Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 154 (2016) 801 – 810

Procedia Engineering

www.elsevier.com/locate/procedia

12th International Conference on Hydroinformatics, HIC 2016

Accuracy and computational efficiency of 2D urban surface flood modelling based on cellular automata

Michael J Gibson[a,b,*], Dragan A Savic[a], Slobodan Djordjevic[a], Albert S Chen[a], Stuart Fraser[b], Tim Watson[b]

[a]*Centre for Water Systems, University of Exeter, Harrison Building, North Park Road, Exeter, EX4 4QF, United Kingdom*
[b]*ICS Consulting Ltd., Peartree House, Main Street, Little Smeaton, WF8 3LG, United Kingdom*

Abstract

There is an emerging abundance of freely available high resolution (one meter or less) LIDAR data due to the advent of remote sensing, which enables wider applications of detailed flood risk modelling and analysis. Digital terrain surface data often comes in raster form, i.e., a square regular grid, and often requires conversion into a specific computational mesh for two-dimensional (2D) flood modelling that adopts triangular irregular meshes. 2D modelling of flood water movement through urban areas requires resolution of complex flow paths around buildings, which requires both high accuracy and computational efficiency. Water distribution and wastewater systems in the UK contain over 700,000 km of water distribution and sewer pipes, which represents a large risk exposure from flooding caused by sewer surcharging or distribution pipe breaks. This makes it important for utilities to understand and predict where clean or dirty water flows will be directed when they leave the system. In order to establish risk assessment many thousands of simulations may be required, calling for the most computational efficient models possible.

Cellular Automata (CA) represents a method of running simulations based on a regular square grid, thus saving set-up time of configuring the terrain data into an irregular triangular mesh. It also offers a more uniform memory pattern for very fast modern, highly parallel hardware, such as general purpose graphical processing units (GPGPU). In this paper the performance of the CADDIES [1], a CA platform and associate flood modelling software caFloodPro, using a square regular grid and Von Neumann neighbourhood, is compared to industry standard software using triangular irregular meshes for similar resolutions. A minimum time step is used to control the computational complexity of the algorithm, which then creates a trade-off between the processing speeds of simulations and the accuracy resulting from the limitations used within the local rule to cope with relatively large time steps. This study shows that using CA based methods on regular square grids offers process speed increases in terms of 5-20 times over that of the industry standard software using irregular triangular meshes, while maintaining 98-99% flooding extent accuracy.

* Corresponding author. Tel.: +44-1392-724075.
E-mail address: M.J.Gibson@exeter.ac.uk

Peer-review under responsibility of the organizing committee of HIC 2016
doi:10.1016/j.proeng.2016.07.409

Peer-review under responsibility of the organizing committee of HIC 2016

Keywords: LIDAR; Irregular triangular mesh; Regular square mesh; GPGPU; CPU; Parallelism; Cell size, Time step, Cellular Automaton, Trade-off.

1. Introduction

Recent legislation in the UK puts the emphasis on the understanding of risks from water infrastructure, and places the responsibility upon the Water Utility companies. It also gives the Environment Agency powers to levy large fines, so it is important for Water Utility companies to understand the risks stemming from their infrastructure and assets. So full coverage is needed to understand the relative consequence of failure. As computing power has increased during recent years, modelling of water distribution (pipe bursts) and wastewater system (collapses and blockages) failures has moved away from one dimensional (1D) flood routing, towards fully dynamic two-dimensional flood paths, and even linked 1D-2D models, which are more computationally complex. The increased availability of high resolution (one meter or less) LIDAR data further facilitates 2D modelling. There are a number of different software commonly used for 2D surface flow modelling. A commercial package InfoWorks ICM by Innovyze [2], which uses an irregular triangular mesh, is capable of linking 1D sewer flows with 2D surface flow (1D-2D linkage). MIKE FLOOD by DHI software [3], which uses regular grids, is capable of multigrid resolutions modelling [4]. Many models either solve the full Shallow Water Equations (SWE) or a reduced complexity version where some of the terms are neglected in order to achieve higher processing speeds while maintain reasonable high accuracy in given situations. This complexity reduction is normally achieved through approximating the less significant terms of the equation [5], such as inertia, which in many urban circumstance can be neglected where frictional forces are largely dominant. The JFLOW model [6], Urban Inundation Model (UIM) [7], and the diffusive version of LISPFLOOD-FP [8] solve the 2D diffusive wave equations that neglect the inertial (local acceleration) and advection (convective acceleration) terms. The inertial version of LISPFLOOD-FP [9] solves the SWEs without the advection term. In all the above models excluding InfoWorks, the flow is decoupled in the Cartesian x, y directions between each pair of cells, whereas InfoWorks calculates flows between each irregular cell/processing element.

CADDIES (standing for: Cellular Automata Dual DraInagE System) is a framework, designed with an Application Programming Interface (API) to facilitate the programming of local state transition rules, which can then deploy the same Cellular Automata (CA) rules to multiple hardware platforms and neighbourhood or grid types [1]. Currently the CADDIES framework can be deployed on a regular square grid using the Von Neumann neighbourhood (although work is underway to extend this to include hexagonal grids). This can be deployed on serial CPU, parallel multi-core CPU (with OpenMP [10]), and parallel many-core GPGPU (using OpenCL [11]). The CADDIES framework also includes an open source application caFlood [12]. The framework also comes in a more advanced commercial and research limited version caFloodPro, which is capable of simulating spatially variable rainfall, roughness values, and infiltration, along with being having improved accuracy and speed.

Another implementation of a CA based storage cell flood model, which is similar to LISPFLOOD-FP, has been developed by Dottori and Todini [12]. It employs the Manning's equation in order to calculate local intercellular discharges (fluxes). The Manning's equation calculates the intercellular flow rate (Q), shown in Equation (1) below, assuming the $(i-1, j)$ cell's water stage/level (h) is less than of cell (i, j) as only outflows are calculated. Equation (1) is shown only in the x direction but is calculated for between the main cell and each neighbour in the Von Neumann neighbourhood. The hydraulic radius $\left(h_{flow}\right)$ is often interpreted as the water depth in the current/main cell, and (n) is the Manning's roughness coefficient, and all outflows in the neighbourhood must not exceed the total volume present. In Equation (2), the new water level/depth is calculated for each cell by removing outflows from this cell and adding the inflows from neighbouring cells (multiplied by the time step). Again time steps must not be large enough to move more water than is present, in order to conserve mass throughout the grid.

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

$$h_{i,j}^{t+\Delta t} = h^t + \left(\sum\left(Q_{i-1,j}, Q_{i+1,j}, Q_{i,j-1}, Q_{i,j+1}\right) \times \Delta t\right) \quad (2)$$

Ghimire et al. [13] developed the first version of CADDIES flooding rule, which used a ranking system between the neighbourhood of local cells to determine intercellular fluxes. The model still used Manning's equation to limit the fluxes for each intercellular connection, but did achieve high performance thought massive parallelism of the GPGPU [1] [13]. The Weighted model is an advancement upon this initial local flooding rule, which uses a simple weighting system based on the water level difference between the local neighbourhood. The new 'weighted' method calculates the Manning's formula only once for each cell and distributes the flow to neighbouring cell using a simple weighting system, as opposed to calculating the Manning's equation for each neighbour, thus reducing computation per cell [14]. Work by Hunter et al. [5] [8] and Bates et al. [9] shows how similar storage cell transition rules have been developed, and limiting factors added to the local rule to attempt to reduce the likelihood of 'checkered board' style oscillations, which can destroy the quality of the overall simulation. The oscillations are caused by relatively large time steps, or very small water depths, or low hydraulic gradients in the local neighbourhood. This means that if the time step for the entire gird is too large in just one cell at any one iteration, then these destructively large flows can lead to the oscillations that spread across the grid. Early methods like UIM checked that the time step is not too large for each iteration, otherwise reducing it and re-running that iteration. If no cell in the grid exceeds the maximum time step, then UIM can increase the time step. However, due to the none-linear nature of the Manning's formula the entire iteration/grid needs re-calculation.

As the primary source of computational complexity in such a cellular system is the number of cells, multiplied by the number of iterations [15]. Such systems as UIM have been found to be very computationally intensive as they re-calculate the entire grid of flow rates multiple times for each time step. Therefore, Hunter et al. [8] put forward the idea of an adaptive time step, which would calculate the flow rates for the entire grid and then calculate largest possible time step within the limitations of the rule and multiply all flow rates by this amount. While such an adaptive time system should automatically choose a time step and use the largest possible, it is by no means a panacea, as very large amounts of processing time can be spent on a proportionally short period of the simulation. The adaptive time step is often bounded in model configuration by a minimum/maximum time step and an additional scaling factor (alpha) in order to limit computational time spent on very short periods of simulation time. Constraining the minimum time step means that the local spatial rule is forced to deal with time steps outside of its normal operations range for the given flow rates. Due to the need for the local rule to deal with larger time steps, flow limiters have been explicitly coded into the local spatial rules. Furthermore, the fact that calculating the time step for every iteration is very computationally expressive, the CADDIES system allows for the specification of an update time, after which a new time step is calculated. With static time steps it was previously necessary just to find a maximum time step at which a rule would operate. However, with adaptive time step this situation is further complicated as it is now necessary to find the largest minimum time step, scaling factor and time step update rate at which local spatial a rule will operate successfully. Due to the problem of excessively large flows in the local neighbourhood, Hunter & Bates use Equation (3) to limit the flow rates relative to the current time step.

$$Q_x^{i,j} = min\left(Q_x^{i,j}, \frac{\Delta x \Delta y\left(h^{i-1,j}-h^{i,j}\right)}{4\Delta t}\right) \quad (3)$$

The weighted model in CADDIES assigns a weight to the central/main cell equal to that of the smallest outflow in order to leave some water in the main cell and cause more gradual drain from each cell. The computational complexity of a simulation is based on the complexity of the local rule, the number of cells and number of iterations of the CA algorithm. Therefore, the complexity of the local spatial rules can be completely dominated by a rule which can operate successfully at higher time steps. Rather, it is best to determine which rule is fastest by being able to run at the highest time step while maintaining accuracy. While flow limitations can serve to lower the chance of destructive oscillations, it also tends to lower the physical realism of the simulation. The comparison of rules' performance should therefore be in terms of their processing speed and the accuracy of the results. In other words, rules should be compared in terms of the trade-off between speed and accuracy, controlled by the interaction of both the temporal rules and constraint settings, and the local spatial rule.

2. Method

An area in Sheffield (United Kingdom) is chosen for the testing. The initial DSM (Digital Surface Model) from the LIDAR data is processed in order to remove items from the data such as trees, cars, buses etc. However, this smoothing process also removes all the building and other structures, which need to be brought back into the DEM (Digital Elevation Model) using the building polygon data from Ordnance Survey Master Maps [17]. For the comparison of two different models the importance of the realism of the terrain is limited. It is more important a DEM represents a fair test of processing time for given flow rate than be realistic in its modelling of the real-world location. Therefore, for modelling simplicity a basic and generic increase of 30cm was added for all building polygons to the smoothed terrain to represent buildings. While a more accurate DEM could be constructed by more detailed representation of building heights and other structures, it is not deemed necessary as both models are tested on the same terrain data. Furthermore, this means that output of models cannot be considered as realistic for the area. Therefore, for ethical reason the results cannot affect any of the residents, but are used purely for the purposes of testing the differences in model results.

A test point (pipe burst) location is selected from which a constant flow rate emerges onto the terrain for the full two hour simulation. The flow rate is dependent on the pipe diameter simulated. Pipe diameters from 100 mm to 1,000 mm are tested at increments of 100 mm, with two much larger diameters of 1,200 mm and 2,000 mm also tested. Outflow rates from the pipe burst are calculated using a pipe velocity of 1.2 m/s, and a full break of each pipe diameters is considered, i.e., multiplying the velocity by the circular area of the given pipe diameter gives the flow for each simulation. Initial 2D tests where run in CADDIES with the largest flow rates from the 2,000 mm bursting main, and GPU for processing speed, in order to estimate the total extent and a sensible size for the testing terrain. A 1.0 km^2 area was initially selected, however, flooding for the largest flow rates slightly exceeded this area, and therefore a 1.5 km^2 area was used.

InfoWorks ICM [2] is utilised as a comparative model using an irregular triangular mesh. However, this mesh must be configured to a specific grid for testing, where a maximum and minimum cell size are needed. This comes in the form of a maximum triangle size, and in order to keep to the shape of the terrain and follow the maximum triangle size, the meshing algorithm may form very small triangles. Small triangles compound the problem which will lead to undesired small time step for modelling. The minimum element size is then used to control the minimum mesh size for computing, by grouping sets of very small triangles, which are used to represent the terrain, together until this minimum element size is reached. Configuration is therefore chosen in order to provide the closest possible resolution and number of processing elements as the input LIDAR data, i.e., a maximum triangle size of 1.0 m^2 is utilised. The minimum element size was then lowered incrementally until the number of processing elements slightly exceeded the number of square-grid cells in the testing area of 1.5 million. That resulted in a minimum processing element size of 0.6 m^2. Much more effort can be put into model construction within InfoWorks, with large cities typically taking 6-12 months [6] to construct the variable resolution mesh over the terrain. However, for the purposes of comparative tests this is not necessary, as the objective is to compare the differences between the two models in terms of accuracy and speed, given a similar resolution. Finally, the output from the InfoWorks runs are translated back into the regular 1.0 m raster format so that comparisons to the CADDIES outputs can be made.

In order to overcome the checkerboard style oscillations a number of improved rules with flow limiters have been develop, including the Hunter-Bates [8] [5] [9], and the weighted model [14]. The limited flow rates avoid complete drying or overfilling of a cell in a single iteration. This avoids the critical condition of producing a dry cell or abnormally high water depth, where the water that left a cell in the previous iteration and, therefore, is present in the neighbours, in turn returns back into the cell in the successive iteration, often from all four directions. As the four inflows are calculated each as separate outflows from neighboring cells, they do not communicate that information in a single iteration. This results in cumulatively a very large depth in the receiving cell, which starts the process all over again, and continues to oscillate further and further. While these limited local rules are capable of operating at much higher time steps than previous none flow limited rules, without causing the checkerboard style oscillations, there is still an associated drop in the accuracy of the simulations as flows are unrealistically limited. This means rather than simply presenting the traditional quality measures and metrics, a trade-off is produced between the processing time of the simulation (controlled by the minimum time step) and the quality of its output. As the primary driver for the adaptive time step control is the maximum flow rate within the grid, the largest flows during the simulation will affect the trade-off. During a pipe burst scenario this is usually driven by the burst flow rate, and therefore a spectrum of different flow rates are tested. For each flow rate a range of different minimum time step values are tested, and the difference between rasterized InfoWorks and the CADDIES outputs measured.

Measures and metrics used include: Mean and Root Mean Squared Error (RMSE) over the entire terrain, but also on just the cells in which both models agree that there is a certain threshold level of flooding (wet both). Furthermore, the same is done for the cells in which either model passes the threshold level (wet either). The Nash Sutcliffe coefficient is also calculated over the entire terrain. Finally, in order to better measure the difference in flooding extents from the pipe bursts, a confusion matrix (shown in Table 1) is created for each simulation using the InfoWorks outputs as the 'true' (Benchmark) output. This determines which cells are flooded or not, and if the CADDIES output successfully predicts the flooding in the correct cells.

Table 1. Confusion matric from cells either positively or negatively flooded, by passing the threshold level.

	Model observation Positive	Model observation Negative
Benchmark Positive	True positive (TP)	False Positive (FP)
Benchmark Negative	False Negative (FN)	True Negative (TN)

From the confusion matrix shown in Table 1, three performance indicators are calculated, as used by Chang et al. [18], including Accuracy (ACC), Sensitivity (True Positive Rate; TPR), and Precision (positive predictive value; PPV). They are all used to evaluate the performance in terms of flood extent difference from InfoWorks models, as defined in Equations (4-6).

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

$$TPR = \frac{TP}{TP+FN} \tag{5}$$

$$PPV = \frac{TP}{TP+FP} \tag{6}$$

3. Experimental set-up

Two different multi-core I7 CPUs are used to benchmark the processing time with 4 cores and hyper threading, and an Nvidia Tesla K20c GPGPU with 2,500 CUDA cores for the many-core tests. InfoWorks does not appear to be able to perform full double precision (8 bytes) runs on the GPU, but only single precision (4 bytes). There is a large difference in performance of the GPGPU and CPU in terms of double and single floating point values, due to the fact that GPGPU are based on graphical technology where a single precision floating number is needed to represent a colour. The modern CPU is based on 64bit architecture and tuned for double precision values. Therefore, the CPU performs slightly slower with single as opposed to double floating point values. Surprisingly, the Nvidia GPGPU performs approximately twice as fast with single as opposed to double precision values [15], as two core are used to perform one double precision calculation. Due to the difference in the level of discretisation from a single floating point value, compared to a double, there is an associated drop in accuracy performance. Therefore, the processing times for CADDIES are shown using both single and double floating point values on the GPGPU. Comparison is also drawn on the difference in performance from a single precision simulation and the same double precision experiments.

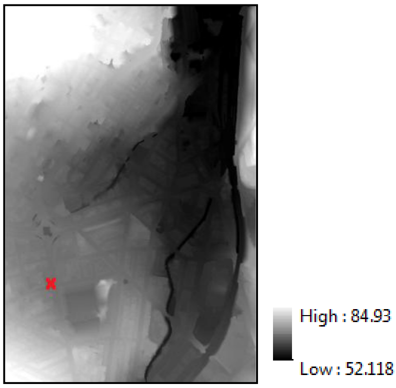


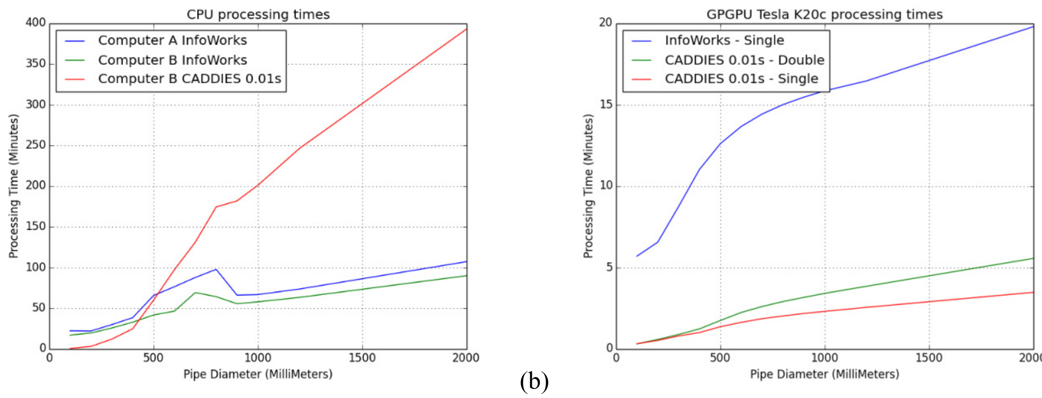
Fig. 1. The chosen test terrain location/extent (1km x 1.5km, totaling 1.5million cells) and burst location identified by the red cross.

Figure 1, shows the chosen terrain extent and the location of the pipe burst from which allowed flow enters the simulation. Using a maximum triangle size of 1.0 m2 and minimum element size of 0.6 m2 InfoWorks produced a mesh consisting of 1.17 million vertices used to create 2.33 million triangles and 1.57 million elements, which is close the 1.5 million cells in the original 1.0 m resolution raster grid. This simple meshing was completed in 2 minutes processing time, executed in a few attempts to narrow down the correct minimum element size. Many months can be spent on tailoring the spatial detail of the meshing with irregular grids, however, for comparative tests and ethical reason it is only necessary to achieve a similar resolution even though the irregular grid will vary slightly over space. The CADDIES caFloodPro application and weighted rule are applied with a maximum time step of one second, and a simulation is run with the minimum time step at 0.01 seconds up to 0.1 seconds at increments of 0.01 seconds, and then up to a 1 second time step at increments of 0.1 seconds.

4. Experimental Results

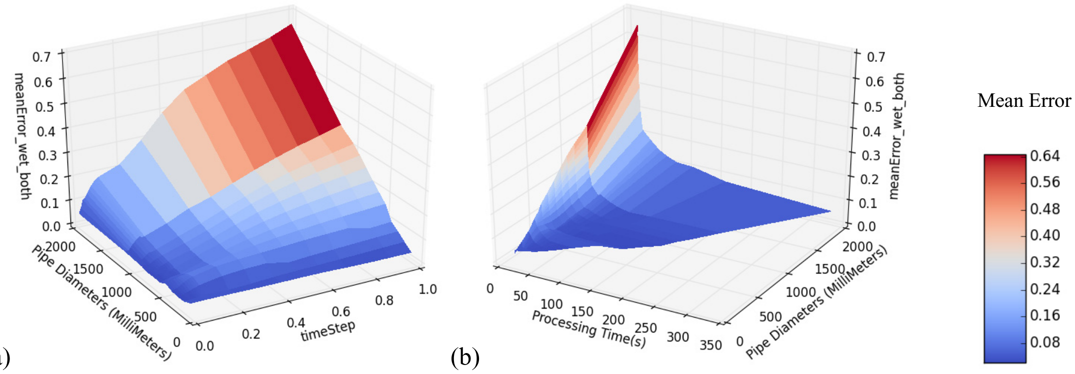
Figure 2 [a] shows how on the most accurate, shortest and slowest time step of 0.01 seconds the CADDIES system on the CPU quickly become slower than InfoWorks after approximately 500mm pipe diameter bursts. It is likely that the InfoWorks system is tuned to use a sensible default time step settings, so not to take too long in processing. As the InfoWorks time step settings have not been altered in these experiments InfoWorks results are presented at only a single processing time and accuracy for each flow rate/pipe diameter. The CADDIES simulations are slower on the CPU for most pipe diameters. However, these result are only for the most accurate (and slowest computationally) minimum time step tested. Figure 3 [b] shows the variation of processing time for each pipe diameter.

Figure 2 [b] shows how the regular nature of the CADDIES grid means that on the GPGPU (Tesla K20c) it outperforms InfoWorks at every pipe diameter, even on double precision compared to InfoWorks using single precision. Furthermore, single precision is nearly twice as fast as double precision on the GPGPU. Figure 3 [b] shows, a trade-off exists between the processing speed and accuracy of the simulation, controlled in this instance by the minimum time step value. Using knowledge of this trade-off it is possible to further accelerate the processing speed, while accepting only marginal losses to accuracy. The possible processing time savings also increase with the increased flow rates within the simulations. For example, given a maximum mean error (for cells where either model determines a flood) of 10 cm, it possible to gain up to twice the speed. As the flow rates are increased the time saving in real terms are also increased. Furthermore, by identifying the 'knee' region of the trade-off it is possible to determine a sensible minimum time step given the flow rates, for the best possible trade-off between speed and accuracy.



(a) (b)

Fig. 2. (a) Processing Time in minutes of the InfoWorks simulation with different inflow rates from the different pipe diameters, on Computers A and B, and the processing times of the CADDIES runs; (b) Processing times in minutes on the Tesla k20c GPGPU, for InfoWorks using single precision, and CADDIES using both single and double precision values. All CADDIES runs on the smallest and conversely slowest time step of 0.01 seconds.



(a) (b)

Fig. 3. (a) Resulting Mean errors over just the cells with either model indicating flooding of 0.01m or more; (b) Trade-off between processing time against mean error for cells where either model is flooded more than 0.01m, for each pipe size. Both figures are from the double precision runs on the Tesla K20c.

The comparisons in Figure 3 are based on the double precision models of both CADDIES and InfoWorks. However, as shown in Figure 2 [b], single precision runs on the many-core GPGPU are approximately twice the processing speed. Table 3 shows the recorded errors metrics and measures between the double precision CPU runs and single precision GPGPU runs for both CADDIES and InfoWorks. While scientific experimentation may call for the most accurate simulation possible, where large number of simulations are called for, industry and practitioners are likely to consider that these very small difference are acceptable. This is especially true considering that nearly twice the speed can be achieved, which over many thousands of simulations could be a sizeable amount of time in real terms. Considering that the most precise values expressible by double precision are in order of 10(+/-)308, compared to single precision values in order of 10(+/-)38, it would seem that the double precision is unnecessary in most circumstances. However, errors multiplied over many millions of cells and iterations of the algorithms could result in tenable errors as shown in Table 3.

Figure 4 [a] shows the InfoWorks output for the 2,000 mm pipe diameter burst. The most accurate CADDIES simulation using a 0.01 second time step is shown in Figure 4 [b], which takes 3 minutes 33 seconds to process. However, by accepting a 10cm error and using a minimum time step of 0.03 seconds, it is possible to gain twice the processing speed and not lose much in terms of accuracy, as shown in Figure 3 [c]. However, the trade-off also demonstrates when pushing the minimum time step too high (to 1.0 second) increases the error in the simulation drastically, as shown in Figure 3 [d].

Table 3. The difference between a simulation using double floating point precision (8 bytes) and single floating point precision (4 bytes); recordings made for the entire terrain, and also just wetted cells where both models agree on cells being wetted, and cells where either model suggests wetted. For uniformity, the comparison is made on CPU double runs, against GPU single precision runs.

Error Metric/Measure	CADDIES	InfoWorks
RMSE (m)	0.0007	0.0004
RMSE wet both (m)	0.0027	0.0004
RMSE wet either (m)	0.0054	0.0467
Mean error (m)	0.0001	0.0000
Mean error wet both (m)	0.0008	0.0000
Mean error wet either (m)	0.0030	0.0283
Accuracy (-)	0.9999	0.9999
Sensitivity (-)	0.9990	0.9998
Precision (-)	0.9996	0.9992
Nash Sutcliff coefficient (-)	1.0	1.0
True Positive (cells)	104,508	108,696
False Positive (cells)	45	90
True Negative (cells)	1,397,848	1,393,691
False Negative (cells)	100	24

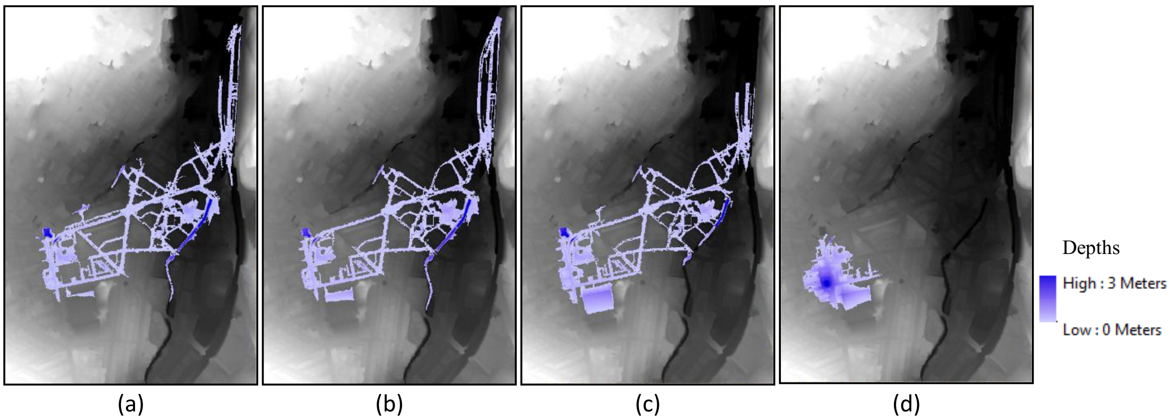


Fig. 4. Shows the varying depths (colour ranges shown in [e] between 0 and 10.92 meters) and extent at 2hours of simulation time for the 2,000 Millimeter pipe burst from (a) InfoWorks output (runtime: 19m47s); (b) CADDIES 0.01s time step output (3m33s); (c) CADDIES 0.03s time step output, Mean error (wetted both models) within tolerance of 0.1m (1m46s); (d) CADDIES 1.0s time step output (15s);

5. Conclusions

The push toward simplified models in recent years has the obvious advantage of huge processing speed-up over the full hydraulic models. The simplified models capture the majority of the flood dynamics, under the caveat that terms like the advective momentum are negligible. However, with the increasing number and variety of simplified models presented in literature, the trade-offs between processing speed and accuracy should be investigated in a more scientific manner. In this paper, the regular grid used by CADDIES allows much greater speed increases on the many-core GPGPU than the irregular mesh system used by InfoWorks. This is due to the regular and tightly knit nature of the many-core GPGPU hardware that fits better the square, regular grid than the irregular mesh. This result is exemplified by the fact that when using the same time step setting the CADDIES runs are much slower on the CPU but faster than InfoWorks on the GPU, even when InfoWorks uses a single precision and CADDIES uses double. The single floating point precision is shown to be nearly twice the processing speed on the GPU with only minor loss of accuracy.

The increase in the time step to manage the temporal discritisation of the simulation, forces the spatial rules to operate outside or beyond their normal range and creates an associated drop in accuracy. A problem with early cellular based rules was the likelihood of wild ‘checkerboard’ style oscillations when operating at a too large time step in relation to the cell size and flow rates. However, more recent CA rules often have more limiting local rules that stop or lower the excessive flow rates. This allows them to continue to operate at higher time steps with smaller errors, and thus operate at higher processing speeds. This paper has investigated this trade-off for the weighted model in the CADDIES framework, compared to the output from InfoWorks. This provides a useful guide to modelers as to which minimum step values can be selected to produce the fastest processing speed given a maximum threshold level of error.

Furthermore, the comparison reveals a sensible level of error given the request for the highest speeds. While this paper has not investigated the additional time step controls within InfoWorks, and the linear time step scaling of the alpha factor within CADDIES, this work begins to demonstrate how the time step controls can affect the trade-off between accuracy and processing speed. Meanwhile, the alpha parameter in the CADDIES system serves a similar purpose to the minimum time step factor, but it applies to the entire temporal domain. Either parameter has a similar effect from the point of view of limiting the temporal discritisation and computational complexity, in order to balance the speed and accuracy of the local spatial rules.

The other main time step control employed by the CADDIES system is the update time at which the change in the time step is calculated. This is done to avoid performing a reduction algorithm to find the maximum flow rate at every iteration. The algorithm retrieves a single maximum flow from the entire grid and calculates a new time step, thus reducing parallelism significantly. This becomes a bottleneck for the many-core GPU implementation. However, oscillations are possible when flow rates increase without the time step being updated at every iteration. In other words, a wider range of discritisations of time is possible when using adaptive time step compared to using a static time step. Furthermore, the computational complexity is less predictable when using adaptive time methods as it is difficult to foresee how many model iterations are necessary.

While this scale of time discritisation is more complex and higher dimensional with the adaptive time steps, altering the minimum time step presents the most interesting trade-off, as the alpha and update rates scale fairly linearly over the entire simulation. However, these parameters have the same effect of testing the local spatial rules balance between correctly propagating the flow and limiting excessive flows which cause destructive chain reactions (checkerboard oscillations).

6. Future work

The CADDIES framework is designed to be able to encompass the majority of cellular based models through its API, and therefore make fair comparisons of the different models performances. Currently there is only a regular square grid system for multiple hardware platforms. However, this is being expanded to include hexagonal grids and in the future could have triangular meshes, and finally the irregular version of each of the tessellatable shapes on a flat surface (Square/rectangle, Hexagon, and Triangles). Furthermore a number of other spatial and temporal rules are presented in literature, for example Hunter-Bates rules [9], which are similar to the weighed rule, but both have novel elements. In terms of the speed and accuracy trade-off and which rule is better; does one rule dominate another, or are there certain circumstances where rule outperforms the other, or vice versa; or is one rule faster while the other is more accurate. These questions are further complicated when the multi-dimensional nature of the adaptive time steps discritisation are also considered. However, one simplifying factor is the desire for faster rules, while maintaining a certain level of accuracy in relation to the real-world flooding phenomena.

Acknowledgements

The authors would like to acknowledge the funding provided by the UK Engineering and Physical Sciences Research Council, grant GR/J09796 (Simplified Dual-Drainage Modelling for Flood Risk Assessment in Urban Areas). The authors would also like to thank the UK Environment Agency for the LIDAR datasets, Ordnance Survey for the Master Maps, Innovyze for InfoWorks licence, and NVIDIA Corporation for the Tesla K20c GPU used in this research. This work is being developed under a UK government Knowledge Transfer Partnership (KTP) research contract 9721, in conjunction with ICS Consulting Limited. ICS are an asset management consulting business providing consulting services to asset intensive industries.

References

[1] M. Guidolin, A. Duncan, B. Ghimire, M. J. Gibson, E. C. Keedwell, A. S. Chen, S. Djordejevic and D. Savic, "CADDIES: A New Framework for Rapid Development of Parallel Cellular Automata Algorithms for Flood Simulation," Proceedings of the 10th International Conference on Hydroinformatics (HIC 2012), Hamburg, Germany, 14-18 July 2012, 2012.

[2] Innovyze, "InfoWorks ICM," [Online]. Available: http://www.innovyze.com/products/infoworks_icm. [Accessed 15 03 2016].

[3] DHI software, "MIKE FLOOD," [Online]. Available: https://www.mikepoweredbydhi.com/products/mike-flood. [Accessed 15 03 2016].

[4] J. Hénonin, Y. Zheng-Yu, J. Hartnack, K. Havnø, P. Gourbesville, O. Mark and M. Hongtao, "Citywide multi-grid urban flood modelling: the July 2012 flood in Beijing," *Urban Water Journal,* vol. 12, no. 1, pp. 52-66, 2015.

[5] N. M. Hunter, P. D. Bates, M. S. Horritt and M. D. Wilson, "Simple spatially-distributed models for predicting flood inundation: a review," *Geomorphology,* vol. 90, no. 3, pp. 208-225, 2007.

[6] K. Bradrook, "JFLOW: a multiscale two-dimensional dynamic flood model," *Water and Environment Journal,* vol. 20, no. 2, pp. 79-86, 2006.

[7] A. S. Chen, S. Djordjevic, J. Leandro and D. Savic, "The urban inundation model with bidirectional flow interaction between 2D overland surface and 1D sewer networks," in *NOVATECH*, Lyon, France, 2007.

[8] N. M. Hunter, M. S. Horritt, P. D. Bates, M. D. Wilson and M. G. Werner, "An adaptive time step solution for raster-based storage cell modelling of floodplain inundation," *Advances in Water Resources,* vol. 28, no. 9, pp. 975-991, 2005.

[9] P. D. Bates, M. S. Horrit and T. J. Fewtrell, "A simple inertial formulation of the shallow water equations for efficient two-dimensional flood inundation modelling," *Journal of Hydrology,* vol. 387, no. 1, pp. 33-45, 2010.

[10] OpenMP.org, "The OpenMP API specification for parallel programming," [Online]. Available: http://openmp.org/. [Accessed 7 04 2016].

[11] K. group and Khronos group, "Open CL - The open stand for parallel programming heterogreneous systems," [Online]. Available: https://www.khronos.org/opencl. [Accessed 7 04 2016].

[12] "CADDIES framework," University of Exeter, April 2016. [Online]. Available: available from emps.exeter.ac.uk/engineering/research/cws/resources/caddies-framework/) . .

[13] F. Dottori and E. Todini, "A 2D flood inundation model based on cellular automata approach," in *Proc. XVIII International Conference on Water Resources. In Carrera J (ed)* , Barcellona, 2010.

[14] B. Ghimire, A. S. Chen, M. Guidolin, E. C. Keedwell, S. Djordjević and D. Savić, "Formulation of fast 2D urban pluvial flood model using a cellular automata approach," *Journal of Hydroinformatics,* vol. 15, no. 3, pp. 676-686, 2013.

[15] M. Guidolin, A. S. Chen, B. Ghimire, E. C. Keedwell, S. Djordjević and D. Savić, "A weighted cellular automata 2D inundation model for rapid flood analysis," *in review,* 2016.

[16] M. J. Gibson, Genetic programming and cellular automata for fast flood modelling on multi-core CPU and many-core GPU computers (Thesis), University of Exeter, 2016.

[17] Ordnace Survey UK, "Ordnace Survey Master Maps," [Online]. Available: https://www.ordnancesurvey.co.uk/business-and-government/products/mastermap-products.html. [Accessed 2016 04 12].

[18] T.-J. Chang, C.-H. Wang and A. S. Chen, "A novel approach to model hynamics flow interactions between storm sewer system and overland surface for different land covers in urban areas," *Journal of Hydrology,* vol. 524, pp. 662-679, 2015.