Catchment-scale multi-process modeling with local time stepping

Bastian van den Bout¹ · V. G. Jetten¹

Received: 6 September 2019 / Accepted: 25 March 2020 / Published online: 10 April 2020
© The Author(s) 2020

Abstract

Numerical modeling is an important tool for prediction, analysis and understanding of the dynamics of land surface processes. To increase the usage and impact of such tools, it is crucial to decrease runtime by increasing computational efficiency. Dynamic processes such as water flow are typically described by higher-order differential equations. Solving these accurately requires numerical integration over time, where numerical errors depend on the time steps taken. Typically, flow simulation use the smallest required time steps in a model's domain to simulate flow. In this paper, we analyze the usage of local time stepping, for catchment-scale simulation of land surface processes such as water flow, infiltration, slope stability and landslide runout. In such a scheme, temporal integration is cell specific, allowing for higher numerical efficiency. The implemented scheme works with fully free local time steps that are synchronized only for visualization. We implement this method in a monotonic upwind scheme for conservation laws (MUSCL). We investigate the influence on stability and the resulting changes in computation time and accuracy in a hydrology-coupled, catchment-scale flood simulation. Results show that local time stepping can be implemented in a total variation diminishing (TVD) numerical scheme that is second-order spatially accurate. Simulation results in both 1D dam-break scenarios and catchment-scale flash flood scenarios show insignificant changes in modeling result, while computation time reduces with over 50%. Finally, the method is successfully implemented in a multi-process lands surface model with hydrology, flooding, slope failure, and runout. The implementation of a local time stepping for computation of dynamic land surface processes could be implemented widely for increased computational efficiency without significant loss of accuracy.

Keywords Flood · Numerical scheme · Landslide · Physically-based modelling

Introduction

Numerical modeling has become the primary tool for understanding and predicting the behavior of water and sediment-related land surface processes. Scientific investigations into these processes similarly use these models to investigate the physical principles that drive their behavior (Navok et al. 2010). Furthermore, models for hazardous land surface processes have been extensively applied in risk assessment, leading to usage by disaster risk reduction experts, policy makers and urban planners (Feng and Lu 2010; Brunner 2010; Chen and Zhang 2015). It is thus crucial that numerical models are numerically efficient, decreasing delays between simulations and decision making.

The complexity of models influences their computational speed, which is one of the key factors that determine usage, application and thus impact of the underlying theories (Navok et al. 2010). The dynamics of hydrological or sediment-related processes generally involve complex second-order differential equations. The progression of time requires integration of these equations. In the past decades, large amounts of research have been focused on high-resolution numerical schemes for hyperbolic conservation laws (Harten 1983; Toro 2013; LeVeque 2002; Sohn 2005). Within numerical integration schemes, the discretization of time primarily determines the magnitude of approximation errors. This discretization must allow for accurate integration of the differential equations, to maintain accurate and stable behavior. Typically, accuracy increases with a smaller numerical integration step (Harten 1983). However, this increases computational time and decreases usability and application of physically based models if sufficient computing resources are not available.

A variety of computational schemes have been developed and used in the simulation of land surface processes. General integrations schemes such as the Euler, Heuns or

✉ Bastian van den Bout
b.vandenbout@utwente.nl

¹ Faculty of Geo-Information Science and Earth Observation,
University of Twente, Enschede, The Netherlands

Runge–Kutta schemes provide different orders of accuracy in time (Bouchut 2004). Other methods, such as the Monotonic Upstream-centered Scheme for Conservation Laws (MUSCL) (Bouchut 2004), have been developed for the purpose of improving the second-order conserving differential equations. Harten (1983) formalized the framework for proving the stability of numerical schemes. In particular, in each temporal step, total variation in the variables does not increase. Such schemes are named Total Variation Diminishing (TVD). TVD schemes, such as MUSCL, have successfully been applied for the Saint–Venant equations for shallow water flow, and are now available open source in the FullSWOF library (Delestre et al. 2014). The most frequently used method of decreasing numerical computations is the dynamic selection of a time step. When the time step is not fixed, but rather dependent on the current state of the simulation, calculations are automatically reduced when possible. A further extension of this method varies the time step both temporarily and spatially.

Local time stepping (LTS), a method where the time step is varied both through time and space, has in the past been applied to a relatively small selection of models when compared to other numerical methods. Finite difference seismic models have utilized the method to successfully improve calculation time (Dumbser et al. 2017; Madec et al. 2009). In models for unsteady internal flows, Sedaghat et al. (1999) and Yee and Harten (1987) have applied LTS with a time step condition based on finite grid-cell volume. They concluded that the method was useful in improving calculation time, depending on the used scenario. In later works, the applicability of a local time stepping scheme to water flow has been investigated, showing significant increases in computational efficiency can be reached. Similar conclusions were reached for turbulence simulations (Muradoglu and Pope 2002). Recently, local time-stepping was applied in a flood model with wetting and drying (Sanders 2008). Finally, Dazzi et al. (2018) implemented LTS in 2D shallow water flow using GPU-integrated methods.

Besides the aforementioned implementations of local time stepping, their application is still limited. Both commercially available software (Flo-2D, Delft3D etc.) and open-source alternatives do not implement such a scheme. Furthermore, the link between this method and broader environmental modeling is limited. In modeling of catchment-scale hydrology, groundwater flow, surface flow and related processes, the method that could potentially provide drastic improvement to computation time has not yet been investigated.

In this paper, we propose a flexible implementation of local time stepping method for solving mass-conserving differential equations for land surface processes. The scheme that is developed combines spatially and temporally dynamic time steps with a MUSCL numerical schemes for simulation

of land surface processes. This is furthermore combined with existing numerical methods such as a Heun's predictor–corrector and a semi-explicit implementation of Darcy–Weisbach friction (Te Chow 1964). In this paper, the numerical method will be applied to a hydrological catchment model which uses the depth-averaged Saint Venant equations for shallow flow. Furthermore, we simulate a multi-hazard event using two-phase mixture flow equations. To investigate the influence of the numerical method on model behavior, the final model was tested on catchments on the Fella River basin and the Caribbean islands of St. Lucia and Dominica.

Methods

To apply and test sophisticated environmental modeling in a numerical solution that uses local time stepping, we use and extend the existing open-Source multi-hazard model OpenLISEM.

The OpenLISEM model implements multiple types of infiltration models such as Smith & Parlange (Smith and Parlange 1978) and a full vertical soil water balance model named SWATRE (Van den Broek et al. 1996). The simulations in this paper use the Green and Ampt infiltration model which assumes a wetting front moving down into the soil due to infiltrating rainfall (Green and Ampt 1911). The resulting potential infiltration is subtracted from the available surface water (Eq. 1).

$$f_{\text{pot}} = -K_s \left(\psi \frac{\theta_s - \theta_i}{F} + 1 \right), \quad (1)$$

with f_{pot} the potential infiltration rate (m s^{-1}), F the cumulative infiltrated water (m), θ_s the porosity ($\text{m}^3 \text{m}^{-3}$), θ_i the initial soil moisture content ($\text{m}^3 \text{m}^{-3}$), ψ the matric pressure at the wetting front ($h_h = \psi + z$) (m), h_h is the hydraulic head (m), z is the elevation (m) and K_s the saturated conductivity (m s^{-1}).

Additionally, a simplified depth-averaged Darcy-type groundwater flow is implemented using an identical approach to the surface flow, but using larger time steps due to the smaller velocities (Eq. 2).

$$Q_{\text{gw}} = -K_s h_{\text{gw}} \left(\frac{dh_{\text{gw}}}{dx} + \frac{dz}{dx} \right), \quad (2)$$

with Q_{gw} the groundwater discharge (m s^{-1}) and h_{gw} the groundwater height (m).

The input data consist of soil, land surface and terrain properties, and can be defined on a sub-cell basis (Fig. 1). The hydrological processes that are simulated within OpenLISEM are extensive and include interception by vegetation, surface micro-ponding and dynamic flow. Further details on the

Fig. 1 Spatially varying time step values for numerical integration of flow equations. Blue curve represents some flow height. Black arrows indicate flow velocity. Vertical axis (time) is locally divided into a number of integration steps (dt) based on the velocity

underlying physical principles of OpenLISEM can be found in Baartman et al. (2013) and Jetten and De Roo (2001).

Equations for flow dynamics

The created numerical method is designed for use with hyperbolic conserving second-order partial differential equations. In our case, mass conservation (Eq. 3) and a momentum advection scheme with source terms for water flow form the basis of the model (Eq. 4).

$$\frac{\partial h}{\partial t} + \frac{\partial(uh)}{\partial x} + \frac{\partial(vh)}{\partial y} = 0 \quad (4)$$

$$\frac{\partial hu}{\partial t} + \frac{\partial(hu^2)}{\partial x} + \frac{\partial(huv)}{\partial y} + \frac{d\left(\frac{gh^2}{2}\right)}{dx} - ghS_x = 0, \quad (4)$$

$$\frac{\partial hv}{\partial t} + \frac{\partial(huv)}{\partial x} + \frac{\partial(hv^2)}{\partial y} + \frac{d\left(\frac{gh^2}{2}\right)}{dy} - ghS_y = 0, \quad (5)$$

with h the flow height (m) u and v the flow velocity in the x and y directions, g the gravitational acceleration and S_x S_y the friction slope terms in the x and y direction ($m\ m^{-1}$).

$$S_{x,f} = \alpha_f \left\{ g \left(\frac{\partial b}{\partial x} \right) - \epsilon \left[\frac{1}{h} \frac{\partial}{\partial x} \left(\frac{h^2}{2} P_{b_f} \right) + P_{b_f} \frac{\partial b}{\partial x} - \frac{1}{\alpha_f N_R} \left(2 \frac{\partial^2 u_f}{\partial x^2} + \frac{\partial^2 v_f}{\partial y \partial x} + \frac{\partial^2 u_f}{\partial y^2} - \frac{\chi u_f}{\epsilon^2 h^2} \right) \right. \right. \\ \left. \left. + \frac{1}{\alpha_f N_R} \left(2 \frac{\partial}{\partial x} \left(\frac{\partial \alpha_s}{\partial x} (u_f - u_s) \right) + \frac{\partial}{\partial y} \left(\frac{\partial \alpha_s}{\partial x} (v_f - v_s) + \frac{\partial \alpha_s}{\partial y} (u_f - u_s) \right) \right) - \frac{\xi \alpha_s (v_f - v_s)}{\epsilon^2 \alpha_f N_{R_A} h^2} \right] \right\} \frac{1}{\gamma} C_{DG} (u_f - u_s) |\vec{u}_f - \vec{u}_s|^{j-1} \quad (11)$$

$$S_{y,f} = \alpha_f \left\{ g \left(\frac{\partial b}{\partial y} \right) - \epsilon \left[\frac{1}{h} \frac{\partial}{\partial y} \left(\frac{h^2}{2} P_{b_f} \right) + P_{b_f} \frac{\partial b}{\partial y} - \frac{1}{\alpha_f N_R} \left(2 \frac{\partial^2 v_f}{\partial y^2} + \frac{\partial^2 u_f}{\partial y \partial x} + \frac{\partial^2 v_f}{\partial x^2} - \frac{\chi v_f}{\epsilon^2 h^2} \right) \right. \right. \\ \left. \left. + \frac{1}{\alpha_f N_R} \left(2 \frac{\partial}{\partial y} \left(\frac{\partial \alpha_s}{\partial y} (v_f - v_s) \right) + \frac{\partial}{\partial x} \left(\frac{\partial \alpha_s}{\partial y} (u_f - u_s) + \frac{\partial \alpha_s}{\partial x} (v_f - v_s) \right) \right) - \frac{\xi \alpha_s (u_f - u_s)}{\epsilon^2 \alpha_f N_{R_A} h^2} \right] \right\} - \frac{1}{\gamma} C_{DG} (u_f - u_s) |\vec{u}_f - \vec{u}_s|^{j-1}, \quad (12)$$

The friction slope terms include gravitational forces and a basal frictional force for shallow depth-averaged water flow (Eqs. 6 and 7).

$$S_x = -S_{0x} + S_{fx}, \quad (6)$$

$$S_y = -S_{0y} + S_{fy}, \quad (7)$$

where S_0 is the bed slope ($-$).

As a friction force, the Darcy–Weisbach type friction law will be used (Eq. 8), based on the value of Manning's roughness coefficient (Te Chow 1964).

$$S_f = \frac{1}{n^2} \frac{|\vec{u}| |\vec{u}|}{h^{\frac{4}{3}}}, \quad (8)$$

with S_f the friction force per unit mass ($m^2 s^{-1}$), $\vec{u}$ a velocity ($m^1 s^{-1}$) and n the Manning's roughness coefficient ($-$).

Hydrology and slope failure are based on infinite-slope depth-averaged assumptions (van Beek 2002). The sub-surface hydrology is taken from the OpenLISEM model and the slope failure calculations are done using the iterative failure model (Bout et al. 2018). In case of multi-phase flow, initiated by slope failure, we implement the Pudasaini (2012) generalized debris flow equations.

$$S_{x,s} = \alpha_s \left(g \left(\frac{\partial b}{\partial x} \right) - \frac{u_s}{|\vec{u}_s|} \tan(\partial P_{b_s}) - \epsilon P_{b_s} \left(\frac{\partial b}{\partial x} \right) \right) \\ - \epsilon \alpha_s \gamma P_{b_f} \left(\frac{\partial h}{\partial x} + \frac{\partial b}{\partial x} \right) + C_{DG} (u_f - u_s) |\vec{u}_f - \vec{u}_s|^{j-1}, \quad (9)$$

$$S_{y,s} = \alpha_s \left(g \left(\frac{\partial b}{\partial y} \right) - \frac{v_s}{|\vec{u}_s|} \tan(\partial P_{b_s}) - \epsilon P_{b_s} \left(\frac{\partial b}{\partial y} \right) \right) \\ - \epsilon \alpha_s \gamma P_{b_f} \left(\frac{\partial h}{\partial y} + \frac{\partial b}{\partial y} \right) + C_{DG} (v_f - v_s) |\vec{u}_f - \vec{u}_s|^{j-1}, \quad (10)$$

with α_s and α_f the volume fraction for solid and fluid phases (-), P_b the pressure at the base surface ($\text{Kg m}^{-1}\text{s}^{-2}$), b the basal surface of the flow (m), N_R the Reynolds number (-), N_{R_A} the quasi-Reynolds number (-), C_{DG} the drag coefficient (-), ρ_f the density of the fluid (kg m^{-3}), ρ_s the density of the solids (kg m^{-3}), γ the density ratio between the fluid and solid phase (-), χ the vertical shearing of fluid velocity (m s^{-1}), ε the aspect ratio of the model (-), ξ the vertical distribution of α_s (m^{-1}).

Timesteps

Current models generally use a Courant–Fredrich–Levy condition to determine the maximum time step that can be taken while maintaining numerical stability. This condition is provided by the requirement that the velocity of a water column must not exceed the cell length during one step of numerical integration, since otherwise cells are skipped and negative flow heights can result (Eq. 13).

$$\frac{v dt}{dx} < c. \quad (13)$$

with v the velocity (m s^{-1}), dx the grid cell size (m), dt the time step (m) and c the Courant factor, generally between 0.1 and 0.4 (-).

Typically, to reduce numerical integration errors, the time step is reduced much more than the strict limit of positive mass conservation by reducing the Courant factor. Extensions of this condition have been developed based on empirical flow velocity (Hunter et al. 2005; Delestre et al. 2014). In this research, a similar adaptation of the Courant–Fredrich–Levy condition will be used. We include the acceleration and Manning's velocity in the condition to automatically adapt the time step to high accelerations (Eq. 14).

$$\frac{\max(v, v + a) dt}{dx} < c, \quad (14)$$

with a the acceleration (m s^{-2}).

From the resulting condition, time step requirements are separately calculated for every grid cell. Once every time step is known, a global minimum time step is determined. Then, for every grid cell, the time step must equal a multiple of the global minimum time step (Eq. 15).

$$dt_{r,c} = \min(dt_{\max}, dt_{\min} n) \quad n \in \mathbb{N}, \quad (15)$$

with $dt_{r,c}$ the time step for a specific cell ($r = \text{row}$, $c = \text{column}$) (s), $dt_{\max}$ the global common time step (s) and $dt_{\min}$ the global minimum time step (s).

The global common time step is a common time step at which all cells synchronize (Fig. 1). Specifically, at every multiple of $dt_{\max}$, every grid cell will have its state at that same temporal position. Grid cells might have a single

step within $dt_{\max}$, or many more. On these synchronized moments, the overarching model is called to solve processes that do not substantially depend on the time step for accuracy, such as interception, sub-surface processes and empirical splash erosion. This allows us to calculate these processes with larger, but appropriate, integration steps.

Influence on numerical stability

Most widely used numerical schemes for solving hyperbolic sets of partial differential equations are well balanced and accurately maintain steady-state solutions by using correction terms that ensure a mass-conserving flux transfer (Kurganov and Petrova 2007). In the case of flow equations, this property is particularly important, as the still water level of lakes (a steady state at rest) should be maintained for an accurate and stable simulation. To work out the influence of a local time stepping on the stability of a numerical scheme, we will now look at the example of a hydrostatic reconstructing using cell boundary fluxes and a MUSCL-approximation for the cell boundary fluxes. This widely used numerical scheme, as adopted in FullSwof (Delestre et al. 2014), is based on the following principles; Cell boundary fluxes are estimated using stepwise linear approximations (Fig. 2).

Water heights for each cell are reconstructed based in total incoming and outgoing flux (Eq. 16):

$$\begin{aligned} \Delta h_i = & 0.5 \times \sum_{dt_i} \frac{DT_i}{DX} (F_{i-\frac{1}{2}} + F_{i+\frac{1}{2}}) + 0.5 \\ & \times \sum_{dt_{i-\frac{1}{2}}} \frac{DT_{i-1}}{DX} (F_{i-\frac{1}{2}}) + 0.5 \times \sum_{dt_{i+\frac{1}{2}}} \frac{DT_{i+1}}{DX} (F_{i+\frac{1}{2}}), \end{aligned} \quad (16)$$

where $h_{i,j}^n$ is the water height at location i, j at time n (m), DT the time step (s), DX is the cell size (m) and F is the

Fig. 2 The definition of cells, boundaries, and cell boundary fluxes. MUSCL uses linearly reconstructed water heights at the cell boundaries to estimate a flux at those boundaries. Superscript L indicates L boundary of cell, R indicates right boundary

approximated flux through a cell boundary in either x or y direction (m^2s^{-1}).

Fluxes are determined based on the linear reconstructions of water heights, velocities, and bed elevations from the cells that touch the cell boundary (Eq. 17).

$$F_{i+\frac{1}{2}} = \mathcal{F}\left(h_{i+1}^L, u_{i+1}^L, z_{i+\frac{1}{2}}^L, h_{i+1}^R, u_{i+1}^R, z_{i+\frac{1}{2}}^R\right). \quad (17)$$

Numerical schemes such as these, when combined with appropriate flux limiters, such as minmod, van Albada, or superbee, are proven to maintain a steady state at rest and furthermore diminish total variations in the solution (Van Albada et al. 1997; Harten 1983; Roe 1986). The actual function that calculates the cell-boundary flux must solve local discontinuous Riemann problem. A variety of Riemann solvers can be used (Delestre et al. 2014); here we implement the Harten–Lax–Van Leer (HLLC) solver (Harten et al. 1983; Toro et al. 1994). For every initial condition, the total variation in the state of the simulation must, without external influences, either remain equal or decrease. The usage of LTS can be done while maintaining the useful properties of such numerical schemes.

In particular, take a numerical scheme to the advection equation used with hyperbolic conserving laws (Eq. 18).

$$u_x^{t+1} = u_x^t + C_{+,x+\frac{1}{2}} \Delta_{x+\frac{1}{2}} u^t - C_{-,x-\frac{1}{2}} \Delta_{x-\frac{1}{2}} u^t, \quad (18)$$

where the local change in variable u is described by Eq. 19.

$$\begin{aligned} \text{TV}(t+1) &= \sum_{-\infty}^{\infty} \left| \Delta_{x+\frac{1}{2}} u^{t+1} \right| \leq \sum_{-\infty}^{\infty} C_{-,x-\frac{1}{2}} \left| \Delta_{x-\frac{1}{2}} u^t \right| + \sum_{-\infty}^{\infty} \left(1 - C_{-,i+\frac{1}{2}} - C_{+,i+\frac{1}{2}} \right) \left| \Delta_{x+\frac{1}{2}} u^t \right| + \sum_{-\infty}^{\infty} C_{-,x+\frac{3}{2}} \left| \Delta_{x+\frac{3}{2}} u^t \right| \\ &= \sum_{-\infty}^{\infty} C_{-,x+\frac{1}{2}} \left| \Delta_{x+\frac{1}{2}} u^t \right| + \sum_{-\infty}^{\infty} \left(1 - C_{-,i+\frac{1}{2}} - C_{+,i+\frac{1}{2}} \right) \left| \Delta_{x+\frac{1}{2}} u^t \right| + \sum_{-\infty}^{\infty} C_{-,x+\frac{1}{2}} \left| \Delta_{x+\frac{1}{2}} u^t \right| = \sum_{-\infty}^{\infty} \left| \Delta_{x+\frac{1}{2}} u^t \right| = \text{TV}(t). \end{aligned} \quad (19)$$

$$\Delta_{x+\frac{1}{2}} u^t = u_{x+1}^t - u_x^t \quad (19)$$

The total variation in the state of the variable u is given by Eq. 20.

$$\text{TV}(t) = \sum_{-\infty}^{\infty} \left| \Delta_{x+\frac{1}{2}} u^t \right|. \quad (20)$$

The condition for a total variation diminishing solution is then given by Eq. 21.

$$\begin{aligned} C_{+,x+\frac{1}{2}} &\geq 0 \\ C_{-,x-\frac{1}{2}} &\geq 0 \\ C_{-,x-\frac{1}{2}} + C_{+,x+\frac{1}{2}} &\leq 1. \end{aligned} \quad (21)$$

Fig. 3 The underestimation of the required time step due to incoming flow and momentum. Left: time step in right cell is large due to small fluxes. Right: time step has decreased due to incoming flow. The time since the last time step is larger than required for stability

The proof of this is provided by Harten (1983). First, he notes the local variation in u^{t+1} (Eq. 22).

$$\begin{aligned} \Delta_{x+\frac{1}{2}} u^{t+1} &= C_{-,x-\frac{1}{2}} \Delta_{x-\frac{1}{2}} u^t + \left(1 - C_{-,i+\frac{1}{2}} - C_{+,i+\frac{1}{2}} \right) \Delta_{x+\frac{1}{2}} u^t \\ &\quad + C_{-,x+\frac{3}{2}} \Delta_{x+\frac{3}{2}} u^t \end{aligned} \quad (22)$$

Then, based on the conditions for the parameters, Eq. 22 results.

$$\begin{aligned} \left| \Delta_{x+\frac{1}{2}} u^{t+1} \right| &\leq C_{-,x-\frac{1}{2}} \left| \Delta_{x-\frac{1}{2}} u^t \right| + \left(1 - C_{-,i+\frac{1}{2}} - C_{+,i+\frac{1}{2}} \right) \left| \Delta_{x+\frac{1}{2}} u^t \right| \\ &\quad + C_{-,x+\frac{3}{2}} \left| \Delta_{x+\frac{3}{2}} u^t \right| \end{aligned} \quad (23)$$

Then, by altering the summation index, the total variation at time t is acquired (Eq. 23).

Thus, finally, based on the conditions in Eq. 24, the total variation must remain either equal or diminish.

$$\text{TV}(t+1) \leq \text{TV}(t). \quad (25)$$

In our implementation, fluxes through a cell interface are immediately taken from one cell to another, and the simulation can be seen as a sequence of cell interactions through cell boundary interfaces. For each activated cell, fluxes for four cell boundaries are calculated, leading to indirect activation of five cells. As stated earlier, fluxes are halved, to compensate for every cell being activated twice. However, within this subset of calculations, the time step of all activated cells is taken to be the same. These small subset of calculations, on five specific cells could, either through an open or closed boundary condition, be

imagined to be an independent simulation. For this tiny, but independent simulation, the properties of the numerical scheme remain valid, since only the flux is multiplied by a factor of a half and the time steps taken for every cell are identical. Now, when applying this logic to a larger sequence of cell-interactions, and identifying a full simulation as a collection of flux interactions between five cells, the properties of the numerical scheme must remain valid for the entire simulation. Thus, the properties that make a numerical scheme both well-balanced or total variation diminishing, are still valid when used with LTS, when the conditions on the coefficients in Eq. 21 are still met.

A possible case where not all conditions for a stable and TVD scheme are met could be a large amount of sudden inflow and increase of velocity. Figure 3 visualizes this process, where the cell on the right experiences incoming flow, leading to a sudden decrease in local time step.

Due to the decrease in local time step, the actual time since its last time step is much larger than the required time step for a stable simulation. Two additional aspects must be implemented in the numerical scheme to prevent these instable large time steps.

Safety region

When using local time stepping, unnatural behavior might arise when there is a substantial difference in time step between neighboring cells due to a difference in velocity. Since the MUSCL scheme uses velocities of both cells to calculate the cell boundary flux, this might lead to an over-estimation of outflow for the low-velocity cell. This cell will drain, leading to a border of flow height underestimations around areas of high velocity. To prevent this, each time step is, for one iteration, compared with the neighboring values, and set to the lowest value (Eq. 26).

$$dt_{r,c} = \min(dt_{r,c}, dt_{r,c+1}, dt_{r,c-1}, dt_{r+1,c}, dt_{r-1,c}). \quad (26)$$

This equation creates a region around cells with a small time step, were a similarly small time step is used. The safety region only needs to include the direct neighbors of a cell that can receive a flux. This allows those cells to receive a possible high-velocity flux with appropriate time steps. The influence of larger areas is tested in later sections of this work. This safety region thus prevents unnatural patterns from forming in the numerical solution.

Calculation order

When using LTS, the activation of cell processes can be done either at the start of its local time step, or at the end. In both cases, the time steps fill the global common time

step. However, if cells are activated at the start, inaccurate behavior can arise. When water flows into the cell, decreasing the required time step, the local time of that cell is already ahead, due to the earlier larger time step. Thus, for the front of a moving wave, the time step is only adapted after a global common time step. For this reason, we choose to activate processes at the end of a time step. This ensures that the time step of other cells can be adapted due to inflow of water.

The movement of interface fluxes is implemented in a strictly mass-conserving manner. When a cell boundary interface is used for a specific cell, water is immediately added or subtracted from the relevant neighbor. Effectively, this causes the method to move water fluxes twice, since cells are activated by neighbors, and by itself. To compensate for this, we move half of the flux during the activation of the cell, and half during the activation by a neighbor cell.

Implementation of friction

In the majority of flow types, the gravitational and frictional force predominantly determine the flow velocity. Due to the non-linearity of the friction law, solving the balance between these forces requires a small high temporal precision. Because of this, the stability of the interaction between these two forces can be increased by using a non-explicit solution (Fiedler and Ramirez 2000; Bristeau and Coussin 2001). A semi-implicit solution that provides a well-balanced and stable estimation of the frictional force was derived by Bristeau and Coussin (2001). This semi-implicit solution estimates friction based on all other acceleration terms (Eqs. 27 and 28).

$$\overline{du^{n+1}} + dt \frac{g}{n^2} \frac{\overline{u^{n+1}} \left| \overline{u^n} \right|}{h^{\frac{4}{3}}} = \overline{u}_{\text{no friction}}^{n+1} = u^n + \sum \text{other acceleration terms}. \quad (27)$$

Thus,

$$\overline{u}^{n+1} = \frac{\overline{u}_{\text{no friction}}^{n+1}}{1 + dt \frac{g}{n^2} \frac{\left| \overline{u^n} \right|}{h^{\frac{4}{3}}}}. \quad (28)$$

While the accuracy of this method is still dependent on the time step, the possible time step while maintaining accuracy is substantially larger.

A disadvantage of this method is, however, that when time steps increase above a stable level, there is increasing oscillation around the analytical solution (Fig. 4a), e.g.: In the case of Fig. 4a, the final velocity of the flow will converge to $1 \frac{3}{4} \text{ m s}^{-1}$. Depending on the time step, the semi-explicit method either shows a correct adjustment of velocity

Fig. 4 **a**, left Over- and underestimations of velocity due to oscillatory behavior of semi-explicit friction at larger time steps. Each curve represents a solution to flow velocity using a constant time

step ($dt=0.05, 0.1, 0.15, \dots, 0.5$). **b**, middle A change in time step ($dt=0.05, 0.1, 0.15, \dots, 0.5$) results in increasing flow velocities when using the semi-explicit solution

in the first 0.3 s, or several oscillating steps that over- and underestimate the flow velocity. These oscillations are especially significant in the case of shallow flow. In such conditions, the relative size of the frictional force increases, while the gravitational acceleration remains identical. Thus, the final stable velocity decreases (Fig. 4b).

When flow height is low, inertial acceleration terms are many orders of magnitude smaller compared to other acceleration terms. Therefore, velocity would in reality be determined almost immediately by the balance of the friction and gravitational forces. Because of this, we counteract these velocity oscillations by limiting the oscillations of the friction solution at the stable velocity. To enable the larger time step for areas with a low water height, we further alter the courant condition (Eqs. 28 and 29)

$$\frac{\max(v, v + ka)dt}{dx} < c \quad (28)$$

With

$$k = 1 - \min \left(1.0, \max \left(0.0, \frac{1}{c_k} (c_k - h) \right) \right) \quad (29)$$

with k the kinematic factor (–) and c_k the kinematic depth (m).

Thus, below water depth c_k , the influence of acceleration on the required time step decreases linearly to zero. The value for this depth must be chosen so that below this water depth, the velocities are well approximated by a kinematic flow. Therefore, we choose a value of 0.25 in later simulations. Finally, shallow overland is thus calculated using large time steps, and velocity is immediately determined by the balance of gravity and friction, effectively mixing a full shallow water equation for the deeper water and a kinematic flow for the shallow runoff.

Numerical tests

Dam-break test

The first scenarios tested with the LTS setup are 1D dam-break simulations. First, a numerical simulation is compared with an analytical solution of Ritter (1892). Here, a homogeneous water level is located in one-half of the simulation, with the other half dry. This initial water volume is released without obstacles, ignoring any frictional forces. In this type of simulation, sudden and large changes in local time step are present, and thus forms a good test for the accuracy and possible influences of LTS. In a second comparison, the height of a point water release is compared to an analytical solution which is based on the dam-break solution by Ancely et al. (2007). For a frictionless water body on a sloped surface, they derived an analytical expression for water height after the release of the water body. Their analytical solution will be compared to a numerical solution with and without LTS. Friction forces will be similarly neglected in this simulation.

Catchment simulations

The created numerical method has undergone testing on catchment models for two study sites. The first catchment is a 1020 ha urban area from the St. Lucia Island (Fig. 5). This catchment has a medium topography, generally with medium slopes (5° – 10°), and altitudes between 0 and 400 m above sea level. While the main land uses are tropical forest and agroforestry, many buildings and roads are present. These are included in the simulations automatically through interception, infiltration and runoff. The hydrological connection to the sea is provided by a small (~ 5 m) channel in the central region of the catchment. The

Fig. 5 An overview of the St. Lucia catchment. Hill shaded elevation, land use and soil texture are shown

climatic conditions of this area are tropical and the majority of the rainfall occurs during the strong monsoon season. Beyond the regular extreme rainfall events, the island of St. Lucia is recurrently hit by tropical hurricanes. These provide rainfall intensities beyond the normal distribution. For this catchment a spatially homogeneous 1 in 5 years design storm will be used, according to the frequency distribution of hurricane events. The dataset of this catchment is available in a resolution of 10 m, was developed as part of the CHARIM project (Caribbean Handbook on Risk Information Management) and is openly available on their website.

The second catchment is a 2000 ha region along the northern Italian Alps that has been investigated by Borga et al (2007) (Fig. 6). The area has a medium topography, generally with medium slopes (5° – 10°), and altitudes between 540 and 2240 m above sea level. Land use in the region consists mainly of multiple types of forest and heathland in the upslope areas, and small build-up regions in the lower. Annual rainfall varies between 1600 and 1800 mm per year. Rainfall data are available for an extreme event on 29 August 2003. During this event, multiple upslope branches of tributaries of the Fella River experienced flash flooding. Continuous discharge data for this event has been provided. The dataset for this catchment has a resolution of

20 m. The dataset for this catchment was made as part of the IncREO project (Increasing Resilience through Earth Observation-IncREO).

For both simulations, we set the Courant factor at 0.25 since this has provided accurate results in previous calibrations where no dynamic time step was used (Bout and Jetten 2018). The global common time step is set at 20 s for the St. Lucia catchment and 60 s at the Fella River catchment. These values depend on the grid cell size and are chosen based on stable numerical steps that can be taken for infiltration, the ground water balance, and slope stability (van Beek 2002).

For the St. Lucia scenario, both spatial patterns of flow and outlet discharge are compared. The results from the simulations without LTS are used as a reference.

Results and discussion

Comparison with analytical solutions

For several moments during the simulation, the flow height and flow velocities are displayed for comparison in Figs. 7 and 8, with the frictionless analytical dam-break solution from Ritter (1892).

Fig. 6 Catchment overview for the Fella Basin, Northern Italy. The elevation model, land use type and soil textures are shown

Fig. 7 Numerical simulation of frictionless dam break using traditional and LTS scheme, compared with analytical solution by Ritter (1892). Cell size for simulation: 0.5 m, Time (T) given in seconds

Fig. 8 Numerical simulation of frictionless dam break using traditional and LTS scheme, compared with analytical solution by Ritter (1892). Cell size for simulation: 0.5 m, Time (T) given in seconds

A comparison of the analytical and numerical solution of a water release, as described by Ancey et al. (2008), is shown in Fig. 9. The peak water height at every second is provided for the analytical solution, and the numerical solutions with and without LTS.

Based on the comparison, the performance of the method numerical methods is similar to the used numerical methods (MUSCL, HLLC Riemann Solver). The accuracy of these numerical schemes have furthermore been proven during their development (Zoppou et al. 1999; Delis and Katsaounis 2003; Delestre et al. 2014). The differences between the numerical solution with and without LTS are minor. Flow

height is altered slightly with values below percent. This difference originates from a change in velocities, since these determine inter-cell fluxes. Velocities in the case of the normal simulation are slightly higher, leading to a slight advance in water flow. Reason for this changes in velocities is found in the approximations made in the numerical scheme. With altering time step, the accuracy of the scheme changes slightly. In general, the normal simulation enforces the lowest found time step on each cell, therefore increasing the accuracy. Typically, the normal simulation should be at least as accurate in its solution as the LTS simulation.

Fig. 9 Comparison of a numerical and analytical solution of water height after a dam break on a sloped surface. Analytical solution by Ancy et al. (2008)

However, where small accuracy errors are accepted for practical reasons in any numerical model, the same holds true for the LTS. The average time steps during the numerical simulations are 0.079 and 0.032 s, respectively, for the normal and LTS simulation. Without significant differences in simulation results, the effective time step and thus numerical efficiency has increased substantially. Finally, note that the inflow of water is sudden, which requires a quick change in local time step. Here, the cell calculation order becomes significant. When larger time steps are calculated first, inflow from faster-moving water is ignored until the cell is activated again, leading to unnatural flow boundaries. However, when small time steps are executed first, cells with incoming flow automatically adapt and are activated when required.

Numerical tests

Results of local time stepping

In Fig. 10, the spatial patterns of maximum flow depth for the St. Lucia scenario, simulated without LTS, are shown. Figure 11 shows these same patterns for the Fella Basin flood scenario. Figure 12 shows the output hydrograph for the St. Lucia catchment, and the relative difference between a normal and LTS scheme. The absolute difference between the resulting maximum flood depth with and without LTS is furthermore provided.

The usage of LTS causes minor changes in simulated maximum flood depth. On average, a deviation of 4.3×10^{-4} percent was found in the St. Lucia scenario. These differences are similar to the results of the sensitivity analysis of other numerical methods (Delestre et al. 2014; Haile and Rientjes 2005; Alcrudo and Garcia-Navarr, 1993; Sanders et al. 2010). Differences in maximum flood depth are higher near higher water depths, but show separate patterns based on topography. On average, the LTS simulation shows higher flood depths, due to slightly smaller fluxes. The output hydrographs of the St. Lucia catchment similarly show minor differences. The relative difference

Fig. 10 Maximum flood depth for the ST. Lucia simulation, and the difference between simulations with a traditional and LTS scheme

is highest at the start of the event. Here, a sudden increase in discharge takes place when the flood event arrives at the outlet. When this occurs in one simulation and not the other, large relative differences occur for a small amount of time. Thus, a slight change in timing of the arrival of the flood event can lead to a larger difference at the start of the hydrographs.

Computation time

For the St. Lucia catchment, the comparison between effective time step throughout the simulation is shown in Fig. 14. A map showing the spatial flow depths and time steps at a fifth of the event are shown in Fig. 13.

A table of simulation times and the information of the used hardware is given in Table 1.

For both simulations, there was a decrease of more than 50% in computation time. The gradual increase and decrease in average time step in Fig. 14 is caused by the gradual increase and decrease of flood area during the simulation. The decrease in computation time is not directly proportional to the decrease in average time step. This is caused by several aspects of the model. Firstly, processes besides flow are computed during the global common time step. Here, processes such as interception, rainfall, infiltration and other slower processes take up computational time. Secondly, the usage of LTS requires the model to keep track of cell-specific time states and time step values. This creates a certain amount of overhead that would not be present in a normal adaptive time step approach.

Fig. 11 Maximum flood depth for the Fella Basin simulation, and the difference between simulations with a traditional and LTSScheme

Fig. 12 The St. Lucia hydrographs using traditional and LTSScheme simulations, and the relative difference

Especially, the comparison with the time step of neighboring cells provides substantial overhead computation.

Application to multi-hazard modeling

To further test the developed implementation of LTS, we perform a multi-hazard simulation using the full two-phase generalized debris flow equations from Pudasaini (2012). We apply the model on a Catchment in Dominica, which was hit by hurricane Maria in 2017. An overview of the catchment, the land use and soil types found in the area are shown in Fig. 15.

The area is mostly covered by dense tropical forests and features very steep slopes prone to slope failure. The active landscape of the volcanic island consists of clay-rich weathering products of volcanic rock and pyroclastic flow deposits. Internal friction angles have been taken from literature to be 26° . Soil parameters are used from pedotransfer functions

Fig. 13 Flow heights and local time step at a fifth of the simulated event

by Saxton et al. (2006). The rainfall input is a measurement of the Hurricane Maria event, with a total of 540 mm during a period of 24 h, and peak intensities of over 130 mm/h. During the simulation, a large number of slope failures are initiated due to increased pore water pressure and weight of the soil columns on slopes. The mass movements originate

Table 1 Simulation times using traditional and LTS schemes

Simulation using intel core i7 hq 6700	Simulation property			
	Simulation time with LTS	Average Δt with LTS	Simulation time without LTS	Average Δt without LTS
St. Lucia	459 min	11.5	1033 min	3.7
Fella Basin	1232 min	37.4	2889 min	3.2

Fig. 14 Graph of the effective time step in the St. Lucia simulation

from shallow failures (< 10 m depth) and typically quickly evolve from shallow landslides into debris flows. We simulate runout based on the slope failures.

Results

The results from the simulation of Hurricane Maria on the southern part of Dominica are shown in Figs. 16, 17 and 18. The differences between simulations with and without LTS are highlighted.

Similar to the case for the earlier simulations of catchment-integrated flash flood behavior using LTS, the model shows minimal differences between simulation results with or without LTS. The primary difference in the simulations treatment of differences induced by LTS is the cascading nature of the processes. Due to the interactions, and threshold-like behaviors in the equations used to estimate slope failure, runout, and two-phase flow behavior, a more chaotic system with a higher number of degrees of freedom is simulated. A most direct example can be found in slope failure itself. Here, a slow increase in soil saturation does not gradually

Fig. 15 An overview of the terrain, land cover and soil texture for a southern catchment on Dominica

Fig. 16 Predicted flash flooding on Dominica for the Hurricane Maria event. Differences with LTS are shown

Fig. 17 Predicted slope failure on Dominica for the Hurricane Maria event. Differences with LTS are shown

Fig. 18 Predicted debris flows on Dominica for the Hurricane Maria event. Differences with LTS are shown

influence the simulation results. Instead, after a threshold value is reached, a strong effect is suddenly activated. Thus, small changes in input can lead to processes reaching these threshold values. Therefore, small changes in the simulation could later cascade into major differences in output. Here, the changes induced by LTS are relatively small so that even later differences in output remain insubstantial. Computation time for the multi-hazard simulation reduced from 1124 to 673 min, with the average time step increased from 2.8 to 10.3 s. Similar to the flood scenarios, the applicability of such methods increases with the efficiency of the model, opening the way for ensemble simulations, or real time application.

Conclusion

We have compared the performance of an adapted friction force implementation for the Saint Venant equations, combined with LTS. The altered implementation of the friction force showed an increase in stability and accuracy, when time steps are forced to be larger than allowed by the Courant condition. With increasing time steps, it took longer before results deviated from a reference simulation. The kinematic time step constant was furthermore varied. Results from these simulations indicated that the assumptions about shallow (<0.1 m) runoff did not significantly alter simulation results.

A comparison with an analytical solution for an ideal dam-break showed insubstantial differences between the numerical

solutions with and without LTS. Numerical simulations in the study catchment of St. Lucia, Dominica and the Fella basin provided identical results, showing that in both small and large-scale simulations, the usage of LTS can allow for a larger time step while maintaining accuracy and stability. Based on the performed test, several varieties of LTS have shown more accurate results. Especially the calculation order substantially influenced the deviation from the regular numerical solution. The usage of a variety of neighborhoods to decrease time step around unstable areas has a minor effect on the simulation results, but is in theory required for a TVD simulation.

While the usage of LTS maintains accuracy, it provides a substantial increase in computational efficiency. Due to the increase in effective time step, computational costs for a stable simulation decreased. For the St. Lucia and Fella Basin study catchments, an average increase in computation time of 1.9 was found. Because of the decrease in computation time, LTS provides a good option for catchment-scale assessment of flood dynamics. Furthermore, we have shown this increase in efficiency translates to multi-processes land surface models for hydrology, flow and multi-hazard events. This indicates the wider potential benefit of LTS for a variety of models in environmental studies.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes

were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Alcrudo F, Garcia-Navarro P (1993) A high-resolution Godunov-type scheme in finite volumes for the 2D shallow-water equations. *Int J Numer Meth Fluids* 16(6):489–505
- Baartman JE, Temme AJ, Veldkamp T, Jetten VG, Schoorl JM (2013) Exploring the role of rainfall variability and extreme events in long-term landscape development. *CATENA* 109:25–38
- Borga M, Boscolo P, Zanon F, Sangati M (2007) Hydrometeorological analysis of the 29 August 2003 flash flood in the Eastern Italian Alps. *J Hydrometeorol* 8(5):1049–1067
- Bouchut F (2004) Nonlinear stability of finite volume methods for hyperbolic conservation laws: and well-balanced schemes for sources. Springer Science & Business Media, Berlin
- Bout B, Jetten VG (2018) The validity of flow approximations when simulating catchment-integrated flash floods. *J Hydrol* 556:674–688
- Bout B, Lombardo L, van Westen CJ, Jetten VG (2018) Integration of two-phase solid fluid equations in a catchment model for flash-floods, debris flows and shallow slope failures. *Environ Model Softw* 105:1–16
- Bristeau MO, Coussin B (2001) Boundary conditions for the shallow water equations solved by kinetic schemes (Doctoral dissertation, INRIA)
- Brunner GW (2010) HEC-RAS (River Analysis System). In: North American Water and Environment Congress & Destructive Water, pp 3782–3787, ASCE.
- Chen HX, Zhang LM (2015) EDDA 1.0: integrated simulation of debris flow erosion, deposition and property changes. *Geosci Model Dev* 8(3):829–844
- Dazzi S, Vacondio R, Dal Palù A, Mignosa P (2018) A local time stepping algorithm for GPU-accelerated 2D shallow water models. *Adv Water Resour* 111:274–288
- Delestre O, Cordier S, Darboux F, Du M, James F, Laguerre C, Planchon O (2014) FullSWOF: a software for overland flow simulation. In: *Advances in hydroinformatics*. Springer, Singapore, pp 221–231
- Delis AI, Katsaounis T (2003) Relaxation schemes for the shallow water equations. *Int J Numer Meth Fluids* 41(7):695–719
- Dumbser M, Käser M, Toro EF (2007) An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes-V. Local time stepping and p-adaptivity. *Geophys J Int* 171(2):695–717
- Feng LH, Lu J (2010) The practical research on flood forecasting based on artificial neural networks. *Expert Syst Appl* 37(4):2974–2977
- Fiedler FR, Ramirez JA (2000) A numerical method for simulating discontinuous shallow flow over an infiltrating surface. *Int J Numer Meth Fluids* 32(2):219–239
- Green WH, Ampt GA (1911) Studies on soil physics. *J Agric Sci* 4(01):1–24
- Haile AT, Rientjes THM (2005) Effects of LiDAR DEM resolution in flood modelling: a model sensitivity study for the city of Tegucigalpa, Honduras. *Isprs wg iii/3, iii/4* 3:12–14
- Harten A (1983) High resolution schemes for hyperbolic conservation laws. *J Comput Phys* 49(3):357–393
- Harten A, Lax PD, van Leer B (1983) On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Rev* 25(1):35–61
- Hunter NM, Horritt MS, Bates PD, Wilson MD, Werner MG (2005) An adaptive time step solution for raster-based storage cell modelling of floodplain inundation. *Adv Water Resour* 28(9):975–991
- Jetten VG, de Roo AP (2001) Spatial analysis of erosion conservation measures with LISEM. In: *Landscape erosion and evolution modeling*. Springer, Boston, pp 429–445
- Kurganov A, Petrova G (2007) A second-order well-balanced positivity preserving central-upwind scheme for the Saint-Venant system. *Commun Math Sci* 5(1):133–160
- LeVeque RJ (2002) Finite volume methods for hyperbolic problems (Vol. 31). Cambridge University Press, Cambridge
- Madec R, Komatitsch D, Diaz J (2009) Energy-conserving local time stepping based on high-order finite elements for seismic wave propagation across a fluid-solid interface. *Comput Model Eng Sci (CMES)* 14(2):163
- Muradoglu M, Pope SB (2002) Local time-stepping algorithm for solving probability density function turbulence model equations. *Aiaa J* 40(9):1755–1763
- Novak P, Guinot V, Jeffrey A, Reeve DE (2010) Hydraulic modelling—an introduction: principles, methods and applications. CRC Press, Boca Raton
- Ritter A (1892) Die fortpflanzung de wasserwellen. *Zeitschrift Verein Deutscher Ingenieure* 36(33):947–954
- Roe PL (1986) Characteristic-based schemes for the Euler equations. *Annu Rev Fluid Mech* 18(1):337–365
- Sanders BF (2008) Integration of a shallow water model with a local time step. *J Hydraul Res* 46(4):466–475
- Sanders BF, Schubert JE, Detwiler RL (2010) ParBreZo: a parallel, unstructured grid, Godunov-type, shallow-water code for high-resolution flood inundation modeling at the regional scale. *Adv Water Resour* 33(12):1456–1467
- Sedaghat A, Ackroyd JAD, Wood NJ (1999) Turbulence modelling for supercritical flows including examples with passive shock control. *Aeronaut J* 103(1020):113–125
- Smith RE, Parlange JY (1978) A parameter-efficient hydrologic infiltration model. *Water Resour Res* 14(3):533–538
- Sohn SI (2005) A new TVD-MUSCL scheme for hyperbolic conservation laws. *Comput Math Appl* 50(1–2):231–248
- Te Chow V (1964) Applied hydrology. Tata McGraw-Hill Education
- Toro EF, Spruce M, Speares W (1994) Restoration of the contact surface in the HLL-Riemann solver. *Shock Waves* 4(1):25–34
- Toro EF (2013) Riemann solvers and numerical methods for fluid dynamics: a practical introduction. Springer Science & Business Media, Berlin
- Van Albada GD, Van Leer B, Roberts W (1997) A comparative study of computational methods in cosmic gas dynamics. In: *Upwind and high-resolution schemes*. Springer, Berlin, Heidelberg, pp 95–103
- van Beek L PH (2002) Assessment of the influence of changes in land use and climate on landslide activity in a Mediterranean environment (Doctoral dissertation).
- Van den Broek B (1996) Dutch experience in irrigation water management modelling. DLO Winand Staring Centre, Wageningen, The Netherlands
- Yee HC, Harten A (1987) Implicit TVD schemes for hyperbolic conservation laws in curvilinear coordinates. *AIAA J* 25(2):266–274
- Zoppou C, Roberts S, Molls T, Molls DF (1999) Space-time conservation method applied to Saint Venant equations. *J Hydraul Eng* 125(8):891–891

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.