

A cellular automata model of surface water flow

Jay A. Parsons* and Mark A. Fonstad

Department of Geography, Texas State University-San Marcos, San Marcos, TX 78666, USA

Abstract:

Previous cellular automata models of surface water flow have been constructed to reflect steady, gradually-varied flow conditions. While these models are extremely important in showing the near-equilibrium forms that result from the interactions of water and boundary material, highly dynamic forms and processes require models that represent unsteady flow conditions. In order to simulate unsteady flow in a cellular model of surface water flow, the conservation of mass and the Manning's equations are coupled with an algorithm to delay the movement of water from one pixel to the next until the correct timing is reached. This approach yields highly realistic flood wave hydrographs when compared with flood observations in the Walnut Gulch Experiment Watershed. Coupling this unsteady flow model with simple laws of sediment erosion, transport, and deposition should allow event-based simulations of watershed and river channel geomorphologic change. Copyright © 2007 John Wiley & Sons, Ltd.

KEY WORDS surface water hydrology; cellular automata; distributed models; calibration

Received 20 August 2006; Accepted 7 June 2006

INTRODUCTION

During the past two centuries, scientists and engineers have sought to develop models to represent and predict the behaviour of water flow within canals, rivers and over land. These models began with simple empirical descriptions of water flow and have progressed through the twentieth century to include incredibly complex models based on discretized, numerical solution of general laws (such as the conservation of energy and momentum). The guiding principle of these developments has been the idea that sufficient complexity in the model rules was necessary to mimic the complexity of actual river dynamics. The following short discussion argues that there is an alternative to this idea; spatial complexity can arise in a model with very simple rules as long as the spatial interactions of these rules are allowed to be sufficiently complex. The aim of this paper is to show how simple physical and semi-empirical rules of surface water motion can be coupled with the cellular automata representation of spatial process to produce realistic unsteady surface flow simulations of both hydrologic and hydraulic behaviour. In particular, this approach produces high-resolution in-channel flow (hydraulic) dynamics.

Traditionally, simulations of watershed runoff have been based on black-box, stochastic transfer functions or spatially-discretized, numerically-solved calculations based on simultaneous solution of the discharge equation and either the momentum equation or energy equation (or some combination). These more physically-based simulations generally simplify the constitutive equations

in order to allow a closed solution. For example, the use of only the steady-uniform part of the momentum equation produces a hydrologic routing procedure known as kinematic routing, and when an additional term for the water surface slope is added, the routing is called diffusion routing (Chaudhry, 1993).

Finite-difference or finite-element techniques that allow for numerical solution of these equations require simplifying assumptions. For example, they often assume a power-function relationship between discharge and flow depth for each discrete region. Computational stability of these numerical schemes is not certain in all flow situations, and the computer processing power needed for large areas is considerable. Typically, the processing necessary is reduced by lumping computational elements into larger spatial sizes. The relative merits of lumped, semi-distributed, or fully spatially distributed models are important concerns in computational hydrology (Mulligan, 2004), but are not the focus of this paper.

The integration of hydrodynamic models with large geographic systems can also be problematic. Despite considerable improvement, most geographic information system (GIS) software still have computational difficulties in integrating numerically iterative, time-varying processes with the geo-algebraic and visualization software components. The result has been a standard of loose-coupling the simulation packages with the GIS, a situation requiring modellers to have knowledge of both.

An alternative grid-based simulation framework allows both time-varying and geo-algebraic solution of spatial dynamics, cellular automata (CA). Instead of numerically integrating the solution to a map-wide global equation (such as a finite-difference solution to the momentum equation), CA systems compute only local processes.

* Correspondence to: Jay A. Parsons, Department of Geography, Texas State University-San Marcos, 601 University Drive, San Marcos, TX 78666, USA. E-mail: jparsons@txstate.edu

Global, map-wide behaviour emerges from these local processes and is not hardwired into the model as in traditional formulations (Nicholas, 2005). CA use in spatial studies has been primarily limited to urban simulations and land cover change modelling. In the last decade, however, methods for describing physically deterministic systems in the CA framework have become much more accurate, and with the introduction of various 'time-chaining' approaches, the possibility now exists to merge separate CA simulations of different environmental systems into unified 'multiautomata' models.

In the following model, the simple river rule known as the Manning's equation is combined with the conservation of mass and allowed to spatially interact throughout a two-dimensional lattice to produce sufficiently complex and real hydrodynamics as would be expected in a real watershed with rivers. The Manning's equation allows the CA to calculate the amount of time each cell must process the local hydrodynamics before allowing the local water volume to move, forming the time-chained basis for unsteady flow routing in this CA formulation of hydrology. This approach is similar to that made by Favis-Mortlock *et al.* (2000) in simulating the growth of small rills in response to hillslope erosion. De Roo *et al.*'s (2000) LISREL approach uses a similar cellular approach for watershed and floodplain modelling, but uses a kinematic wave approximation for in-channel flows. Coulthard *et al.* (2002) and Coulthard *et al.* (2005) have recently introduced a gradually-varied CA for catchment response modelling that produce realistic sediment transport dynamics despite not including unsteady watershed hydrology. The original CA river model was developed by Murray and Paola (1994), which included both hydrology and geomorphology, but whose representation did not include explicit time or real physical scaling directly in the model. Thomas and Nicholas (2002) extended this original model to simulate more realistic water flow dynamics in braided river systems. As with other semi-distributed or fully distributed models of hydrology, correctly parameterizing the model for making predictions is an important, difficult process, but this CA formulation simplifies part of that calibration procedure by simplifying the hydrodynamic representations to a few locally-applied rules. The possible spatio-temporal dynamics in even the simplest CA's is very similar to the level of dynamics allowed in the continuum mechanical description of the universe (Toffoli, 1984).

The CA formulation of surface water hydrology immediately invites the extension of this class of models to include sediment transport, groundwater flow, channel change, and watershed vegetation dynamics, all of which could be represented in a similar manner and directly coupled with the following basic hydrological model. While the model currently runs independently, it could be easily ported to many GIS packages, many of which would allow simpler and more elegant visualization of the model output.

DATA INPUTS

The CA flow model requires minimal data sets to simulate a study area's surface water flow. These inputs are entered primarily as raster layers, allowing the model to interface easily with GIS software outputs. A digital elevation model (DEM) defines the watershed and facilitates calculating slope and flow direction. An infiltration layer or an average value representing the average infiltration rate for the entire basin determines the amount of surface water absorbed through the soil per time step. A roughness layer or an average roughness value for the entire site enables the calculation of water velocity during each simulation time step. Precipitation duration and intensity describe the simulated storm event.

DEMs are the primary data input for a study site. High resolution DEMs, 10 m or better, are necessary to accurately predict surface water flow over the given terrain, particularly if in-channel hydraulics are to be visualized at the same time as overall basin hydrology. Once acquired, DEMs for the study area can be cropped and corrected in a variety of GIS or remote sensing software packages. The DEM must encompass the entire watershed; however, minimizing the number of cells eliminates unnecessary calculations and reduces computation time.

The model requires an infiltration value or values for the study area to determine the water lost per cell per time interval. This value describes the maximal amount of surface water lost to infiltration (soil absorption) per time unit. This value may be input as one average value for the entire study area or an 'infiltration layer' may be created based on soil classification. While specifically termed infiltration, this value can also represent all loss including infiltration and evapo-transpiration. Though this step relies on infiltration-limited overland flow, saturation-limited behaviour could also be used with this model, but not in this introductory version.

A hydraulic roughness coefficient is necessary to calculate surface water flow velocity. One form of this input is a standard (average) Manning's n value for the entire study area. This should be accurate enough for research in most simple, uniform watersheds. More diversely covered research areas would require specific roughness values for different sections of the study area. A 'friction' data layer can be developed by classifying ground cover in the study area and associating a roughness value for each ground cover type. For example, areas with exposed rock will have a low friction value of 0.011 while areas with dense growth may have a friction value of 0.2.

Basic precipitation events are modelled with a constant precipitation rate for a specified duration. To model a specific storm with varying rates over time, a control program is customized to input varying precipitation amounts into the model over time.

CA CONSTRUCTION

Models based on CA theory consist of four primary components: a lattice of cells, the definition of a local

'neighbourhood' area, rules determining the changes in cell properties ('transition rules'), and boundary conditions. Each of these components influences the state of an individual cell and in turn the overall behaviour of the system.

A two-dimensional lattice of cells for this model is created from a digital elevation model and encompasses the entire watershed. The lattice is generally created from a 10 m spatial resolution DEM, but higher resolution DEMs may be used if available. Each cell is instantiated with inherent spatial coordinates, an elevation value, a friction coefficient, and an infiltration rate. The initial model uses uniform friction coefficient and infiltration rate values across the CA lattice, these values can be spatially varied in later versions of the model.

Neighbourhoods in CA models define areas of process influence. The state of each cell during time $t + 1$ is determined, in part, by the state in a 'neighbour' cell at time t . A neighbourhood is the group of cells around an individual centre cell that affect the centre cell's properties at a later time. The Von Neumann neighbourhood, for example, is the group of four cells in the four cardinal directions from the centre cell (Manhattan neighbours), whereas the Moore neighbourhood also includes the diagonal neighbours. Neighbourhoods including a wider radius than these simple neighbourhoods may also be employed, termed the extended Von Neumann and extended Moore neighbourhoods. Hexagonal grids employ the most symmetric two-dimensional neighbourhood, where each neighbour is the same distance around the centre cell and (unlike the Von Neumann neighbourhood) totally surrounds each part of the centre cell. Unfortunately, the extended and hexagonal neighbourhoods can be much more difficult to integrate with known physical laws and/or with GIS databases than simple neighbourhoods. In addition, the diagonal neighbours in the Moore neighbourhood are a different distance from the centre cell than are the cardinal neighbours, which cause some difficulties when producing accurate timed water flow. For this reason, the Von Neumann neighbourhood has been chosen to be used as the basis of the CA model at this early stage of the model. Not only do the four neighbourhood cells affect the state of the centre cell but they are also the cells that can receive water from the centre cell. Because the transition rules described later in this section produce a certain amount of symmetric water diffusion through time, the simplification of water flow to four of the surrounding cells (and not the diagonals) should not excessively affect the overall performance of the model, especially if the water flow spans at least several cells.

The CA rules dictating surface water flow between cells are based on physical processes involved in water movement but are modified to work in the simulated world of the CA lattice of cells. The rules dictate the addition of water to each cell through precipitation as well as the subtraction of water due to infiltration (loss) in order to maintain the conservation of mass. By incorporating rules for unsteady flow, this model does

not require flow field stabilization as would be needed in a steady-state model. A rule based on the Manning's equation calculates water velocity (Equation (1)) in each cell, where the velocity (vel) is a function of the cell's calculated water depth (volume divided by pixel area), water surface slope S , and the input roughness n . In Equation (1), the water depth is an approximate measure of the normally-used hydraulic radius. This is the same approximation typically used in overland flow uses of the Manning's equation. The water surface slope(s) is calculated using the Zevenbergen and Thorne's method (1987). This approach first calculates the water depth, and then calculates the water surface slope in each cell based on the new water depth and elevation. This allows the computed depths, slopes, and velocities to change dynamically with varying flow conditions, an improvement over many water flow CA models. The program uses this velocity to calculate how many seconds (T_t) it will take for the water to traverse the cell by dividing the velocity rate by cell width (Equation (2)). The water in the cell is held there until this time condition is met. Because the calculated traverse time in seconds is likely to be not a multiple of the time step length chosen (a second or fraction of a second) by the software operator, the traverse time is rounded up or down to the nearest whole time step length. Once the water in a cell is released by the program to its neighbours the time condition is reset.

$$\text{vel} = \frac{\text{depth}^{2/3} \cdot S^{1/2}}{n} \quad (1)$$

$$T_t = \text{width} / \text{vel} \quad (2)$$

The final rule executes when the water has traversed the cell and can be transferred to the surrounding cells in its neighbourhood as long as the cells have a lower water surface elevation. First, the model calculates which portion of the cell's water volume is 'moveable', meaning what volume of the cell's water is above the neighbour with the lowest water surface elevation. If all of the neighbours have a water surface elevation greater than the centre pixel's, there is no 'moveable' water. Second, the model calculates which of the neighbour cells have water surface elevations lower than the centre cell's water surface elevation (in other words, downstream neighbours). Finally, the model transfers the 'moveable' water volume to the neighbour cells in proportion to the differences in water surface elevations. For example, if two neighbours have lower water surface elevations than the centre cell, but one of the two neighbours is twice as low as the other, then two-thirds of the centre cell's 'moveable' water would go to the lowest downstream neighbor and a third would go to the other downstream neighbour.

For each time step, the model applies the CA rules to each cell in the following specific order. (1) The length of the time step is chosen by the user so that water will not cross a pixel in less than one time step. (2) Precipitation is added to cells then infiltration (loss)

is subtracted to get the current water depth in the cell. (3) The water surface slope is then calculated using the Zevenbergen and Thorne's method (1987). (4) Water velocity is then calculated using depth, surface slope, and a friction coefficient. (5) The final cell property calculation determines the amount of time required for the water to traverse the cell and sets the traverse time value. (6) Then each cell containing water is checked to determine if the traverse time has been completed, and if not, the counter is simply incremented one additional time step. (7) If the traverse time has been met then the water can leave the cell. (8) Flow direction and quantity is determined by surface elevation of the centre cell and its neighbours. (9) The difference in surface water elevation between the lowest neighbour and the centre cell determines the amount of water available to move. The 'moved' amount of water is then stored in a buffer layer until the entire lattice completes processing, then that water is added to the cell's water depth.

The final component of a CA model is the boundary condition. The boundary condition of a CA determines what happens at the outer most cells of the lattice. Since these outer cells do not have a complete neighbourhood (by definition), special conditions apply to their behaviour. CA models may incorporate several types of well-known boundaries, such as 'reflective' (virtual 'walls' around the modelled area), 'periodic' (water flowing off one edge re-enters the grid on the opposite side from where it left), or 'absorbing' (water flowing off the edge disappears). This model uses an absorbing boundary to remove the water from the study area. The outermost cells of the lattice of cells absorb any water moved into them by moving the water into a storage value that does not affect water movement in the rest of the CA. To reduce boundary slope errors, it is possible to artificially add a water depth value to the boundary corresponding to a neighbouring cell's water depth. While this is not necessary for water movement in the CA, the values are stored to track the volume of water leaving the study area.

MODEL IMPLEMENTATION

The CA for modelling surface water flow was developed in the Java programming language. Java was selected as the development language due to its object-oriented nature and inherent multi-platform support. Logical components of the model are encapsulated in objects that can be easily extended or replaced with new objects to change input formats or enhance model behaviour. The core modelling code requires only a standard Java installation. The visualization component (Figure 1) requires an additional visualization library, VisAD (VisAD, 2004). Data input into the model are raster layers exported from GIS as text files or as constant values from configuration files. Other programming languages may provide better faster performance for the calculation intensive CA model but performance was not a consideration for this initial implementation.

CALIBRATION AND TESTING

The Walnut Gulch Experimental Watershed (WGEW) near Tucson, Arizona is an excellent study site because of its wealth of precipitation and flow gauges and gauge records. Researchers from the Southwest Watershed Research Center (SWRC) provided several of the necessary inputs including a DEM and storm event data to validate the model. A sub-watershed within the WGEW provided validation data for the model. The watershed is approximately 8.24 km² and includes a precipitation gauge near its centre that recorded the event that was used for initial model testing. The sub-watershed conditions are described as 20% covered by desert shrubs and 80% desert grasses (SWRC, 2005).

The summer rain events that occur in this area are typical of semi-arid areas; the storms are usually very intense and have a short duration. A storm of this type occurred on August 4, 1980, with 4.27 cm (1.68 in) of total precipitation. The majority of the precipitation fell in the first 47 min of the event.

Multiple simulations were performed to determine the impact of varying the unknown parameters of roughness and infiltration rates. Given the size of the watershed pixels, a time step equal to 0.25 s was chosen to be used in order to avoid the possibility that water could cross the pixel in less than one time step, and to more accurately calculate water movement. The Manning's n value for roughness varied between 0.01 and 0.02 while the infiltration rate varied between 0 mm/h and 10.8 mm/h. These n values are quite low; this can be explained by existing desert pavement, low vegetation, and likely a lack of diagonal water movement in the model. The results from each simulation are then compared to the observed flow recorded from the flume. The observed flow is shown in Figures 2, 3 and 4. A peak of 25.29 m³ s⁻¹ occurred at 53 min after the start of the storm event.

The results of the first simulation (Sim 1), shown in Figure 2, demonstrate how the model reacts to a constant infiltration rate of 7.2 mm/h and 0.01 for a Manning's n roughness value. In Sim 1, the peak discharge of 26.59 m³ s⁻¹ occurred at a time of 36 min after the storm event began. The simulated hydrograph is similar to the observed hydrograph; the peak values are close but the peak in the simulation occurs 17 min earlier than in the observed.

Simulation 2 (Sim 2), shown in Figure 3, shows the result of a constant infiltration rate of 7.2 mm/h and a Manning's n value of 0.02. In this simulation, a peak discharge of 18.46 m³ s⁻¹ occurred at 42 min after the event started. The hydrograph from this simulation is much 'flatter' than the observed. The peak discharge of the simulation is 6.83 m³ s⁻¹ less than the observed.

Simulation 3, depicted in Figure 4, shows the results of an infiltration rate of 10.8 mm/h and a Manning's n value of 0.01. In this simulation, a peak discharge of 32.13 m³ s⁻¹ occurred at 40 min. The hydrograph from this simulation maintains the shape of the observed

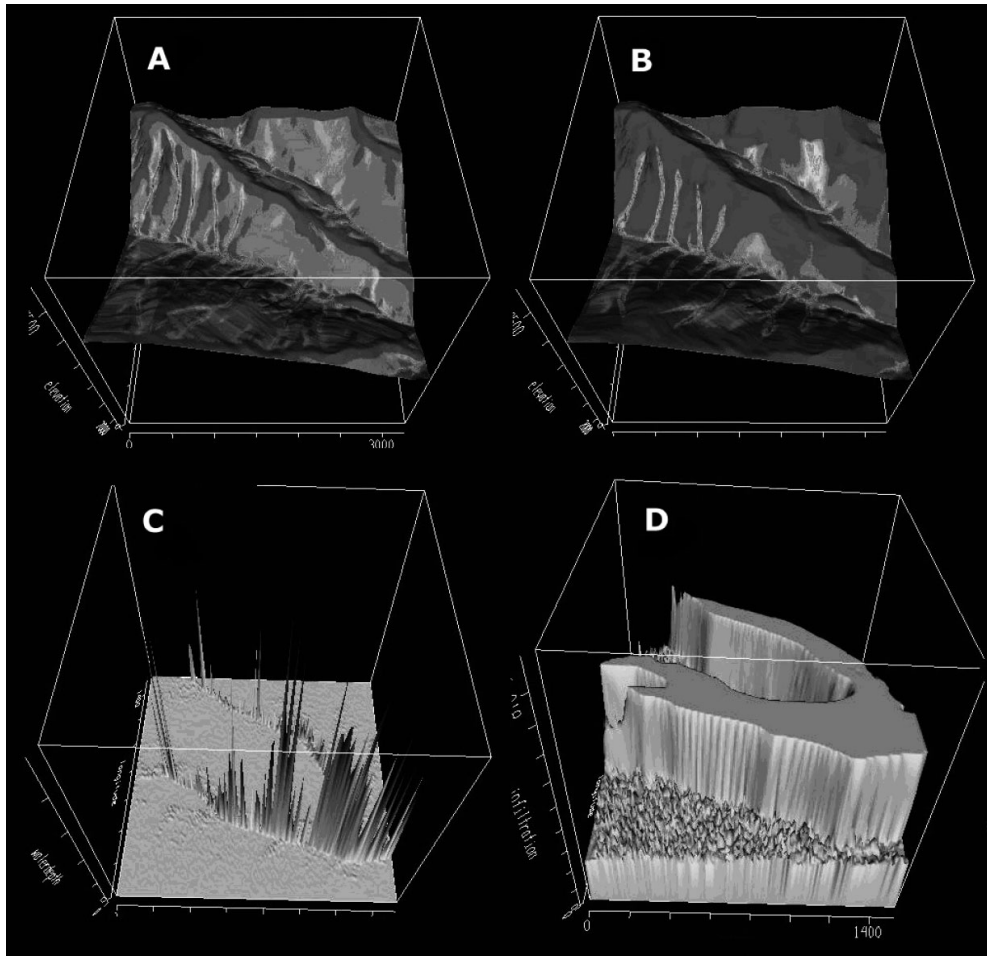


Figure 1. Three-dimensional views of a CA rainfall-runoff model. (a and b) Water depth during an ongoing rainstorm, with lighter colours representing deeper water (images made at times 5 and 20 min into the precipitation event), (c) a three-dimensional exaggerated view of water depth in cells at one time during the runoff period, (d) a three-dimensional exaggerated view showing the distribution of accumulated infiltrated water within the basin

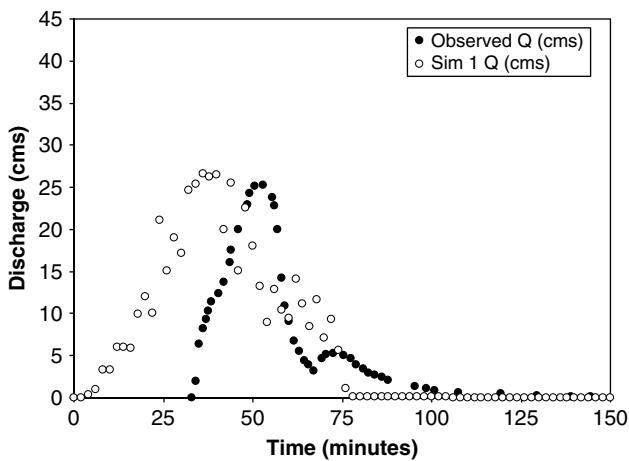


Figure 2. Comparison of Sim 1 hydrograph to observed hydrograph (cubic meters per second (cms) ($m^3 s^{-1}$))

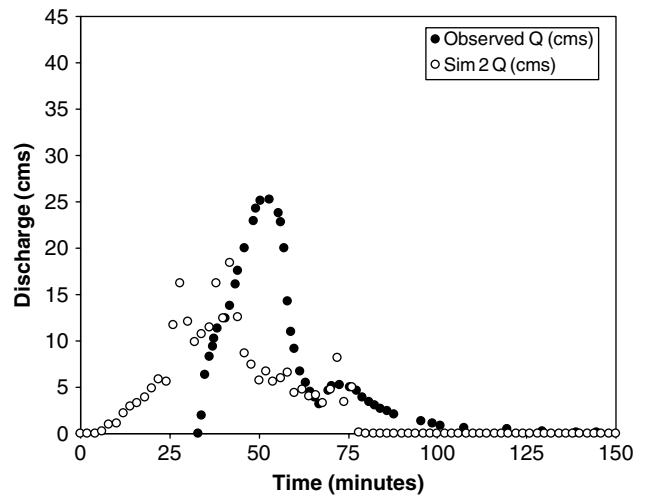


Figure 3. Comparison of Sim 2 hydrograph to observed hydrograph

hydrograph but peaks 13 min early for an additional $6.84 m^3 s^{-1}$.

Overall, the performance of the CA model appears reasonably realistic, and with appropriate calibration of the input parameters (such as roughness), yields runoff discharges within a few percentage of the observed

discharges. Often rising limb discharge reaches the gauge in these simulations sooner than was actually observed. This is likely due either to precipitation not being uniform (as in this simulation) but rather concentrated in the upper portion of the basin during the actual event, or the

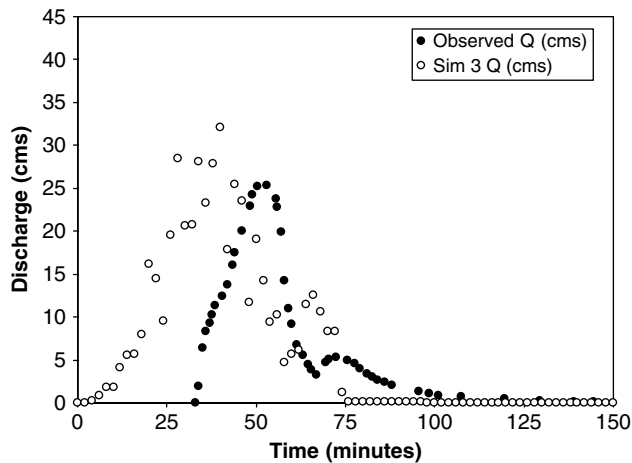


Figure 4. Comparison of Sim 3 hydrograph and observed hydrograph

infiltration rule used is too simple to accurately portray the rising limb magnitudes.

Because the Walnut Gulch test areas have very small channels, it is difficult to see how well the CA model demonstrates floodwave routing across a floodplain. As a further example, Figure 5 shows the simulated spatially distributed flood water depths below Canyon Lake Dam, Texas after a flood in 2002. A $1400 \text{ m}^3 \text{ s}^{-1}$ flow over the dam's emergency spillway produces a wide range of water depths, showing the normally expected distribution of shallower depths near the margins of the progressing floodwave.

DISCUSSION

A major consideration of CA models is the computational requirements needed to run simulations. For example, for a hypothetical grid of 100×100 cells, assuming five calculations per cell per time step, this equals 50 000 calculations per time step, summing to 4 320 000 000 calculations for a 24-h event simulation. This requirement grows as a power function of the grid size. The simulations described previously took up to 6 h on a low-end notebook computer. In principle, multiprocessor, parallel, or distributed computing could greatly reduce computational time in this model formulation, but most end users will not have access to these advantages.

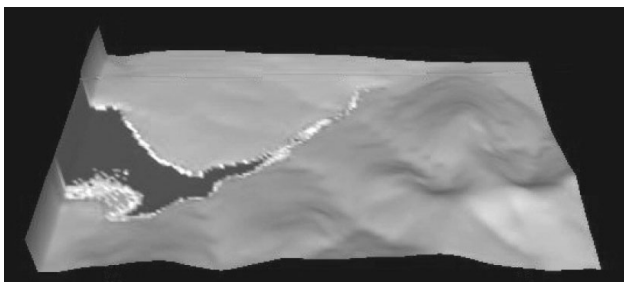


Figure 5. Floodplain flow dynamics below Canyon Lake Dam, Texas. Flow is from left to right. The darker shades of the water represent deeper water areas

Some space and time asymmetry in the water movement may occur as the result of using the Von Neumann neighbourhood in the model's calculations. Switching to the Moore neighbourhood would be an advance over our model, but this will require the addition of a timing scheme that includes diagonals because of the greater length from the centre cell to the diagonals compared to that of the cardinal cells.

The method used to 'hold' the water in the cell until the correct length of time (the traverse time) has been reached creates an irregular, 'jerky' water movement at the time scale of individual model time steps. This is inherent to the model and smaller time steps do not remove the effect but only make them visible at smaller intervals. This high-frequency signal can be artificially smoothed after the simulation is finished in order to produce standard hydrographs, but it does not fundamentally affect the model performance. Other approaches to building true time into the CA may be feasible in the future.

Figures 2, 3, and 4 show that the responses of the simulated events are fairly sensitive to input parameters such as roughness and infiltration rate. From this standpoint, the model's parameter sensitivity is in line with other spatially distributed models such as those based on distributed kinematic wave modelling. In these various simulations it was meant to be shown the general sensitivity of the model to these parameters, but these values were also used in a 'calibration' (Baird, 2004) manner in order to best match the simulated to the observed discharge. It is possible the use of the Von Neumann neighbourhood (no diagonal movement) causes the calibrated roughness to be too low, water velocities to be too high and water depths to be too low.

Future additions or improvements to this CA model fall into four categories: (1) changes in the CA neighbourhood, (2) alterations to the transition rules, (3) improvements in the number of modelled components, and (4) improvements in the input representation of the watershed environment. Improving the Von Neumann neighbourhood will require either altering the underlying tessellation of the grid to produce a hexagonal lattice, or more likely incorporating two sets of time rules so that the Moore neighbourhood can be used.

The transition rules were developed to be simple and easy to implement. Significant improvements could come from improving the physicality of certain model components. These could include adding infiltration rules such as capacity limits. Additional model components with their own transition rules might be useful in model realism. For example, the CA model that has been described has no baseflow component, which could be added if a model component was available that described shallow groundwater flow, perhaps using the Darcy's flow equation and the conservation of mass.

Additional processing of DEMs, for example, changing the pixel elevations to include known-channel geometry, should improve the model's realism. For example, Lidar, stereo-photogrammetry, and field surveys could greatly improve channel elevation precision.

Finally, there are many methods for improving the overall computing performance required to run CA simulations over large areas or long time spans. These methods include both algorithmic enhancements and alterations in the program architecture (such as multi-threading). Improving distributed model computational performance is a research area in need of active work.

CONCLUSIONS

The CA model described in this paper is computationally straightforward, simple in conception, and produces highly realistic watershed simulations with only a modicum of parameterization. The Walnut Gulch simulations show that the time-chaining approach is reasonable in stormflow situations, although it is somewhat sensitive to the time step length used in the simulation. The sensitivity of the model to variations in pixel size and parameter uncertainty is quite similar to classical distributed models of watershed runoff. Hydrographs are implicitly generated for every pixel in the watershed, making three-dimensional visualization of the hydrologic response simple and elegant.

Generally, the limitations of the CA approach to hydrology and hydraulics as have been described are those of omission. Baseflow, evaporation losses, or saturation-capacity limits have not been explicitly described. Although the precipitation inputs used were homogeneous for the test basins, there is no restriction from using spatially-varying rain data, such as radar-derived precipitation maps. In some extreme hydraulic situations such as rapidly varied flow and highly three-dimensional flow, the Manning's equation will fail to represent velocity accurately, and alternate rules could be substituted, such as a locally-operating momentum conservation rule. Computational power required for applying this model to large areas is considerable, but future increases in processing power and the use of parallelized code applied to multi-processor computers will make this problem essentially one of time and money.

Perhaps the most useful attribute of CA hydrologic and hydraulic modelling is its sheer computational simplicity, easily adapted to many environments and GIS packages.

The simplicity makes future extension of this approach straightforward, and allows a tighter integration between modelling, altering parameters, and visualizing results.

ACKNOWLEDGEMENTS

The authors wish to thank W. Andrew Marcus for insightful conversations about this modelling approach and for reviewing an early draft of this manuscript. Goeff Jacques and Carl Legleiter similarly proved to be ideal sounding boards during the development of this hydrological approach.

REFERENCES

- Baird A. 2004. Soil and hillslope hydrology. In *Environmental Modelling*, Wainwright J, Mulligan M (eds). John Wiley & Sons: Chichester; 93–106.
- Chaudhry MH. 1993. *Open-Channel Flow*. Prentice-Hall: New York; 640.
- Coulthard TJ, Macklin MG, Kirkby MJ. 2002. A cellular model of Holocene upland river basin and alluvial fan formation. *Earth Surface Processes and Landforms* **27**: 269–288.
- Coulthard TJ, Lewin J, Macklin MG. 2005. Modelling differential catchment response to environmental change. *Geomorphology* **67**: 222–241.
- De Roo A, Wesseling C, Van Deursen W. 2000. Physically-based river basin modeling within a GIS: the LISFLOOD model. *Hydrological Processes* **14**(11–12): 1981–1992.
- Favis-Mortlock DT, Boardman J, Parsons AJ, Lascelles B. 2000. Emergence and erosion: a model for rill initiation and development. *Hydrological Processes* **14**(11–12): 2173–2205.
- Mulligan M. 2004. Modelling catchment hydrology. In *Environmental Modelling*, Wainwright J, Mulligan M (eds). John Wiley & Sons: Chichester; 107–122.
- Murray AB, Paola C. 1994. A cellular model of braided rivers. *Nature* **371**(6492): 54–57.
- Nicholas AP. 2005. Cellular modeling in fluvial geomorphology. *Earth Surface Processes and Landforms* **30**: 645–649.
- Southwest Watershed Research Center(SWRC). *Walnut Gulch Experimental Watershed*. Available: <http://www.tucson.ars.ag.gov/unit/Watersheds/WGEW.htm> [accessed August 29, 2005].
- Thomas R, Nicholas AP. 2002. Simulation of braided river flow using a new cellular routing scheme. *Geomorphology* **43**: 179–195.
- Toffoli T. 1984. Cellular automata as an alternative to (rather than an approximation of) differential equations in modeling physics. *Physica D* **10**: 117–127.
- VisAD. 2004. VisAD Home Page. Available online at <http://www.ssec.wisc.edu/~billh/visad.html>. Accessed May 10, 2005.
- Zevenbergen LW, Thorne CR. 1987. Quantitative analysis of land surface topography. *Earth Surface Processes and Landforms* **12**: 47–56.