



A simple inertial formulation of the shallow water equations for efficient two-dimensional flood inundation modelling

Paul D. Bates^{a,*}, Matthew S. Horritt^b, Timothy J. Fewtrell^a

^a School of Geographical Sciences, University of Bristol, University Road, Bristol BS8 1SS, UK

^b Halcrow Ltd., Burderop Park, Swindon, Wiltshire SN4 0QD, UK

ARTICLE INFO

Article history:

Received 27 May 2009

Received in revised form 19 October 2009

Accepted 22 March 2010

This manuscript was handled by K. Georgakakos, Editor-in-Chief, with the assistance of Ehab A. Meselhe, Associate Editor

Keywords:

Shallow water flow
Flood propagation
Inundation modelling
Hydraulic modelling

SUMMARY

This paper describes the development of a new set of equations derived from 1D shallow water theory for use in 2D storage cell inundation models where flows in the x and y Cartesian directions are decoupled. The new equation set is designed to be solved explicitly at very low computational cost, and is here tested against a suite of four test cases of increasing complexity. In each case the predicted water depths compare favourably to analytical solutions or to simulation results from the diffusive storage cell code of Hunter et al. (2005). For the most complex test involving the fine spatial resolution simulation of flow in a topographically complex urban area the Root Mean Squared Difference between the new formulation and the model of Hunter et al. is ~ 1 cm. However, unlike diffusive storage cell codes where the stable time step scales with $(1/\Delta x)^2$, the new equation set developed here represents shallow water wave propagation and so the stability is controlled by the Courant–Freidrichs–Lewy condition such that the stable time step instead scales with $1/\Delta x$. This allows use of a stable time step that is 1–3 orders of magnitude greater for typical cell sizes than that possible with diffusive storage cell models and results in commensurate reductions in model run times. For the tests reported in this paper the maximum speed up achieved over a diffusive storage cell model was 1120 \times , although the actual value seen will depend on model resolution and water surface gradient. Solutions using the new equation set are shown to be grid-independent for the conditions considered and to have an intuitively correct sensitivity to friction, however small instabilities and increased errors on predicted depth were noted when Manning's $n = 0.01$. The new equations are likely to find widespread application in many types of flood inundation modelling and should provide a useful additional tool, alongside more established model formulations, for a variety of flood risk management studies.

© 2010 Elsevier B.V. All rights reserved.

Introduction

Since first proposed by Zanobetti et al. (1970) methods to predict floodplain inundation using storage cell approaches have become justifiably popular. Initially, such methods discretized floodplains into irregular polygonal units representing large (surface areas of 10^0 – 10^1 km²) natural storage compartments and calculated the fluxes of water between these according to some uniform flow formulae such as the weir or Manning's equations. For many such models in-channel flows are calculated using some form of the 1D Saint–Venant equations, and when bankfull flow is exceeded water is routed into and between the floodplain storage units. Most commercial 1D codes now include such a floodplain representation. More recently the availability of increased computing power and detailed descriptions of floodplain topography available through remote sensing (e.g. LiDAR data, Marks and Bates, 2000) has

allowed a move away from large, irregular storage units to the discretization of the floodplain as a fine spatial resolution regular grid (cell areas of 10^{-2} – 10^{-3} km²). Here each cell within the grid is a storage area for which the mass balance is updated at each time step according to the fluxes of water into and out of each cell. Similar to polygonal storage cell models, fluxes are calculated analytically using uniform flow formulae but with the advantage of higher resolution predictions and removal of the need for the modeller to make explicit decisions about the location of storage compartments and the linkages between these. Numerous such models are now available (e.g. Estrela and Quintas, 1994; Bechteler et al., 1994; Bates and De Roo, 2000) and a similar blueprint has increasingly been adopted in commercial modelling packages (e.g. JFLOW by JBA Ltd., FlowRoute by Ambiental and the RMS Ltd., UK Flood Risk Model). Such models therefore solve a continuity equation relating flow into a cell and its change in volume:

$$\frac{\Delta h}{\Delta t} = \frac{\Delta Q}{\Delta x \Delta y} \quad (1)$$

* Corresponding author. Tel.: +44 117 928 9108; fax: +44 117 928 7878.
E-mail address: paul.bates@bristol.ac.uk (P.D. Bates).

and a flux equation for each direction where flow between cells is calculated according to Manning's law (only the x direction is given here):

$$Q_x^{ij} = \frac{h_{\text{flow}}^{5/3}}{n} \left(\frac{h^{i-1j} - h^{ij}}{\Delta x} \right)^{1/2} \Delta y \quad (2)$$

where h^{ij} is the water free surface height [L] at the node (i, j) , Δx and Δy are the cell dimensions [L], t is the time [T], n is the Manning's friction coefficient [$L^{-1/3} T$], and Q_x and Q_y describe the volumetric flow rates between floodplain cells [$L^3 T^{-1}$]. Q_y is defined analogously to Eq. (2). The flow depth, h_{flow} , represents the depth through which water can flow between two cells, and is defined as the difference between the highest water free surface in the two cells and the highest bed elevation. These equations are solved explicitly using a finite difference discretization of the time derivative term:

$$\frac{{}^{t+\Delta t}h^{ij} - {}^th^{ij}}{\Delta t} = \frac{{}^tQ_x^{i-1j} - {}^tQ_x^{ij} + {}^tQ_y^{ij-1} - {}^tQ_y^{ij}}{\Delta x \Delta y} \quad (3)$$

where th and tQ represent depth and volumetric flow rate at time t respectively, and Δt is the model time step which is held constant throughout the simulation.

The advantage of the storage cell formulation is that fluxes are calculated analytically so the computational costs per time step are potentially much lower than in equivalent numerical solutions of the full shallow water equations. The method is also simple in concept and it is therefore relatively easy to develop and maintain code that can perform the calculations. Such methods also interface readily with newly available remotely sensed terrain data which typically arrives in the form of a regular grid. For this reason the number of research and commercial codes based on these techniques has proliferated over the last decade (for a review see Hunter et al. (2007)). Whilst the method can only be applied to gradually varied flows and does not include inertia or the ability to capture supercritical effects, for many floodplain inundation problems the representation is appropriate.

Such storage cell models were originally conceived for application at relatively coarse grid resolutions (25–100 m) and early applications showed that at these scales there was a distinct computational advantage over full solutions of the 2D Saint-Venant equations (see for example Horritt and Bates, 2001, 2002). This allowed new applications of hydraulic models to be considered including Monte Carlo uncertainty analysis (Aronica et al., 2002), inclusion of hydraulic models in ensemble forecasting chains (Pappenberger et al., 2005) and model applications to domain scales orders of magnitude larger than anything previously attempted (e.g. Wilson et al., 2007). Despite these successes, a number of concerns became apparent. First, unless the constant time step used to solve Eq. (3) was small, simulations with storage cell models quickly developed 'chequerboard' type instabilities as all the water in a particular cell drained into the adjacent ones in a single (large) time step (Cunge et al., 1980). At the next time step, this situation would reverse and all the water would flow back. To solve this problem many modellers introduced some kind of 'flow limiter' to prevent the solution over-shooting and too much water leaving a given cell in a single time step. The flow limiter sets the maximum flow that can occur between cells and is typically a function of flow depth, grid cell size and time step. In LISFLOOD-FP, for example, the flow limiter used is:

$$Q_x^{ij} = \min \left(Q_x^{ij}, \frac{\Delta x \Delta y (h^{ij} - h^{i-1j})}{4 \Delta t} \right) \quad (4)$$

This value is determined by considering the change in depth of a cell, and ensuring it is not large enough to reverse the flow in or out

of the cell at the next time step. This limiter replaces fluxes calculated using Manning's equation with values dependent on model parameters, and hence when the flow limiter is in use floodplain flows are sensitive to grid cell size and time step, and insensitive to Manning's n .

Flow limiters were rarely discussed in journal publications at the time as their significance was not appreciated, but it was clear from the cell sizes and time steps used in these early applications that for many cells at each time step a flow limiter was being invoked. As a result flow-limited storage cell models often showed very little sensitivity to floodplain friction and their results were strongly dependent on the grid size and time step selected. There is a legitimate debate over the degree of sensitivity to floodplain friction one should expect in an inundation model given that floodplain velocities are usually very small, but the almost complete lack of such sensitivity in certain applications of flow-limited storage cell models appeared to be more than could be explained in physical terms.

A solution to this problem was provided by Hunter et al. (2005) based on adaptive time-stepping. This approach seeks to remove the need to invoke the flow limiter (Eq. (4)) by finding the optimum time step (large enough for computational efficiency, small enough for stability) at each iteration. This optimum time step is obtained using an analysis of the governing equations and their analogy to a diffusion system which gives the following expression for Δt :

$$\Delta t = \frac{\Delta x^2}{4} \min \left(\frac{2n}{h_{\text{flow}}^{5/3}} \left| \frac{\partial h}{\partial x} \right|^{1/2}, \frac{2n}{h_{\text{flow}}^{5/3}} \left| \frac{\partial h}{\partial y} \right|^{1/2} \right) \quad (5)$$

A scheme that uses this criterion can be implemented by searching the domain for the minimum time step value and using this to update h in Eq. (3). The time step will thus be adaptive and change during the course of a simulation, but is uniform in space at each time step. Hunter et al. (2005) tested this new unconditionally stable time step formulation against analytical solutions for wave propagation over flat and planar slopes and showed a considerable improvement over the classical fixed time-step version of the model. Moreover, the adaptive scheme was shown to yield results that were independent of grid size or choice of initial time step and which showed an intuitively correct sensitivity to floodplain friction over spatially-complex topography. Hunter et al. (2006) went on to test the new version of the LISFLOOD-FP model against real world flood extent and wave travel time data for the upper River Severn in the UK for a model at 60 m spatial resolution. The adaptive time step model showed a better absolute performance than the classical fixed time-step version at this spatial resolution, but at approximately six times the computational cost. In particular the adaptive model appeared able to simulate floodplain wetting and drying more realistically.

Despite this success, the results obtained by Hunter et al. (2006) identified a fundamental problem with Eq. (5), namely that the optimum stable time step reduces quadratically with decreasing grid size. For an explicit code this means that the computational cost will increase as $(1/\Delta x)^4$. For applications with grid sizes in the range for which LISFLOOD-FP was originally designed (25–100 m, Bates and De Roo, 2000) this led to a 2–10 \times increase in simulation times which could be offset through advances in processor speed. Any residual cost increases could then be justified easily as simulations were more realistic. However, for the finer resolution (1–10 m) grids required for application of hydraulic models to urban areas (Fewtrell et al., 2008) simulation costs increased by several orders of magnitude such that at these scales adaptive time step storage cell codes actually proved slower than full 2D solutions of the shallow water equations. Hunter et al. (2008) found during benchmark testing of six 2D models applied at 2 m spatial

resolution to a 0.4 km² area of Glasgow UK that numerical solutions of the full shallow water equations completed a 2 h real time simulation in around 1 h (depending on code complexity and processor architecture), whilst the storage cell codes took approximately an order of magnitude longer.

Although a less serious problem, the dependence of the time step on the water surface slope in Eq. (5) also means that the time step is reduced for areas with flat water surfaces, where intuitively we would expect the governing equations to be easier to solve. In the limit of a horizontal water surface, the time step is forced to zero, whereas the solution (zero flow in all directions) is trivial. In practice, the divergence of computation times in time-adaptive storage cell models as the water comes to rest is avoided by applying a linearization for small surface slopes (see Hunter et al., 2005).

Adaptive time step storage cell codes are therefore incompatible with the fine spatial resolution grids increasingly required for urban flood modelling. The only solution to date is to invoke a flow limiter, but this leads to a poor representation of flow dynamics. Whilst for fine grids full 2D models give shorter simulation times at current processor speeds, for practical applications they are still only able to treat small (<1 km²) areas at the required level of detail. It is clear that to allow wide area urban flood modelling at fine spatial resolution a new hydraulic model formulation is required. Development and testing of such an approach is the fundamental aim of this paper where we describe a set of flow equations for adaptive time step storage cell models which can overcome the quadratic dependency on grid size in Eq. (5) yet which can be solved analytically with approximately the same computational cost as Eq. (2). The new scheme therefore retains all the computational advantages of storage cell models over full 2D codes whose equations require expensive numerical solution, yet with none of the previous disadvantages. Below we describe the derivation of the new set of equations from first principles, and then the new formulation is subject to a number of analytical and benchmark tests of increasing complexity. Finally, results are discussed and conclusions drawn.

Derivation of an inertial formulation of the shallow water equations

The route to a new set of equations for fast inundation modelling in two dimensions was identified in the urban model benchmarking study of Hunter et al. (2008). It was clear from this comparison that the lack of mass and inertia in Eq. (2) was the key reason why storage cell models required the strict time step control implied by Eq. (5). In gradually varying shallow water flows the effect of inertia is to reduce fluxes between cells, yet in Eq. (5) flux is merely a function of gravity and friction. Eq. (5) therefore overestimates fluxes, particularly, as noted above, in areas of deep water where there is only a small free surface gradient. Hunter et al. (2008) suggested that the solution was to modify explicit storage cell codes to include inertial terms (or simple approximations to these) that may allow the use of a larger stable time step, and hence quicker run times. In addition, inclusion of inertial effects may also be important to represent the flow physics in particular environmental settings.

Our starting point for derivation of such an equation is therefore the momentum equation from the quasi-linearized one-dimensional Saint–Venant or Shallow Water equations:

$$\underbrace{\frac{\partial Q}{\partial t}}_{\text{acceleration}} + \underbrace{\frac{\partial}{\partial x} \left[\frac{Q^2}{A} \right]}_{\text{advection}} + \underbrace{\frac{gA\partial(h+z)}{\partial x}}_{\text{water slope}} + \underbrace{\frac{gn^2Q^2}{R^{4/3}A}}_{\text{friction slope}} = 0 \quad (6)$$

where Q [L³ T⁻¹] is the discharge, A is the flow cross section area [L²], z is the bed elevation [L], R is the hydraulic radius [L], g is

the acceleration due to gravity [L T⁻²] and all other terms are defined as above. For many floodplains flows advection is relatively unimportant (see Hunter et al. (2007) for a discussion of the magnitude of terms in the shallow water equations) so we neglect this term, assume a rectangular channel and divide through by a constant flow width, w [L], to obtain an equation in terms of flow per unit width, q [L² T⁻¹]:

$$\frac{\partial q}{\partial t} + \frac{gh\partial(h+z)}{\partial x} + \frac{gn^2q^2}{R^{4/3}h} = 0 \quad (7)$$

For wide, shallow flows we can approximate the hydraulic radius, R , with the flow depth, h . We can now discretize Eq. (7) with respect to the time step, Δt , to give:

$$\left(\frac{q_{t+\Delta t} - q_t}{\Delta t} \right) + \frac{gh_t\partial(h_t+z)}{\partial x} + \frac{gn^2q_t^2}{h_t^{7/3}} = 0 \quad (8)$$

And rearrange to give an explicit equation for q at time $t + \Delta t$:

$$q_{t+\Delta t} = q_t - gh_t\Delta t \left[\frac{\partial(h_t+z)}{\partial x} + \frac{n^2q_t^2}{h_t^{10/3}} \right] \quad (9)$$

This gives an equation for the unit flow at the next time step, $q_{t+\Delta t}$, in terms of q_t , h_t and z and hence can be solved explicitly at a very similar cost to Eq. (2), as it contains only a single additional term. The advantage of this formulation is that since the acceleration term is now included, the water being modelled has some mass, and it is therefore less likely to generate the rapid reversals in flow which lead to a chequerboard oscillation. Shallow water wave propagation will also be represented, rather than the diffusive behaviour typical of previous storage cell models.

Eq. (9) can be improved further, since instabilities may still arise at shallow depths when the friction term becomes large. Replacing a q_t in the friction term by a $q_{t+\Delta t}$ leads to an equation linear in the unknown $q_{t+\Delta t}$ but which has some of the improved convergence properties of an implicit time stepping scheme:

$$q_{t+\Delta t} = q_t - gh_t\Delta t \left[\frac{\partial(h_t+z)}{\partial x} + \frac{n^2q_tq_{t+\Delta t}}{h_t^{10/3}} \right] \quad (10)$$

Eq. (10) can be rearranged into an explicit form for calculation of flows at the new time step in the model:

$$q_{t+\Delta t} = \frac{q_t - gh_t\Delta t \frac{\partial(h_t+z)}{\partial x}}{\left(1 + gh_t\Delta t n^2 q_t / h_t^{10/3} \right)} \quad (11)$$

The enhanced stability of Eq. (11) stems from the increase in the denominator as the friction term increases, forcing the flow to zero, as would be expected for shallow depths. A similar approach is used in Liang et al. (2006) to improve the stability of a full 2D shallow water model.

Unlike (2), Eq. (11) includes shallow water wave propagation so while stability is improved, it is still subject to the Courant–Freidrichs–Levy condition:

$$C_r = \frac{V\Delta t}{\Delta x} \quad (12)$$

where the non-dimensional Courant number, C_r , needs to be less than 1 for stability and V is a characteristic velocity [L T⁻¹]. In the case of a shallow water flow where advection is ignored this characteristic velocity is:

$$\sqrt{gh} \quad (13)$$

where \sqrt{gh} is the celerity of a long wavelength, small amplitude gravity wave. Eq. (12) gives a necessary but not sufficient condition

for model stability, and is used to estimate a suitable model time step at $t + \Delta t$:

$$\Delta t_{\max} = \alpha \frac{\Delta x}{\sqrt{gh_t}} \quad (14)$$

where α is a coefficient in the range 0.2–0.7 used to produce a stable simulation for most floodplain flow situations. The parameter α is included because Eq. (12) is not sufficient to ensure model stability, because the assumption of small amplitude in calculating the wave celerity is not always valid, and because of the inclusion of friction terms in the model. The stable time step is therefore often somewhat less than that indicated by the Courant–Friedrichs–Lewy condition, and so the parameter α is introduced to reduce the time step. Despite these limitations in the application of the condition, Eq. (14) represents a useful approach to time step selection for a wide range of flow conditions, subject to an appropriate choice of α .

This time step is typically 1–3 orders of magnitude larger than the stable time step for the purely diffusive scheme of Eq. (5). Moreover, within this range, proportionally larger time step differences become apparent as the grid size decreases, as for Eq. (14) time step scales with $1/\Delta x$ rather than $(1/\Delta x)^2$. Hence, we expect the new flux equation and adaptive time step constraint to be significantly more computationally efficient than previous storage cell models. The performance of this new set of equations and the extent of this potential improvement is analysed in the following section.

Model testing and results

Eqs. (11) and (14) were implemented within the LISFLOOD-FP hydraulic model of Bates and De Roo (2000). This code has been developed extensively since conception from a simple storage cell model written in the PC-RASTER language (Bates and De Roo, 2000), to a flow limited code written in C++ (Horritt and Bates, 2001) and finally to an adaptive time step code using Eq. (5) (Hunter et al., 2005, 2006). At each step previous variants were retained in the code and can be readily switched on or off. Finally, the LISFLOOD-FP code has recently been placed within a version control system, re-written in modular form (Fewtrell, 2009) and parallelized using Open-MP (Neal et al., 2009). Implementing and comparing new code variants is thus relatively straightforward given previous work on refining the code structure and bug fixing.

The new inertial formulation of LISFLOOD-FP was assessed against a structured sequence of numerical experiments that provide a rigorous test of its numerical and computational performance. Three of these tests are taken from Hunter et al. (2005), and for the final test we simulate the urban flooding problem used in the benchmarking studies of Hunter et al. (2008) and Fewtrell et al. (2008). Specifically these tests are:

Test 1: Non-breaking wave propagation over a horizontal plane and comparison to an analytical solution.

Test 2: Non-breaking wave run-up on a planar beach and comparison to an analytical solution.

Test 3: Wetting and drying of a planar beach (i.e. a full tidal cycle).

Test 4: Simulation of flood propagation through a complex street and building network at fine spatial resolution.

In each case the new inertial formulation is compared to the adaptive time step diffusive model (Eqs. (2) and (5)) of Hunter et al. (2005) in terms of root mean squared error (or difference), % volume error, minimum time step during the simulation and total computational time. Tests 1–3 were run on a single 2.66 GHz node of a dual-core Intel Core2 Duo processor with 3 Gb of RAM,

whilst Test 4 was run on a single 2.8 GHz node of a quad-core Intel Xeon Harpertown E5462 processor with 12 Mb of cache memory. The executables for both processors were built using the Intel C++ compiler. Test 4 has also been used by Hunter et al. (2008) to benchmark the performance of six 2D inundation models, by Lamb et al. (2009) to evaluate an implementation of the JFLOW adaptive time step diffusive storage cell code designed to run on massively parallel Graphics Processor Units (GPUs) and by Schubert et al. (2008) to test an unstructured finite element model for urban applications. These latter studies also report data on computational efficiency which provides important information on the likely comparative speed of models built with our new equation set.

Test 1: non-breaking wave propagation over a horizontal plane

Hunter et al. (2005) developed a one-dimensional analytical solution for inundation model testing where the full Saint–Venant equations can be simplified to yield an ordinary non-linear differential equation. This can then be solved analytically to provide rigorous validation solutions. The analytical solution presented below is for the propagation of a non-breaking wave over a horizontal plane which allows us to test the ability of inundation models to simulate wave movement correctly in the absence of a bed slope term (i.e. $S_o = 0$). In fact, this is not a true analytical solution to the inertial equation solved by the model and thus we would not expect the model results to fully converge to the analytical solution although at fine grid resolutions it should be a close approximation. The derivation of the analytical solution is given in Hunter et al. (2005) and is not repeated here, however the final equation for the water depth, h , at any point in space x or at any time, t , is:

$$h_{x,t} = \left[\frac{7}{3} (C - n^2 u^3 (x - ut)) \right]^{7/3} \quad (15)$$

where u is the component of depth-averaged velocity in the x direction, C is a constant of integration, which can be fixed by referring to the initial conditions of the problem (i.e. h at $x = 0$ and $t = 0$) and all other terms are defined as previously. Eq. (15) can now be used to provide an analytical solution against which the model can be tested with boundary conditions in the form of $h(t)$ at $x = 0$, and initial conditions in the form of $h(x)$ at $t = 0$. Here this solution was

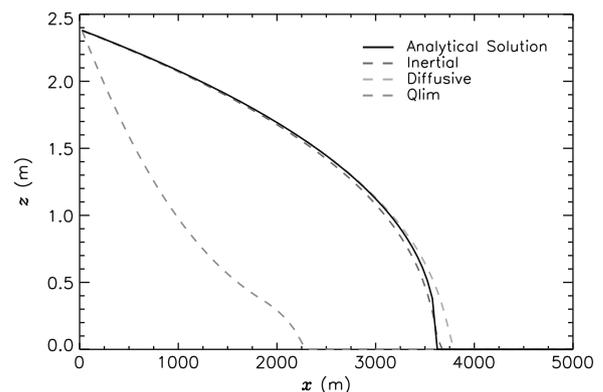


Fig. 1. Predicted water surface elevation (z) at $t = 3600$ s for wave propagation over a horizontal beach simulated at 50 m spatial resolution and $n = 0.03$ using: (a) a fixed time step ($\Delta t = 1$ s) flow limited diffusive model (Eqs. (2)–(4), light grey dashed line); (b) an adaptive time step diffusive model (Eqs. (2), (3), and (5), mid grey dashed line); and (c) the new adaptive time step inertial model developed in this paper (Eqs. (3), (11), and (14), dark grey dashed line). Each model is compared to the analytical solution of Eq. (15) (solid black line). Summary numerical results for these simulations are reported in Table 1.

Table 1
Summary of numerical and computational efficiency results from Tests 1, 2, 3 and 4.

Test case	Model	Root Mean Square Error (RMSE, in m) from analytical solution, or Root Mean Squared Difference (RMSD, in m), from diffusive model	Volume error from analytical solution (%)	Minimum time step during simulation (s)	Total computation time (min)	Speed up using inertial version
Test 1: horizontal beach, $\Delta x = 50$ m, $n = 0.03$	Diffusive	0.06	1.27	0.15	0.33	17×
	Inertial	0.03	-1.25	7.25	0.02	
Test 2: planar beach, $\Delta x = 50$ m, $n = 0.03$	Diffusive	0.02	-0.17	0.02	1.22	61×
	Inertial	0.11	-1.16	4.93	0.02	
Test 3: wetting and drying of a planar beach, $\Delta x = 50$ m, $n = 0.03$	Diffusive	-	-	0.03	1.80	60×
	Inertial	See Fig. 7	-	5.59	0.03	
Test 4: Glasgow flooding, 2 m resolution, spatially uniform friction	Diffusive	-	-	0.003	155.0	105×
	Inertial	0.01	-	0.43	1.47	

implemented using parameter values of $u = 1 \text{ ms}^{-1}$, $\Delta x = 50$ m and $n = 0.03 \text{ m}^{-1/3}\text{s}$ for a simulation of duration 3600 s.

Fig. 1 shows the water surface elevations predicted by the diffusive and inertial models at the end of the simulation (i.e. at $t = 3600$ s) compared to the analytical solution (solid black line) given by Eq. (15). For this case only, Fig. 1 also shows the water elevations predicted by a flow limited diffusive model with a fixed time step of 1 s. The Root Mean Square Error (RMSE), volume error, minimum time step during the simulation and the total computation time for each of these simulations are also summarised in Table 1. These results show that the new inertial formulation is able to match the analytical solution well, with an RMSE half that of the diffusive solution (0.03 m compared to 0.06 m) and similar volume errors. By contrast, the flow limited diffusive model simulates wave propagation, wave front position and water depths poorly and therefore is not considered here further. The minimum time step achieved during the inertial model run is also $\sim 48\times$ larger than that for the diffusive model, and this translates to a $17\times$ speed up in computational time. The gearing of minimum time step to computation time is due to the fact that: (a) for short simulations the fixed costs of running LISFLOOD-FP (data input and output, data consistency checks, etc.) actually become a relatively large proportion of the total simulation time; and (b) timings for simulations which last only a few seconds may not necessarily be sufficiently precise. Hence the full speed up potential may not be seen for this particular test case.

To test whether these results were sensitive to grid resolution, identical simulations were also run with the diffusive and inertial models at $\Delta x = 5, 10, 25, 100$ and 200 m. Predicted water eleva-

tions from these simulations at $t = 3600$ s are shown in Fig. 2 and summarised in Table 2. The inertial model outperformed the diffusive model in terms of RMSE and volume error at $\Delta x = 50, 100$ and 200 m, whereas the diffusive model was marginally better at $\Delta x = 5, 10$ and 25 m. Errors are low for all resolution models apart from $\Delta x = 200$ m where the solution quality for both schemes is dominated by the effect of the coarse grid resolution which is

Table 2
Impact of grid resolution on RMSE and volume error for simulations of non-breaking wave propagation over a horizontal plane with $n = 0.03$.

Grid resolution (m)	Model	Root Mean Square Error (RMSE, in m) from analytical solution	Volume error from analytical solution (%)	Minimum time step during simulation (s)	Total computation time (min)
5	Diffusive	0.006	-0.099	0.002	514.7
	Inertial	0.07	-2.37	0.73	2.33
10	Diffusive	0.013	0.071	0.006	32.02
	Inertial	0.065	-2.25	1.45	0.35
25	Diffusive	0.03	0.67	0.04	2.60
	Inertial	0.05	-1.89	3.62	0.12
50	Diffusive	0.06	1.27	0.15	0.33
	Inertial	0.03	-1.25	7.25	0.02
100	Diffusive	0.09	2.67	0.60	0.05
	Inertial	0.05	0.25	14.51	0.02
200	Diffusive	0.15	4.94	2.41	0.03
	Inertial	0.11	2.95	29.04	0.01

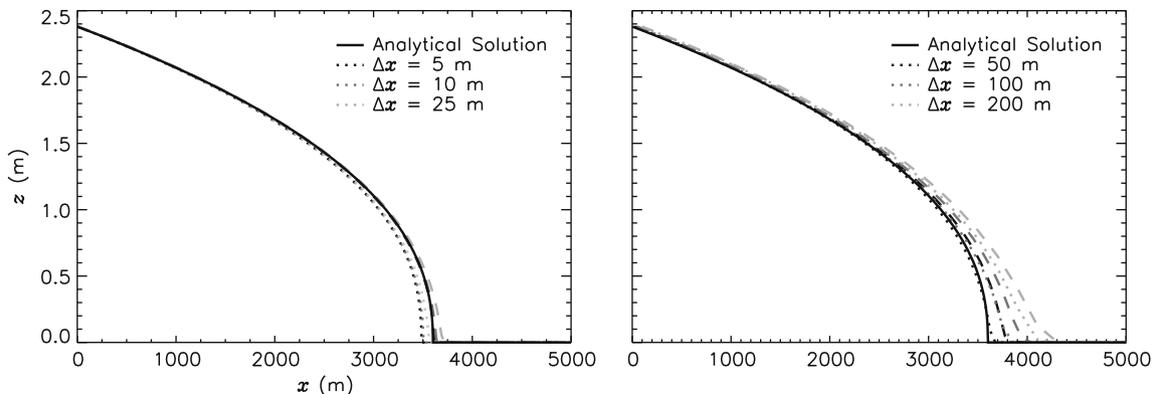


Fig. 2. Predicted water surface elevation (z) at $t = 3600$ s for wave propagation over a horizontal beach simulated at $\Delta x = 5, 10, 25, 50, 100$ and 200 m spatial resolution (denoted with dark to light grey lines respectively) and $n = 0.03$ using: (a) an adaptive time step diffusive model (dashed lines); and (b) the new adaptive time step inertial model (dotted lines). Each model is compared to the analytical solution (solid black line).

not able to resolve the steep water surface slopes at the wave front. At resolutions greater than $\Delta x = 25$ m the improvement in computation time also becomes more apparent as expected. Notably, at $\Delta x = 5$ m, the minimum time step achieved was $\sim 360\times$ larger for

the inertial formulation (0.785 s compared to 0.002 s) and this resulted in a $\sim 220\times$ decrease in run time. Fig. 3 shows the evolution of the time step Δt over the simulation and shows the quadratic reduction in the time step with decreasing grid size in the diffusive

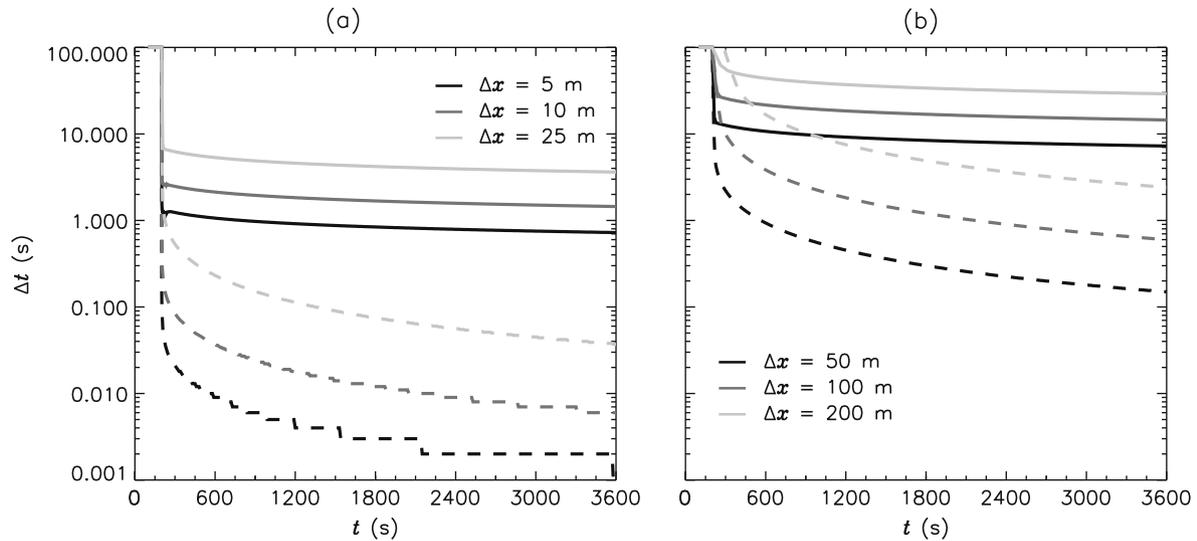


Fig. 3. Time step evolution for simulations of wave propagations over a horizontal beach at $\Delta x = 5, 10, 25, 50, 100$ and 200 m spatial resolution (denoted with dark to light grey lines respectively) and $n = 0.03$ using: (a) an adaptive time step diffusive model (dashed lines); and (b) the new adaptive time step inertial model (solid lines).

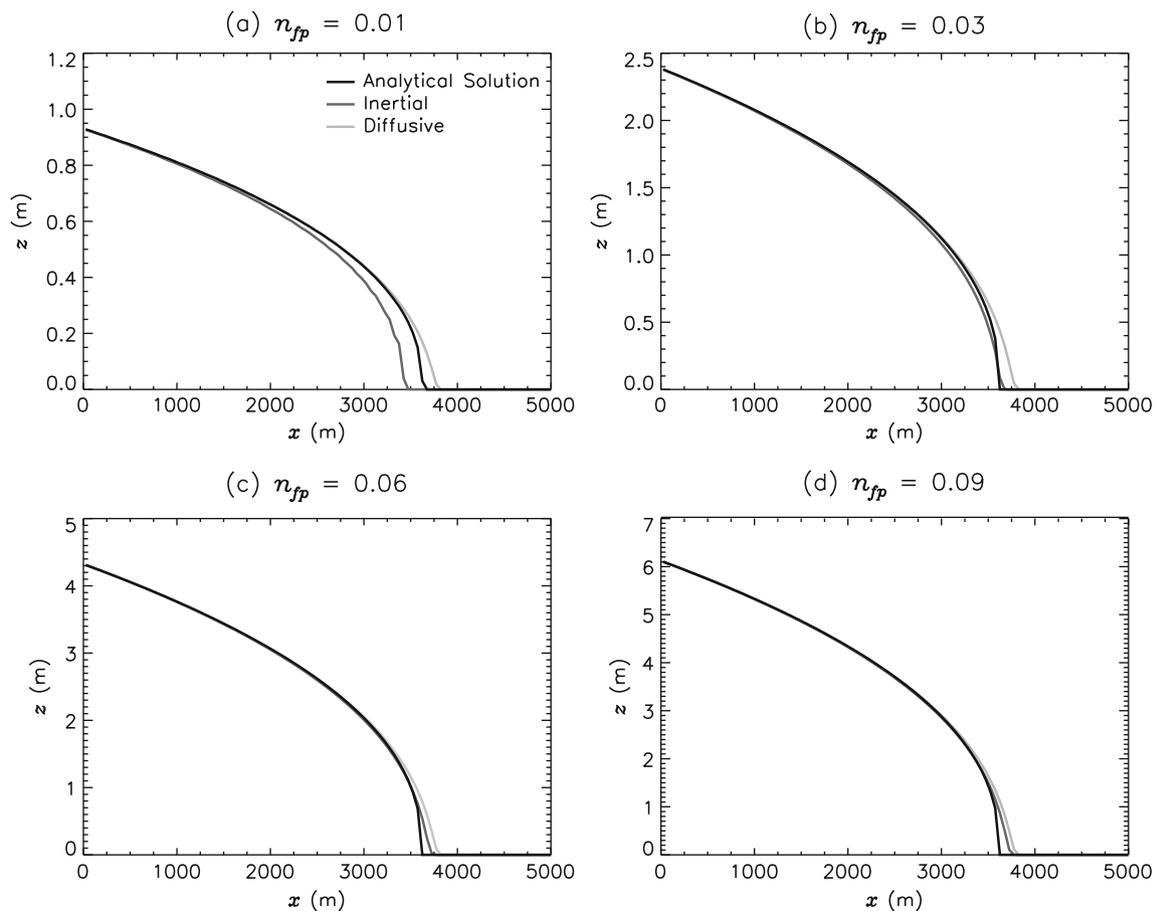


Fig. 4. Predicted water surface elevation (z) at $t = 3600$ s for wave propagation over a horizontal beach simulated at $\Delta x = 50$ m using the adaptive time step diffusive model (light grey lines) and new adaptive time step inertial model (mid grey lines) for $n = 0.01, 0.03, 0.06$ and 0.09 . In each case the model is compared to the appropriate analytical solution (solid black lines, note the changing range of the y axis).

models as a result of the Δx^2 term in Eq. (5). Fig. 3 also shows that with the diffusive model the time step continues to decrease over the simulation as depths increase and water surface slopes reduce. By contrast the reduction of time step with grid size in the inertial formulation is more linear, and after an initial period of evolution stabilizes to a near uniform value as one would expect given the velocity in the analytical solution is fixed at 1 ms^{-1} .

Lastly, sensitivity to friction was assessed by running the diffusive and inertial models for $n = 0.01, 0.03, 0.06$ and 0.09 for a model with $\Delta x = 50 \text{ m}$. Results from these simulations are presented in Fig. 4 and Table 3 and show the inertial model outperforming the diffusive scheme at frictions above $n = 0.03$, but the performance advantage switching to the diffusive scheme for $n = 0.01$. Why the behaviour should change for low values of n is unknown at this stage. One possible explanation is that for low Manning's numbers there is insufficient dissipation inherently in the numerical scheme to dissipate the energy of the flow. Hence, at $n = 0.01$ the acceleration terms in the shallow water equations start to dominate and we get wave-like behaviour, whereas inertial LISFLOOD-FP is designed for situations where there is a dominance of surface slope and friction terms. Thus for model domains dominated by very low surface friction a full shallow water model may give more accurate results. Even with this divergence from the analytical solution at $n = 0.01$, the RMSE for the inertial model is still only 0.05 m and this is probably acceptable for many applications as this is less than the vertical error in typically available terrain data (e.g. LiDAR). For both models RMSE errors increase with increasing friction, to a maximum in these tests of 0.1 m for the inertial scheme and 0.14 m for the diffusive scheme at $n = 0.09$.

Test 2: non-breaking wave run-up on a planar beach

The second test case developed by Hunter et al. (2005) consists of solving Eq. (15) for a planar beach (i.e. where $S_o \neq 0$). Here no direct analytical solution exists, but an accurate numerical solution to Eq. (15) with uniform velocity can be obtained using a 4th order Runge–Kutta scheme. Again this can be used to develop initial conditions, boundary conditions and a numerical solution against which the model can be tested. This numerical solution was implemented using parameter values of $u = 1 \text{ m s}^{-1}$, $\Delta x = 50 \text{ m}$, $S_o = 10^{-3} \text{ mm}^{-1}$ and $n = 0.03 \text{ m}^{-1/3} \text{ s}$ for a simulation of duration 3600 s . Again the models are compared in terms of Root Mean Square Error (RMSE), volume error, minimum time step during the simulation and the total computation time. These the results are summarised in Table 1. In this case the 50 m diffusive model has lower RMSE than the inertial model (0.02 m compared to 0.11 m), but at a significantly greater computational cost. The minimum time step for the diffusive model is $\sim 250\times$ smaller than that for the inertial model and this translates into a simulation time that is $\sim 61\times$ longer. The greater speed up with the inertial model here is due to the fact that in this case the minimum water surface

slope is much smaller and hence Eq. (5) leads to smaller stable time steps for the diffusive model. In fact the minimum time step achieved over the simulation with the diffusive model is $\sim 20\times$ smaller for a planar beach compared to the horizontal case at 5 m resolution. This compares to a difference of only $\sim 1.5\times$ for the inertial model at the same scale. Despite its lower accuracy, the RMSE for the inertial model is still within the typical vertical error of high resolution floodplain topographic surveys (e.g. those derived using airborne laser altimetry) and likely to be less than uncertainties induced by boundary condition errors. It should also be noted that the increase in depth for this test case is relatively rapid compared to that which would be typical for most dynamic floodplain inundation in lowland rivers. In reality in such situations flow evolves much more gradually and hence Test 2 actually represents rather a stringent case. It may therefore be that for many practical applications the minor increase in errors in predicted depth are acceptable given the large improvement in computational efficiency, and that these can, like any other model structural error, be compensated for during calibration.

In Fig. 5 we show the impact of changing model resolution on the prediction of non-breaking wave run-up. This shows predicted water surface elevations at $t = 3600 \text{ s}$ for the diffusive (dashed lines) and inertial models (dotted lines) at $\Delta x = 5, 10, 25, 50, 100$ and 200 m . Here there is no impact of resolution on the model results and the only differences are generated by the choice of model formulation. All the diffusive model runs overlay the analytical solution, apart from where $h \rightarrow 0$ where the impact of grid size can be seen. All the inertial models also over plot, but lag the numerical solution by a short distance. This result, in combination with the accuracy assessment in Table 4, suggests that the inertial models are in fact tending to a subtly different numerical solution, as noted above. The computation time for the inertial model at $\Delta x = 5 \text{ m}$ is $1120\times$ shorter than for the diffusive model as a result of the large difference in stable time step (0.493 s compared to 0.0001 s) which in turn is a consequence of the shallow water surface slope generated by this test case. Furthermore, Fig. 6 shows that the under-prediction of wave front position is related to friction and that the effect has largely disappeared when $n = 0.06$, at and above which both diffusive and inertial models perform equally well. For the simulations at $n = 0.01$ there is also a suggestion of some minor instabilities with the inertial model.

Table 3
Impact of friction on RMSE and volume error for simulations of non-breaking wave propagation over a horizontal plane with $\Delta x = 50 \text{ m}$.

Friction (Manning's n)	Model	Root Mean Square Error (RMSE, in m) from analytical solution	Volume error from analytical solution (%)
0.01	Diffusive	0.02	1.15
	Inertial	0.05	4.98
0.03	Diffusive	0.06	1.27
	Inertial	0.03	-1.25
0.06	Diffusive	0.10	1.28
	Inertial	0.06	0.00
0.09	Diffusive	0.14	1.27
	Inertial	0.10	0.31

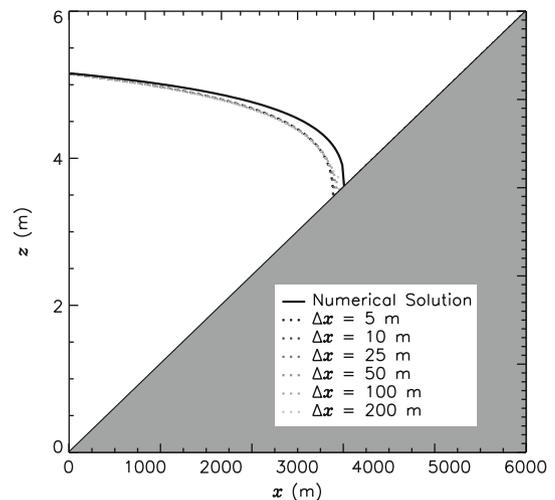


Fig. 5. Predicted water surface elevation (z) at $t = 3600 \text{ s}$ for wave propagation up a planar beach simulated at $\Delta x = 5, 10, 25, 50, 100$ and 200 m spatial resolution (denoted with dark to light grey lines respectively) and $n = 0.03$ using: (a) an adaptive time step diffusive model (dashed lines); and (b) the new adaptive time step inertial model (dotted lines). Each model is compared to the numerical solution (solid black line).

Table 4
Impact of grid resolution on RMSE and volume error for simulations of non-breaking wave run-up on a planar beach with $n = 0.03$.

Grid resolution (m)	Model	Root Mean Square Error (RMSE, in m) from analytical solution	Volume error from analytical solution (%)	Minimum time step during simulation (s)	Total computation time (min)
5	Diffusive	0.030	-0.464	0.0001	4834.8
	Inertial	0.107	-1.069	0.493	4.3
10	Diffusive	0.031	-0.467	0.001	303.5
	Inertial	0.107	-1.083	0.986	0.65
25	Diffusive	0.036	-0.536	0.004	9.12
	Inertial	0.109	-1.148	2.465	0.07
50	Diffusive	0.040	-0.553	0.018	0.73
	Inertial	0.108	-1.158	4.931	0.05
100	Diffusive	0.047	-0.590	0.072	0.1
	Inertial	0.118	-1.349	9.876	0.02
200	Diffusive	0.060	-0.674	0.296	0.05
	Inertial	0.127	-1.494	19.78	0.02

Test 3: wetting and drying of a planar beach

Tests 1 and 2 have tested the ability of the new model formulation to simulate wave propagation over flat and sloping floodplains, however correct simulation of floodplain inundation over whole events also requires the accurate representation of flow reversals and inundation front recession during floodplain drying. The ability of the model to do this can be evaluated by extending Test 2 to represent wave run-up and drying through the imple-

mentation of a sinusoidal wave boundary condition at $x = 0$ (see for example Leendertse and Gritton, 1971; Falconer and Chen, 1991). In this case we use a wave of amplitude 4 m, period 4 h, and $S_0 = 10^{-3} \text{ m m}^{-1}$ for a simulation of duration 7200 s. This gives boundary conditions comparable to those used in Test 2. As far as the authors are aware, there is no analytical solution to this problem so in this case we simply look at differences between the diffusive and inertial formulations. These are shown in Fig. 7 for a simulation using $\Delta x = 50 \text{ m}$ and $n = 0.03$. This shows smaller differences between the inertial and diffusive formulations than for Test 2. For example, the RMSD at the end of flow advance at 3600 s is only 0.074 m, compared to 0.11 m for Test 2 at the same time. Overall, Fig. 7 shows wave front position during wetting and drying simulations to differ only marginally between the two formulations, and for wave shape to show noticeable differences only at the start of flood wave recession (maximum RMSD of 0.137 m at 4500 s). Compared to Test 2 the differences between the inertial and diffusive formulations at lower frictions are also less pronounced. This is shown in Fig. 8 where we compare the output from dynamic wetting and drying simulations at $t = 1800, 3600, 5400$ and 7200 s for $n = 0.01, 0.03, 0.06$ and 0.09 for both inertial and diffusive models. Maximum differences occur at $t = 3600 \text{ s}$ and 4500 s for $n = 0.01$ and $n = 0.03$, whilst at other times and for other frictions the results are only marginally different. This suggests that the differences between diffusive and inertial formulations shown in Figs. 5 and 6 may be a worst case scenario and that differences may be less marked for more realistic cases. Fig. 8 also confirms the presence of small instabilities during the initial phase of the simulation with the inertial model when $n = 0.01$, however these die out as the simulation proceeds.

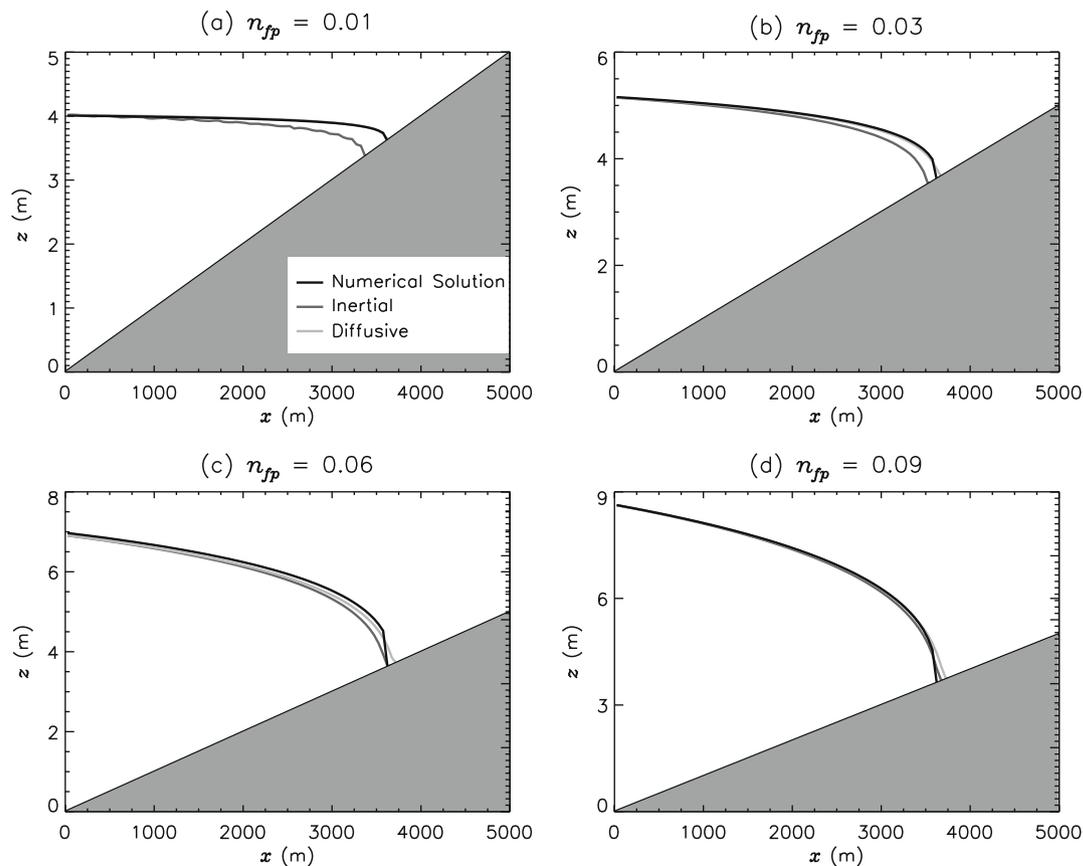


Fig. 6. Predicted water surface elevation (z) at $t = 3600 \text{ s}$ for wave propagation up a planar beach simulated at $\Delta x = 50 \text{ m}$ using the adaptive time step diffusive model (light grey lines) and new adaptive time step inertial model (mid grey lines) for $n = 0.01, 0.03, 0.06$ and 0.09 . In each case the model is compared to the appropriate numerical solution (solid black lines, note the changing range of the y axis).

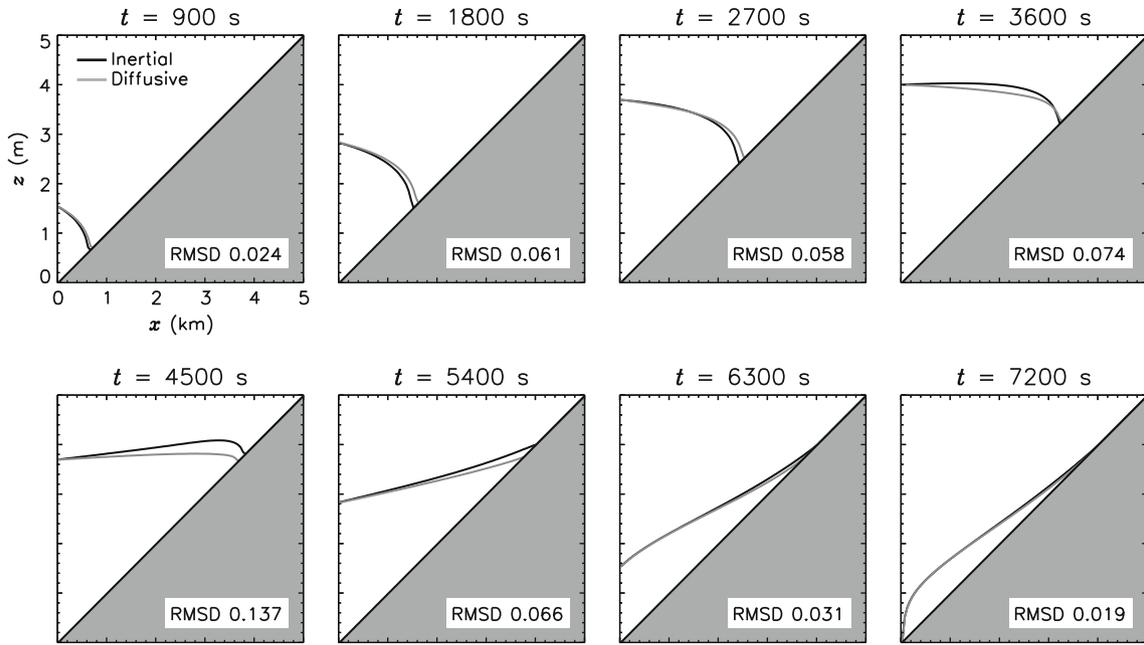


Fig. 7. Predicted water surface elevation (z) during wetting and drying of a planar beach simulated using $\Delta x = 50$ m and $n = 0.03$ for: (a) an adaptive time step diffusive model (grey lines); and (b) the new adaptive time step inertial model (black lines).

In terms of computational cost, Test 3 should be relatively expensive for the diffusive model to solve as during flow reversal and the start of wave front recession the water surface profile becomes near horizontal. Fig. 7 clearly shows this happening between 4500 and 5400 s. Flow reversals are a necessary feature of any dynamic flood simulation and cause the minimum time step in a diffusive model to become very small because of the presence of the free surface gradient terms in Eq. (5). Hence we would expect greater computational savings with the inertial model for Test 3 than for Tests 1 or 2. This is clearly shown in Table 1 where the inertial scheme results in a minimum time step $\sim 186\times$ larger than that for the diffusive model and a total simulation time that is

$\sim 60\times$ shorter. Fig. 9 shows the time step evolution for the diffusive and inertial simulations for Test 3 and highlights the fact that the minimum stable time step differs by 1–2 orders of magnitude over the majority of the simulation. Moreover, as expected the time step evolution for the inertial model is quasi-linear and after an initial period of evolution stabilizes to a near uniform value over the whole simulation.

Test 4: fine spatial resolution simulation of urban inundation

Tests 1–3 have demonstrated the numerical and computational performance of the inertial formulation in a series of idealised

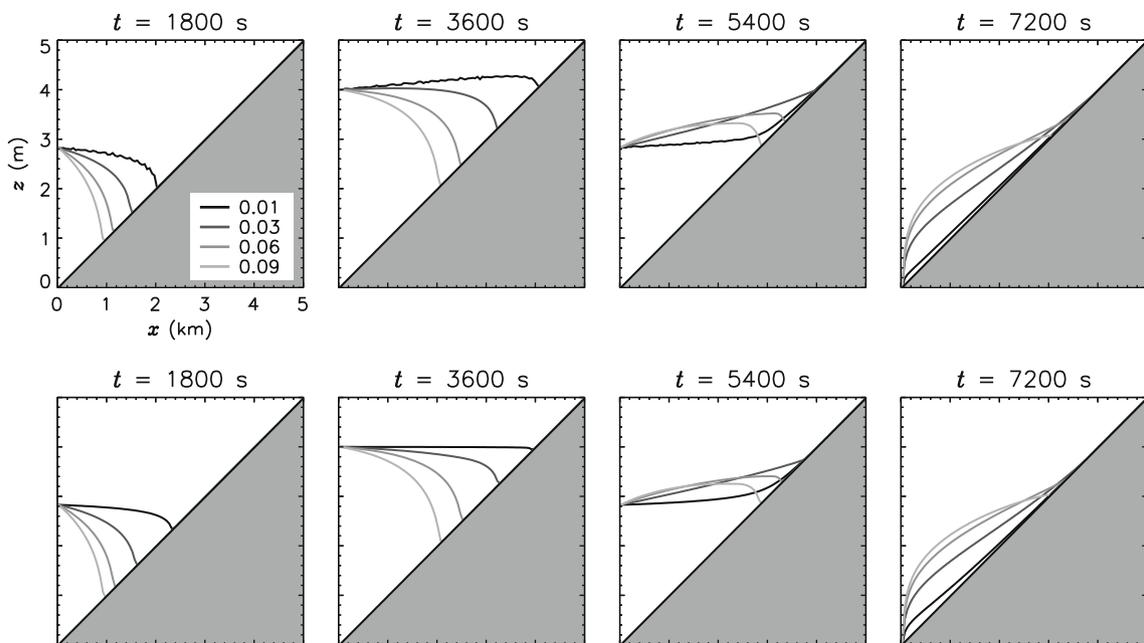


Fig. 8. Predicted water surface elevation (z) during wetting and drying of a planar beach simulated using $\Delta x = 50$ m and $n = 0.01, 0.03, 0.06$ and 0.09 (black to light grey lines respectively) for: (a) the new adaptive time step inertial model (top four panels) and (b) an adaptive time step diffusive model (bottom four panels).

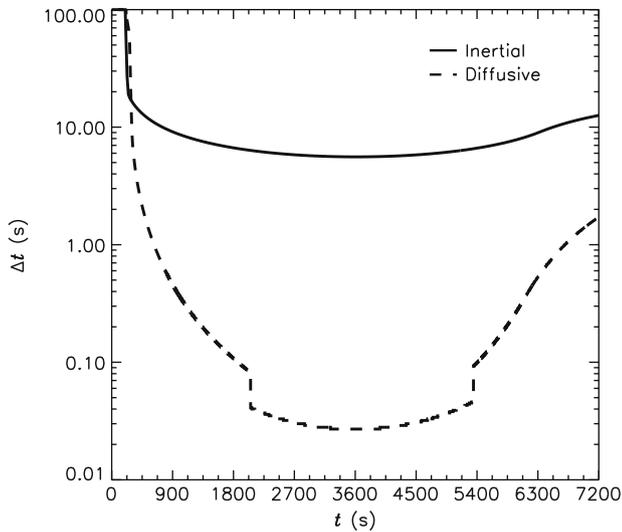


Fig. 9. Time step evolution during wetting and drying of a planar beach simulated using $\Delta x = 50$ m and $n = 0.03$ for: (a) an adaptive time step diffusive model (dashed line); and (b) the new adaptive time step inertial model (solid line).

cases of increasing complexity. However, the critical test of the model is whether it is able to simulate flood propagation over complex topography and reduce the long run times of storage cell codes when applied at fine spatial resolution. To test this we used the new inertial formulation of LISFLOOD-FP to simulate the urban inundation test case used in Hunter et al. (2008) and Fewtrell et al. (2008). This comprises a 1.0×0.4 km domain in the Greenfields area of Glasgow, UK which has been observed to flood in response to heavy rainfall in the small (~ 5 km²) upstream catchment. For this site digital elevation data at 1 m resolution were available from an airborne laser altimetry survey that can be used to build high resolution inundation models. This was supplemented by Hunter et al. (2008) with Ordnance Survey (OS) Mastermap[®] digital map data that defined building locations, the road network and land-use type as vector layers. The LiDAR data acquired for this study had already been filtered to remove vegetation and building features to leave a 'bare earth' digital elevation model (DEM) with horizontal and vertical accuracies less than 50 cm and 15 cm Root Mean Square Error (RMSE) respectively. For hydraulic modelling Hunter et al. (2008) aggregated the 'bare earth' LiDAR data to 2 m and reinserted buildings, kerbs and roads based on their locations in the digital map layer. Fig. 10 shows: (a) the road and building layout at this study site overlaid onto the surface height (z) from the benchmark DEM and (b) a high resolution aerial photo of the same area.

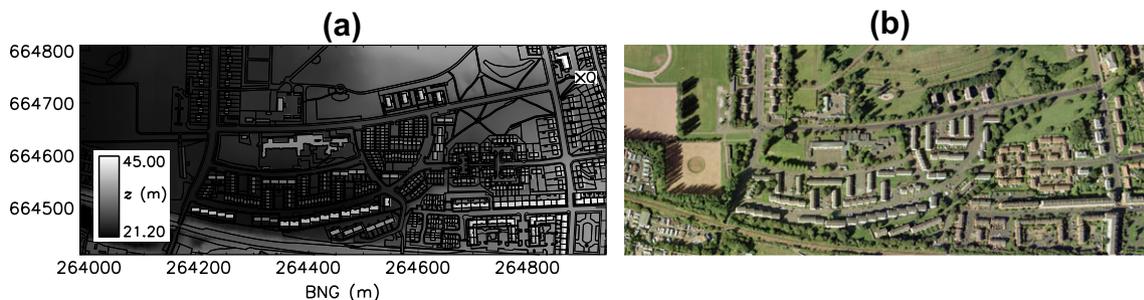


Fig. 10. The Greenfields study site (Test 4): (a) building and road topology derived from Ordnance Survey Mastermap[®] data with the surface height (z) from the benchmark DEM shown as a grey scale and (b) high resolution aerial photo of the study site. All map plots are in Cartesian coordinates where east–west is oriented along the x axis and north–south along the y axis. Dimensions are in m.

Flooding at this site is caused, at least in part, by a small (~ 1 m wide) stream that enters near the north-east corner of the domain (located at point XQ on Fig. 10a) and almost immediately enters a culvert that runs under the entire site. Flooding has been observed to occur here as a result of flow exceeding the capacity of the culvert and spilling into the street network. Once the capacity of the culvert is exceeded water flows along two main east–west oriented streets before converging and ponding in low-lying areas in the southern part of the domain.

The flow event simulated is based on a real flood that occurred at this site on 30 July 2002. The inflow boundary condition consisted of the hydrograph shown in Fig. 11, which was imposed as a point source internal to the model domain at location XQ. This hydrograph represents the water volume overflowing the culvert and lasts <60 min, but as in Hunter et al. (2008) simulations were continued for 120 min to allow water to come to rest and pond in depressions. All external boundaries for each model were closed as mass flux across the external boundary is negligible. Lastly, friction was represented as in Fewtrell et al. (2008) using a single composite value of $n = 0.035$. This was selected based on the spatial distribution of land use within the domain as determined from the OS Mastermap[®] data.

Simulations were run with these data using the adaptive time step storage cell formulation of Hunter et al. (2005) and the new inertial formulation developed in this paper. Fig. 12 shows the predicted water depths at the end of the simulation (120 min) for both formulations and a map of the absolute differences in water depth (diffusive minus inertial) at this time. Overall flood extents differ only marginally and the Root Mean Squared Difference in water

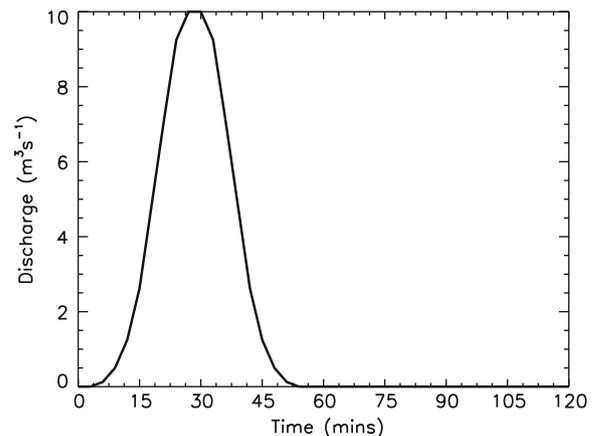


Fig. 11. Event hydrograph simulated in Test 4. The vertical dashed lines at 30 and 60 min represent instances for which model results are presented in later figures.

depth is only 0.01 m (see Table 1). Maximum differences in water depth are ± 0.1 m, but these are very localized and over the majority of the domain water depth differences are close to the RMSD. The largest difference occurs at $x = 180, y = 250$ where the flood extent is slightly greater in the diffusive version and fills up a small (~ 0.1 m deep) depression not flooded by the inertial model. Other areas of difference, such as the higher predicted depths in the inertial model at $x = 700, y = 200$ at the end of the simulation, may be a result of the additional physics in this scheme obtained by including the acceleration terms from the full shallow water equations. The run time for this test is $\sim 105\times$ shorter for the inertial model (1.47 min compared to 155 min for the diffusive model) which is in line with theoretical expectations. The total computation time for the inertial model also compares favourably with the run times reported for other classes of model applied at this test site. These include Hunter et al. (2008) who report run times of around 60 min for various structured grid, full shallow water models, Schubert et al. (2008) who report a ~ 18 min run time for a 2 m res-

olution unstructured grid, full shallow water model and Lamb et al. (2009) who report a ~ 9 min run time for a structured grid, adaptive time step diffusive model (JFLOW) run on a massively parallel Graphics Processor Unit (GPU). Whilst these are not controlled tests conducted with identical processors and compilers (for example the processors used for the simulations reported by Hunter et al. are now relatively old), these results do suggest that the new inertial formulation would be faster for this case than any previously applied code. This is a significant advance on diffusive storage cell models which were shown by Hunter et al. (2008) to be an order of magnitude slower than full shallow water models at this resolution because of the quadratic dependency on grid size in Eq. (5). Moreover, when one couples the speed up achieved here with the 5–6 \times speed up in LISFLOOD-FP run times achieved by Neal et al. (2009) using Open-MP parallelization on an 8 core processor, the potential for a $\sim 600\times$ reduction in run times for this test case (i.e. down to ~ 0.25 min) becomes a realistic expectation.

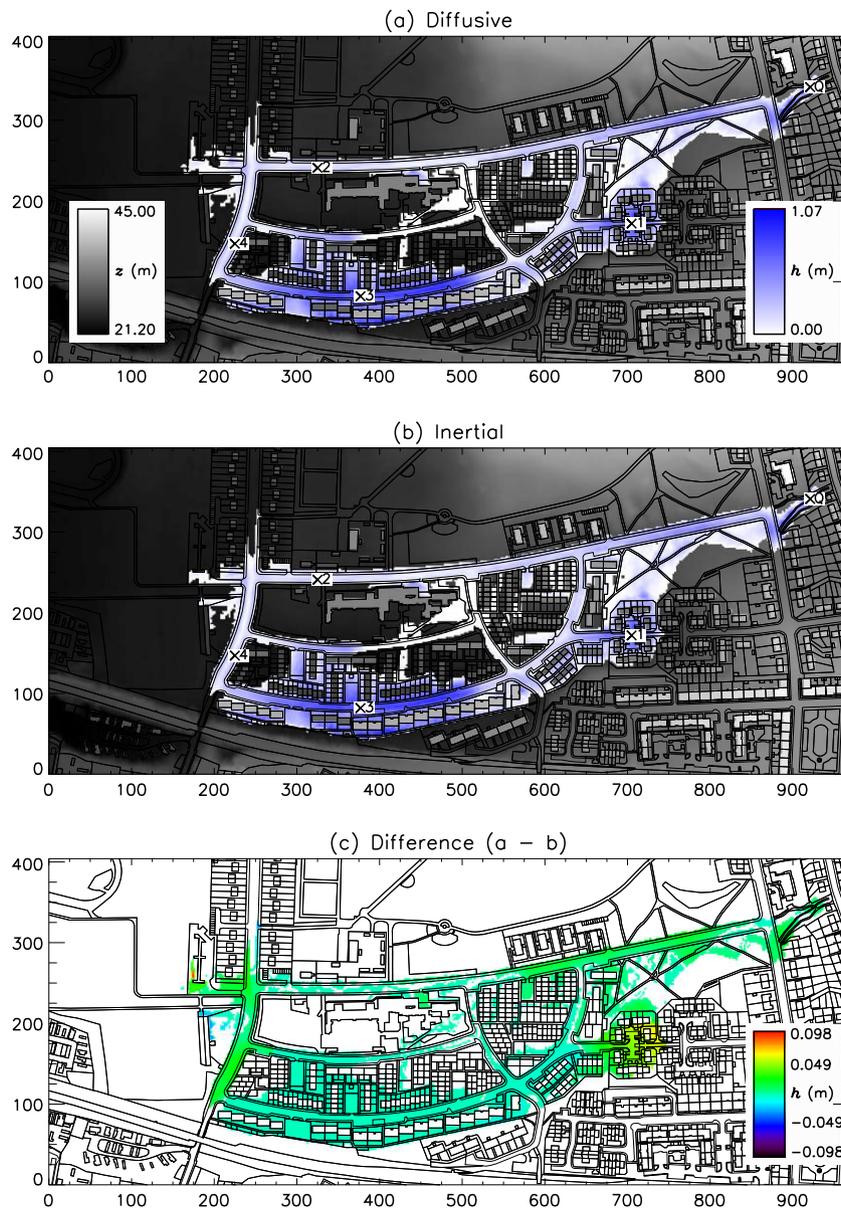


Fig. 12. Predicted water depths for at the end of the simulation for Test 4 (120 min) using a grid resolution of $\Delta x = 2$ m for: (a) an adaptive time step diffusive mode and (b) the new adaptive time step inertial model. Panel (c) shows the difference in predicted water depths (diffusive minus inertial) between (a) and (b).

Discussion and conclusions

The results outlined in the preceding section show the new inertial formulation of the shallow water equations developed in this paper to produce flow predictions that compare favourably with analytical solutions for non-breaking wave propagation over horizontal and planar beaches and to the results from an adaptive time step diffusive storage cell code. Like the diffusive model, the inertial code is relatively insensitive to grid resolution and displays an intuitively correct response to changing friction.

The inertial code performs slightly better than the diffusive code for a horizontal beach and slightly worse for a planar beach when compared to the analytical solution. However for either code the Root Mean Squared Error is always less than the typical vertical error (~ 0.1 m) in high resolution terrain data (such as airborne laser altimetry or LiDAR) used for inundation modelling. Water depth errors are independent of grid scale but do vary with friction, with the inertial model tending to perform worse than the diffusive model when $n = 0.01$. Differences between the inertial and diffusive codes become less marked for more realistic cases involving dynamic wetting and drying and for higher frictions, and for a real test case involving the fine spatial resolution simulation of flow in a topographically complex urban area the Root Mean Squared Difference is ~ 1 cm. Whilst water depths predicted by the new formulation are similar to benchmark solutions, these results are achieved at a significantly reduced computational cost because the minimum stable time step scales with Δx , rather than with $(1/\Delta x)^2$ as would be the case for a purely diffusive scheme. The exact speed up over a diffusive code will depend on grid resolution and water surface gradients within the flow domain but from theoretical considerations is likely to be 1–3 orders of magnitude. The maximum speed up achieved for the tests reported here is $1120\times$ and as a result the inertial code will most likely be faster than either diffusive or full shallow water models at any given spatial resolution. Moreover, as the explicit equation we here describe is relatively easy to code it should be simple to parallelize using Open-MP techniques (e.g. Neal et al., 2009) or software tools such as NVIDIA's Kuda which allow models to run on massively parallel GPU cards (e.g. Lamb et al., 2009). This will pave the way to further substantial reductions in run times and make possible a whole range of new applications of hydraulic models. These could include whole city risk analyses at the native spatial resolution of LiDAR data (i.e. 0.25–1 m), real time two-dimensional inundation forecasting using ensemble data assimilation, multi-year simulations of flows at continental scales (e.g. the Amazon River basin, Wilson et al., 2007), and explorations of model uncertainty using Monte Carlo analyses with orders of magnitude more realisations than has hitherto been possible.

Care should be taken when using the new inertial formulation for domains where large areas of low friction land use dominate (i.e. where n is equal to ~ 0.01), as here predicted water depth errors increase and small instabilities can creep into the solution. The instability seen at low Manning's n results from the hybrid nature of the model, and shortcomings in the stability criteria applied. The Courant–Friedrichs–Lewy condition is more usually applied to advective problems where upwinding is applied, and thus may not be strictly applicable to the centred difference approach adopted here. Hunter et al. (2005) showed that without the inertia terms, stable time step is proportional to n (Eq. (5)), and hence will be driven to zero if the friction is neglected. Friction is acting to stabilise the scheme in this respect, with stability increased further by addition of the inertia term as described in this paper. A rigorous analysis of the stability of a non-linear model including both friction and wave propagation behaviour will be complex, if not impossible, but the analysis presented here is a pragmatic approach taking into account the limiting behaviours understood by the Courant–

Friedrichs–Lewy condition and the approach of Hunter et al. (2005). This approach has been shown to be reasonable through the test cases presented in this paper, for models of typical natural floodplains. While the absence of a rigorous analysis of stability is a drawback compared to other numerical schemes, in some circumstances this is outweighed by the advantages of simplicity, which make implementation on parallel and non-standard architectures as discussed above far easier. However, for model domains characterised by very low surface friction a full shallow water model may give more accurate results.

Future research should test the new inertial formulation for further test cases and seek to benchmark model performance (in terms of both predictions and computational times) against a variety of other model types in controlled experiments such as those described by Hunter et al. (2008). Further work should also be conducted to try to improve stability of the new equation set at low friction or examine the possibility of developing models capable of using different physical formulations in different parts of the model domain depending on changing flow dynamics. A spatially varying time step may also improve model efficiency, although previous work on this for 1D models shows that the potential speed up is reduced significantly by other computational overheads associated with changing the time step for different parts of the model (Wright et al., 2003). Despite this need for ongoing research, this paper has demonstrated the utility of the new inertial formulation against a series of stringent tests of increasing complexity. The new equations are therefore likely to find widespread application in many types of flood inundation modelling and should provide a useful additional tool, alongside more established model formulations, for a variety flood risk management studies.

Acknowledgements

Timothy Fewtrell's work on this paper forms part of his Research Fellowship funded by the Willis Research Network (<http://www.willisresearchnetwork.com>). The University of Bristol Blue-Crystal high performance computer was used to undertake the simulations reported in this paper.

References

- Aronica, G., Bates, P.D., Horritt, M.S., 2002. Assessing the uncertainty in distributed model predictions using observed binary pattern information within GLUE. *Hydrological Processes* 16 (10), 2001–2016.
- Bates, P.D., De Roo, A.P.J., 2000. A simple raster-based model for flood inundation simulation. *Journal of Hydrology* 236 (1–2), 54–77.
- Bechteler, W., Hartmann, S., Otto, A.J., 1994. Coupling of 2D and 1D models and integration into Geographic Information Systems (GIS). In: White, W.R., Watts, J. (Eds.), 2nd International Conference on River Flood Hydraulics. John Wiley and Sons Ltd., Chichester, pp. 155–166.
- Cunge, J.A., Holly, F.M., Verwey, A., 1980. *Practical Aspects of Computational River Hydraulics*. Pitman Publishing, London, 420 pp.
- Estrela, T., Quintas, L., 1994. Use of a GIS in the modelling of flows on floodplains. In: White, W.R., Watts, J. (Eds.), 2nd International Conference on River Flood Hydraulics. John Wiley and Sons Ltd., Chichester, pp. 177–190.
- Falconer, R.A., Chen, Y., 1991. An improved representation of flooding and drying and wind stress effects in a 2-D tidal numerical model. *Proceedings of Institution of Civil Engineers: Research and Theory* 91 (12), 659–678.
- Fewtrell, T.J., 2009. Development of simple numerical methods for improving two-dimensional hydraulic models of urban flooding. Unpublished Ph.D. Thesis, University of Bristol.
- Fewtrell, T.J., Bates, P.D., Horritt, M., Hunter, N., 2008. Evaluating the effect of scale in flood inundation modelling in urban environments. *Hydrological Processes* 22, 5107–5118.
- Horritt, M.S., Bates, P.D., 2001. Predicting floodplain inundation: raster-based modelling versus the finite-element approach. *Hydrological Processes* 15 (5), 825–842.
- Horritt, M.S., Bates, P.D., 2002. Evaluation of 1-D and 2-D numerical models for predicting river flood inundation. *Journal of Hydrology* 268 (1–4), 87–99.
- Hunter, N.M., Horritt, M.S., Bates, P.D., Wilson, M.D., Werner, M.G.F., 2005. An adaptive time step solution for raster-based storage cell modelling of floodplain inundation. *Advances in Water Resources* 28, 975–991.

- Hunter, N.M., Bates, P.D., Horritt, M.S., Wilson, M.D., 2006. Improved simulation of flood flows using storage cell models. *Proceedings of the Institution of Civil Engineers, Water Management* 159 (1), 9–18.
- Hunter, N.M., Bates, P.D., Horritt, M.S., Wilson, M.D., 2007. Simple spatially-distributed models for predicting flood inundation: a review. *Geomorphology* 90, 208–225.
- Hunter, N.M., Bates, P.D., Neelz, S., Pender, G., Villanueva, I., Wright, N.G., Liang, D., Falconer, R.A., Lin, B., Waller, S., Crossley, A.J., Mason, D., 2008. Benchmarking 2D hydraulic models for urban flood simulations. *Proceedings of the Institution of Civil Engineers, Water Management* 161 (1), 13–30.
- Lamb, R., Crossley, A., Waller, S., 2009. A fast 2D floodplain inundation model. *Proceedings of the Institution of Civil Engineers, Water Management* 162 (1), 1–9.
- Leendertse, J.J., Gritton, E.C., 1971. *A Water Quality Simulation Model for Well-mixed Estuaries and Coastal Seas*, vol. 2. Computation Procedures. The Rand Corporation, Santa Monica, Report R-708-NYC.
- Liang, D., Falconer, R.A., Lin, B., 2006. Improved numerical modelling of estuarine flows. *Proceedings of the Institution of Civil Engineers, Maritime Engineering* 159 (1), 25–35.
- Marks, K., Bates, P.D., 2000. Integration of high-resolution topographic data with floodplain flow models. *Hydrological Processes* 14, 2109–2122.
- Neal, J.C., Fewtrell, T.J., Trigg, M.A., 2009. Parallelisation of storage cell flood models using OpenMP. *Environmental Modelling and Software* 24 (7), 872–877.
- Pappenberger, F., Beven, K.J., Hunter, N., Bates, P.D., Gouweleeuw, B., Thielen, J., De Roo, A.P.J., 2005. Cascading model uncertainty from medium range weather forecasts (10 days) through a rainfall–runoff model to flood inundation predictions within the European Flood Forecasting System (EFFS). *Hydrology and Earth System Science* 9 (4), 381–393.
- Schubert, J.E., Sanders, B.F., Smith, M.J., Wright, N.G., 2008. Unstructured mesh generation and landcover-based resistance for hydrodynamic modeling of urban flooding. *Advances in Water Resources* 31 (12), 1603–1621.
- Wilson, M.D., Bates, P.D., Alsdorf, D., Forsberg, B., Horritt, M., Melack, J., Frappart, F., Famiglietti, J., 2007. Modeling large-scale inundation of Amazonian seasonally flooded wetlands. *Geophysical Research Letters* 34 (Paper No. L15404).
- Wright, N.G., Whitlow, C., Crossley, A., 2003. Local time stepping for modeling open channel flows. *American Society of Civil Engineers, Journal of Hydraulic Engineering* 129 (6), 455–462.
- Zanobetti, D., Longère, H., Preissmann, A., Cunge, J.A., 1970. Mekong delta mathematical model program construction. *American Society of Civil Engineers, Journal of the Waterways and Harbors Division* 96 (WW2), 181–199.