HENRY
Hydraulic Engineering Repository

Ein Service der Bundesanstalt für Wasserbau

Conference Paper, Published Version

Villanueva, Ignacio; Wright, Nigel G.

Multi-Block Computation for Flood Inundation Studies

Zur Verfügung gestellt in Kooperation mit/Provided in Cooperation with:
Kuratorium für Forschung im Küsteningenieurwesen (KFKI)

Verfügbar unter/Available at: https://hdl.handle.net/20.500.11970/110081

Vorgeschlagene Zitierweise/Suggested citation:
Villanueva, Ignacio; Wright, Nigel G. (2008): Multi-Block Computation for Flood Inundation Studies. In: Wang, Sam S. Y. (Hg.): ICHE 2008. Proceedings of the 8th International Conference on Hydro-Science and Engineering, September 9-12, 2008, Nagoya, Japan. Nagoya: Nagoya Hydraulic Research Institute for River Basin Management.

Standardnutzungsbedingungen/Terms of Use:

Die Dokumente in HENRY stehen unter der Creative Commons Lizenz CC BY 4.0, sofern keine abweichenden Nutzungsbedingungen getroffen wurden. Damit ist sowohl die kommerzielle Nutzung als auch das Teilen, die Weiterbearbeitung und Speicherung erlaubt. Das Verwenden und das Bearbeiten stehen unter der Bedingung der Namensnennung. Im Einzelfall kann eine restriktivere Lizenz gelten; dann gelten abweichend von den obigen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Documents in HENRY are made available under the Creative Commons License CC BY 4.0, if no other license is applicable. Under CC BY 4.0 commercial use and sharing, remixing, transforming, and building upon the material of the work is permitted. In some cases a different, more restrictive license may apply; if applicable the terms of the restrictive license will be binding.

Multi-block Computation for Flood Inundation Studies

Ignacio Villanueva[1] and Nigel Wright[2]

[1] Research Associate, School of Civil Engineering, University of Nottingham
University Park, Nottingham, NG7 2RD, UK, e-mail: ignacio.villanueva@nottingham.ac.uk
[2] Professor of Hydraulic Engineering and River Basin Development, UNESCO-IHE Institute for Water Education,
Delft, The Netherlands, e-mail: n.wright@unesco-ihe.org

Abstract

It is expected that in the next decade a large number of flood inundation studies will be required in order to fulfil the European Flood Directive and similar statutory requirements introduced as a response to increased climate variability and rapid urbanisation. These exercises will require many simulations to be produced in a timely manner.

One way to approach this issue is to use simplified models. In some cases this involves very coarse grids for a 2D model or a 1D model with representation of the urban floodplain through a network of large storage basins (Vasquez-Lara 2008). In another approach Bates and de Roo (Bates and de Roo 2000) proposed the LISFLOOD-FP model which uses a simplification of the governing 2D shallow water equations. In a test case in the City of Glasgow the latter approach was found to not work as well or as fast as solutions based on the 2D Shallow Water Equations (Hunter et al. 2008). This has prompted the authors to investigate further the use of 2D Shallow Water equation models in urban areas. In this paper an approach that does not simplify the equations or mesh is adopted. Instead use is made of a parallel computing environment to solve the equations more efficiently in order to address the significant computational demand, particularly when resolving buildings.

The concept of parallelising an algorithm is relatively simple: the domain is divided into sub-domains which are allocated to different processors in the system. However, there are a number of details that require careful attention (Hervouet 2000; Castro et al. 2006; Pau and Sanders 2006). The code used here for parallelisation implements a robust scheme based on the finite volume discretisation of depth-averaged equations in two dimensions using the Roe approximate Riemann solver and improves it to allow faster calculations. The domain of interest is split into different blocks or domains. These are integrated explicitly in time which allows for easy over-laying between neighbours. Equations in each block are solved by one assigned processor within a computer cluster. The numerical engine uses the MPI libraries for parallelisation and works both on dedicated HPC grids and more affordable multi-core processor PCs.

Test cases in Glasgow and Carlisle in the UK are presented.

1. MODEL EQUATIONS AND SOLUTION ALGORITHM

The present study considers flooding in urban areas and models this with the hydrodynamic equations of mass and momentum conservation over a given topography and therefore using the well-known shallow water approximation. Given below are the one-dimensional and two-dimensional versions:

1D mass conservation

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = -q_l \qquad (1)$$

1D momentum conservation

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A} + gI_1\right) = gI_2 + gA(S_0 - S_f) \qquad (2)$$

where A is the wetted area, Q the discharge, I_1 and I_2, hydrostatic pressure terms, S_o the slope and S_f the friction slope where the Manning formulation is used. The term q_l represents lateral outflow of volume per unit length (m^2 s^{-1}). The hydrostatic pressure terms are given by

$$I_1 = \int_0^h (h - \eta)\sigma d\eta \qquad (3)$$

$$I_2 = \int_0^h (h - \eta)\frac{\partial \sigma}{\partial x} d\eta \qquad (4)$$

Where h is the water depth and σ is the channel width at a distance η above the bed.

2D mass conservation:

$$\frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = Q_l \qquad (5)$$

2D momentum conservation:

$$\frac{\partial hu}{\partial t} + \frac{\partial hu^2}{\partial x} + \frac{\partial huv}{\partial y} + gh\frac{\partial h}{\partial x} = gh(S_{ox} - S_{fx}) \qquad (6)$$

$$\frac{\partial hv}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial hv^2}{\partial y} + gh\frac{\partial h}{\partial y} = gh(S_{oy} - S_{fy}) \qquad (7)$$

Where h is water depth, u the x component of velocity, v the y component of velocity, and Q_l transfer of mass to a 2D cell, (discharge per unit area, m s^{-1}).

In the cases under consideration here, it is necessary to track transient features generated by irregular topography. This requires an explicit shock-capturing scheme such as that based on the approximate Riemann solver of Roe. Such methods have been reported by many authors and have been reviewed by Toro and García-Navarro (Toro and García-Navarro 2007). This approach has been adopted here. The scheme implemented solves the explicit equations using a finite volume technique for Cartesian grids that is first order in time and space which allows an easy parallelisation in 2D as explained below.

Both 1D and 2D versions of the TRENT (Total Riemann Explicit NoTtingham) code have been developed along with a combined 1D/2D model (Villanueva and Wright 2006). This has been validated in different scenarios and scales: for a rural case with a 2D grid of 18m calibrated against ASAR images (Villanueva and Wright 2006); for a urban scenario with a 2D grid of 2m calibrated against other hydrodynamic and simplified schemes (Hunter et al. 2008) and for a massive inundation caused by a quaternary ice dam collapse involving a

2D grid of 50m and extreme peaks of 10 million cubic meters per second, calibrated with sediment bars (Carling et al. 2008).

2. ARCHITECTURE OF THE MODEL

Implementing a fluid dynamics solver in parallel can be done by parallelising the matrix operations or it may be done by breaking the domain into sub-domains. In this work the latter is adopted as it is readily done in a 2D domain. The domain is divided into sub-domains which are allocated to different processors in the system. However, there are a number of details that require careful attention, see the work of (Hervouet, 2000) for the finite element method formulation of the shallow water equations, and the works (Pau and Sanders, 2006) and (Castro et al. 2006) for explicit shock-capturing finite volume schemes based on the approximate Riemann solver of Roe. It should be noted that these three were tested in non-urban cases.

2.1 Regular domain division

The initial domain is divided into regular, adjacent sub-domains. These clearly have interdependent solutions and thus communication between the sub-domains is necessary. This is implemented through the use of halo cells: each sub-domain has cells around its edges that duplicate cells in the adjacent sub-domains. At every timestep values in these cells are communicated to the processors working on the adjacent sub-domains so that they can use the data in their calculations. This process must be carefully implemented as bottlenecks in communication can affect parallel efficiency significantly.

With a regular Cartesian domain decomposition halo cells are readily set up and standard *MPI* (Message Passage Interface) libraries are available (e.g *MPICH2* at http://www.mcs.anl.gov/research/projects/mpich2) which provide dedicated communication subroutines and synchronisation calls (MacDonald et al. 2005).

2.3 HPC versus specialised desktops

High Performance Computing installations are expensive to maintain and require dedicated system administration that is not affordable for small research groups or consultancy groups. On the other hand the GNU *gcc* C compiler (http://gcc.gnu.org) and *MPICH2* libraries are easy to implement on GNU *Linux* SMP (symmetrical multi-processor) platforms which nowadays allow for several processors in a single, cheap box. In the authors' experience using more affordable desktop machines based on dual-core processors is advantageous. Both hardware approaches are demonstrated here and comparisons made later in the paper.

2.4 Time step and advanced multi-block domain division

The main restriction of the explicit time discretisation is the limit on the time step Δt governed by the Courant-Friedrichs-Lewy criteria for a grid resolution of Δx:

$$\Delta t < \min\left(\frac{\Delta x}{|u| + c}, \frac{\Delta x}{|v| + c}\right) \qquad (8)$$

Where c is the local celerity at every wet cell defined as: $c = \sqrt{gh}$ (9)

2.5 General Performance of the model

Here we define the parallel efficiency by the equation:

$$Efficiency = \frac{T_1}{T_2} \times \frac{N_1}{N_2} \times \left(\frac{\Delta x_1}{\Delta x_2}\right)^3$$

Where T is the time taken, N is the number of processors used and Δx is the grid size for each of two model runs denoted by *1* and *2 – 2* is taken to be the finer resolution. The ratio $\Delta x_1 / \Delta x_2$ is raised to the power of three to account for the different number of cells and the time step which is proportional to Δx (Equation 8). Optimal efficiency gives a value of 1, but a value less than this is usually found due to messaging overheads and algorithmic factors.

The sub-domains must be carefully defined in order to ensure a balance of the computation model across processors (the load-balancing problem). With a flood inundation code this is just as straightforward as ensuring the same number of cells for each sub-domain as a predominantly dry sub-domain will require much less computation than for a completely wet sub-domain even if they have the same number of cells. Efficiency therefore decreases as the number of processors increases due to the effect of a higher number of dry processors which are difficult to synchronise with the rest of wet neighbour processors. Tables of efficiency are shown later.

3. PRACTICAL PROBLEMS

One practical advantage of the algorithm described here is that it allows the use of the very fine grids necessary for the complete 2D modelling of floodplains and channels. This means that models can be constructed without topological 1D channels or branches as long as the bathymetry data is accurate enough to resolve embankments and main channels. However, in other research in the particular case of Carlisle (described later), initial tests by the authors (Neal et al. 2008) using multi-block grids of 1-5 m resolution indicates that use of a hybrid 1D-2D model is still required. Further practical problems related to hybrid-modelling or coupling 1D-2D models with topography and resolution scale are cited in other work (Villanueva 2007; Vojinovic and Tutulic 2008; Wright et al. 2008).

4. CALIBRATION WITH OTHER NUMERICAL MODELS

The first scenario for validation is an area of the City of Glasgow which has flooded in the recent past. The modelled domain is 1.0km by 0.4km, with dense urban development on either side of two main streets and a network of minor roads. A 1m resolution airborne laser altimetry (LiDAR) data was fused with Ordnance Survey (OS) Mastermap® digital map data which gave building locations, the road network and land use. Figure 1 shows the road and building layout at this study site.

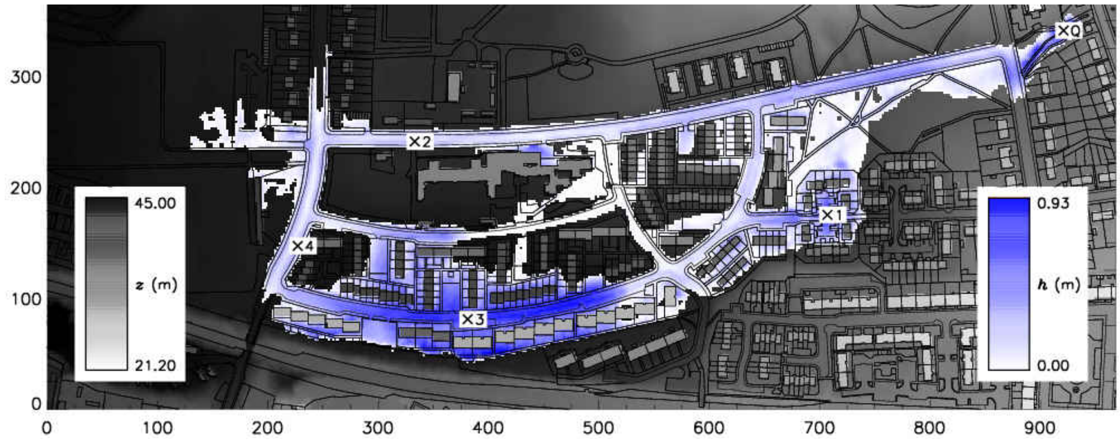

Figure 1: Urban scenario in Glasgow, DTM shown as a grey scale overlain by water depths (h) from a typical model simulation as in Hunter et al., 2008.

Flooding at this site is caused by a stream in the northeast corner of the domain which can exceed the capacity of a culvert in this location. This stream is prone to flash floods. The test case is described more fully elsewhere (Hunter et al. 2008), and in that paper the authors observe that the domain complexity and the high velocity, low depth flow on steep slopes and depression ponding is typical of many urban flooding situations.

The flow event simulated in this analysis is based on a real flood which occurred at this site on the 30th July 2002. The flood event lasts less than 60 minutes, but the simulation was continued for 120 minutes to allow water to come to rest and pond in depressions. Different models with particular assumptions are compared showing a complete battery of graphs (Hunter et al. 2008). The proposed model TRENT was in good agreement, particularly considering the computational cost using a single processor, which is usually considered the main restriction while using explicit shock-capturing schemes.

An interesting validation more related to the interest of this paper was the comparison of the results using 2 different DTM resolutions: a $2m \times 2m$ single processor and a $0.25m \times 0.25m$ multi-processor one. The computational cost time of the latter is theoretically 512 times the cost of the former, which is reduced using the regular domain division with multi-processors, but with a decreasing efficiency as the number of processors increases as can be seen in Table 1. Figure 2 shows a comparison of water elevations at the gauge number 3.

From Figure 2, it can be seen that there are slight differences in arrival times and peak values (less than 5 cm) due to the differences of slopes and fine details like kerbs and other overtopping heights that each DTM considers particularly.

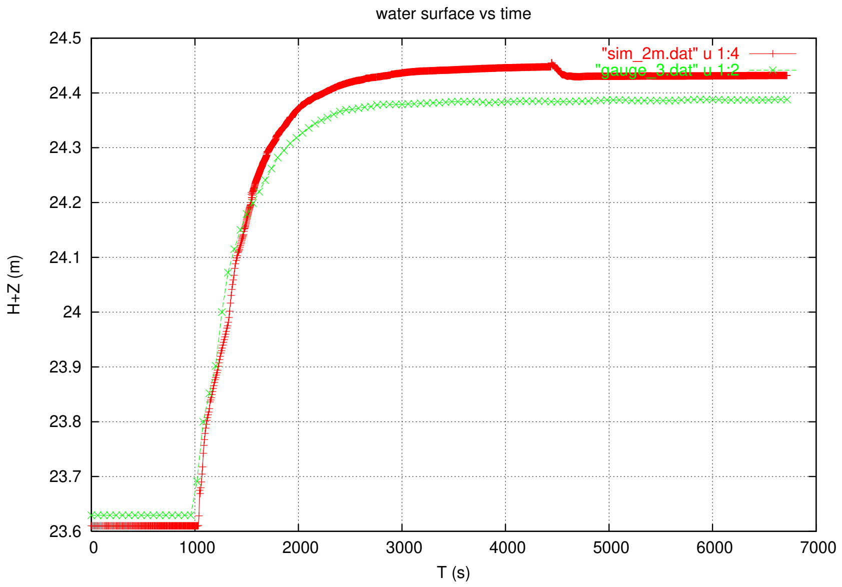

Figure 2: Computed water stage elevation against time using TRENT, for two different resolutions at gauge number 3, red colour is for the 2m resolution and green colour for the 0.25m.

Table 1: Glasgow computational costs for a 2-hour long hydrograph, tested with a 2 m resolution grid against 6, 12, 18 processors at 0.25m resolution grids, the efficiency decreases with the number of processors. All the processors were at 2.2 GHz speed in a dedicated cluster.

Grid size	Processors	Computational cost (HPC)	Efficiency against 1 processor
2m (98 K cells)	1	1 h 37 m	
0.25m (6.3 M cells)	6	198 h 51 m	0.69
	12	141 h 9m	0.49
	18	112 h 42 m	0.41

5. CALIBRATION WITH REAL DATA: CARLISLE

The city of Carlisle in the Northwest of England was flooded in an extreme event in January 2005. The extent of the domain is shown in Figure 3. This encompasses a number of different land use types: both urban and rural. Additionally there are three separate tributaries with the Rivers Peterill (bottom right) and Caldew (bottom left) flowing into the River Eden (top one, flowing from right to left).

A detailed description of the scenario and calibration data has been first exposed in Neal et al., 2008. As a brief summary an airborne laser altimetry (LiDAR) survey undertaken by the Environment Agency of England and Wales (EA) in March 2002, and updated along the River Caldew in November 2005, was used to provide a 1m resolution digital surface model (DSM) of the study site. For the model simulations run here the DEM was re-sampled to 10m and 5m resolution raster grids. The differential GPS survey conducted soon after the flooding event by the University of Bristol is very extensive both in terms of the number of data points and spatial coverage for an urban inundation event of this magnitude, which provides many validation points and tools.

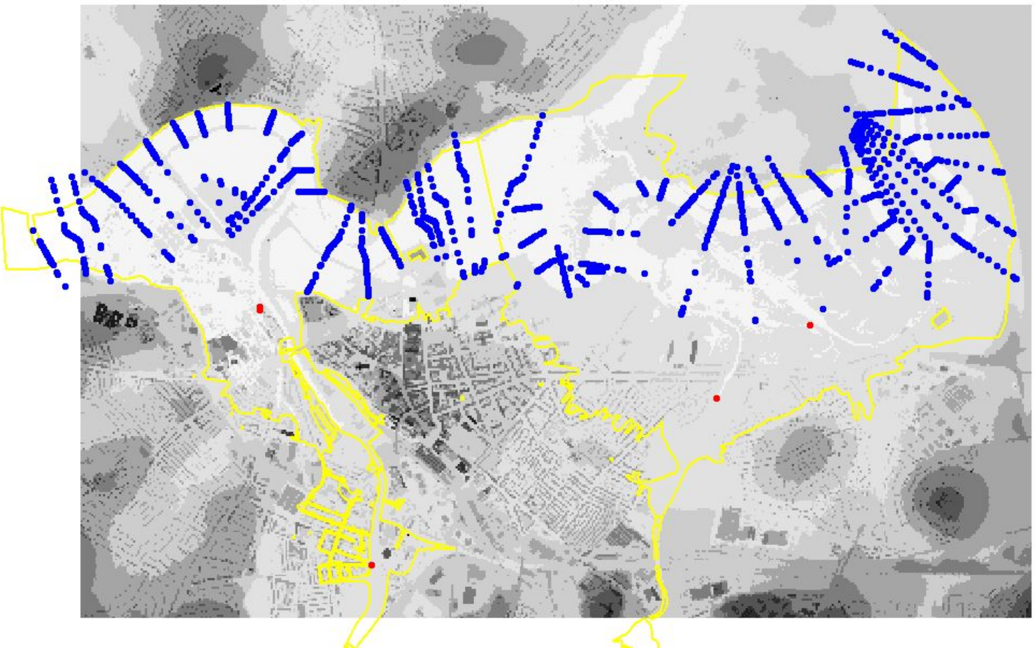

Figure 3: Carlisle DTM in grey with the maximum flooding extent outline (yellow) and available 1D cross sections for the main River Eden (blue points), monitoring gauges are displayed in red. The total area covered is approximately 5×3 km.

The most accurate and feasible computed maximum flooding extent with TRENT 2D is displayed in Figure 4, for a $10m \times 10m$ grid size and uniform Manning coefficient of 0.015. There is a clear over estimation in the Caldew tributary (bottom left) which it is believed would be improved if a one-dimensional model with a proper bathymetry were coupled to the flood plain, as finer 2D results obtained with the original 1m grid did not reduce the flooding extent to the observed outline.

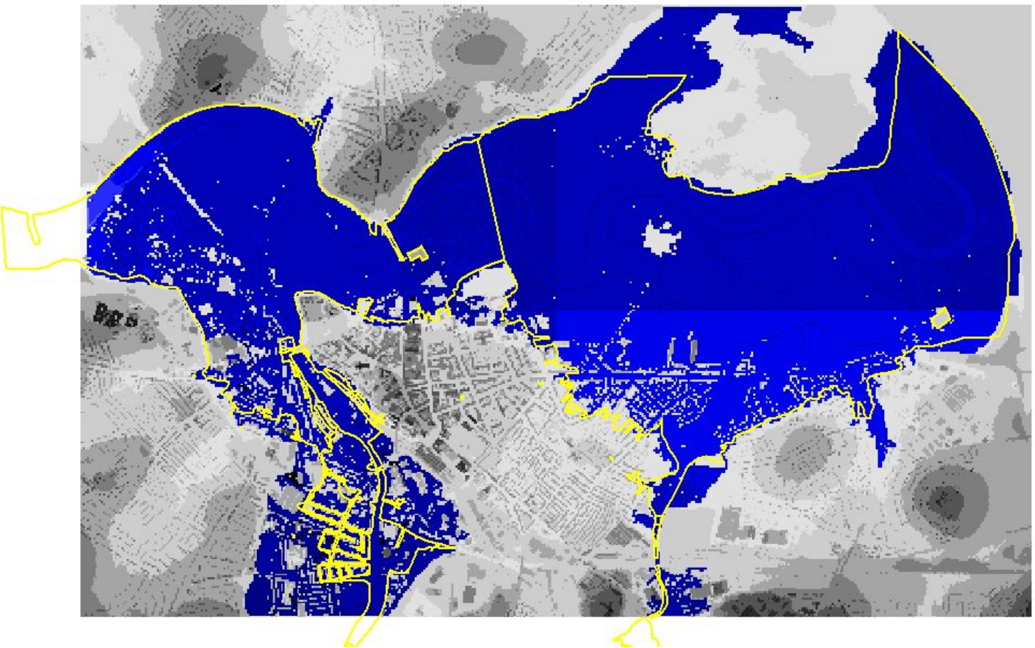

Figure 4: Maximum flooding extent map calculated by TRENT for the 4 processors simulation at $10m \times 10m$ grid (blue) against observed outline (yellow).

Table 2: Computation times for the Carlisle test case at 5m resolution.

Grid size	Number of cells	Number of processors and speed	CPU time	Efficiency
10m	24,000	4, 3.2GHz (desktop)	38h 46m	1
5m	960,000	6, 3.2 GHz (desktop)	493h 21m	0.38
		24, 2.2 GHz (HPC)	181h 55m	0.37

As many water marks are available for calibration, histograms of errors between measurements and simulations can be plotted, see Figure 5 which shows a standard over-estimation (a more detailed analysis is offered in Neal et al. (Neal et al. 2008)). It is expected that the fit would be improved by coupling 1D channels and 2D flood plains by careful linking through embankment lines.

The duration of the computation for Figures 4 and 5 was 38 hours and 46 minutes with 4 processors at 3.2 GHz. The resulting efficiencies for simulations at finer resolution are given in Table 2 and are slightly lower than those found in the Glasgow test case. These are lower than might be desired due to the extensive wetting and drying in the system. It is interesting to note that the HPC offers to significant efficiency gain over the desktop system.

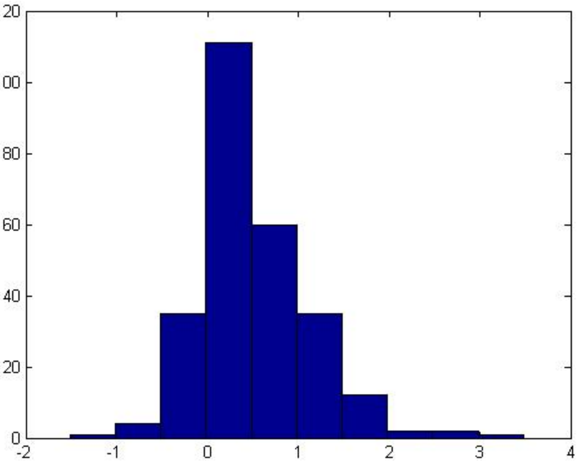

Figure 5: Histogram of errors (vertical axis) between simulations and measurements at $10m \times 10m$ for TRENT in f (horizontal axis).

6. CONCLUSIONS

A parallel version of a shallow water equation code based on the Godunov method has been implemented and successfully tested. Good efficiency has been obtained inspite of the challenges for load-balancing presented by the wetting and drying phenomenon. It has also been seen that using a standard PC with dual-core processors can deliver efficient results even in comparison to more powerful cluster-based machines.

Further work will focus on more detailed validation and comparison with other methods and tools for describing practical rules for selection of feasible mesh sizes and coupling of models, depending on the available number of processors, expected accuracy for prediction in urban areas and uncertainty associated to validation data or DTM data processing. Additional work with sub-domains of different resolution and possibly local timestepping (Crossley and Wright 2005) is underway.

ACKNOWLEDGMENTS

The research reported in this paper was conducted as part of the Flood Risk Management Research Consortium. The FRMRC is supported by grant GR/S76304 from the Engineering and Physical Sciences Research Council, in partnership with the Natural Environment Research Council, the DEFRA/EA Joint Research Programme on Flood and Coastal Defence, the Scottish Executive and the Rivers Agency (Northern Ireland). The data collection, processing of the DTM and results for Carlisle event were provided by the University of Bristol Hydrology group supervised by Prof. Paul Bates. Special thanks are given to Dr. Jeff Neal of the University of Bristol for useful discussions and assistance with the graphical output.

REFERENCES

Bates, P. D. and de Roo, A. P. J. (2000). "A simple raster-based model for floodplain inundation." Journal of Hydrology **236**(1-2): 54-77.

Carling, P., Villanueva, I., Herget, J., Wright, N., Borodavko, P. and Morvan, H. (2008). "Unsteady 1-D and 2-D hydraulic models with ice-dam break for Quaternary megaflood, Altai Mountains, southern Siberia." Global and Planetary Change **accepted for publication**.

Castro, J. M., García-Rodríguez, J. A., González-Vida, J. M. and Parés, C. (2006). "A parallel 2D finite volume scheme for solving systems of balance laws with nonconservative products: Application to shallow flows." Compt. Methods Appl. Mech. Engrg, **195**: 2788-2815.

Crossley, A. J. and Wright, N. G. (2005). "Time accurate local time stepping for the unsteady shallow water equations." International Journal For Numerical Methods In Fluids **48**(7): 775-799.

Hervouet, J. M. (2000). "A high resolution 2-D dam-break model using parallelization." Journal of Hydrological Process **14**: 2211-2230.

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. and Mason, D. (2008). "Benchmarking 2D hydraulic models for urban flood simulations." Proceedings of the Institution of Civil Engineers: Water Management **161**(1): 13-30.

MacDonald, N., Minty, E., Malard, J., Harding, T., Brown, S., Antonioletti, M. and Henty, D. (2005). Writing Message-Passing Parallel Programs with MPI: introductory course, Edinburgh Parallel Computing Centre.

Neal, J. C., Bates, P. D., Fewtrell, T. J., Wright, N. G., Villanueva, I., Hunter, N. M. and Horritt, M. S. (2008). Modelling the 2005 Carlisle flood event using LISFLOOD-FP and TRENT. Flood Risk 2008, Oxford, UK.

Pau, J. C. and Sanders, B. F. (2006). "Performance of parallel implementations of an explicit finite-volume shallow-water model." Journal of Computing in Civil Engineering **20**(2): 99-110.

Toro, E. F. and García-Navarro, P. (2007). "Godunov-type methods for free-surface shallow flows: a review." Journal of Hydraulic Research **45**(6): 736-751.

Vasquez-Lara, C. (2008). Risk Based Flood Model Selection: A case study of Carlisle, UK. Department of Water Engineering. Delft, UNESCO-IHE Institute for Water Education. **MSc**.

Villanueva, I. (2007). Modelling and real time flash flood forecasting in a Mediterranean basin. International Workshop on Numerical Modelling of Hydrodynamics for Water Resources, Zaragoza, Spain, Taylor and Francis.

Villanueva, I. and Wright, N. G. (2006). "Linking Riemann and storage cell models for flood prediction." ICE Journal of Water Management **159**(1): 27-33.

Villanueva, I. and Wright, N. G. (2006). Performance of several hybrid numerical schemes to determine flooding extent. River Flow 2006, Lisbon, Portugal.

Vojinovic, Z. and Tutulic, D. (2008). "On the use of 1D and coupled 1D-2D modelling approaches for assessment of floods in urban areas." Journal of Hydrology **Accepted**.

Wright, N. G., Villanueva, I., Bates, P. D., Mason, D. C., Wilson, M. D., Pender, G. and Neelz, S. (2008). "A Case Study of the Use of Remotely-sensed Data for Modelling Flood Inundation on the River Severn, UK,." Journal of Hydraulic Engineering **134**(5): 533-540.