Local time stepping applied to mixed flow modelling

Local time stepping applied to mixed flow modelling

SUSANNA DAZZI, PhD Student, *Department of Civil and Environmental Engineering and Architecture, University of Parma, Parco Area delle Scienze 181/A, 43124 Parma, Italy*

Email: susanna.dazzi@studenti.unipr.it (author for correspondence)

ANDREA MARANZONI, Research Assistant, *Department of Civil and Environmental Engineering and Architecture, University of Parma, Parco Area delle Scienze 181/A, 43124 Parma, Italy*

Email: andrea.maranzoni@unipr.it

PAOLO MIGNOSA, Full Professor, *Department of Civil and Environmental Engineering and Architecture, University of Parma, Parco Area delle Scienze 181/A, 43124 Parma, Italy*

Email: paolo.mignosa@unipr.it

Local time stepping applied to mixed flow modelling

ABSTRACT

Mixed flows in closed conduits are characterized by waves whose celerity values lie within a range of up to two orders of magnitude due to the simultaneous occurrence of free-surface and pressurized flow. If an explicit numerical scheme is used to simulate these phenomena, the time step necessary to guarantee stability is considerably restricted by pressure wave celerity, and thus the computational efficiency is reduced. In order to address this specific problem this paper proposes the application of the local time stepping strategy to a finite-volume scheme for mixed flow modelling. The Preissmann slot approach is adopted to handle pressurization. The results of the simulations of several tests show that local time stepping reduces run time significantly, compared to the conventional global time stepping, especially when only a small region of the domain is surcharged. Accuracy and mass conservation are not impaired. Besides, accuracy slightly improves in the free-surface region of the flow field.

Keywords: 1D shallow water equations; finite volume; local time stepping; mixed flow; numerical modelling; Preissmann slot

1 Introduction

Mixed flows take place in sewers or other closed pipe systems when transitions from free-surface to pressurized flow and vice versa occur as a result of variations in inflow discharge or operations on pumps and gates. The correct modelling of this type of flow is important for both the design and maintenance of these hydraulic structures. A number of studies have therefore focused on this topic from both experimental and numerical point of view (e.g. Wiggert, 1972; Cardle, Song, & Yuan, 1989; Vasconcelos & Wright, 2005; Bourdarias & Gerbi, 2007; León, Ghidaoui, Schmidt, & Garcia, 2009; Ferreri, Ciraolo, & Lo Re, 2014; Aureli, Dazzi, Maranzoni, & Mignosa, 2015).

The numerical modelling of mixed flows is particularly challenging since the two flow regimes, although governed by the same basic physical principles, are traditionally described by two different sets of governing equations and, in addition, are characterized by very different wave celerity values within a range of up to two orders of magnitude. A variety of mathematical approaches is available in literature to address the former problem (for a general review, see Bousso, Daynou, & Fuamba, 2013); these include the Preissmann slot technique (Cunge & Wegner, 1964), the Two-component Pressure Approach (TPA) (Vasconcelos, Wright, & Roe, 2006), interface tracking models (e.g. Politano, Odgaard, & Klecan, 2007), and dual models (Bourdarias & Gerbi, 2007; León, Ghidaoui, Schmidt, & Garcia, 2010). However, the sharp jump in wave celerity at the transition between the two

different flow regimes can induce spurious post-shock oscillations in the numerical solution when a shock-capturing scheme is adopted (Vasconcelos, Wright, & Roe, 2009).

A further consequence of the occurrence of wave celerity values varying considerably in the flow field is the fact that, if an explicit scheme is adopted, the pressurized cells (characterized by higher values of wave celerity) will determine the computational time step to be used to update the solution in the whole domain, according to the Courant-Friedrichs-Lewy (CFL) stability condition. This restriction has two main negative effects: firstly, the computational effort increases greatly compared to free-surface flow simulations; secondly, the numerical solution in the free-surface portion of the domain is updated with a much smaller time step than its maximum allowable, resulting in an expected reduction in solution accuracy (Zhang, Trépanier, Reggio, & Camarero, 1994). The problem is similar to the one encountered in free-surface flow applications when the mesh is locally refined, in which case the smallest grid cell will dictate the size of the time step, thereby drastically reducing model efficiency, especially if the mesh refinement is restricted to only a small portion of the entire computational domain.

The local time stepping (LTS) strategy (or temporal adaptivity) has been proposed in literature since the 1980s (Osher & Sanders, 1983) to increase computational efficiency when a non-uniform spatial discretization is used. It is based on the idea of updating each computational cell with a local time step as close as possible to its maximum allowable to satisfy the stability criterion, in contrast with the global time stepping (GTS) strategy, which advances the solution by using the minimum permissible time step in the whole domain. Most of the LTS schemes available are based on local time steps defined as integer multiples of a common reference time step (e.g. Kleb, Batina, & Williams, 1992). However, some authors presented LTS algorithms where each cell can be advanced at its own time step without introducing any defined time step ratio (e.g. Dumbser, 2014). LTS methods have been satisfactorily applied to Euler and Navier-Stokes equations (e.g. Pervaiz & Baron, 1988; Kleb et al., 1992; Coquel, Nguyen, Postel, & Tran, 2010), but also to other hyperbolic systems (e.g. Dumbser, Käser, & Toro, 2007; Grote & Mitkova, 2010). Moreover, LTS schemes have also been adopted in combination with adaptive mesh refinement (AMR) techniques (e.g. Berger & LeVeque, 1998; Dumbser, Zanotti, Hidalgo, & Balsara, 2013).

Applications of the LTS strategy to the one-dimensional (1D) and two-dimensional (2D) shallow water equations can also be found in literature (e.g. Crossley & Wright, 2005; Sanders, 2008; Trahan & Dawson, 2012; Kesserwani & Liang, 2014; Maleki & Khan, 2015). The results reported in these works show that considerable saving in computational time can be achieved without impairing model accuracy when a non-uniform mesh in used. However, despite the fact that LTS methods can be expected to be profitable not only when a local mesh refinement is employed, but also in applications involving significantly variable wave

celerity, to the best of the authors' knowledge the applicability of the LTS methodology to unsteady mixed flows has not been thoroughly assessed to date. This strategy can actually be very useful and effective in practical applications, especially in those cases where only a small region of the domain is pressurized in an otherwise fully free surface flow field. This occurs, for example, when a single branch of a pipe network is surcharged, or when a culvert, a bridge or an inverted siphon inserted in an open channel system are partially or totally surcharged. Indeed, in the context of pipe network modeling, Sanders & Bradford (2011) adopted a TPA model in which the computational cells within each pipe were updated by using the minimum time step permissible in that pipe, while a larger global time step was used to synchronize all pipes and junctions.

This paper presents the application of a local time stepping method modified from previous works (Kleb et al., 1992; Crossley & Wright, 2005), to the numerical modelling of 1D unsteady mixed flows. Of the different approaches available to simulate mixed flows, the Preissmann slot model is adopted here due to its intrinsic simplicity, which has fostered its wide use in both literature (e.g. Capart, Sillen, & Zech, 1997; Trajkovic, Ivetic, Calomino, & D'Ippolito, 1999; León et al., 2009) and commercial software packages (e.g. DHI, 2002; HEC, 2010). The basic idea is to add a narrow slot above the pipe crown to enable the model to artificially preserve the free-surface regime even when the water surface reaches the pipe crown. Thus, de Saint-Venant equations can be used throughout the whole computational domain. Furthermore, the 'negative' slot concept recently introduced by Kerger, Archambeau, Erpicum, Dewals, & Pirotton (2011b) overcame the main limitation of the model, i.e. its inability to handle sub-atmospheric pressurized flows. The governing equations are solved by a Godunov-type finite volume shock-capturing scheme coupled with an LTS method. Several simulations of idealized numerical tests, of an experimental test, and of a simplified field-scale test are performed to assess the effectiveness and robustness of the LTS method in mixed flow applications and to evaluate the improvement achievable in model efficiency and solution accuracy in comparison with the conventional GTS scheme.

The paper is structured as follows. Section 2 describes the numerical model and the LTS strategy. In Section 3 the results of the numerical tests are presented and discussed. Conclusions are drawn in the last section.

2 Numerical model

2.1 Preissmann slot model

The Preissmann slot approach is based on the de Saint-Venant equations for open channel flows. For a prismatic channel, the system of equations can be written in vector conservative

form as (e.g. Cunge, Holly, & Verwey, 1980):

$$\frac{\partial \mathbf{U}}{\partial t}+\frac{\partial \mathbf{F}}{\partial x}=\mathbf{S},$$

$$\mathbf{U}=\begin{bmatrix} A \\ Q \end{bmatrix}, \quad \mathbf{F}=\begin{bmatrix} Q \\ \dfrac{Q^2}{A}+gI_1 \end{bmatrix}, \quad \mathbf{S}=\begin{bmatrix} 0 \\ gA\left(S_0-S_f\right) \end{bmatrix}, \tag{1}$$

where Q is the flow discharge, A is the flow area, x is the distance along the channel (positive downstream), t is time, g is acceleration due to gravity, $S_0 = -\mathrm{d}z_b/\mathrm{d}x$ is the bottom slope (z_b is the bottom elevation), and S_f is the friction slope, which is evaluated here according to the Manning formula as $S_f = n_m^2 u|u| R_h^{-4/3}$, where n_m is Manning's roughness coefficient, u is the cross-sectional average velocity, and R_h is the hydraulic radius. In Eq. (1), the hydrostatic pressure force term I_1 is defined as: $I_1 = \int_0^h (h-\eta)\sigma(h,\eta)\mathrm{d}\eta$, where h is the water depth and $\sigma(x, \eta)$ is the cross-sectional width at an elevation η above the bottom.

If a fictitious narrow vertical slot is added above the top of the closed conduit according to the Preissmann slot model (Cunge & Wegner, 1964), Eq. (1) can also describe pressurized flow. Water depth exceeding the pipe crown represents the pressure head (with reference to the bottom) of an equivalent surcharged flow. The slot width must be narrow enough for the water volume stored in the slot to be negligible and the mass balance not to be significantly affected (Cunge et al., 1980). Furthermore, in order to model the physics of the phenomenon correctly, the slot width T_s should be set according to the relation:

$$T_s = \frac{gA}{a^2} \tag{2}$$

where a is the pressure wave celerity which, in practical applications, depends on many factors, including fluid compressibility, pipe deformability, and also air entrainment. In general, if the physical values of pressure wave celerity need to be modelled, the constraint stated by Eq. (2) leads to the estimate of very narrow slots, which often induce numerical oscillations at flow transitions. To overcome this problem, some authors (e.g. Trajkovic et al., 1999; DHI, 2002) set the slot width at approximately 1−2% of the pipe diameter, accepting a significant reduction in the pressure wave celerity value, but still obtaining satisfactory results in the simulation of experimental test cases.

2.2 Numerical scheme

Equation (1) is solved by using an explicit Godunov-type finite volume numerical scheme coupled with a LTS strategy. Hence, the following algorithm is adopted to advance in time the solution in each computational cell i (with size Δx_i):

$$\mathbf{U}_i^{k+1} = \mathbf{U}_i^k - \frac{\Delta t_i}{\Delta x_i}\left(\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}\right) + \Delta t_i\, \mathbf{S}_i, \tag{3}$$

where $\mathbf{U}_i$ and $\mathbf{S}_i$ are the cell-averaged values of the vector of conserved variables and of the source term respectively, while $\mathbf{F}$ denotes the vector of numerical fluxes at the interface between neighbouring cells. Superscripts k and $k+1$ refer to current and updated time levels respectively, and Δt_i is the local permissible time step, which must be computed according to the CFL stability relation:

$$\Delta t_i = \mathrm{Cr}\,\frac{\Delta x_i}{|u_i| + c_i}, \tag{4}$$

where Cr is the Courant number (≤ 1). If a GTS strategy were applied, all cells would be updated with the same time step Δt, selected as the minimum allowable in the whole domain:

$$\Delta t = \min_i\left(\Delta t_i\right). \tag{5}$$

Numerical fluxes are estimated by using the HLL approximate Riemann solver (Toro, 2001). The left- and right-hand states at each cell edge are evaluated through the first-order accurate reconstruction procedure suggested by Sanders & Bradford (2011), while wave speeds are estimated following León et al. (2009). The source term treatment is accomplished according to Sanders & Bradford (2011), thus ensuring that the *C*-property (which is the ability of the scheme to preserve a stationary condition in a non-horizontal pipe) is satisfied and that unphysical flow inversions induced by flow resistance are prevented. Boundary conditions are implemented on the basis of the ghost cell approach (LeVeque, 2002).

The effectiveness of this numerical scheme with GTS in mixed flow modelling was previously validated by comparison with experimental data (Aureli et al., 2015). The LTS update procedure adopted here is explained in detail in the next subsection. A uniform mesh is assumed in the following for the sake of simplicity.

2.3 Local time stepping

The LTS technique proposed by Kleb et al. (1992), and then successfully applied to shallow water equations by different authors (e.g. Crossley & Wright, 2005; Krámer & Józsa, 2007; Sanders, 2008), is adopted in this paper with modifications. According to this method, the local time step Δt_i in each cell is set as a power of two multiples of the minimum time step Δt throughout the computational domain:

$$\Delta t_i = 2^{m_i}\,\Delta t, \tag{6}$$

where m_i is an integer parameter computed for each cell as:

$$m_i = \mathrm{int}\left(\frac{\log(\Delta t_{i,\mathrm{CFL}} / \Delta t)}{\log(2)} \right), \tag{7}$$

in which $\Delta t_{i,\mathrm{CFL}}$ denotes the maximum time step allowable in cell i to satisfy the CFL condition.

Following Sanders (2008), the cell edges are also marked with a specific parameter m_f which is defined as $m_{f,\,i+1/2} = \min(m_i, m_{i+1})$, and a new integer parameter m^* is associated to each cell based on the following criterion: $m_i^* = \min(m_{f,\,i-1/2}, m_{f,\,i+1/2})$; the local time step is then redefined according to Eq. (6) on the basis of the parameter m_i^*. Finally, the value $m_{\max} = \max_i(m_i^*)$ is computed, and the maximum time step is defined as $\Delta t_{\max} = 2^{m_{\max}}\,\Delta t$.

The time advancing of the computational cells during each update from the common time level t^n to the new common time level $t^{n+1} = t^n + \Delta t_{\max}$ is asynchronous: actually, the updating cycle from t^n to t^{n+1} is divided into a sequence of $2^{m_{\max}}$ intermediate steps (identified with the step number $p = 0, 1, \ldots, 2^{m_{\max}} - 1$); during each of these steps, only the cells for which $(p+1)$ is a multiple of 2^{m^*} are advanced via Eq. (3) on the basis of their local time step Δt_i. Consequently, in the overall updating cycle, each computational cell is advanced by $2^{m_{\max} - m^*}$ steps. In particular, cells marked with $m^* = m_{\max}$ are advanced by a single time step of size $\Delta t_{\max}$, while those labelled with $m^* = 0$ are updated by $2^{m_{\max}}$ unit steps of size Δt. The update algorithm is sketched in Fig. 1 for $m_{\max} = 2$, corresponding to a sequence of four intermediate steps. Black dots indicate the known states at the beginning of each intermediate step, while white and black arrows represent cell updates and intercell flux evaluations performed during the current step, respectively.

The LTS algorithm for the flux evaluation at the cell boundaries follows the procedure proposed by Sanders (2008). Accordingly, at each intermediate step, numerical fluxes are evaluated only at those intercells labelled with an m_f value for which p is an integer multiple of 2^{m_f}. In particular, at the first intermediate step ($p = 0$) fluxes are computed at all cell boundaries and stored in a dedicated array; fluxes at intercells marked with $m_f = 0$ are re-evaluated at all intermediate steps (see Fig. 1).

Special attention must be devoted to the case of neighboring cells characterized by different local time steps (see, for example, cells $i-1$ and i in Fig. 1). Although the permissible time step of cell i is $2\Delta t$, the update procedure based on the parameter m^* dictates that this cell is advanced by two distinct intermediate steps of size Δt. Only the flux at intercell $i-1/2$ is re-computed during the second step ($p = 1$), while the flux at $i+1/2$ is kept unchanged during the two steps $p = 0$ and $p = 1$. The added intermediate state is represented with dashed lines in Fig. 1. The use of m^* instead of m to identify which cells must be advanced during each intermediate step ensures that flux evaluation is carried out correctly at

the edges between adjacent cells characterized by different values of the parameter m. It is to be noticed that the original method (Kleb et al., 1992; Crossley & Wright, 2005) employed a different strategy based on an interpolation procedure at interfaces, and a correction was necessary to ensure conservation properties (Tan, Zhang, Huang, & Tang, 2004; Krámer & Józsa, 2007). The update procedure adopted here instead is intrinsically conservative (Sanders, 2008).

Some authors suggest that in order to increase the robustness and accuracy of the method the jump in the m value between neighbouring cells should not exceed one (e.g. Krámer & Józsa, 2007). Moreover, the correct propagation of information at interfaces between cells updated with different time steps requires that a suitable interface region, where m varies gradually, separates areas characterized by highly different values of m (Crossley & Wright, 2005). In applications with non-uniform grids, this requirement is often satisfied thanks to an appropriate mesh design, which should guarantee smooth transitions from small to large cells. In the context of mixed flows, instead, transitions from free-surface to pressurized flow induce abrupt changes in the time step size permissible, thus values of m must be artificially re-assigned in a few cells near the flow regime transition prior to updating in order to ensure the correct propagation of travelling waves. The procedure outlined in Crossley & Wright (2005) to define this interface region is adopted in this work. In short, if neighbouring regions are characterized by two different values m_1 and m_2 (with $\Delta m = m_2 - m_1 > 0$), the interface is composed of $2^{\Delta m} + 3$ cells. In particular, value m_1 is assigned to the first four cells originally belonging to the m_2 region; then, a gradual transition is created by assigning value $m_1 + l$ to 2^l cells, with l being an integer counter ranging from 0 to $\Delta m - 1$. For example, at the interface between $m_1 = 0$ and $m_2 = 3$ (which could reasonably be the case of a flow regime transition), the original m vector […, 0, 0, 0, **3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3**, 3, 3, …] becomes […, 0, 0, 0, **0, 0, 0, 0, 0, 1, 1, 2, 2, 2, 2,** 3, 3, …], where the numbers in bold type refer to the cells belonging to the interface zone.

A possible source of numerical instability is due to the fact that, throughout a global update from t^n to t^{n+1}, the local time step in each cell is kept constant, while, after a partial update, wave speed may increase in some cells, with consequent reduction of the permissible time step, and the CFL condition might then be no longer fulfilled in these cells. Possible strategies to face this problem include the use of a slightly reduced value of the Courant number (e.g., in Sanders (2008) Cr is set at 0.8), and the artificial limitation of the maximum value of m (see, for example, Crossley & Wright (2005) and Krámer & Józsa (2007), where $m_{max} = 3$). A different strategy is implemented in this work. After each intermediate update from t^k to t^{k+1}, the local values assumed by the Courant number in the next intermediate time step are calculated as:

$$\mathrm{Cr}_i^{k+1} = \Delta t_i \frac{\left|u_i^{k+1}\right| + c_i^{k+1}}{\Delta x_i}. \tag{8}$$

If the Courant number exceeds unity somewhere, the usual update procedure is interrupted, and all cells are advanced to the current intermediate time level t^{k+1} (see Fig. 2). After this synchronization, local time steps are re-evaluated in all cells and a new updating cycle can start. During the simulation of the laboratory test presented in the next section and used for validation, it was observed that stopping the update procedure and re-evaluating the local time steps improved the robustness of the model, unlike limiting the maximum time step. This strategy actually proves to be effective when some cells become pressurized starting from a free-surface condition; in this case celerity values increase suddenly, and instabilities may arise (and are further compounded by the well-known spurious oscillations near sudden flow regime transitions) if local time steps are not re-evaluated correctly.

3 Numerical tests

This section shows the results of some numerical simulations concerning different test cases, and compares the LTS and GTS schemes in terms of accuracy and CPU time. First, the simple case of water at rest in a pipe is simulated; then, idealized test cases with analytical solution and a simple test case of gradual pipe filling are considered in order to assess the effectiveness and efficiency of the two time stepping techniques; finally, a laboratory test and a real-field test are analyzed as applications of practical significance. All simulations were performed on a workstation with a 2.67 GHz Intel Core i7 processor and 8 GB RAM.

3.1 Test cases with water at rest

The first test case concerns a 100 m-long sloped pipe with diameter D = 1 m. Both pipe ends are closed, and the pipe is partially filled with still water up to a certain level. Different pipe slopes ranging from +0.5% to +5% (positive downwards) are considered. The water depth at the upstream end (x = 0 m) is assumed equal to 0.5 m for all slopes. Consequently, the upstream portion of the pipe is in free-surface conditions, while the downstream part is pressurized; the length of this latter portion increases with the pipe slope, and the pressure head at the downstream end of the pipe ranges between 1.0 m and 5.5 m.

The computational domain is discretized with N = 200 cells of uniform size Δx = 0.5 m, and the Courant number is set at 0.9. Dimensionless slot widths T_s/D equal to 0.01 and 0.001 are assumed, corresponding to 27 m/s and 87 m/s pressure wave celerity values, and to ratios of maximum to minimum local time step equal to 8 (i.e. $m_{\max}$ = 3) and 32 (i.e. $m_{\max}$ = 5), respectively. In both cases the solution is advanced for 3,000 time steps in the GTS

simulation and for the corresponding duration in the LTS simulation (approximately 50 s for $T_s/D = 0.01$, and 16 s for $T_s/D = 0.001$).

Both LTS and GTS schemes are able to preserve the static condition, except for round-off and truncation errors. As an example, at the end of the simulation with the pipe slope equal to +1%, the maximum discharge is in the order of 10^{-14} m^3/s in both cases.

Table 1 reports the speed-up factors calculated for simulations carried out with different pipe slopes and with both values of the slot width. The speed-up factor is the ratio of the execution time of the GTS simulation to that of the LTS simulation, thus representing an estimate of the performance improvement in terms of computational effort that the LTS scheme can achieve compared to the GTS scheme. The speed-up factor decreases with the increase in number of pressurized cells and, theoretically, tends to unity when the whole computational domain is pressurized. Moreover, the narrower the slot is, the greater the value of the speed-up factor is. These trends can be explained by estimating the achievable speed-up, as suggested by Kleb et al. (1992). If $\alpha\,(0 < \alpha \leq 1)$ is the percentage of pressurized cells (which is known *a priori* and does not change in time in this simple application), then a dimensionless quantification of the computational loads associated with the GTS and LTS schemes (assumed proportional to the total number of updates carried out until the computational time t) and the achievable speed-up are roughly given by:

$$W_{\mathrm{GTS}} \propto \sum_{i=1}^{N} \frac{t}{\Delta t} = N \frac{t}{\Delta t}, \qquad W_{\mathrm{LTS}} \propto \sum_{i=1}^{N} \frac{t}{\Delta t_i} = \alpha\, N \frac{t}{\Delta t} + (1-\alpha) N \frac{t}{2^{m_{\max}} \Delta t}, \tag{9a}$$

$$\text{speed - up} = \frac{W_{\mathrm{GTS}}}{W_{\mathrm{LTS}}} \propto \left[\alpha + \frac{(1-\alpha)}{2^{m_{\max}}} \right]^{-1}. \tag{9b}$$

It follows that the LTS procedure basically reduces the number of update operations in the free-surface portion of the domain with a factor $2^{m_{\max}}$. Thus, other conditions being equal, the more cells belong to the free-surface region, the more update operations are skipped, and the faster the LTS simulation run time is, compared to the GTS simulation. Furthermore, narrower slots, corresponding to higher pressure wave celerity values, lead to higher values of the parameter $m_{\max}$, further increasing the achievable speed-up. The speed-up factors expected according to Eq. (9b) are also reported in Table 1 for comparison. The theoretical and actual values do not coincide, since the estimate of Eq. (9b) does not take into account the presence of an interface region with gradually varying values of m and other implementation details. Moreover, geometrical relations in free-surface conditions may require iterations or interpolations even for common cross-section shapes (e.g. circular), while in pressurized conditions explicit calculations can be done, and the CPU time necessary to perform such operations may differ for the two regimes. In any case, Eq. (9b) can be used as a

rule of thumb to predict whether the use of the LTS strategy can be beneficial for a certain application, even if in general the percentage of pressurized cells changes in time and may be difficult to estimate *a priori*.

Equation (9b) also suggests that the achievable speed-up should not depend on the total number of cells. This was verified by simulating the test case characterized by T_s/D = 0.01 and a pipe slope equal to +1% (which led to the pressurization of exactly half of the domain) with different grid sizes (N = 200, 400, and 800 cells). The speed-up factor is almost constant, ranging between 1.9 and 2.1.

3.2 Test cases with analytical solution

Idealized test cases with analytical solution are presented in this subsection. The exact Riemann solver proposed by Kerger, Archambeau, Erpicum, Dewals, & Pirotton (2011a) is used here to calculate reference solutions for initial value problems with piecewise constant initial states $\mathbf{U}_L$ (on the left-hand side) and $\mathbf{U}_R$ (on the right-hand side). All tests are performed in a frictionless 500 m-long horizontal conduit with rectangular cross-section characterized by width B = 1 m and height H = 1 m. The slot width T_s is set at 0.01B, which corresponds to pressure wave celerity approximately equal to 31 m/s. The computational domain is defined by the interval [0, 500] m and the initial discontinuity is located at x = 250 m. Test conditions are summarized in Table 2.

The solution of both Tests T1 and T2 is characterized by a rarefaction wave propagating to the left with transition between pressurized and free-surface conditions; the wave on the right instead is a free-surface shock in Test T1 and a free-surface rarefaction in Test T2. For the numerical simulations of both tests, the computational domain is discretized by means of N = 500 cells and the Courant number is set at 0.9. Figure 3 compares the GTS and LTS numerical pressure head and velocity profiles with the analytical solution at t = 6 s. Both schemes correctly predict the speed of waves travelling both left and right and the intermediate state. However, the LTS scheme provides slightly less diffusive results in the free-surface portion of the solution (see the inserts in Fig. 3 at $x \approx 270$ m).

The error based on the L_2-norm (LeVeque, 2002) is used to quantify the accuracy of the numerical solution. Accordingly, the error at a fixed time can be assessed in term of conserved variables as:

$$E_{L_2}(A) = \left(\frac{1}{N} \sum_{i=1}^{N} \left(A_i - A_{i,\mathrm{ref}} \right)^2 \right)^{0.5}, \quad E_{L_2}(Q) = \left(\frac{1}{N} \sum_{i=1}^{N} \left(Q_i - Q_{i,\mathrm{ref}} \right)^2 \right)^{0.5}, \tag{10}$$

where the subscript "ref" refers to the analytical solution. Table 3 reports the L_2-norms calculated for flow area and discharge profiles at t = 6 s with progressively halved mesh size (N = 500, 1,000, and 2,000 cells). The LTS scheme gives lower error norms than the GTS

scheme for all mesh sizes and for both test cases. The improvement in accuracy mainly concerns the free-surface portion of the flow field, where error norms decrease by about 20−30% switching from the GTS to the LTS scheme. Indeed, in the free-surface region the Courant number is locally closer to unity, whilst in the pressurized portion of the flow field error norms increase slightly (only by a few percent) for the LTS numerical solution. This is probably due to the fact that the LTS technique adopted here employs the same time step over a certain number of intermediate updates, potentially making the scheme slightly more diffusive, especially where celerity decreases, such as near the transition from pressurized to free-surface flow; conversely, the time step is recomputed after each time advancement in the GTS scheme.

Table 3 also reports the speed-up factor for test cases T1 and T2. In general, the speed-up factor is not expected to change considerably with the total number of cells, and this is confirmed by the fact that it varies from 1.6 to 1.9 (corresponding to CPU time reduction between 37% and 47%) if N increases from 500 to 2,000. In this case, the percentage of pressurized cells is time-dependent; considering an average value in the time interval between $t = 0$ s and $t = 6$ s, the speed-up estimated according to Eq. (9b) is approximately equal to 1.9 for both tests.

Finally, the numerical scheme does not deteriorate mass conservation; in fact, for Test T1, which involves no inflows or outflows in the time interval considered, the relative mass conservation error is in the order of 10^{-15} (with reference to the mass initially present in the conduit) for both the LTS and GTS scheme at the end of the simulation (after approximately 10^3 updates in the GTS simulation with $N = 2{,}000$).

Test T3 is characterized by an initial condition representing two approaching free-surface uniform flows; the impact produces two pipe filling bores propagating in opposite directions. It is well-known in literature that the propagation of an abrupt free-surface/pressurized flow transition is a challenging problem for numerical schemes (e.g. Bousso et al., 2013). In fact, numerical solutions are systematically affected by post-shock oscillations at flow regime transitions due to the sudden jump in the wave speed value (e.g. Vasconcelos et al., 2009). The computational domain is discretized by means of $N = 1{,}000$ cells and simulations are repeated for three different values of the slot width ($T_s/B = 0.04$, 0.01, and 0.001), corresponding to pressure wave celerity values of approximately 15, 31, and 100 m/s. The Courant number is set at 0.7. Figure 4 shows the pressure head and velocity profiles at $t = 20$ s; only LTS results are reported, since the LTS and GTS profiles practically overlap. The speed-up factor increases with the reduction of the slot width, and ranges from 1.4 to 1.9 to 2.4.

3.3 Gradual pipe filling

Pipe pressurization can be induced not only by a transition bore (as in test T3), but also by gradual rising of the water level (e.g. Capart et al., 1997; Ferreri et al., 2014). The numerical simulation of gradual pressurization is presented in this subsection in order to further verify the effectiveness of the LTS scheme.

The test is performed in a 100 m-long circular pipe with 1 m diameter and 0.4% slope. The Manning roughness coefficient is set at 0.015 $m^{-1/3}s$. A subcritical uniform free-surface flow with 0.7 m^3/s discharge and approximately 0.52 m normal depth is assumed as initial condition. The sudden closure (at $t = 0$) of a gate located at the downstream end of the pipe causes an abrupt increase in the water depth which propagates upstream as a free-surface shock wave. Due to pipe slope, after a few seconds the downstream portion of the pipe becomes pressurized and a smooth flow regime transition travelling upward arises.

The pipe is discretized by $N = 100$ computational cells ($\Delta x = 1$ m), and the slot width is set at $0.01D$. The Courant number is assumed equal to 0.9, and the solution is advanced for 60 s. A constant discharge is specified at the upstream boundary, while the reflective boundary condition is imposed downstream. Figure 5 shows the pressure head profiles at $t =$ 5, 20, and 40 s for the GTS and LTS schemes, along with a reference solution obtained with a very fine mesh ($N = 2{,}000$ cells, $\Delta x = 0.05$ m). Moreover, Table 4 reports the L_2-error norms of flow area and discharge profiles calculated via Eq. (10) at selected times (the same as in Fig. 5) with reference to the fine mesh predictions. The results confirm that the overall numerical scheme is able to describe smooth free surface-pressurized interfaces, and that the LTS scheme reproduces the moving free-surface discontinuities in a slightly more accurate way than the GTS model. The speed-up factor is approximately 1.6 in this test.

3.4 Laboratory test case

The robustness and effectiveness of the LTS scheme are assessed in this subsection on the basis of the laboratory test proposed by Aureli et al. (2015) as a benchmark for the validation of mixed flow models. In particular, this test case is adopted here to compare the performances of the GTS and LTS schemes in terms of accuracy and computational efficiency.

Experiments were performed in a Plexiglas pipe almost 12 m-long with inner diameter D equal to 0.192 m, characterized by an abrupt slope change. The slopes of the upstream and downstream branches are about +8.4% (downward) and −27.7% (upward) respectively, and the change in slope is located approximately at $x = 7$ m (with x denoting the distance along the pipe axis, starting from the upstream end). A sluice gate, located at $x = 5$ m, retains still water upstream with 0.225 m pressure head at $x = 1$ m, while the conduit is

initially dry downstream the gate. The sudden removal of the gate triggers a rapid transient characterized by the propagation of wetting and drying fronts, and the occurrence of repeated transitions from free-surface to pressurized flows and, in the last stage, of a damped oscillatory flow. Aeration is guaranteed at both pipe ends, hence air entrapment does not influence the phenomenon. Pressure head and velocity measurements are available at six selected cross-sections.

The computational domain is discretized by means of 300 grid cells (Δx = 0.04 m) and the Manning coefficient is set at 0.009 $m^{-1/3}s$ according to Aureli et al. (2015). The slot width is assumed to be equal to 0.01D and the Courant number is set at 0.9. A reflective boundary condition is imposed at the upstream end, whereas the downstream end is never reached by the flow.

Figure 6 shows GTS and LTS numerical pressure head profiles at selected times, along with the local experimental measurements available. Moreover, Fig. 7 compares the numerical and experimental pressure head time series at Sections G1 (x = 1.0 m) and G5 (x = 7.32 m), as well as velocity time series at Section V5 (x = 7.28 m). The LTS scheme yields almost identical numerical results to the ones provided by the GTS scheme. Table 5 reports the GTS and LTS L_2-errors of the flow area and discharge profiles calculated via Eq. (10) at selected times (the same as in Fig. 6) with reference to the GTS results obtained with a very fine mesh (Δx = 0.005 m). The results confirm that the accuracy of the two schemes is comparable. Moreover, for both schemes the dimensionless mass conservation error remains in the order of 10^{-15} at the end of the simulation (corresponding to almost 10^4 updates in the GTS simulation). Thus, LTS does not deteriorate mass conservation, even in the presence of wetting and drying. Finally, the speed-up factor varies approximately from 2.0 to 2.1 (corresponding to about 50% CPU time reduction) as the mesh is refined from Δx = 0.04 m to Δx = 0.005 m.

The sensitivity of numerical results to the slot width is also analyzed. However, it is well-known that too narrow slots produce spurious oscillations which may unacceptably compromise results, hence in practical applications slot widths corresponding to realistic pressure wave celerity values cannot be set. The simulation of this test case is repeated with dimensionless slot widths decreasing from 0.01 to 0.0005 (correspondingly, the pressure wave celerity increases from 12 m/s to 54 m/s), maintaining Δx = 0.04 m and assuming Cr = 0.5. The speed-up factor increases from 2.2 to 2.5. As an example, numerical and experimental pressure head time series at cross-section G4 (x = 6.8 m) are reported in Fig. 8 for the simulation performed with T_s/D = 0.0005. It must be considered that GTS and LTS time series are characterized by different temporal resolutions, since output results are available only at synchronization times for the LTS simulation, whereas GTS results can be obtained after each update. In this simulation, m_{max} is expected to be equal to 6, hence in Fig.

8 only one value out of 2^6 is represented in the GTS time series, in order to compare the numerical results consistently.

3.5 A field-scale application

The simulation of an idealized field-scale test is finally presented. A 2,000 m-long sewer pipe, characterized by a circular cross-section (D = 1.2 m) and by 0.1% slope, intersects a subway structure at x = 1000 m, and an inverted siphon is used to bypass the obstacle from below. This element is simply schematized as a vertical deviation in the pipe axis, which is 50% sloped (downward) for 8 m, horizontal for 5 m, and −50% sloped (upward) for 7 m. This geometry is certainly a simplification, since these structures often consist of multiple pipes, inlet and outlet chambers, and control devices. However, the main purpose of this test case is to demonstrate the efficiency of the LTS scheme in practical applications where only a small portion of the flow field is surcharged.

The pipe is initially dry, the only exception being the siphon, filled with static water, as if it was left from a previous rainfall event. At t = 0 min, a triangular inflow hydrograph, with 30 min duration and 1.2 m^3/s peak discharge at t = 9 min, is imposed at the upstream section. A free outfall is assumed downstream. The domain is discretized by means of N = 2,000 cells of length Δx = 1 m; moreover, the Courant number is assumed equal to 0.9, the Manning coefficient is set at 0.014 $m^{-1/3}s$, and the dimensionless slot width is set at 0.01 (corresponding to a 30 m/s pressure wave celerity value). The flood propagation in the sewer is simulated for 60 min.

As an example of model results, Fig. 9 shows pressure head profiles at selected times (Fig. 9a), and hydrographs of discharge (Fig. 9b) and pressure head (Fig. 9c) at three cross-sections. Only LTS results are represented, since GTS results are almost identical. The maximum water depth predicted is 0.84 m, thus the flow in the pipe always remains in free-surface conditions, except for the few cells (less than 1% of the total) representing the siphon. Hence, this case is suitable to be simulated with the LTS scheme, since the few pressurized cells would dictate the time step for the whole domain if the GTS scheme were used. The achieved speed-up is in fact equal to 5.7, corresponding to 82% CPU time reduction. If the simulation is repeated with “wet” initial conditions (for example, 0.05 m in the whole pipe), the speed-up grows to 6.5 (−85% CPU time). This is due to the fact that, for both schemes, flux evaluation in dry cells is computationally inexpensive compared to the case of wet cells, thus making the improved performance of the LTS scheme more evident.

4 Conclusions

In this paper the local time stepping strategy was applied for the first time to the numerical modelling of unsteady mixed flows. The LTS method adopted, coupled with a Godunov-type finite-volume solver based on the Preissmann slot approach, was found to be effective in simulating these phenomena as regards both computational efficiency and solution accuracy. In particular, the results of the numerical simulations showed the following:

- the local time stepping reduces run time and improves model efficiency in comparison with the conventional global time stepping, without impairing overall solution accuracy and without deteriorating mass conservation. In the transient test cases presented in this paper, speed-up factors between 1.4 and 6.5 were achieved (corresponding to a reduction in CPU time between 29% and 85%). Moreover, the LTS strategy ensures improved accuracy in the free-surface region of the flow field;
- the speed-up factor increases with the percentage of free-surface cells in the domain (which is time-dependent in unsteady applications); this suggests that LTS is particularly suitable for applications where flow pressurization is expected to involve only a small region of the flow field;
- the run time improvement achievable by the LTS scheme (compared to the GTS scheme) is slightly enhanced when narrow slot widths, corresponding to more realistic values of pressure wave celerity, are employed;
- the novel procedure based on the check of the local Courant number, which interrupts the series of intermediate updates and imposes synchronization to a suitable limited computational time, is useful to prevent numerical instabilities.

The application of the LTS method presented in this paper can easily be extended to finite volume mixed flow models based on different approaches, such as the TPA model or the dual model, and to numerical schemes of higher order of accuracy.

Finally, the recent 2D extension of the Preissmann slot approach (Maranzoni, Dazzi, Aureli, & Mignosa, 2015) suggests the possibility of applying the LTS method to 2D mixed flows modelling, with a potential improvement in efficiency, especially in the simulation of large free-surface flow fields with small pressurized regions.

Acknowledgements

We are grateful to the Associate Editor and the Reviewers, whose valuable comments greatly contributed to improving the paper.

Notation

a = pressure wave celerity (ms^{-1})

A = flow area (m^2)

B = pipe width for a rectangular cross-section (m)

c = speed of gravity waves (ms^{-1})

Cr = Courant number (−)

D = pipe diameter (m)

E_{L_2} = error based on the L_2-norm

F = flux vector

g = acceleration due to gravity (ms^{-2})

h = water depth or pressure head (m)

H = pipe height for a rectangular cross-section (m)

I_1 = first moment of flow area with reference to the water surface (m^3)

m = parameter for LTS update

N = number of computational elements (−)

n_m = Manning roughness coefficient ($m^{-1/3}s$)

p = intermediate step number

Q = flow discharge (m^3s^{-1})

R_h = hydraulic radius (m)

S = source term vector

S_0 = bed slope (−)

S_f = friction slope (−)

t = time (s)

T_s = slot width (m)

U = vector of conserved variables

u = mean cross-sectional velocity (ms^{-1})

x = spatial coordinate (m)

z = elevation (m)

Δt = time step (s)

Δx = mesh size (m)

α = percentage of pressurized cells (−)

Superscripts

k = generic time level

n = synchronization time level

Subscripts

b = bottom

i = computational cell index

f = intercell value

max = maximum value

ref = reference value

References

Aureli, F., Dazzi, S., Maranzoni, A., & Mignosa, P. (2015). Validation of single- and two-equation models for transient mixed flows: a laboratory test case. *Journal of Hydraulic Research*, in press.

Berger, M. J., & LeVeque, R. J. (1998). Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems, *SIAM Journal of Numerical Analysis*, *35*, 2298−2316.

Bourdarias, C., & Gerbi, S. (2007). A finite volume scheme for a model coupling free surface and pressurised flows in pipes. *Journal of Computational and Applied Mathematics*, *209*, 109–131.

Bousso, S., Daynou, M., & Fuamba, M. (2013). Numerical modelling of mixed flows in storm water systems: Critical review of literature. *Journal of Hydraulic Engineering*, *139*, 385–396.

Capart, H., Sillen, X., & Zech, Y. (1997). Numerical and experimental water transients in sewer pipes. *Journal of Hydraulic Research*, *35*, 659–672.

Cardle, J. A., Song, C. C. S., & Yuan, M. (1989). Measurements of mixed transient flows. *Journal of Hydraulic Engineering*, *115*, 169–182.

Coquel, F., Nguyen Q. L., Postel, M., & Tran, Q. H. (2010). Local time stepping applied to implicit-explicit methods for hyperbolic systems. *Multiscale Modeling & Simulation*, *8*, 540−570.

Crossley, A. J., & Wright N. G. (2005). Time accurate local time stepping for the unsteady shallow water equations. *International Journal of Numerical Methods in Fluids*, *48*, 775−799.

Cunge, J. A., Holly, Jr. , F. M., & Verwey, A. (1980). *Practical aspects of computational river hydraulics.* London UK: Pitman.

Cunge, J. A., & Wegner, M. (1964). Numerical integration of Barré de Saint-Venant's flow equations by means of an implicit scheme of finite differences. *La Houille Blanche*, *1*, 33–39.

DHI (2002). *MOUSE pipe flow reference manual*. Horsolm, Denmark: DHI Software.

Dumbser, M. (2014). Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume schemes with time-accurate local time stepping for hyperbolic conservation laws. *Computer Methods in Applied Mechanics and Engineering*, *280*, 57–83.

Dumbser, M., Käser, M., & Toro, E. F. (2007). An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes – V. Local time stepping and *p*-adaptivity. *Geophysical Journal International*, *171*, 695−717.

Dumbser, M., Zanotti, O., Hidalgo, A., & Balsara, D. S. (2013). ADER-WENO finite volume schemes with space-time adaptive mesh refinement. *Journal of Computational Physics*, *248*, 257–286.

Ferreri, G. B., Ciraolo, G., & Lo Re, C. (2014). Storm sewer pressurization transient – an experimental investigation. *Journal of Hydraulic Research*, *52*, 666–675.

Grote, M. J. & Mitkova, T. (2010). Explicit local time-stepping methods for Maxwell's equations. *Journal of Computational and Applied Mathematics*, *234*, 3283–3302.

Kerger, F., Archambeau, P., Erpicum, S., Dewals, B. J., & Pirotton, M. (2011a). An exact Riemann solver and a Godunov scheme for simulating highly transient mixed flows. *Journal of Computational and Applied Mathematics*, *235*, 2030–2040.

Kerger, F., Archambeau, P., Erpicum, S., Dewals, B. J., & Pirotton, M. (2011b). A fast universal solver for 1D continuous and discontinuous steady flows in rivers and pipes. *International Journal for Numerical Methods in Fluids*, *66*, 38–48.

Kesserwani, G., & Liang, Q. (2015). RKDG2 shallow-water solver on non-uniform grids with local time steps: Application to 1D and 2D hydrodynamics. *Applied Mathematical Modeling*, *39*, 1317−1340.

Kleb, W. L., Batina, J. T., & Williams, M. H. (1992). Temporal adaptive Euler/Navier-Stokes algorithm involving unstructured dynamic meshes. *AIAA Journal*, *30*, 1980−1985.

Krámer, T., & Józsa, J. (2007). Solution-adaptivity in modelling complex shallow flows. *Computers & Fluids*, *36*, 562−570.

León, A. S., Ghidaoui, M. S., Schmidt, A. R., & Garcia, M. H. (2009). Application of Godunov-type schemes to transient mixed flows. *Journal of Hydraulic Research*, *47*, 147–156.

León, A. S., Ghidaoui, M. S., Schmidt, A. R., & Garcia, M. H. (2010). A robust two-equation model for transient-mixed flows. *Journal of Hydraulic Research*, *48*, 44–56.

LeVeque, R. J. (2002). *Finite Volume Methods for Hyperbolic Problems*. Cambridge, UK: Cambridge University Press.

Maleki F. S., & Khan, A. A. (2015). A novel Local Time Stepping algorithm for shallow water flow simulation in the discontinuous Galerkin framework. *Applied Mathematical Modelling*, in press, doi: http://dx.doi.org/10.1016/j.apm.2015.04.038.

Maranzoni, A., Dazzi, S., Aureli, F., & Mignosa, P. (2015). Extension and application of the Preissmann slot model to 2D transient mixed flows. *Advances in Water Resources*, *82*, 70−82.

Osher, S., & Sanders, R. (1983). Numerical approximations to nonlinear conservation laws with locally varying time and space grids. *Mathematics of Computation*, *41*, 321−336.

Pervaiz, M. M., & Baron, J. R. (1988). Temporal and spatial adaptive algorithm for reacting flows. *Communications in Applied Numerical Methods*, *4*, 97−111.

Politano, M., Odgaard, A. J., & Klecan, W. (2007). Case study: Numerical evaluation of hydraulic transients in a combined sewer overflow tunnel system. *Journal of Hydraulic Engineering*, *133*, 1103–1110.

Sanders, B. F. (2008). Integration of a shallow water model with a local time step. *Journal of Hydraulic Research*, *46*, 466−475.

Sanders, B. F., & Bradford, S. F. (2011). Network implementation of the two-component pressure approach for transient flow in storm sewers. *Journal of Hydraulic Engineering*, *137*, 158–172.

Tan, Z., Zhang, Z., Huang, Y., & Tang, T. (2004). Moving mesh methods with locally varying time steps. *Journal of Computational Physics*, *200*, 347−367.

Toro, E. F. (2001). *Shock-capturing methods for free-surface shallow flows*. Chichester, UK: Wiley.

Trahan, C. J., & Dawson, C. (2012). Local time-stepping in Runge-Kutta discontinuous Galerkin finite element methods applied to the shallow-water equations. *Computer Methods in Applied Mechanics and Engineering*, *217−220*, 139−152.

Trajkovic, B., Ivetic, M., Calomino, F., & D'Ippolito, A. (1999). Investigation of transition from free surface to pressurized flow in a circular pipe. *Water Science and Technology*, *39*, 105–112.

HEC (2010). HEC-RAS, *River Analysis System Hydraulic Reference Manual, version 4.1*. Davis, California, USA: U.S. Army Corps of Engineers, Hydrologic Engineering Center.

Vasconcelos, J. G., & Wright, S. J. (2005). Experimental investigation of surges in a stormwater storage tunnel. *Journal of Hydraulic Engineering*, *131*, 853–861.

Vasconcelos, J. G., Wright, S. J., & Roe, P. L. (2006). Improved simulation of flow regime transition in sewers: The two-component pressure approach. *Journal of Hydraulic Engineering*, *132*, 553–562.

Vasconcelos, J. G., Wright, S. J., & Roe, P. L. (2009). Numerical oscillations in pipe-filling bore predictions by shock-capturing models. *Journal of Hydraulic Engineering*, *135*, 296–305.

Wiggert, D. C. (1972). Transient flow in free-surface, pressurized systems. *Journal of the Hydraulic Division*, *98*, 11–27.

Zhang, X. D., Trépanier, J. Y., Reggio, M., & Camarero, R. (1994). Time-accurate local time stepping method based on flux updating. *AIAA Journal*, *32*, 1926−1929.

Table 1 Sensitivity of the speed-up factor to the percentage of pressurized cells and to the slot width for the test case of water at rest in a circular sloped pipe (N = 200 cells)

Slope (%)	α (%)	T_s/D = 0.01		T_s/D = 0.001	
		Speed-up (actual)	Speed-up [Eq. (8b)]	Speed-up (actual)	Speed-up [Eq. (8b)]
+0.5	1	4.2	7.5	6.4	24.4
+0.553	10	3.2	4.7	4.3	7.8
+0.66	25	2.7	2.9	3.7	3.7
+1.0	50	1.9	1.8	2.4	1.9
+2.0	75	1.5	1.3	1.7	1.3
+5.0	90	1.3	1.1	1.5	1.1

Table 2 Test conditions for three discontinuous initial-value problems with exact solution

Test	h_L (m)	u_L (m/s)	h_R (m)	u_R (m/s)	Description
T1	3.0	0.0	0.5	0.0	Left transition rarefaction and right shock
T2	1.8	–0.9	0.8	1.0	Left transition rarefaction and right rarefaction
T3	0.8	2.0	0.8	–2.0	Two transition shocks

Table 3 Comparison between GTS and LTS accuracy and efficiency at t = 6 s for different mesh sizes for test cases T1 and T2

Test	N	$E_{L_2}(A)$ (m^2)		$E_{L_2}(Q)$ (m^{3}s^{-1})		Speed-up factor
		GTS	LTS	GTS	LTS	
T1	500	1.2×10^{-2}	1.1×10^{-2}	4.0×10^{-2}	3.6×10^{-2}	1.6
	1000	8.5×10^{-3}	6.4×10^{-3}	3.0×10^{-2}	2.5×10^{-2}	1.6
	2000	6.1×10^{-3}	4.4×10^{-3}	2.2×10^{-2}	1.8×10^{-2}	1.9
T2	500	9.9×10^{-3}	7.4×10^{-3}	3.2×10^{-2}	2.4×10^{-2}	1.7
	1000	7.2×10^{-3}	4.8×10^{-3}	2.3×10^{-2}	1.7×10^{-2}	1.6
	2000	5.0×10^{-3}	3.1×10^{-3}	1.7×10^{-2}	1.1×10^{-2}	1.9

Table 4 Gradual pipe filling: LTS and GTS L_2-error norms of flow area and discharge profiles at selected times

time (s)	$E_{L_2}(A)$ (m^2)		$E_{L_2}(Q)$ (m^{3}s^{-1})	
	GTS	LTS	GTS	LTS
5	1.4×10^{-2}	1.4×10^{-2}	3.8×10^{-2}	3.8×10^{-2}
20	1.5×10^{-2}	1.3×10^{-2}	3.7×10^{-2}	3.4×10^{-2}
40	1.6×10^{-2}	1.4×10^{-2}	2.9×10^{-2}	2.6×10^{-2}

Table 5 Experimental test case: LTS and GTS L_2-error norms of flow area and discharge at selected times with reference to a simulation performed with a very fine mesh ($\Delta x = 0.005$ m)

time (s)	$E_{L_2}(A)$ (m^2)		$E_{L_2}(Q)$ (m^{3}s^{-1})	
	GTS	LTS	GTS	LTS
1.5	2.8×10^{-4}	2.8×10^{-4}	2.5×10^{-4}	2.6×10^{-4}
2.5	5.7×10^{-4}	5.3×10^{-4}	7.2×10^{-4}	7.3×10^{-4}
5.0	4.9×10^{-4}	4.3×10^{-4}	3.3×10^{-4}	2.8×10^{-4}
7.5	3.8×10^{-4}	4.6×10^{-4}	3.0×10^{-4}	2.8×10^{-4}
10.0	3.7×10^{-4}	2.8×10^{-4}	3.4×10^{-4}	4.3×10^{-4}
15.0	4.1×10^{-4}	3.0×10^{-4}	2.1×10^{-4}	1.0×10^{-4}
20.0	3.1×10^{-4}	4.0×10^{-4}	2.5×10^{-4}	2.4×10^{-4}

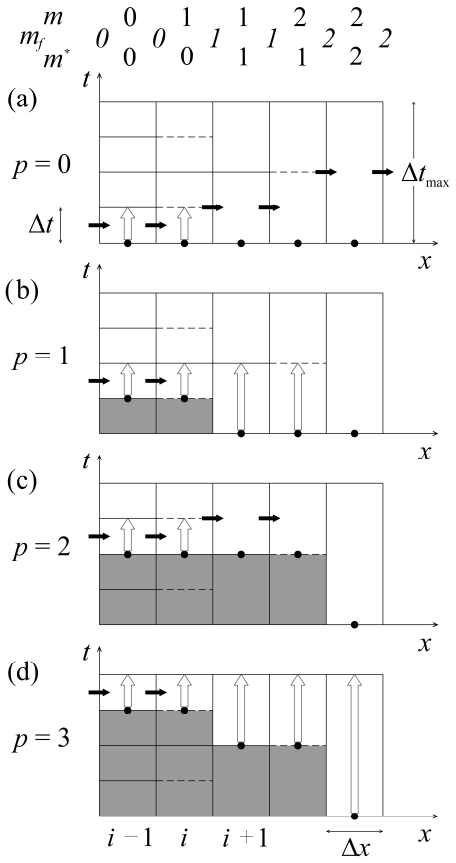

Figure 1 Sketch of the LTS update procedure for $m_{\max} = 2$ (corresponding to a sequence of four intermediate steps)

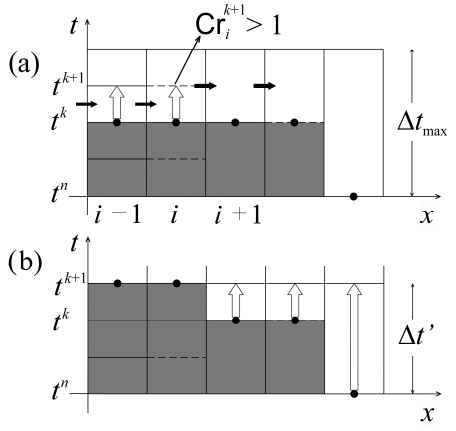

Figure 2 Sketch of the synchronization procedure introduced to avoid instabilities: (a) if $\mathrm{Cr}_i^{k+1} > 1$, the update procedure is interrupted, and (b) all cells are synchronized at t^{k+1}

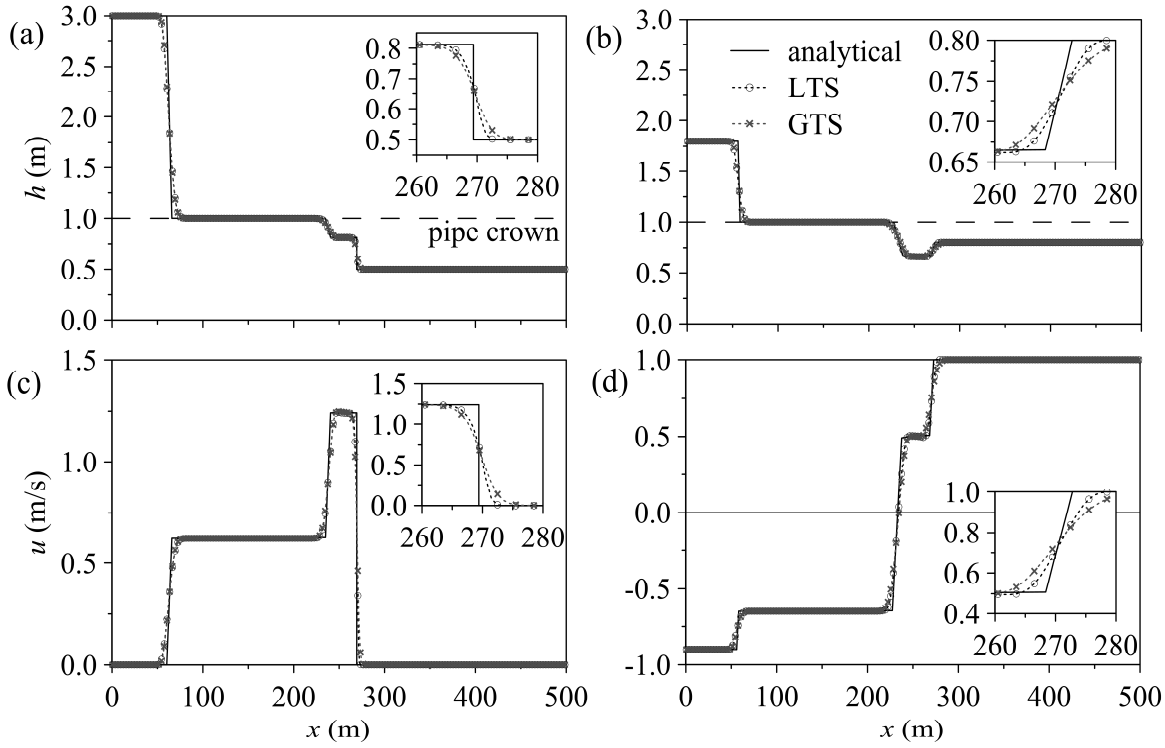

Figure 3 GTS and LTS numerical profiles (N = 500 cells, T_s/B = 0.01, Cr = 0.9) of pressure head and velocity at t = 6 s for tests T1 (a-c) and T2 (b-d), compared with analytical solutions. In numerical profiles, one symbol out of three is represented for clarity

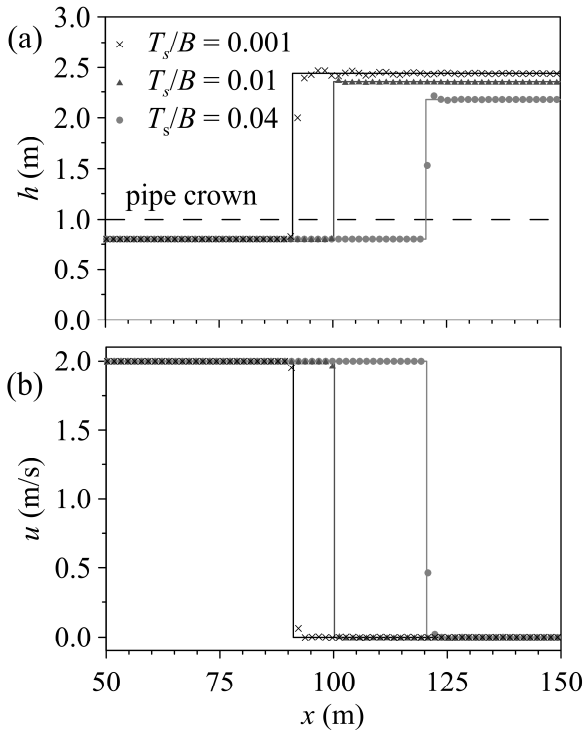

Figure 4 Numerical (LTS) and analytical (solid lines) pressure head (a) and velocity (b) profiles at t = 20 s for Test T3 (N = 1000 cells, Cr = 0.7). The x-axis is restricted to [50, 150] m for clarity. In numerical profiles, one symbol out of three is represented

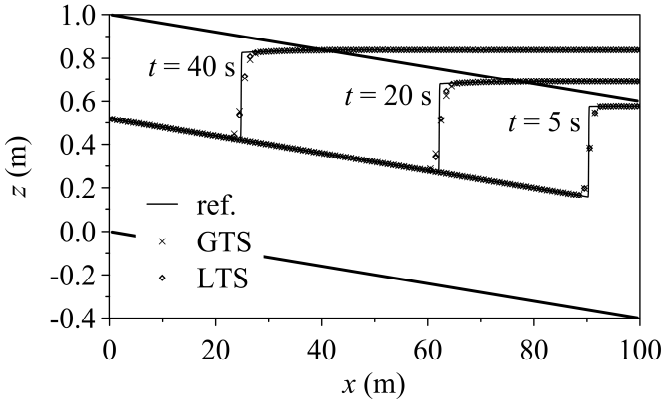

Figure 5 Numerical (LTS and GTS, N = 100 cells) and reference (N = 2000 cells) pressure head profiles at t = 5, 20, and 40 s for the gradual flow transition

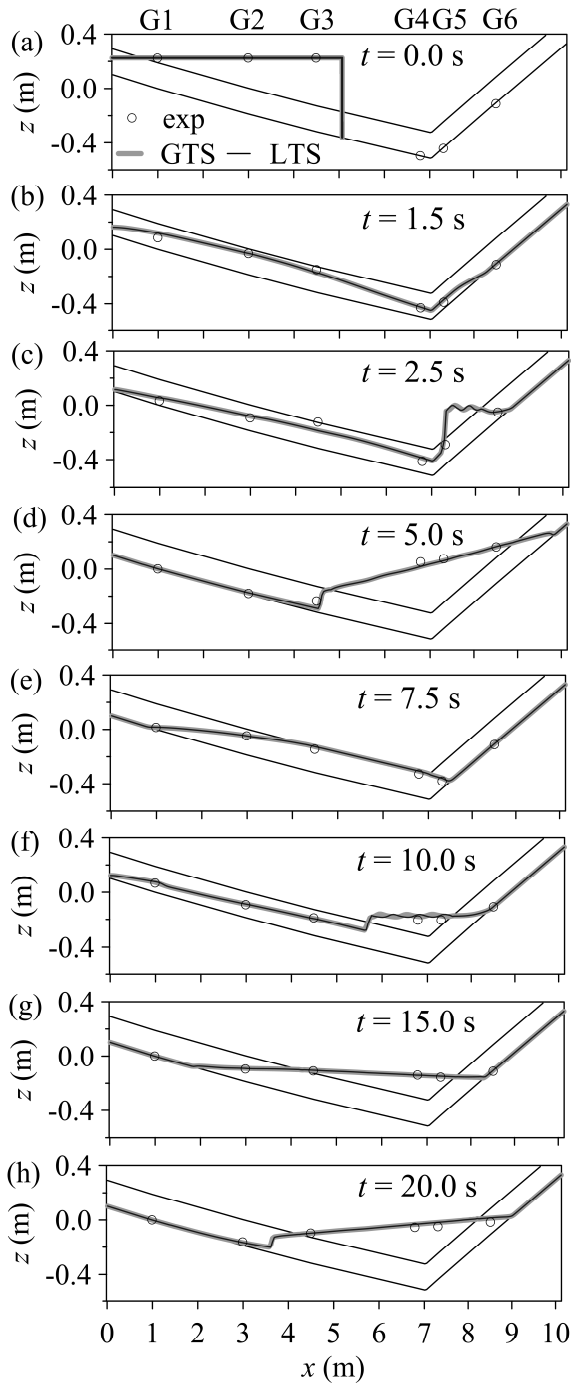

Figure 6 GTS and LTS numerical pressure head profiles at selected stages for the experimental test case (Δx = 0.04 m, T_s/D = 0.01, Cr = 0.9). Experimental data are also reported. The spatial variable x represents the distance along the pipe. Elevation z is referred to the bottom of section G1

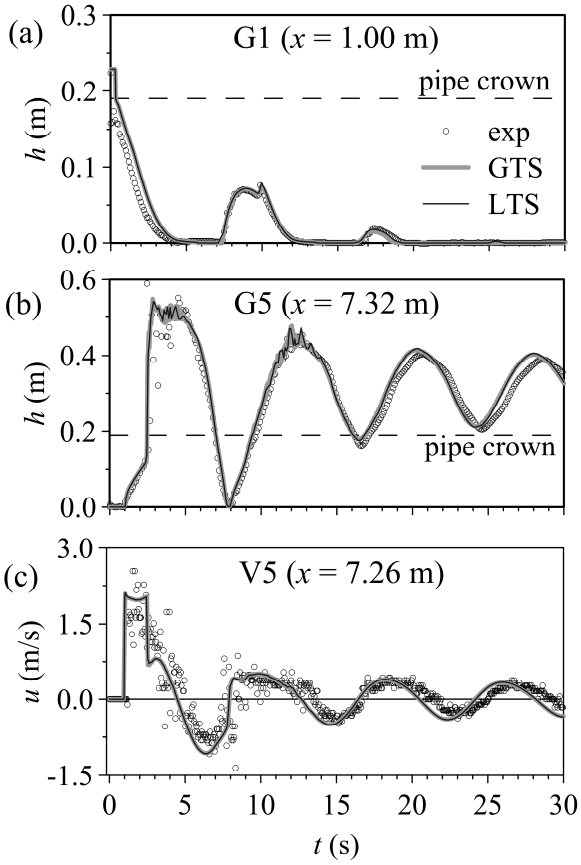

Figure 7 Numerical (GTS and LTS) and experimental pressure head and velocity time series at selected cross-sections (Δx = 0.04 m, T_s/D = 0.01, Cr = 0.9)

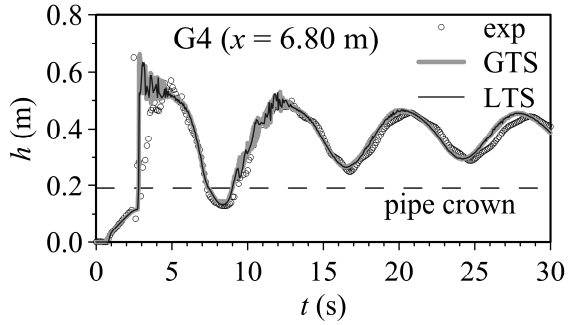

Figure 8 Numerical (GTS and LTS) and experimental pressure head time series at cross-section G4 (Δx = 0.04 m, T_s/D = 0.0005, Cr = 0.5). Only one value out of 64 is considered in the GTS time series

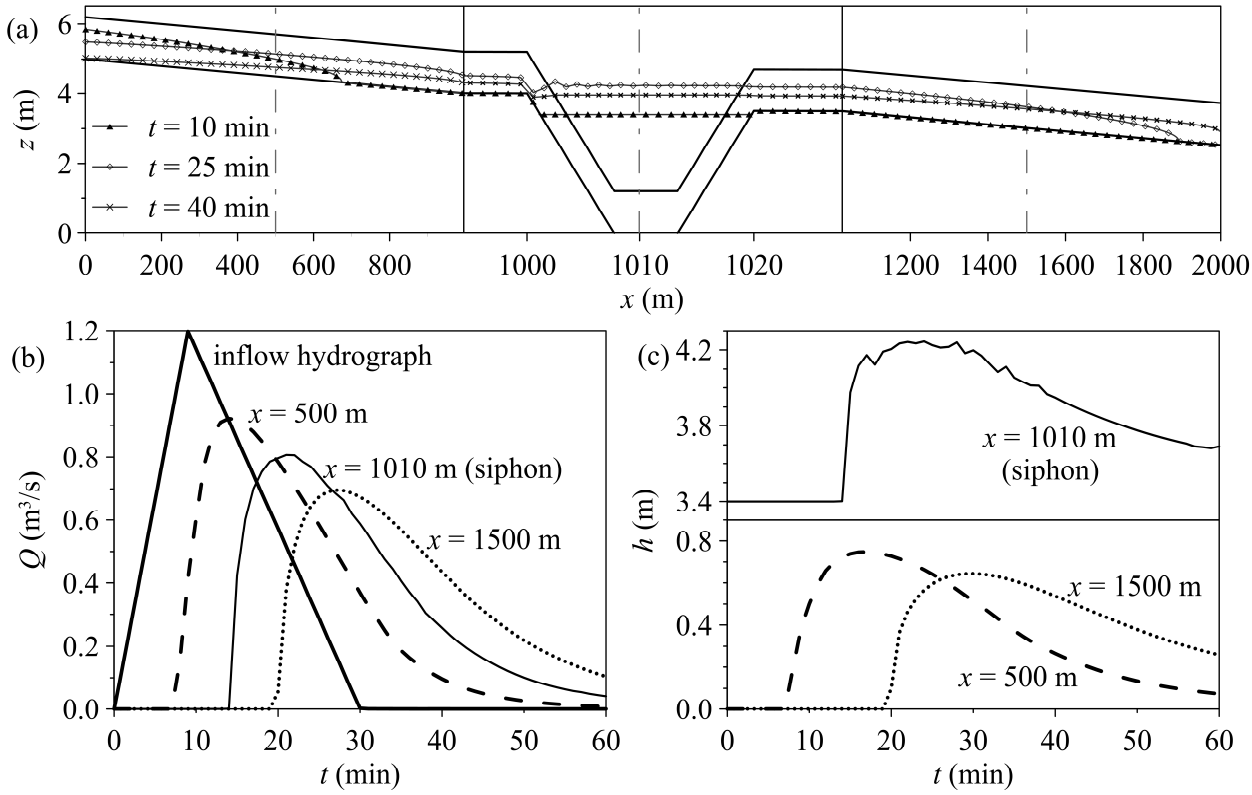

Figure 9 LTS numerical results for the inverted siphon simulation: (a) pressure head profiles at selected times (elevation z is referred to the siphon invert, the representation is distorted near the siphon to improve readability); (b) discharge hydrograph at inflow and at three selected sections; (c) pressure head time series at three sections