values of the Riemann problem. The entropy fix proposed by Harten [13] is adopted whereby the value of $|\tilde{\lambda}|$ in Equation (3) is re-evaluated using

$$|\tilde{\lambda}| = \begin{cases} |\tilde{\lambda}| & \text{if } |\tilde{\lambda}| \geqslant \varepsilon \\ \varepsilon & \text{if } |\tilde{\lambda}| < \varepsilon \end{cases}$$

where

$$\varepsilon = \max(0, \tilde{\lambda}(\mathbf{U}_L, \mathbf{U}_R) - \lambda(\mathbf{U}_L), \lambda(\mathbf{U}_R) - \tilde{\lambda}(\mathbf{U}_L, \mathbf{U}_R))$$

and the update is represented by

$$\boldsymbol{U}_i^{n+1} = \boldsymbol{U}_i^n - \frac{\Delta t}{\Delta x}(\Delta \boldsymbol{F}^*_{i+1/2} - \Delta \boldsymbol{F}^*_{i-1/2}) \tag{4}$$

The scheme may also be extended to be second-order in space and time, and this is achieved through modifying the flux definition using the formulation

$$\boldsymbol{F}^*_{i+1/2} = (\boldsymbol{F}_{i+1} + \boldsymbol{F}_i) - \frac{1}{2}\sum_{k=1}^{2} \tilde{\alpha}^k_{i+1/2}|\tilde{\lambda}^k_{i+1/2}|\tilde{\boldsymbol{e}}^k_{i+1/2}$$
$$+ \frac{1}{2}\sum_{k=1}^{2} \varphi(r^k_{i+1/2})\tilde{\alpha}^k_{i+1/2}|\tilde{\lambda}^k_{i+1/2}|\left(1 - \frac{\Delta t}{\Delta x}|\tilde{\lambda}^k_{i+1/2}|\right)\tilde{\boldsymbol{e}}^k_{i+1/2}$$

Cast in this form, the numerical flux can be seen as the first-order Roe flux coupled with a second-order correction term which is limited via the flux limiter function, φ, with the argument

$$r^k_{i+1/2} = \frac{\tilde{\alpha}^k_{i+1/2-s}}{\tilde{\alpha}^k_{i+1/2}} \quad \text{and} \quad s = \text{sign}(\tilde{\lambda}^k_{i+1/2})$$

A number of limiter functions [14] are available which serve the purpose of limiting the amount of the second-order correction applied in regions of steep gradients. In the results shown in which the second-order form of the scheme is utilized, the Superbee limiter is used i.e.

$$\varphi(r) = \frac{r + |r|}{1 + r}$$

4. SOURCE TERM TREATMENT

The benefits of employing an upwind treatment to evaluate the numerical flux have been established for some time and generally a less sophisticated approach has been adopted to evaluate the source terms. However, more recently it has been demonstrated that it can be beneficial to incorporate upwinding into the source term treatment which in turn balances the flux and the source terms [15–17]. In this work the concept originally proposed by Bermúdez and Vázquez [16] is adopted which demonstrated how to construct an upwind treatment in the case of non-uniform beds (varying S_o) on the basis of not perturbing an equilibrium solution. The treatment is developed in conjunction with the scheme used to evaluate the numerical flux and hence the upwinding can be introduced by utilizing the coefficients within the Riemann solver. The original method developed for the first-order Roe's scheme has been extended by Vázquez-Cendón [18] and García-Navarro and Vázquez-Cendón [12] to non-prismatic rectangular channels and non-uniform grids. Burguete and García-Navarro [19] further developed the method for second-order formulations of Roe's scheme.

The basis of the approach is to determine an expression for $\boldsymbol{R}_i$ to be used in the update formula

$$\boldsymbol{U}_i^{n+1} = \boldsymbol{U}_i^n - \frac{\Delta t}{\Delta x_i}(\Delta \boldsymbol{F}^*_{i+1/2} - \Delta \boldsymbol{F}^*_{i-1/2}) + \Delta t \boldsymbol{R}_i$$

where $\boldsymbol{R}_i$ is such that in the case of zero discharge, the numerical flux and source terms balance so that the solution is not perturbed, deemed ‘Property C’ whereby

$$(\Delta\boldsymbol{F}^*_{i+1/2} - \Delta\boldsymbol{F}^*_{i-1/2}) = \boldsymbol{R}_i \Delta x_i$$

and the source term is defined as a cell average value

$$\boldsymbol{R}_i = \frac{1}{\Delta x_i}\int_{C_i} \boldsymbol{R}(x, \boldsymbol{U})\,\mathrm{d}x$$

given that cell C_i corresponds to

$$C_i = \left(x_i - \frac{x_i - x_{i-1}}{2},\ x_i + \frac{x_i - x_{i-1}}{2}\right)$$

To introduce upwinding into the source term representation, the numerical source term is constructed from left and right contributions, i.e.

$$\boldsymbol{R}_i = \frac{1}{\Delta x_i}\left[\frac{x_i - x_{i-1}}{2}\,\psi_{L_{i-1/2}} + \frac{x_{i+1} - x_i}{2}\,\psi_{R_{i+1/2}}\right]$$

In the case of a uniform mesh, where the mesh spacing is Δx, following the decomposition, the source term within cell i is evaluated using left and right contributions where using the contribution from the left is calculated using the values at the left interface $(i - \frac{1}{2})$ and the contribution from right uses the $(i + \frac{1}{2})$ values, i.e.

$$\boldsymbol{R}_i \Delta x = \frac{1}{2}\left((H_\mathrm{L}\boldsymbol{\beta})_{i-1/2} + (H_\mathrm{R}\boldsymbol{\beta})_{i+1/2}\right) = \frac{1}{\Delta x}\left(\frac{\Delta x}{2}\,(\psi_\mathrm{L})_{i-1/2} + \frac{\Delta x}{2}\,(\psi_\mathrm{R})_{i+1/2}\right)$$

where $\psi_\mathrm{L} = H_\mathrm{L}\boldsymbol{\beta}$, $\psi_\mathrm{R} = H_\mathrm{R}\boldsymbol{\beta}$ and $\boldsymbol{\beta} = (\tilde{\beta}_1, \tilde{\beta}_2)^\mathrm{T}$. The exact form of $\boldsymbol{\beta}$ depends upon the geometry of the problem under consideration. In the case of a rectangular channel $(B = b)$ it can be shown that $\tilde{\beta}_1 = -\tilde{\beta}_2 = \tilde{\beta}$ where

$$\tilde{\beta} = g\,\frac{\Delta x}{2\tilde{c}}\left(-\tilde{A}z_x - \tilde{A}\tilde{S}_\mathrm{f} + \frac{\tilde{A}^2}{2\tilde{b}^2}\,b_x\right) = \frac{g}{2\tilde{c}}\left(-\tilde{A}\Delta z - \tilde{A}\tilde{S}_\mathrm{f} + \frac{\tilde{A}^2}{2\tilde{b}^2}\,\Delta b\right)$$

and the values $\tilde{A}$, $\tilde{B}$, $\tilde{S}_\mathrm{f}$, Δz, and Δb need to be defined. It has been shown [12] that in the case of a smooth rectangular channel with varying bottom topography, using the definitions

$$\Delta z = z_{i+1} - z_i, \quad \tilde{A} = \tilde{h}\tilde{b}, \quad \tilde{h} = \tfrac{1}{2}(h_{i+1} + h_i), \quad \tilde{b} = b \tag{5}$$

Property C is satisfied exactly. For a non-prismatic rectangular channel, where I_2 is non-zero using the preceding definitions does not lead to Property C being satisfied. This stems from the fact that in this instance

$$\Delta\boldsymbol{F} \neq A\Delta\boldsymbol{U}$$

as now

$$\Delta\boldsymbol{F} = \begin{pmatrix} \Delta Q \\ \Delta\left(\dfrac{Q^2}{A} + gI_1\right) \end{pmatrix} = \begin{pmatrix} \Delta Q \\ \left(g\,\dfrac{A}{b} - \dfrac{Q^2}{A^2}\right)\Delta A + 2\,\dfrac{Q}{A}\,\Delta Q - \dfrac{gA^2}{2b^2}\,\Delta b \end{pmatrix}$$

such that

$$\Delta \boldsymbol{F} = A\Delta \boldsymbol{U} + \boldsymbol{V} \quad \text{where } \boldsymbol{V} = \begin{pmatrix} 0 \\ -\dfrac{gA^2}{2b^2}\,\Delta b \end{pmatrix}$$

García-Navarro and Váquez-Cendón [12] accommodate $\boldsymbol{V}$ as an additional source term by modifying $\boldsymbol{R}$ such that

$$\hat{\boldsymbol{R}} = \boldsymbol{R} - \boldsymbol{V}$$

which is incorporated into $\tilde{\beta}$ whereby

$$\tilde{\beta} = \frac{g}{2\tilde{c}}\left(-\tilde{A}\Delta z - \tilde{A}\tilde{S}_f + \frac{\tilde{A}^2}{\tilde{b}^2}\,\Delta b\right)$$

using

$$\Delta b = b_{i+1} - b_i \quad \text{and} \quad \tilde{b} = \tfrac{1}{2}\,(b_{i+1} - b_i)$$

However, analysis shows that although this results in the flux and source terms balancing for the mass equation, the solution to momentum equation will be perturbed. This can be avoided by using the original definition of $\tilde{\beta}$ together with $\tilde{h} = \sqrt{h_i h_{i+1}}$ and $\tilde{A} = \frac{1}{2}(b_i h_i + b_{i+1} h_{i+1})$ for the momentum equation. An alternative approach which permits one set of definitions for $\tilde{\beta}$ and the average values in Equation (5) to be used and leads to Property C being satisfied for both equations is to modify the numerical flux which becomes

$$\boldsymbol{F}^*_{i-1/2} = \frac{1}{2}\,(\boldsymbol{F}_{i-1} + \boldsymbol{F}_i) + \boldsymbol{V}_{i-1/2} - \frac{1}{2}\sum_{k=1}^{2} \tilde{\alpha}^k_{i-1/2}|\tilde{\lambda}^k_{i-1/2}|\tilde{\boldsymbol{e}}^k_{i-1/2}$$

$$\boldsymbol{F}^*_{i+1/2} = \frac{1}{2}\,(\boldsymbol{F}_i + \boldsymbol{F}_{i+1}) - \boldsymbol{V}_{i+1/2} - \frac{1}{2}\sum_{k=1}^{2} \tilde{\alpha}^k_{i+1/2}|\tilde{\lambda}^k_{i+1/2}|\tilde{\boldsymbol{e}}^k_{i+1/2}$$

where

$$\boldsymbol{V}_{i-1/2} = \begin{pmatrix} 0 \\ -\dfrac{h_{i-1}h_i}{2}\,(b_i - b_{i-1}) \end{pmatrix} \quad \text{and} \quad \boldsymbol{V}_{i+1/2} = \begin{pmatrix} 0 \\ -\dfrac{h_i h_{i+1}}{2}\,(b_{i+1} - b_i) \end{pmatrix}$$

Further consideration is necessary to apply the treatment to non-rectangular channels and this has been conducted by the authors of this paper. In the case of a prismatic trapezoidal channel, a logical choice for the average quantities would be

$$\tilde{h} = \tfrac{1}{2}\,(h_i + h_{i+1}), \quad \tilde{B} = b + 2m\tilde{h}, \quad \tilde{A} = \tilde{h}(b + 2m\tilde{h})$$

where m is the gradient of the side walls of the channel. Following this selection, which is consistent with the rectangular values with $m = 0$, it can be shown that Property C is maintained exactly in the case of zero discharge for the continuity equation. However, this is not the case for the momentum equation, for which analysis of the update formula shows there is a discrepancy of the form

$$\text{LHS} - \text{RHS} = \frac{\Delta t}{2\Delta x}\,\frac{gm}{12}[(h_i - h_{i-1})^3 + (h_{i+1} - h_i)^3]$$

Following the approach for non-prismatic channels, this is accommodated by introducing an additional term to the momentum equation such that an equilibrium solution will be maintained.

5. LOCAL TIME STEPPING

With a normal time integration technique (subsequently referred to as global time stepping or GTS) a global time step is established that guarantees stability for all cells and the solution in every cell is advanced to the same point in time. The strategy behind local time stepping (LTS) is to use each cell's maximum stable time step rather than the global value resulting in different cells being advanced to different points in time. Theoretically this should prove more efficient and reduce the computation time. The difficulty with this approach lies in ensuring that the correct integration procedure is followed such that information is correctly propagated between different regions of the flow in a time accurate fashion. If an efficient means of determining the sequence cannot be devised, then the advantages of using such a technique disappear, as the implementation costs may outweigh the potential savings. It should be noted that in the case of steady flows, the temporal transmission of information is not a consideration, and local time stepping can be implemented without the need for a time integration strategy when a pseudo transient algorithm is adopted. Thus local time stepping is commonly used to accelerate the convergence of solutions in steady state problems.

To illustrate the concept of local time stepping, consider a set of homogeneous conservation laws. For any given cell, i, the update from time level n to $n+1$ can be written as

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta x_i}(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n)$$

When global time stepping is used the value of Δt is calculated based upon stability criteria and the same value is used throughout the mesh. For a given explicit scheme, for example Roe's scheme, the time step must be chosen such that the Courant–Friedrichs–Lewy (CFL) number does not exceed a specified value. For many methods this results in a condition of the form

$$v\,\frac{\Delta t}{\Delta x_i} \leqslant 1$$

where v is a characteristic wave speed representing the speed of propagation. In the case of the Saint Venant equations, the CFL condition takes the form

$$(|u_i| + c_i)\,\frac{\Delta t}{\Delta x_i} \leqslant 1 \tag{6}$$

when Roe's scheme is used. To advance from time level n to $n+1$, a time step value denoted Δt_i is calculated for each cell based on Equation (6). The minimum of all the values is then selected as the time increment such that $\Delta t = \min(\Delta t_i)$ and each cell is advanced by this amount. In contrast to this, the basis of local time stepping is to advance individual cells by the local permissible value Δt_i such that the update becomes

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t_i}{\Delta x_i}(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n)$$

In this paper two procedures for applying local time stepping to time-dependent flows are outlined. These strategies were originally demonstrated to be successful for the Euler equations and have been further developed by the authors for application to the Saint Venant equations.

5.1. Local time stepping using a frozen flux (LTS1)

The basis of the procedure outlined by Zhang *et al.* [20, 21] is to perform a series of temporal updates in the usual fashion, i.e. following a GTS approach, but using the same 'frozen' flux value as calculated from a previous time level where possible. In essence the method is a pseudo local time stepping strategy. The technique reduces the number of flux evaluations necessary and hence reduces the computer costs and run time, even though the same number of temporal updates is being performed. For an individual cell, the frozen flux can be used until the cell has been integrated to a point in time corresponding to the cells maximum permissible time step (calculated at the time the flux evaluation was performed). Once that point is reached a new flux is calculated. The technique was originally illustrated for homogeneous equations, as the method is only concerned with modifying the flux calculations. It was noted by the original authors that it became more beneficial to employ the technique as the level of complexity of the flux function increased.

Following the traditional approach and again considering a set of homogeneous equations, given that Δt represents the global minimum time step, then for each cell i there will be some integer value of k which satisfies

$$k_i \Delta t \leqslant \Delta t_i < (k_i + 1)\Delta t$$

On stability grounds, it is then possible to advance specific cells to the point in time $k_i \Delta t$ using

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{k_i \Delta t}{\Delta x_i}(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n)$$

This may also be achieved using a series notation

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta x_i}\sum_1^{k_i}(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n)$$

or equivalently by a series of updates which can be represented as

$$\mathbf{U}_i^{n+j/k_i} = \mathbf{U}_i^{n+(j-1)/k_i} - \frac{\Delta t}{\Delta x_i}(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n) \tag{7}$$

where $j = 1, 2, \ldots, k_i$, and the flux values used in the calculations are frozen at time level n. In contrast to this, any cells that have a k value of 1 can only be advanced by a time increment of Δt on the basis of stability. In an analogous form to Equation (7), these cells can be advanced to a specific point in time using a series of updates whereby

$$\mathbf{U}_i^{n+j/k_i} = \mathbf{U}_i^{n+(j-1)/k_i} - \frac{\Delta t}{\Delta x_i}(\mathbf{F}_{i+1/2}^{n+(j-1)/k_i} - \mathbf{F}_{i-1/2}^{n+(j-1)/k_i})$$

and the fluxes are re-evaluated for each calculation. Note that for the purposes of illustration, Δt has been considered constant, however in practise it must be calculated after each update, i.e. for each value of j.

Denoting $k_{\max}$ as the maximum k value throughout the grid, then a single global time step advances the solution from time level n to $n+1$, which is equivalent to a time increment of $k_{\max}\Delta t$. This is achieved through a series of j local time steps where $j = 1, 2, \ldots, k_{\max}$. For the first local time step, the solution is advanced by Δt. Subsequently, a new value of Δt is calculated after each local time step and the solution is advanced by successive steps until the end of the global time step is reached.

To assess which cells can use a frozen flux, individual cells are assigned to two groups at the beginning of the global time step. The first group, G_1 consists of cells which cannot utilize a frozen flux, and so the flux must be evaluated at each local time step. The second group, G_2 includes the remaining cells to which are able to use a frozen flux. From a stability viewpoint, it is necessary that an individual cell's permissible time step be at least twice the global minimum value if a frozen flux value is to be used, thus the groups are decided by the criteria

$$G_1 \quad \text{if } \Delta t_i \leqslant 2\Delta t$$

or

$$G_2 \quad \text{if } \Delta t_i > 2\Delta t$$

After each update j (local time step), the group distributions must be re-evaluated and the flux values calculated for the G_1 cells. In addition, consideration must be given to the interaction between cells contained within any G_1/G_2 interface. This can be achieved by treating any G_2 cells contained within the computational stencil of G_1 cells as G_1 cells (such that the flux is re-evaluated), and by limiting the time step of these modified G_2 cells to the G_1 cell value. Once the cells have been advanced to the point in time $k_{\max}\Delta t$, the global time step is complete and all of the flux values are recalculated and the process is repeated.

5.2. Local time stepping using full time integration (LTS2)

Unlike the previous approach, Kleb *et al.* [22] presented a local time stepping technique that was based upon advancing individual cells to a level near that allowed by the CFL limit, which is in essence much more of a local time stepping strategy. Linear interpolation was then used at the interface regions between cells at different levels to extract the information at the correct point in time. Although this method is more efficient than the previous scheme in that fewer updates are performed, the algorithm needed to perform the integration is more complex.

To initiate the local time stepping procedure, the local time step values Δt_i are calculated and the corresponding minimum value, Δt is found. Each cell is then assigned a value of m_i

which corresponds to the cells local time step's power of 2 multiple of Δt_i calculated from

$$m_i = \text{int}\left[\frac{\log(\Delta t_i/\Delta t)}{\log(2)}\right]$$

This relationship is such that

$$2^{m_i} \leqslant \frac{\Delta t_i}{\Delta t} < 2^{m_i+1}$$

Following this assignment, the local time steps are re-evaluated in terms of power of two multiples of the minimum time step. This is accomplished by defining new local time step values, Δt_i^*, whereby

$$\Delta t_i^* = \Delta t 2^{m_i}$$

The basis of the integration procedure is to form a series of 'passes' over the mesh and to update particular cells to their permitted point in time. On each pass, cells with a particular m-value will be integrated. On the first pass, pass 0, all of the cells are updated by their respective time steps. On subsequent passes, those cells for which 2^m is an integer multiple of the pass number, are integrated. This is illustrated in Figure 1, where the solution is advanced from time t^n to t^{n+1} corresponding to a global time step and the cells shown have m-values ranging from 0 to 3. Given that the maximum m-value is 3, the corresponding time interval is $8\Delta t$. The figure serves to demonstrate the integration sequence for specific m-values rather than the distribution of m-values which is discussed later. The sequence is shown and the numbers for each update correspond to the pass number at which the update takes place. As can be seen, cells for which $m = 0$ are always updated. If the cells which have a particular m-value are to be integrated, then any cells having a lower value of m will also be updated. The total number of passes which take place over a global time step depends on the maximum value of m. If $m_{\max} = \max(m_i)$, then if P_{total} is the total number of passes

$$P_{\text{total}} = 2^{m_{\max}}$$

where the pass number takes on the values $P_{\text{number}} = 0, 1, \ldots, 2^{m_{\max}} - 1$. In Figure 1, $m_{\max}$ is 3 and a total of 8 passes are made.

Pervaiz and Baron [23] applied a similar strategy for chemically reactive flows, although a different integration procedure was followed. The authors noted that it was necessary to place an upper limit on the m-values, and that the distribution should be such that no time step should be greater than four times that of the neighbouring cells. In tests conducted by the authors, it was found necessary to limit the maximum m-value used on the basis of stability in one of the test cases reported. As the local time stepping strategy employs a fixed value of Δt throughout a single global time step, which is based on the minimum acceptable value (in terms of the CFL condition) at the start of the integration sequence, the conditions may be such that at some point during the integration, Δt may exceed the current permissible time step a particular cell(s), noting that the permissible time step values will change during update procedure. In such an instance the solution will become numerically unstable. In addition to ensure the information between regions of cells of differing m-values is correctly propagated, a strategy has been developed to create an interface to connect the regions with cells of

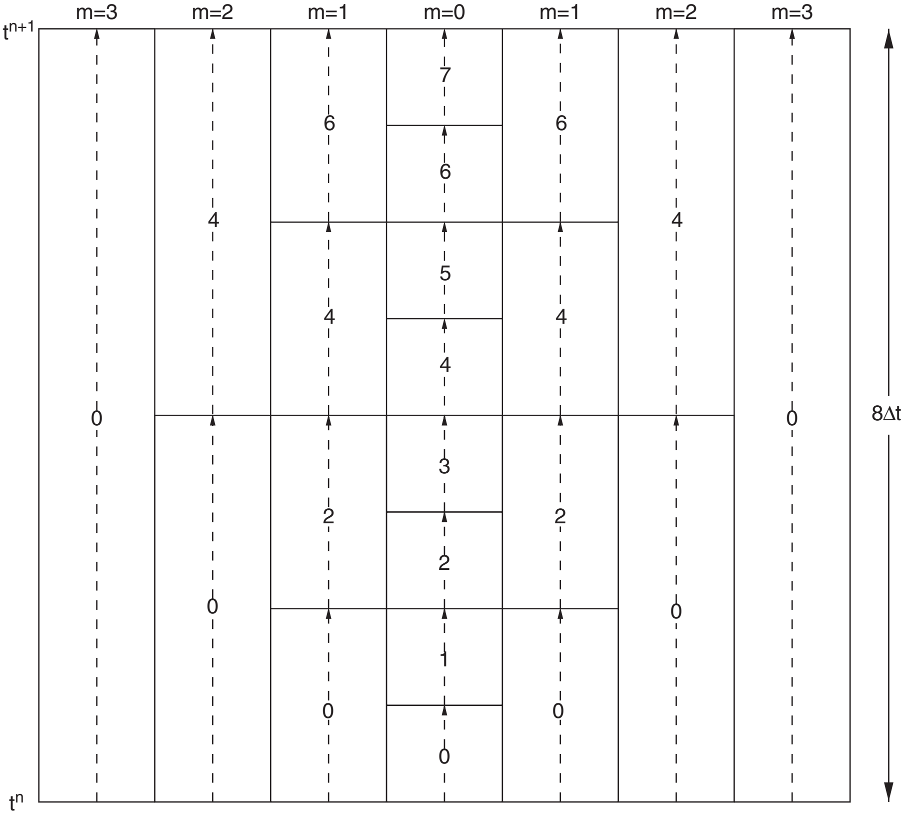

Figure 1. Integration strategy for LTS2.

intermediate m-values. To illustrate the methodology, consider the case where a region of cells with $m=m_1$ adjoins another region of cells with $m=m_2$, where $m_2>m_1$. It is then necessary to mark a number of the m_2 cells as the interface and modify the m-values accordingly. The strategy developed is based on the information travelling at a constant speed over a uniform mesh of grid size Δx. Assuming a CFL number of 1 then

$$\text{Total distance travelled} = \text{number of time steps} \times \Delta x$$

Following this notion, then the number of cells transversed is given by $2^n - 1$. Conceptually, it is then necessary to redefine the m-values for a total of 2^n-1 cells. For example in the case of a region of cells with m-values of 0 adjoining cells with m set to 2, the first neighbouring 2 cell would be marked as a 0 cell, and the next two 2 cells become 1 cells, as illustrated in Figure 2. Furthermore, this can be generalized to say that the first (2^0) m_2 cell should become a m_1 cell, the next two (2^1) cells should become m_1+1 cells, the next four (2^2) cells become m_1+2 cells, and in general the number of cells marked as m_1+b cells is given by 2^b.

The strategy presented above is based on an analogy of information propagating with a constant speed over a uniformly spaced mesh. The procedure outlined was used as a starting

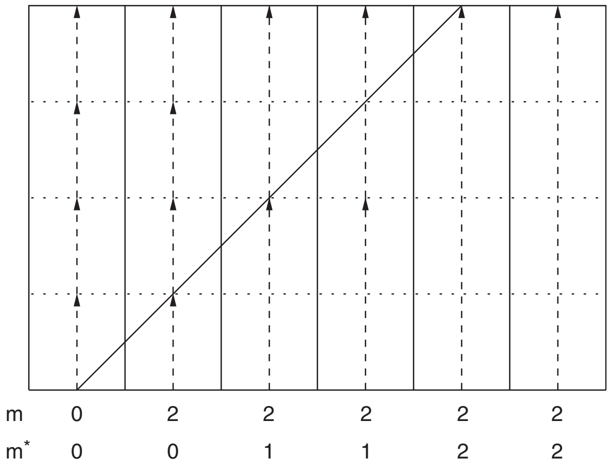

Figure 2. Construction of the interface region for LTS2.

point to develop an algorithm for constructing the interface region on irregular grids. In experiments conducted by the authors [24] it was found that in considering how many cells should be used to construct the interface region, the most crucial factor was the number of cells to be redefined as m_1 cells. It was observed that different results were obtained when the total number of cells used in the interface region remained constant, but the distribution was varied. In practice it was found that given a region of m_1 cells which adjoins a region of m_2 cells, (where $m_2 > m_1$), it was necessary to redefine a minimum of two of the m_2 cells to m_1 cells and the distribution detailed above could be followed for the remainder of the interface region. To allow for a wide range of varying conditions, it was decided to set the first four cells of any interface region to be m_1 cells and follow the trend described above such that a total of $2^{n-1}+3$ cells were modified. In the case of a region of 0 cells adjoining a group of 2 cells, this would mean modifying the first four 2 cells to 0 cells and then changing the next two cells to 1 cells. This strategy was found to give good results in the range of problems considered [24].

In the work being presented, the treatment was adapted to channels with trapezoidal cross-sections and was incorporated into the local time stepping philosophy. In particular, the source term values were frozen in a manner analogous to the frozen flux when the LTS1 method was used.

6. TEST CASES

The test cases presented were all selected as analytic solutions are available. Results from two steady state and two transient cases are shown. In the case of the steady state problems, a run time was established that led to a converged solution for each mesh, and comparative results were obtained using this value.

6.1. S1 flow through a converging/diverging channel

This test case was introduced by García-Navarro *et al.* [25]. The channel is constructed from a series of rectangular cross-sections which form a sinusoidal width contraction and expansion at the centre of the channel. The width decreases from a maximum width of 5 m down to the critical width (b_{crit}) of 3.587 m, which induces critical flow. The width profile is defined by

$$b(x) = \begin{cases} 5, & x \leqslant 100 \\ 5 - (5 - b_{\text{crit}})\cos^2\left[\dfrac{(x-250)}{300}\pi\right], & 100 < x < 400 \\ 5, & x \geqslant 400 \end{cases}$$

The initial conditions are a uniform depth of 2 m and zero discharge everywhere except at the upstream boundary, where the discharge is maintained at $20\,\text{m}^3\,\text{s}^{-1}$ throughout the simulation. A weir boundary condition with a height of 0.1 m is employed at the downstream end of the reach such that the relationship between the velocity and the water depth is

$$u = \tfrac{2}{3}\, C_{\text{d}}(2gH_{\text{w}})^{1/2}$$

where C_{d} is the coefficient of discharge and is taken to be 0.6, and H_{w} is the depth of the water above the level of the weir. An analytic solution is generated by considering the depth profiles obtained by applying Bernoulli's equation to the upstream and downstream sections which are connected via a jump relationship.

6.2. S2 flow through a trapezoidal channel

This test case was proposed by MacDonald *et al.* [26] and forms part of a series of cases considered in which problems with analytical solutions are generated for rough channels with non-flat beds. The channel considered is a 1 km prismatic trapezoidal reach in which the flow is subcritical at the inflow and outflow and contains a hydraulic jump. A discharge of $20\,\text{m}^3\,\text{s}^{-1}$ is imposed at the upstream boundary and the downstream depth is fixed at 1.3449963 m throughout the simulation. The Manning's coefficient is set to 0.02 and the bed slope is given by

$$S_0 = \left(1 - \frac{400[10 + 2\hat{y}(x)]}{g[10 + \hat{y}(x)]^3\,\hat{y}(x)^3}\right)\hat{y}'(x) + 0.16\,\frac{[10 + 2\hat{y}(x)\sqrt{2}]^{4/3}}{[10 + \hat{y}(x)]^{10/3}\,\hat{y}(x)^{10/3}}$$

where $\hat{y}$ is the analytical depth profile

$$\hat{y}(x) = \begin{cases} 0.723449\left[1 - \tanh\left(\dfrac{x}{1000} - \dfrac{3}{10}\right)\right], & 0 \leqslant x \leqslant 300 \\ 0.723449\left\{1 - \dfrac{1}{6}\tanh\left[6\left(\dfrac{x}{1000} - \dfrac{3}{10}\right)\right]\right\}, & 300 < x \leqslant 600 \\ \dfrac{3}{4} + \displaystyle\sum_{k=1}^{3} a_k \exp\left[-20k\left(\dfrac{x}{1000} - \dfrac{3}{5}\right)\right] + \dfrac{3}{5}\exp\left(\dfrac{x}{1000} - 1\right), & 600 < x \leqslant 1000 \end{cases}$$

and

$$\hat{y}'(x)=\begin{cases} -0.723449\times 10^{-3}\mathrm{sech}^2\left(\dfrac{x}{1000}-\dfrac{3}{10}\right), & 0\leqslant x\leqslant 300\\[2ex] -0.723449\times 10^{-3}\,\mathrm{sech}^2\left[6\left(\dfrac{x}{1000}-\dfrac{3}{10}\right)\right], & 300<x\leqslant 600\\[2ex] -\dfrac{1}{50}\sum\limits_{k=1}^{3} ka_k\exp\left[-20k\left(\dfrac{x}{1000}-\dfrac{3}{5}\right)\right]+\dfrac{3}{5000}\exp\left(\dfrac{x}{1000}-1\right), & 600<x\leqslant 1000\end{cases}$$

with $a_1=-0.111051$, $a_2=0.026876$ and $a_3=-0.217567$. The width at the base of the channel is 10 m and the walls have a side slope of unity. A uniform depth of 1.3449963 m is employed as the initial condition together with zero discharge everywhere except at the upstream boundary.

6.3. U1 dam-break

The standard dam-break problem was considered with an upstream to downstream depth ratio of 100:1, whereby the depth to the left of the dam is set at 100 m and the depth to the right at 1 m. The initial velocity is $0\,\mathrm{m\,s^{-1}}$ throughout the channel. The length of the channel is 1000 m and the width is set to 1 m. In this example, transmissive boundary conditions are used at both the upstream and downstream boundaries. As no source terms are present in this problem, the different time step procedures can be contrasted with the source term calculations removed. The results shown correspond to $t=10\,\mathrm{s}$ and the analytical solution is produced using the procedure outline by Glaister [27].

6.4. U2 a tidal flow over steps

The last test case illustrated was proposed at a workshop on dam-break wave simulations and the details were presented in a paper by Zhou *et al.* [28]. The problem considers the flow of a tidal wave over a step where the channel bed is defined as

$$z(x)=\begin{cases} 8 & \text{if } \left|x-\dfrac{1500}{2}\right|\leqslant\dfrac{1500}{8}\\[2ex] 0 & \text{otherwise}\end{cases}$$

with the initial and boundary conditions

$$h(x,0)=H(x)$$

$$u(x,0)=0$$

$$h(0,t)=H(0)+4-4\sin\left[\pi\left(\frac{4t}{86\,400}+\frac{1}{2}\right)\right]$$

$$u(L,t)=0$$

The analytical depth and velocity profiles are

$$h(x,t) = H(x) + 4 - 4\sin\left[\pi\left(\frac{4t}{86\,400} + \frac{1}{2}\right)\right]$$

$$u(x,t) = \frac{(x-L)\pi}{5400h(x,t)}\cos\left[\pi\left(\frac{4t}{86\,400} + \frac{1}{2}\right)\right]$$

where $H(x) = H(0) - z(x)$, $H(0) = 16\,\text{m}$, $L = 1500\,\text{m}$ and the channel is smooth. The results shown are at time $t = 32\,400\,\text{s}$.

7. RESULTS

In order to illustrate the benefits of utilizing local time stepping, solutions were obtained on a symmetric irregular mesh whereby the cells at the beginning and end of the channel were a set number of times larger than the cells at the channel centre. Tests were conducted where this scaling factor varied from 1 (a regular mesh) to 128 [24]. It was observed that as the scale factor increased, so too did the gains from using a local time stepping approach. The results presented in this paper were generated on the 128 mesh. For the S1, S2 and U2 results, the maximum m value utilized by the LTS2 algorithm was 5. In the case of the dam-break problem, U1, the value had to be reduced to 3 on stability grounds. Note also that for the dam-break problem, the source term calculations were removed from the update procedure in order to analyse the run time results on the basis of only the flux calculations.

From Figures 3 and 4, it can be seen that both LTS procedures produce a depth profile which is indistinguishable from the GTS results. In all cases the solutions match the

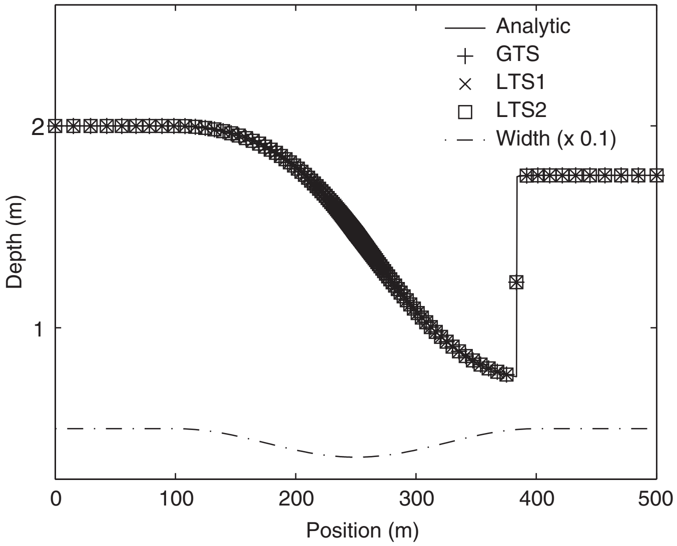

Figure 3. Depth profile for S1.

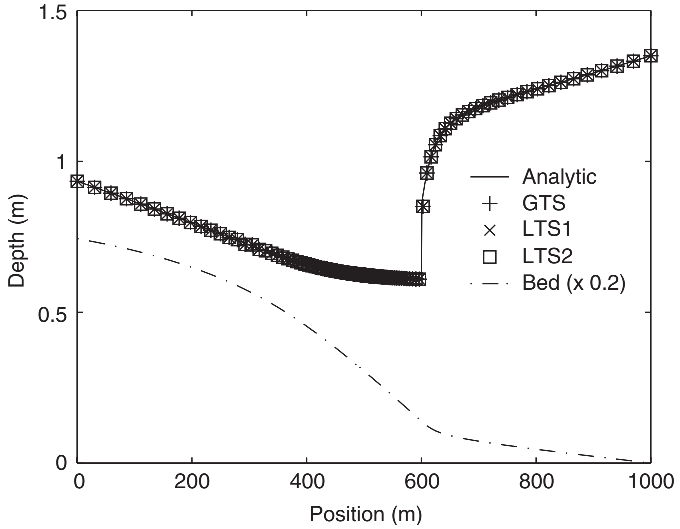

Figure 4. Depth profile for S2.

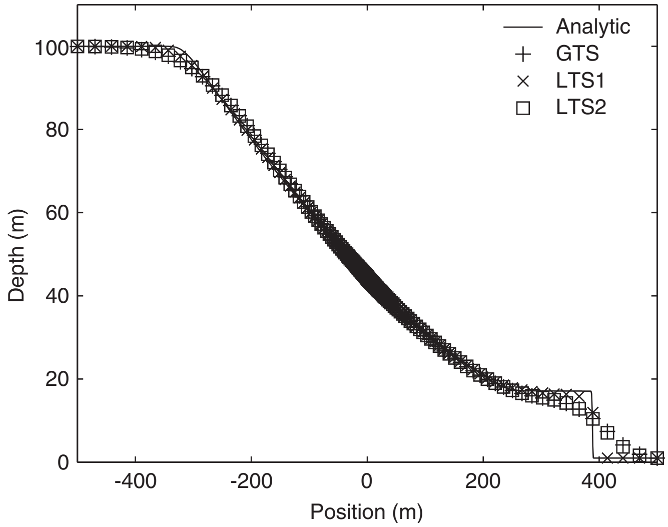

Figure 5. Depth profile for U1.

analytical results and the discontinuity is well resolved. Figure 5 shows the results obtained for the dam-break problem, and the bore region is highlighted in Figure 6. Over most of the domain the numerical solutions are in close agreement to the analytical results. Note that the bore is resolved over a greater number of cells than is generally considered acceptable for a shock capturing scheme, and this is a consequence of the chosen mesh and the implementation of a first-order scheme. Such a combination would not be used in practise but was chosen here to illustrate the differences between global and local time stepping. In this instance the

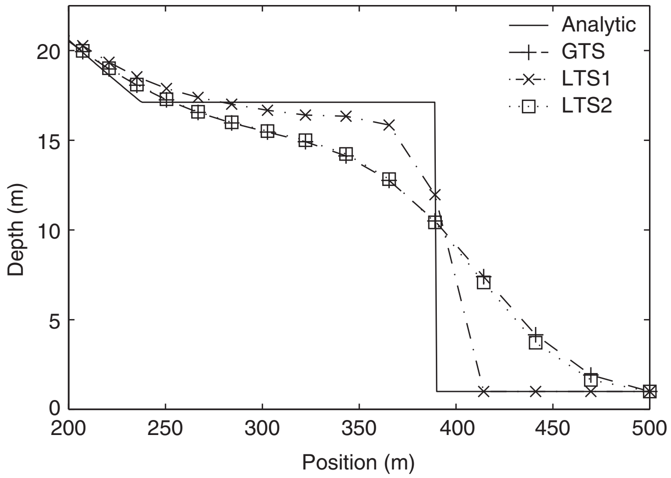

Figure 6. Depth profile for U1—bore region.

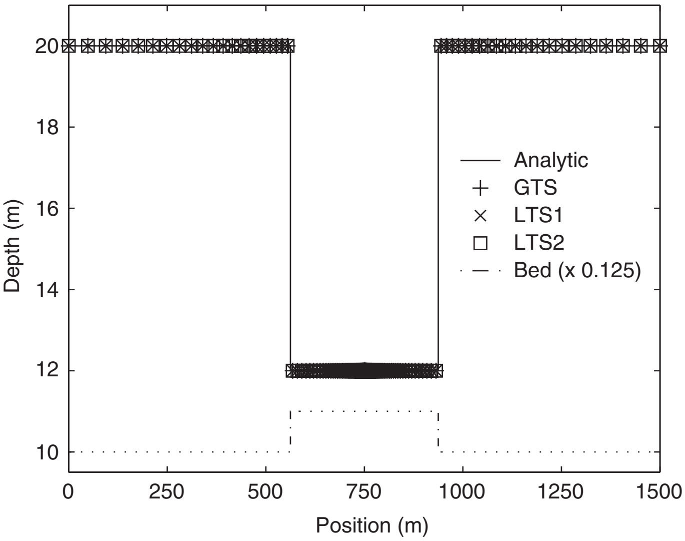

Figure 7. Depth profile for U2.

LTS1 method resolves the bore over the fewest number of cells and the GTS and LTS2 results are fairly similar. The need to restrict the LTS2 algorithm to four levels of temporal embedding ($m_{\max} = 3$) on stability grounds reduced the improvements attainable using the approach. Figure 7 shows the results obtained for problem U2 for which the solution should be a uniform elevation of 20 m. In this instance the results are visually identical and the discontinuities in the profile are well resolved.

Table I summarizes the results of the run time comparison, relative to the GTS calculations. Note that two factors have been considered, the recorded CPU time which was used to

Table I. Comparative run time and flux count efficiency gains using LTS.

	Local time stepping using frozen flux		Local time stepping using full time integration	
Test case	Time reduction (%)	Flux evaluation reduction (%)	Time reduction (%)	Flux evaluation reduction (%)
S1	29.9	69.7	54.0	65.2
S2	28.2	67.6	53.7	65.1
U1	17.0	67.8	44.7	62.0
U2	36.4	66.7	56.0	64.5

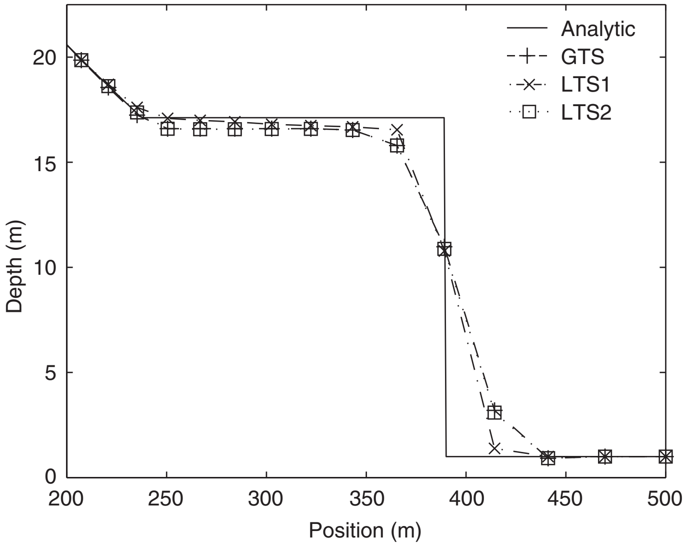

Figure 8. Depth profile for U1 with Superbee—bore region.

obtain the percentage time reduction, and a flux evaluation reduction calculation in which the solver procedures were modified to count the number of flux calculations. This measure gives a reflection of the improvement without the overhead costs of implementing the LTS procedures. It can be seen that there is a marked difference between the recorded time gains for the two procedures but that the flux gain values are quite similar for all of the cases. The time gains are significantly less for the LTS1 method over the LTS2 scheme which is in line with expectations given that the LTS1 method updates the solution at each cell for every time step. It is also noticeable that for both methods, the time gains are significantly less for the dam-break problem where the source term calculations were removed. The data in Table I also suggests that the benefits of employing such a method increase as the complexity of the calculations increase such that the implementation costs reflect an overall lower percentage of

Table II. Comparative run time and flux count efficiency gains using LTS including a higher order scheme and different source term treatment.

	Local time stepping using frozen flux		Local time stepping using full time integration	
Test case	Time reduction (%)	Flux evaluation reduction (%)	Time reduction (%)	Flux evaluation reduction (%)
U1 - Superbee	30.6	68.0	53.6	65.5
S2 - pointwise	22.2	67.3	51.8	65.1
S2 - pointwise and superbee	32.3	65.1	51.7	65.2
U2 - pointwise	30.5	66.7	53.8	64.5

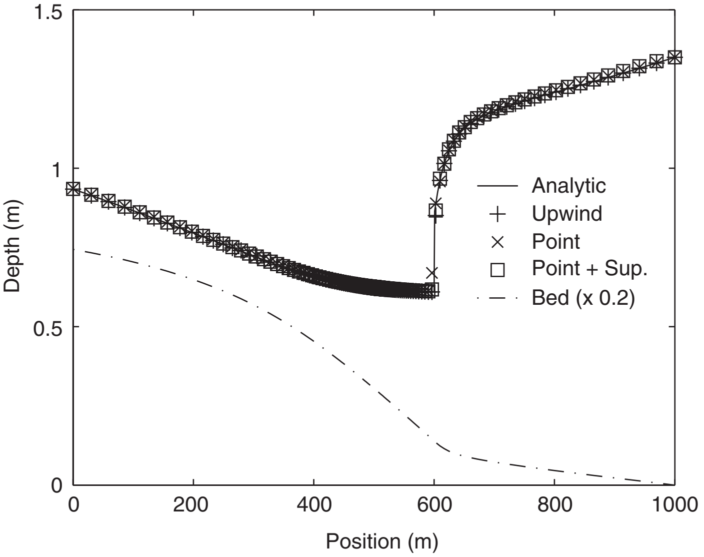

Figure 9. Depth profile for S2—comparison of source terms.

the total run time. This has been verified in another experiment in which the solutions to the dam-break problem were obtained on the same mesh using a second-order version of Roe's scheme utilizing the Superbee flux limiter. The solutions obtained in the region of the bore are shown in Figure 8. In comparison to the first-order results it can be seen that overall the results are more similar and resolve the bore more sharply over much fewer cells. As before, the GTS and LTS2 methods produce similar results whilst the LTS1 profile is marginally better. Table II highlights the fact that the recorded time gains for both LTS approaches are significantly greater in the case of the second-order solver algorithm and more in line with the values noted for the test cases in which source terms were included in the calculation.

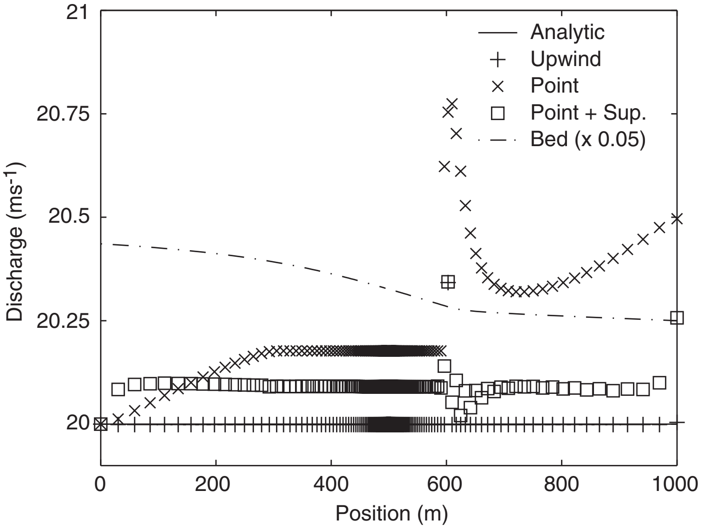

Figure 10. Dishcarge profile for S2—comparison of source terms.

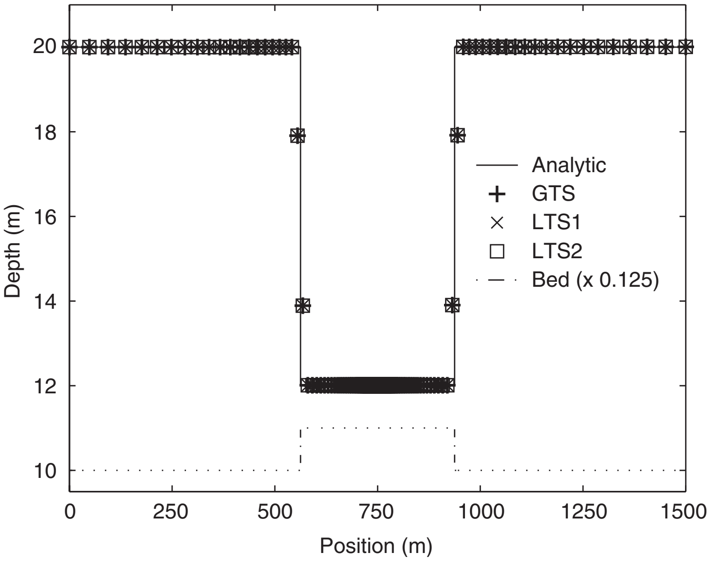

Figure 11. Depth profile for U2—pointwise source terms.

The results presented were all obtained using the upwind treatment for the source terms outlined in Section 4. A comparison has also been made between this treatment and the standard pointwise implementation for a range of problems [24, 29]. It has been observed that the upwind treatment consistently produces better results in flows where source terms are present. This has been particularly evident in steady state flows where, although depth

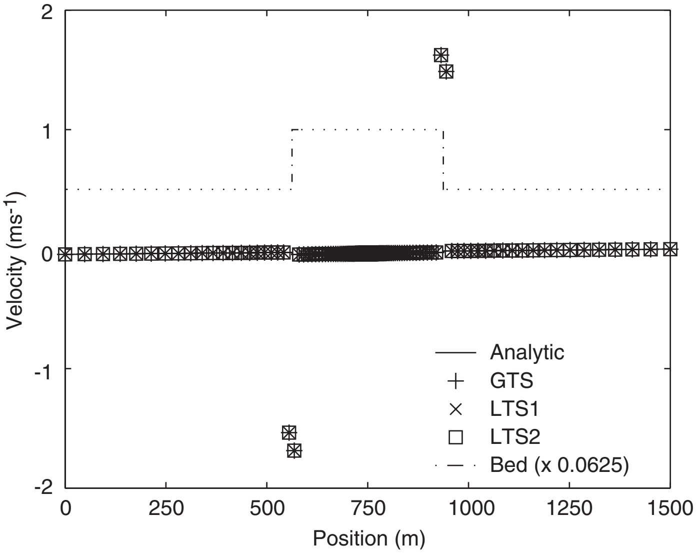

Figure 12. Velocity profile for U2 velocity—pointwise source terms.

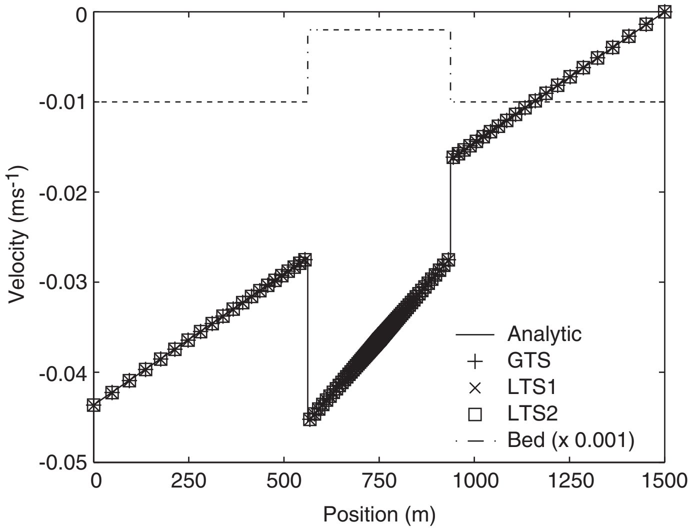

Figure 13. Velocity profile for U2 velocity—upwind source terms.

profile may be satisfactory, discrepancies in the discharge solution (where a constant value should be obtained) can occur when a pointwise treatment is used. In comparing the upwind results with those obtained when a pointwise treatment is used together with a second-order version of Roe's scheme (through using the Superbee flux limiter), the use of a limiter function reduces the perturbations from the uniform value though the results are still generally not as good as those produced by the upwind treatment. This is illustrated in Figures 9 and 10, which show the depth and discharge profiles obtained for test case S2 using the upwind source terms, pointwise source terms and pointwise source terms in conjunction with the Superbee limiter. Note only the GTS results are shown for clarity, as for this case the LTS1 and LTS2 results with the respective source term and flux functions were indistinguishable from the GTS results. This is also illustrated for the unsteady case U2 in Figures 11 and 12, where the depth and velocity profiles obtained using a pointwise approach are shown. Note that GTS, LTS1 and LTS2 results are identical. Figure 12 can be compared to Figure 13, which shows the velocity results when the upwind treatment is used. Erroneous spikes can be seen in the results obtained using the pointwise treatment which are not present in the upwind results (note the change in scale between the figures). In this instance it can be seen that the upwind treatment gives better resolution of the flow characteristics over the steps.

8. CONCLUSIONS

Two LTS strategies developed for the Euler equations have been presented and applied to the shallow water equations of open channel flow. The strategies were implemented within an explicit finite volume framework using Roe's scheme which was used in conjunction with an upwind treatment for the source terms. Further consideration was given to the implementation of the two LTS strategies by developing interface treatments to ensure that information was correctly propagated between cells at different temporal levels. A comparison was performed between conventional global time stepping and using an LTS approach though a series of test cases. The results illustrate that using LTS can lead to a reduction in run times for both steady and transient problems. It was also illustrated how LTS can improve solution accuracy in the vicinity of bores, which was demonstrated for the dam-break problem. In addition it was highlighted how using an upwind source term treatment can be beneficial for flows dominated by the geometry.

In view of these results the LTS strategies are believed to be of use in reducing run times and increasing accuracy. Research is currently underway to apply these techniques to the two-dimensional shallow water equations and for these the LTS is expected to offer more significant savings as the regions requiring small time steps will represent an even smaller proportion of the total area than in one dimension. Commercial software used in this field often adopts the Preismann or Abbott-Ionescu schemes [1], rather than the Riemann-based one presented here. These formulations are implicit, conceptually simple and allow for the incorporation of hydraulic units such as sluices, bridges, weirs, etc. However, the Riemann-based approach has the advantage of dealing correctly with transitions in the flow. The Priesmann and Abbott-Ionescu schemes have difficulty with this [30] and commercial codes use a variety of fixes to deal with this problem, often by neglecting the non-linear term as the Froude number approaches unity. The results of the research presented here still have implications for commercial software either through the implementation of LTS with the Preismann or Abbott-Ionescu schemes or the development of Riemann-based methods for irregular geometries [31] and hydraulic structures that would allow their application in practical situations.

ACKNOWLEDGEMENTS

Thanks are due to Dr Pilar García-Navarro for helpful discussions during the course of this work. Financial support from the University of Nottingham was also gratefully appreciated.

REFERENCES

1. Cunge J, Holly F, Verwey A. *Practical Aspects of Computational River Hydraulics.* Pitman: London, 1980.
2. Tan W. *Shallow Water Hydrodynamics.* Elsevier: Amsterdam, 1992.
3. Delis AI, Skeels CP, Ryrie SC. Evaluation of some approximate Riemann solvers for transient open channel flows. *Journal of Hydraulic Research* 2000; **38**(3):217–231.
4. Erduran KS, Kutija V, Hewett CJM. Performance of finite volume solutions to the shallow water equations with shock capturing schemes. *International Journal for Numerical Methods in Fluids* 2002; **40**:1237–1273.
5. Glaister P. Approximate Riemann solutions of the shallow water equations. *Journal of Hydraulic Research* 1988; **26**(3):293–306.
6. Alcrudo F, García-Navarro P, Savirón J-M. Flux difference splitting for 1D open channel flow equations. *International Journal for Numerical Methods in Fluids* 1992; **14**:1009–1018.
7. Ambrosi D. Approximation of shallow water equations by Roe's Riemann solver. *International Journal for Numerical Methods in Fluids* 1995; **20**:157–168.
8. Jha AK, Akiyama J, Ura M. First- and second-order flux difference splitting schemes for dam-break problem. *Journal of Hydraulic Engineering* 1995; **121**(12):877–884.
9. Goutal N, Maurel F. A finite volume solver for 1D shallow water equations applied to an actual river. *International Journal for Numerical Methods in Fluids* 2002; **38**:1–19.
10. Goutal N. A rough scheme for dam break wave simulation. In *River Flow 2002*, Bousmar D, Zech Y (eds). Swets & Zeitlinger: Lisse, 2002; 475–480.
11. Roe PL. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.
12. García-Navarro P, Vázquez-Cendón ME. On numerical treatment of the source terms in shallow water equations. *Computers and Fluids* 2000; **29**:951–979.
13. Harten A. Self adjusting grid methods for one-dimensional hyperbolic conservation laws. *Journal of Computational Physics* 1983; **50**:235–269.
14. Sweby PK. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis* 1984; **21**(5):995–1011.
15. LeVeque RJ. Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. *Journal of Computational Physics* 1998; **146**:346–365.
16. Bermúdez A, Vázquez ME. Upwind methods for hyperbolic conservation laws with source terms. *Computers and Fluids* 1994; **8**:1049–1071.
17. Rebollo TC, Nieto EDF, Mármol MG. A flux-splitting solver for shallow water equations with source terms. *International Journal for Numerical Methods in Fluids* 2003; **42**:23–55.
18. Vázquez-Cendón ME. Improved treatment of source terms in upwind schemes for the shallow water equations in channels with irregular geometry. *Journal of Computational Physics* 1999; **148**:497–526.
19. Burguete J, García-Navarro P. Efficient construction of high-resolution TVD conservative schemes for equations with source terms: application to shallow water flows. *International Journal for Numerical Methods in Fluids* 2001; **37**:209–248.
20. Zhang XD, Trépanier J-Y, Reggio M, Camarero R. A new local time stepping approach for the unsteady Euler equations. *AIAA Paper 94-0525*, 1994.
21. Zhang XD, Trépanier J-Y, Reggio M, Camarero R. Time-accurate local time stepping method based on flux updating. *AIAA Journal* 1994; **32**(9):1926–1928.
22. Kleb WL, Batina JT, Williams MH. Temporal adaptive Euler/Navier–Stokes algorithm involving unstructured dynamic meshes. *AIAA Journal* 1992; **30**(8):1980–1985.
23. Pervaiz MM, Baron JR. Temporal and spatial adaptive algorithm for reacting flows. *Communications in Applied Numerical Methods* 1988; **4**:97–111.
24. Crossley AJ. Accurate and efficient numerical solutions for the Saint Venant equations of open channel flow. *Ph.D. Thesis*, University of Nottingham, U.K., 1999.
25. García-Navarro P, Alcrudo F, Savirón JM. 1-D open channel flow simulation using TVD-McCormack scheme. *Journal of Hydraulic Engineering* 1992; **118**(10):1359–1372.
26. MacDonald I, Baines MJ, Nichols NK, Samuels PG. Analytic benchmark solutions for open-channel flows. *Journal of Hydraulic Engineering* 1997; **123**(11):1041–1045.
27. Glaister P. Difference schemes for the shallow water equations. *Numerical Analysis Report 9/87*, University of Reading, U.K., 1987.

28. Zhou JG, Causon DM, Ingram DM, Mingham CG. Numerical solutions of the shallow water equations with discontinuous bed topography. *International Journal for Numerical Methods in Fluids* 2002; **38**:769–788.
29. Crossley AJ, Wright NG, Whitlow CD. Local time stepping for modelling open channel flows. *Journal of Hydraulic Engineering* 2003; **129**(6):455–462.
30. Crowder R, Pepper A, Whitlow C. Testing and benchmarking 1D hydraulic river modelling packages. *38th Flood and Coastal Management Conference*, DEFRA: London, 2003, 09.1.1–09.1.11.
31. García-Navarro P, Fras A, Villanueva I. Dam-break simulations: some results for one-dimensional models of real cases. *Journal of Hydrology* 1999; **216**(3–4):227–247.